

Отчёт по лабораторной работе №6

Информационная безопасность

Арбатова Варвара Петровна

Содержание

1	Цель работы	5
2	Теоретическое введение	6
3	Выполнение лабораторной работы	8
4	Выводы	15
	Список литературы	16

Список таблиц

Список иллюстраций

3.1	Установка	8
3.2	Выполняю настройки	8
3.3	Убеждаюсь в работе	9
3.4	Запуск сервера	9
3.5	Статус работы	9
3.6	Ищу контекст безопасности	10
3.7	Просмотр текущего состояния	10
3.8	Просмотр статистики по политике	11
3.9	Смотрю права	11
3.10	Файлы директории	12
3.11	Создание файла	12
3.12	Заполняю файл	12
3.13	Сайт	13
3.14	Изменение контекста	13
3.15	Сайт	14
3.16	Проверка	14

1 Цель работы

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

2 Теоретическое введение

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 [1].

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.

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

Более подробно см. в [a].

3 Выполнение лабораторной работы

Перехожу в корневую директорию и устанавливаю httpd

```
[vparbatova@vparbatova ~]$ su root
Пароль:
[root@vparbatova vparbatova]# yum install httpd
packages for the GitHub CLI 5.6 kB/s | 3.0 kB 00:00
packages for the GitHub CLI 7.0 kB/s | 2.8 kB 00:00
Rocky Linux 9 - BaseOS 12 kB/s | 4.1 kB 00:00
```

Рис. 3.1: Установка

Перехожу в директорию /etc/httpd, чтобы настроить веб-сервер. Добавляю строку `ServerName test.ru` в файл `httpd.conf`, указывая имя сервера. Очищаю правила `iptables`, чтобы сбросить настройки фаервола. Устанавливаю политику АСЦЕРТ для входящих подключений, разрешая входящий трафик. Устанавливаю политику АСЦЕРТ для исходящих подключений, разрешая исходящий трафик.

```
[root@vparbatova vparbatova]# cd /etc/httpd
[root@vparbatova httpd]# echo "ServerName test.ru" >> httpd.conf
[root@vparbatova httpd]# iptables -F
[root@vparbatova httpd]# iptables -P INPUT ACCEPT
[root@vparbatova httpd]# iptables -P OUTPUT ACCEPT
[root@vparbatova httpd]#
```

Рис. 3.2: Выполняю настройки

Убедилась, что SELinux работает в режиме `enforcing` политики `targeted` с помощью команд `getenforce` и `sestatus`


```
[root@vparbatova httpd]# getenforce
Enforcing
[root@vparbatova httpd]# 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@vparbatova httpd]# 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)
lines 1-4/4 (END)
```

Рис. 3.3: Убеждаюсь в работе

Запускаю сервер apache, далее обращаюсь с помощью браузера к веб-серверу, запущенному на компьютере, он работает, что видно из вывода команды `service httpd status` (рисунки 4-5)

```
[root@vparbatova httpd]# service httpd start
Redirecting to /bin/systemctl start httpd.service
```

Рис. 3.4: Запуск сервера

```
[root@vparbatova httpd]# 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: active (running) since Wed 2025-04-23 19:13:36 MSK; 23s ago
     Docs: man:httpd.service(8)
  Main PID: 41610 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served: 0"
    Tasks: 177 (limit: 12178)
   Memory: 21.9M
      CPU: 86ms
   CGroup: /system.slice/httpd.service
           └─41610 /usr/sbin/httpd -DFOREGROUND
             └─41611 /usr/sbin/httpd -DFOREGROUND
               └─41615 /usr/sbin/httpd -DFOREGROUND
                 └─41616 /usr/sbin/httpd -DFOREGROUND
                   └─41617 /usr/sbin/httpd -DFOREGROUND

anp 23 19:13:36 vparbatova systemd[1]: Starting The Apache HTTP Server...
anp 23 19:13:36 vparbatova httpd[41610]: AH00558: httpd: Could not reliably determine the server's fully qualified domain name, because the 'ServerName' directive has not been set yet.
anp 23 19:13:36 vparbatova systemd[1]: Started The Apache HTTP Server.
anp 23 19:13:36 vparbatova httpd[41610]: Server configured, listening on: port 80
lines 1-20/20 (END)
```

