

Отчёт по лабораторной работе №6

Знакомство с SELinux

Куденко Максим Русланович

Содержание

1	Цель работы	4
2	Выполнение лабораторной работы	5
2.1	Подготовка	5
2.2	Изучение механики SetUID	5
3	Выводы	13
	Список литературы	14

List of Figures

2.1	запуск http	6
2.2	контекст безопасности http	6
2.3	переключатели SELinux для http	7
2.4	создание html-файла и доступ по http	8
2.5	ошибка доступа после изменения контекста	9
2.6	лог ошибок	10
2.7	переключение порта	11
2.8	доступ по http на 81 порт	12

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

2 Выполнение лабораторной работы

2.1 Подготовка

1. Установили httpd
2. Задали имя сервера
3. Открыли порты для работы с протоколом http

2.2 Изучение механики SetUID

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`.

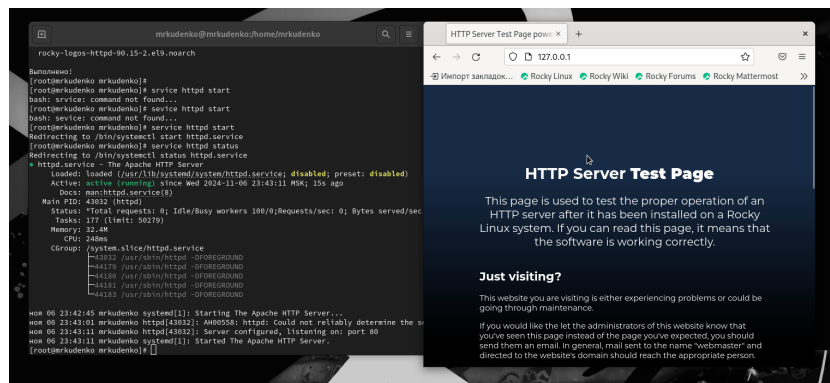


Figure 2.1: запуск http

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`

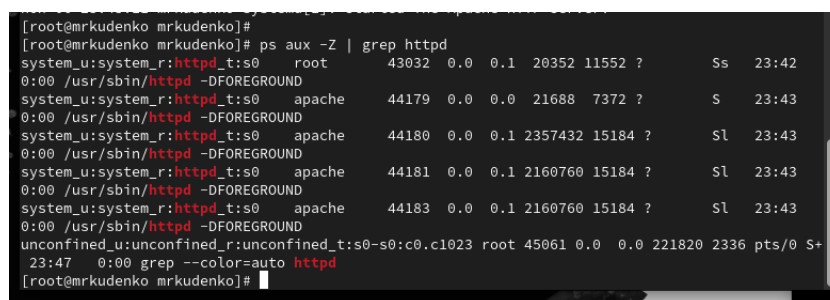


Figure 2.2: контекст безопасности http

4. Посмотрите текущее состояние переключателей SELinux для Apache с помощью команды `sestatus -bigrep httpd`. Обратите внимание, что многие из них находятся в положении «off».

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: Test
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.

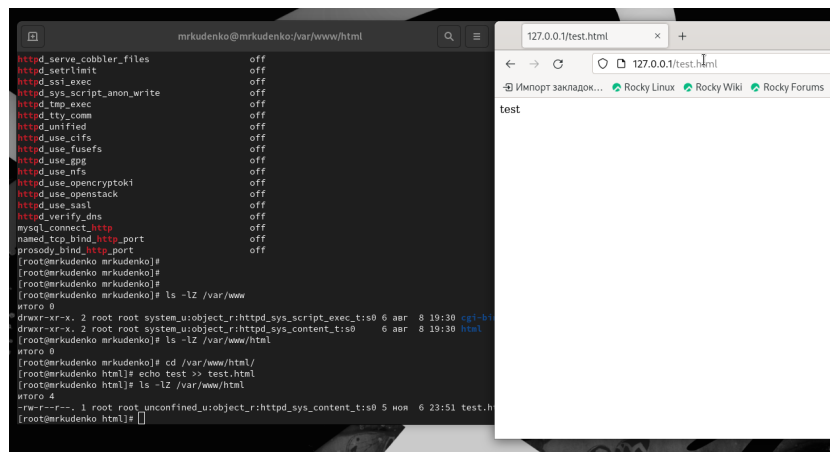


Figure 2.4: создание html-файла и доступ по http

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`. Основным контекстом является `httpd_sys_content_t`, его мы и увидели в выводе команды.
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.` При изменении контекста файл стал считаться чужим для `http` и программа не может его прочитать.

