

Отчёт по лабораторной работе №6

Знакомство с SELinux

Деакиссим Манн Орнела НФИбд-01-19

Содержание

1	Цель работы	4
2	Выполнение лабораторной работы	5
2.1	Подготовка	5
2.2	Изучение механики SetUID	5
3	Выводы	12
	Список литературы	13

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	10
2.8	доступ по http на 81 порт	11

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`.

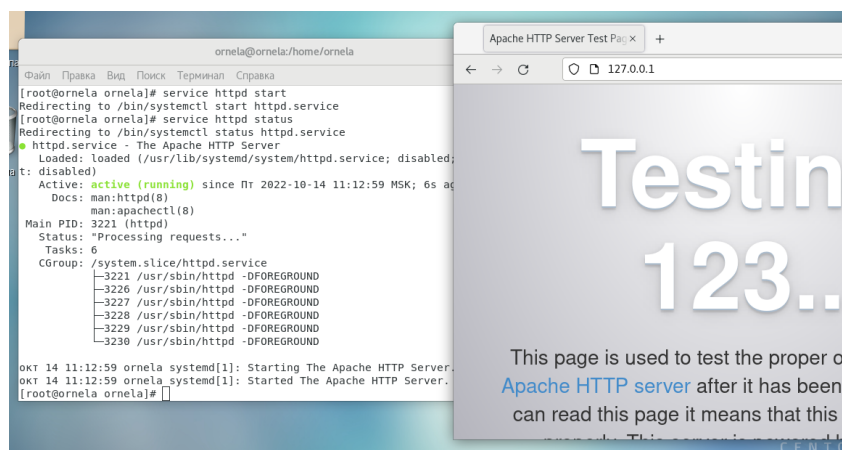


Figure 2.1: запуск http

- 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`

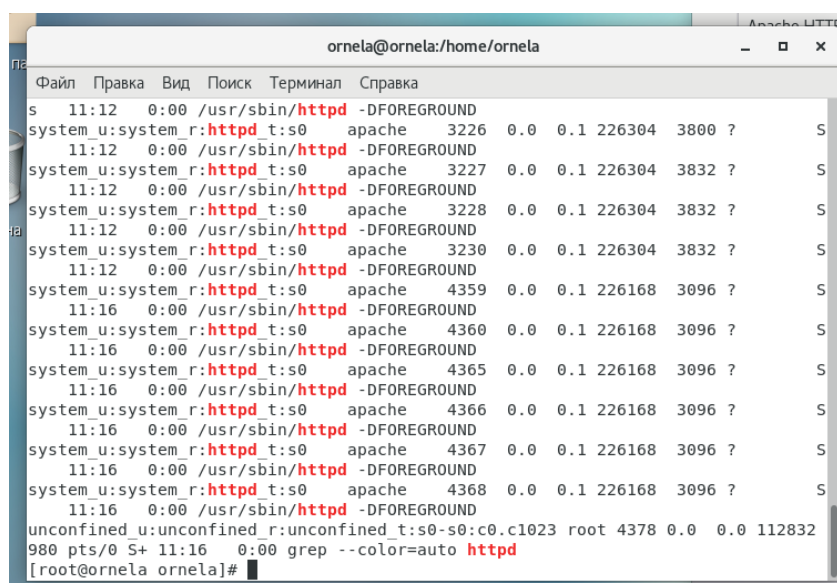
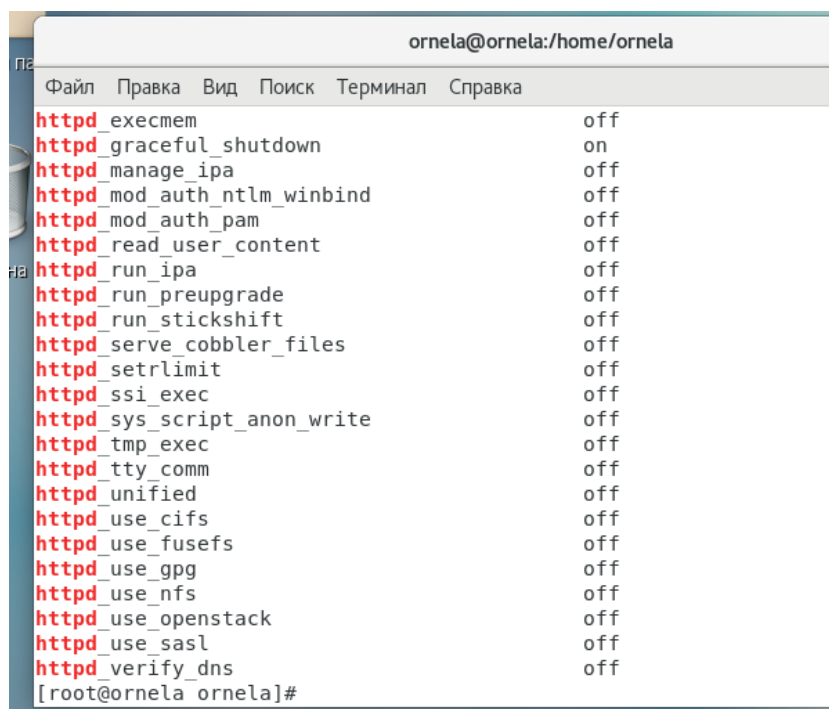