Рис. 3.5: Статус работы

С помощью команды `ps auxZ | grep httpd` нашла веб-сервер Apache в списке процессов. Его контекст безопасности - `httpd_t`

```
[root@vparbatova httpd]# ps auxZ | grep httpd
system_u:system_r:httpd_t:s0 root 41610 0.0 0.5 21240 11584 ?
Ss 19:13 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 41611 0.0 0.3 22972 7660 ?
S 19:13 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 41615 0.0 0.5 982524 11516 ?
Sl 19:13 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 41616 0.0 0.5 982524 11512 ?
Sl 19:13 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 41617 0.0 0.6 1113660 13836 ?
Sl 19:13 0:00 /usr/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 41810 0.0 0.1 221688
2432 pts/0 S+ 19:16 0:00 grep --color=auto httpd
[root@vparbatova httpd]#
```

Рис. 3.6: Ищу контекст безопасности

Просмотрела текущее состояние переключателей SELinux для Apache с помощью команды `sestatus -b | grep httpd`

```
[root@vparbatova httpd]# sestatus -b | grep httpd
httpd_anon_write off
httpd_builtin_scripting on
httpd_can_check_spam off
httpd_can_connect_ftp off
httpd_can_connect_ldap off
httpd_can_connect_mythtv off
httpd_can_connect_zabbix off
httpd_can_manage_courier_spool off
httpd_can_network_connect off
httpd_can_network_connect_cobbler off
httpd_can_network_connect_db off
httpd_can_network_memcache off
httpd_can_network_relay off
httpd_can_sendmail off
httpd_dbus_avahi off
httpd_dbus_sssd off
httpd_dontaudit_search_dirs off
httpd_enable_cgi on
httpd_enable_ftp_server off
httpd_enable_homedirs off
httpd_execmem off
httpd_graceful_shutdown off
```

Рис. 3.7: Просмотр текущего состояния

Просмотрела статистику по политике с помощью команды `seinfo`. Множество

пользователей - 8, ролей - 39, типов - 5135

```
[root@vparbatova httpd]# 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:            5169     Attributes:         259
Users:            8        Roles:              15
Booleans:         358     Cond. Expr.:       390
Allow:            65633    Neverallow:         0
Auditallow:       176     Dontaudit:         8703
Type_trans:       271851   Type_change:        94
Type_member:      37       Range_trans:        5931
Role allow:       40       Role_trans:         417
Constraints:      70       Validatetrans:      0
MLS Constrains:  72       MLS Val. Tran:      0
Permissives:      1       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@vparbatova httpd]#
```

Рис. 3.8: Просмотр статистики 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

```
[root@vparbatova httpd]# ls -lZ /var/www
итого 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 янв 22 03:
25 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0 6 янв 22 03:
25 html
[root@vparbatova httpd]#
```

Рис. 3.9: Смотрю права

В директории /var/www/html нет файлов.

```
[root@vparbatova httpd]# ls -lZ /var/www
итого 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 янв 22 03:
25 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0 6 янв 22 03:
25 html
[root@vparbatova httpd]# ls -lZ /var/www/html
итого 0
```

Рис. 3.10: Файлы директории

Создаю файл

```
[root@vparbatova httpd]# touch /var/www/html/test.html
```

Рис. 3.11: Создание файла

Создать файл может только суперпользователь, поэтому от его имени созда-
ем файл `touch.html` со следующим содержанием

```
[root@vparbatova httpd]# echo '<html>' >> /var/www/html/test.html
[root@vparbatova httpd]# echo '<bode> test </body>' >> /var/www/html/test.html
[root@vparbatova httpd]# echo '<body> test </body>' >> /var/www/html/test.html
[root@vparbatova httpd]# ls -lZ /var/www/html/test.html
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 48 янв 23 19
:32 /var/www/html/test.html
[root@vparbatova httpd]#
```