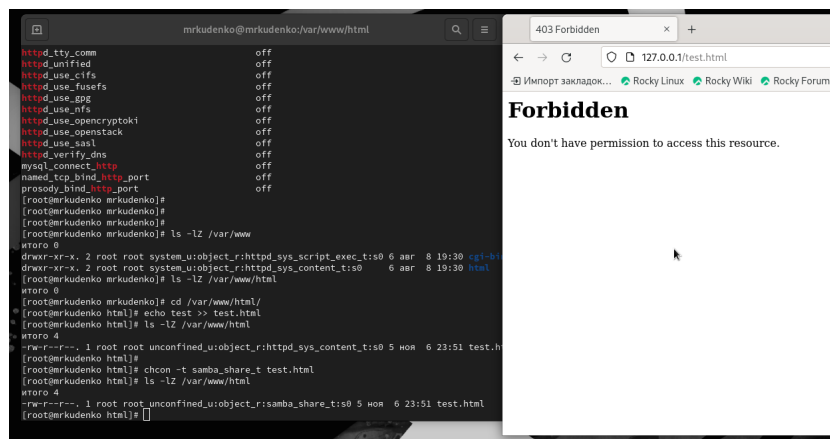


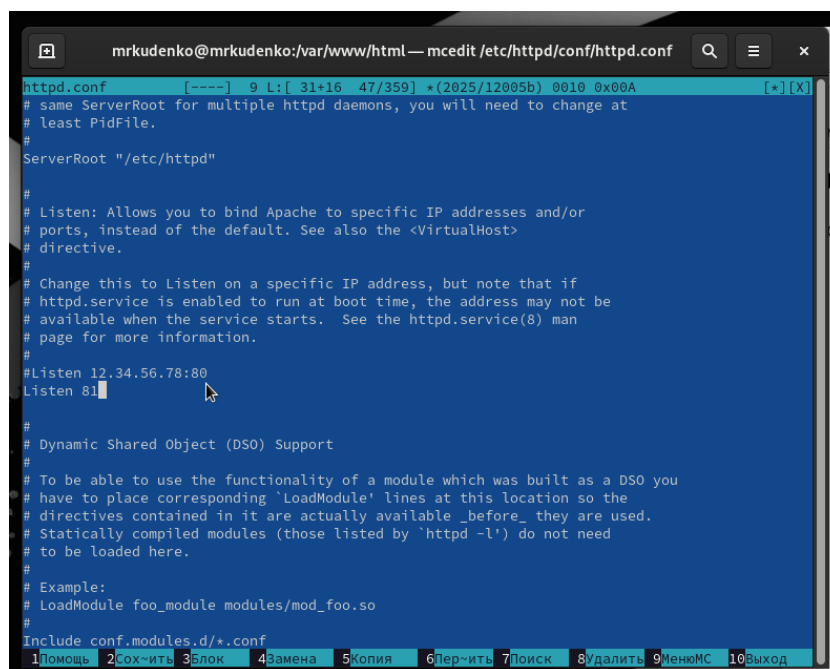
Figure 2.5: ошибка доступа после изменения контекста