Figure 2.2: контекст безопасности http

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



```
ornela@ornela:/home/ornela
Файл Правка Вид Поиск Терминал Справка
httpd_execmem off
httpd_graceful_shutdown on
httpd_manage_ipa off
httpd_mod_auth_ntlm_winbind off
httpd_mod_auth_pam off
httpd_read_user_content off
httpd_run_ipa off
httpd_run_preupgrade off
httpd_run_stickshift off
httpd_serve_cobbler_files off
httpd_setrlimit off
httpd_ssi_exec off
httpd_sys_script_anon_write off
httpd_tmp_exec off
httpd_tty_comm off
httpd_unified off
httpd_use_cifs off
httpd_use_fusefs off
httpd_use_gpg off
httpd_use_nfs off
httpd_use_openstack off
httpd_use_sasl off
httpd_verify_dns off
[root@ornela ornela]#
```

Figure 2.3: переключатели SELinux для http

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`. В поддиректориях могут располагаться системные скрипты и контент для http.
7. Определите тип файлов, находящихся в директории `/var/www/html`: `ls -lZ /var/www/html`. В директории изначально нет файлов.
8. Определите круг пользователей, которым разрешено создание файлов в директории `/var/www/html`. Создавать файлы может только root.
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.

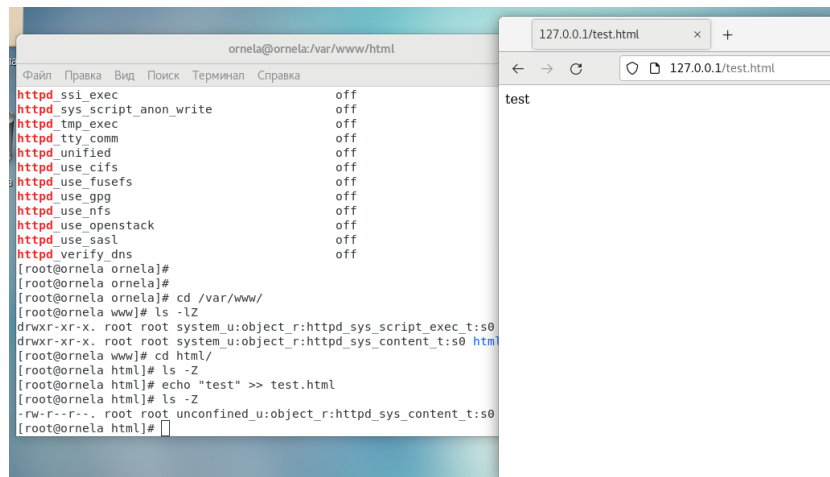


Figure 2.4: создание html-файла и доступ по http

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 -lZ`. `ls -lZ /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 -lZ /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.