Рис. 3.12: Заполняю файл

Перехожу на сайт и смотрю. Всё удачно

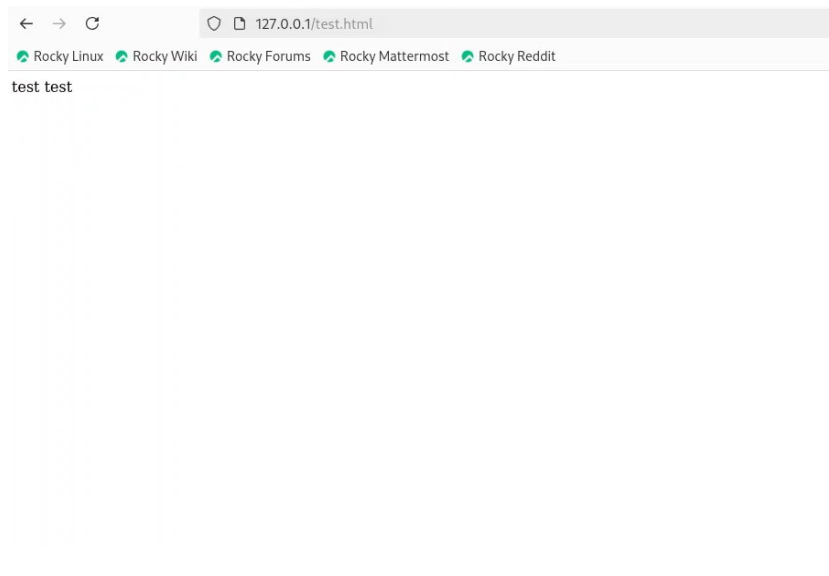


Рис. 3.13: Сайт

Изменяю 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` Контекст действительно поменялся

```
[root@vparbatova httpd]# chcon -t samba_share_t /var/www/html/test.html
[root@vparbatova httpd]# ls -lZ /var/www/html/test.html
-rw-r--r--. 1 root root unconfined_u:object_r:samba_share_t:s0 48 anp 23 19:32 /var/www/html/test.html
[root@vparbatova httpd]# ls -Z /var/www/html/test.html
unconfined_u:object_r:samba_share_t:s0 /var/www/html/test.html
[root@vparbatova httpd]#
```

Рис. 3.14: Изменение контекста

Доступ запрещен

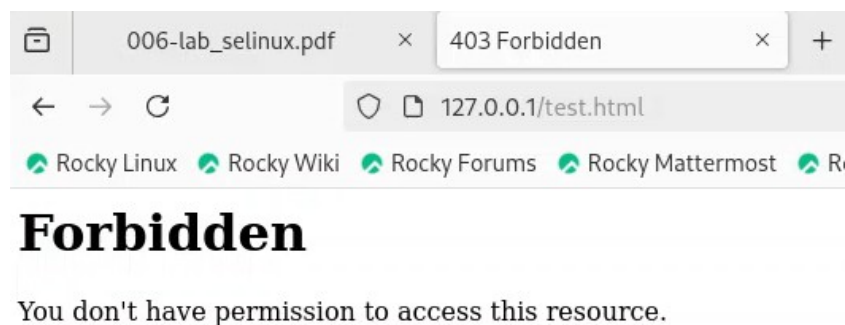


Рис. 3.15: Сайт

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 `log`-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`

```
[root@vparbatova httpd]# ls -l /var/www/html/test.html
-rw-r--r--. 1 root root 48 anp 23 19:32 /var/www/html/test.html
[root@vparbatova httpd]# tail /var/log/messages
Apr 23 19:38:56 vparbatova setroubleshoot[42764]: SELinux запрещает /usr/sbin/httpd
доступ getattr к файл /var/www/html/test.html.#012#012***** Модуль restorecon
предлагает (точность 92.2) *****#012#012Если вы хотите испр
авить метку.$TARGETзнак _PATH по умолчанию должен быть httpd_sys_content_t#012To
вы можете запустить restorecon. Возможно, попытка доступа была остановлена из-за
недостаточных разрешений для доступа к родительскому каталогу, и в этом случае по
пытаться соответствующим образом изменить следующую команду.#012Сделать#012# /sb
in/restorecon -v /var/www/html/test.html#012#012***** Модуль public_content пред
лагает (точность 7.83) *****#012#012Если вы хотите лечить test.h
tml как общедоступный контент#012То необходимо изменить метку test.html с public_
```

Рис. 3.16: Проверка

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.

Список литературы