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.

```
mrkudenko@mrkudenko:/var/www/html
Сделать#012# semanage fcontext -a -t public_content_t '/var/www/html/test.html'#012# restoreco
n -v '/var/www/html/test.html'#012#012***** Модуль catchall предлагает (точность 1.41) *****
*****#012#012Если вы считаете, что httpd должно быть разрешено getattr доступ
к test.html file по умолчанию.#012То рекомендуется создать отчет об ошибке.#012Чтобы разрешит
ь доступ, можно создать локальный модуль политики.#012Сделать#012разрешить этот доступ сейчас,
выполнив:#012# ausearch -c 'httpd' --raw | audit2allow -M my-httpd#012# semodule -X 300 -i my
-httpd.pp#012
Nov  6 23:51:57 mrkudenko setroubleshoot[45227]: SELinux запрещает /usr/sbin/httpd доступ geta
ttr к файлу /var/www/html/test.html. Для выполнения всех сообщений SELinux: sealert -l 92f8632e
-bcbb-4021-80be-6ff49897fa99
Nov  6 23:51:57 mrkudenko setroubleshoot[45227]: SELinux запрещает /usr/sbin/httpd доступ geta
ttr к файлу /var/www/html/test.html.#012#012***** Модуль restorecon предлагает (точность 92.2)
*****#012#012Если вы хотите исправить метку.$TARGETЗнак _PATH по умолчан
ию должен быть httpd_sys_content_t#012То вы можете запустить restorecon. Возможно, попытка дос
тупа была остановлена из-за недостаточных разрешений для доступа к родительскому каталогу, и в
этом случае попытайтесь соответствующим образом изменить следующую команду.#012Сделать#012# /
sbin/restorecon -v /var/www/html/test.html#012#012***** Модуль public_content предлагает (точ
ность 7.83) *****#012#012Если вы хотите лечить test.html как общедоступный ко
нтейнер#012То необходимо изменить метку test.html с public_content_t на public_content_rw_t.#012
Сделать#012# semanage fcontext -a -t public_content_t '/var/www/html/test.html'#012# restoreco
n -v '/var/www/html/test.html'#012#012***** Модуль catchall предлагает (точность 1.41) *****
*****#012#012Если вы считаете, что httpd должно быть разрешено getattr доступ
к test.html file по умолчанию.#012То рекомендуется создать отчет об ошибке.#012Чтобы разрешит
ь доступ, можно создать локальный модуль политики.#012Сделать#012разрешить этот доступ сейчас,
выполнив:#012# ausearch -c 'httpd' --raw | audit2allow -M my-httpd#012# semodule -X 300 -i my
-httpd.pp#012
Nov  6 23:52:07 mrkudenko systemd[1]: dbus-:1.1-org.fedoraproject.SetroubleshootPrivileged@1.s
ervice: Deactivated successfully.
Nov  6 23:52:07 mrkudenko systemd[1]: dbus-:1.1-org.fedoraproject.SetroubleshootPrivileged@1.s
ervice: Consumed 1.126s CPU time.
Nov  6 23:52:07 mrkudenko systemd[1]: setroubleshootd.service: Deactivated successfully.
Nov  6 23:52:07 mrkudenko systemd[1]: setroubleshootd.service: Consumed 1.080s CPU time.
[root@mrkudenko html]#
```

Figure 2.6: лог ошибок

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.



```
mrkudenko@mrkudenko:/var/www/html — mcedit /etc/httpd/conf/httpd.conf
httpd.conf [----] 9 L: [ 31+16 47/359] *(2025/12005b) 0010 0x00A [*][X]
# same ServerRoot for multiple httpd daemons, you will need to change at
# least PidFile.
#
ServerRoot "/etc/httpd"
#
# 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
#
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding 'LoadModule' lines at this location so the
# directives contained in it are actually available _before_ they are used.
# Statically compiled modules (those listed by 'httpd -l') do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
Include conf.modules.d/*.conf
1 Помощь 2 Сохранить 3 Блок 4 Замена 5 Копия 6 Перенести 7 Поиск 8 Удалить 9 Меню MS 10 Выход
```

Figure 2.7: переключение порта

17. Выполните перезапуск веб-сервера Apache. Произошёл сбой? Поясните почему? Сбой не происходит, порт 81 уже вписан в разрешенные
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.
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».

3 Выводы

В процессе выполнения лабораторной работы мною были получены базовые навыки работы с технологией seLinux.

Список литературы

1. SELinux в CentOS
2. Веб-сервер Apache