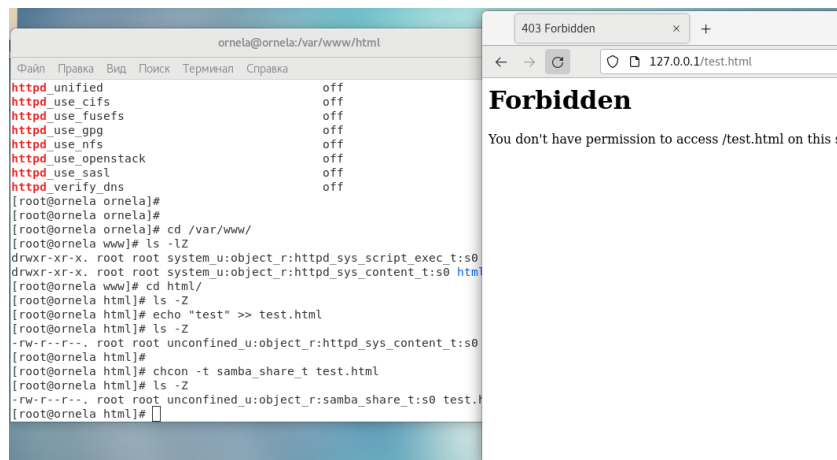
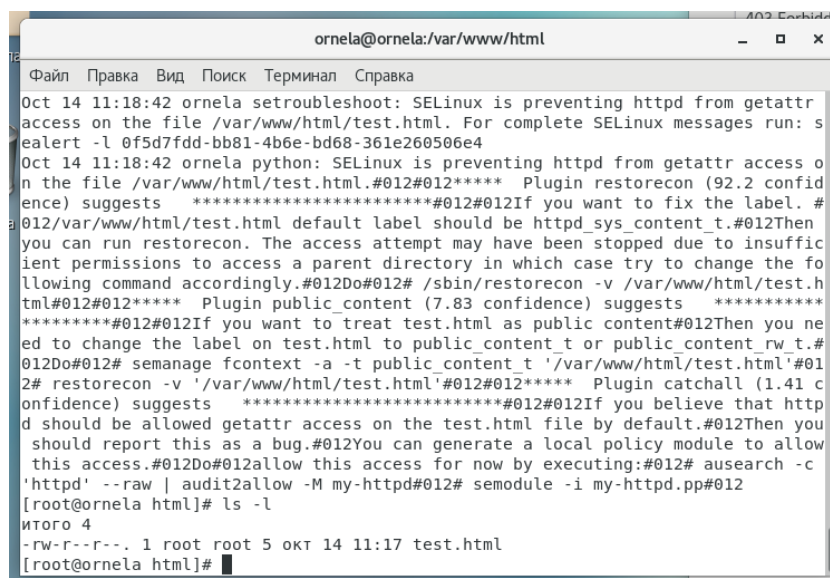


Figure 2.5: ошибка доступа после изменения контекста

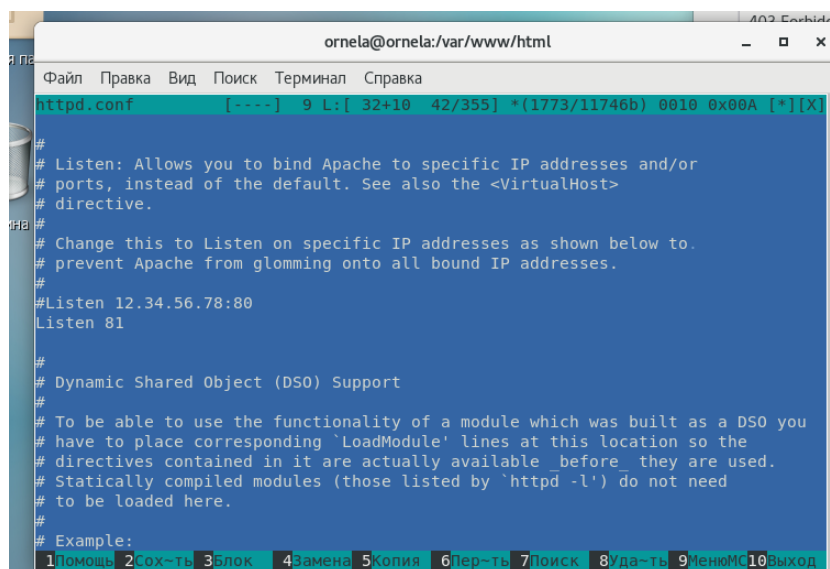
15. Проанализируйте ситуацию. Почему файл не был отображён, если права доступа позволяют читать этот файл любому пользователю? `ls -l /var/www/html/test.html` Просмотрите log-файлы веб-сервера Apache. Также просмотрите системный лог-файл: `tail /var/log/messages` Если в системе окажутся запущенными процессы `setroubleshootd` и `audtd`, то вы также сможете увидеть ошибки, аналогичные указанным выше, в файле `/var/log/audit/audit.log`. Проверьте это утверждение самостоятельно.



```
ornela@ornela: /var/www/html
Файл Правка Вид Поиск Терминал Справка
Oct 14 11:18:42 ornela setroubleshoot: SELinux is preventing httpd from getatrr
access on the file /var/www/html/test.html. For complete SELinux messages run: s
ealert -l 0f5d7fdd-bb81-4b6e-bd68-361e260506e4
Oct 14 11:18:42 ornela python: SELinux is preventing httpd from getatrr access o
n the file /var/www/html/test.html.#012#012***** Plugin restorecon (92.2 confid
ence) 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 insuffic
ient permissions to access a parent directory in which case try to change the fo
llowing command accordingly.#012Do#012# /sbin/restorecon -v /var/www/html/test.h
tml#012#012***** Plugin public_content (7.83 confidence) suggests *****
*****#012#012If you want to treat test.html as public content#012Then you ne
ed 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'#01
2# restorecon -v '/var/www/html/test.html'#012#012***** Plugin catchall (1.41 c
onfidence) suggests *****#012#012If you believe that http
d should be allowed getatrr 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 -i my-httpd.pp#012
[root@ornela html]# ls -l
итого 4
-rw-r--r--. 1 root root 5 окт 14 11:17 test.html
[root@ornela 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.



```
ornela@ornela: /var/www/html
Файл Правка Вид Поиск Терминал Справка
httpd.conf [----] 9 L: [ 32+10 42/355] *(1773/11746b) 0010 0x00A [*][X]
#
# 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 specific IP addresses as shown below to
# prevent Apache from glomming onto all bound IP addresses.
#
#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:
1Помощь 2Сох-ть 3Блок 4Замена 5Копия 6Пер-ть 7Поиск 8Ид-ть 9Имя 10Выход
```

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».

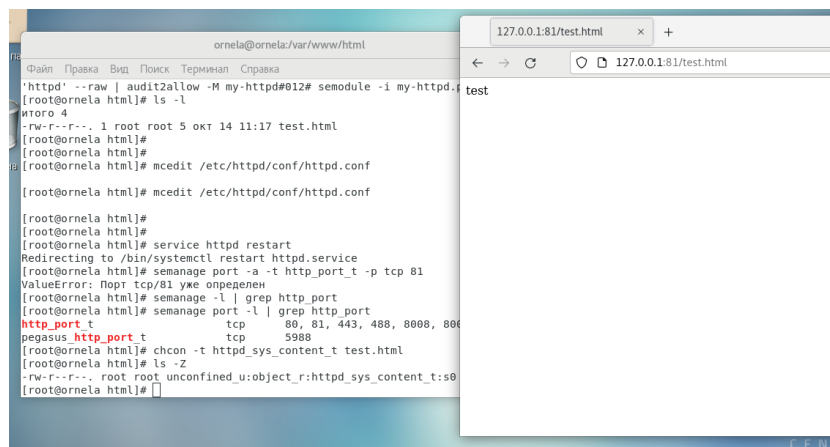


Figure 2.8: доступ по http на 81 порт

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`

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