

# **Отчёт по лабораторной работе №6**

**Знакомство с SELinux**

Билолиддин Набиев

# Содержание

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

# 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`.

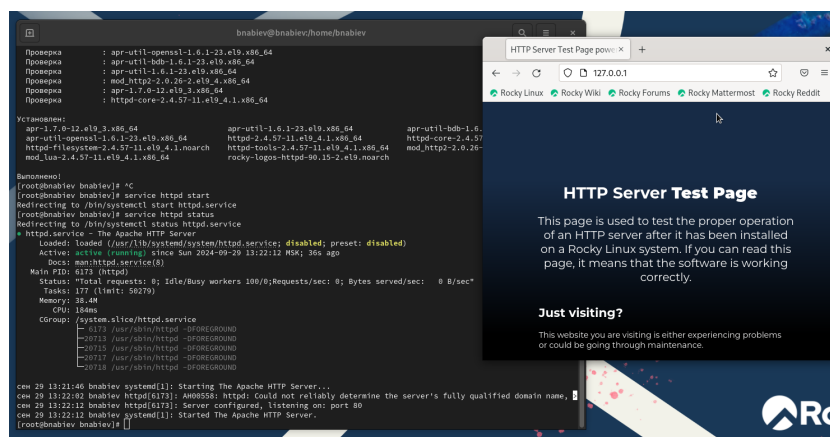


Figure 2.1: запуск http

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`

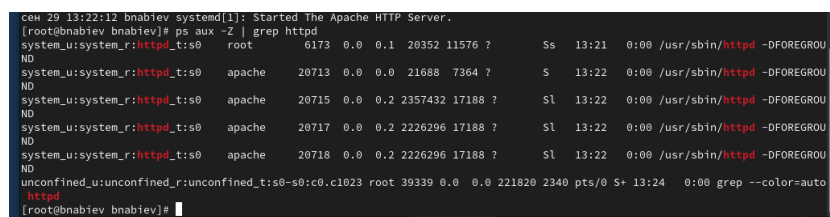


Figure 2.2: контекст безопасности http

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

```
bnabiev@bnabiev:/hom
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
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_openscryptoki       off
httpd_use_openstack           off
httpd_use_sasl                 off
httpd_verify_dns              off
[root@bnabiev bnabiev]#
```

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.

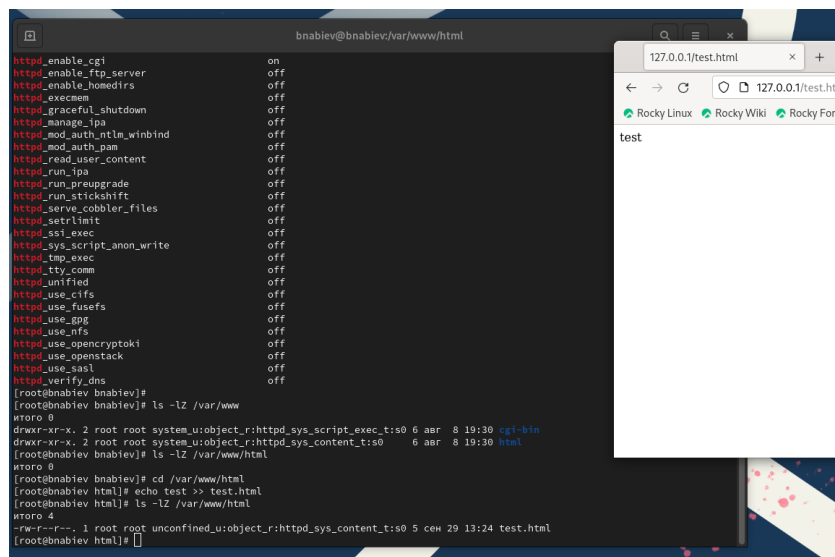


Figure 2.4: создание `html`-файла и доступ по `http`

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 -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.

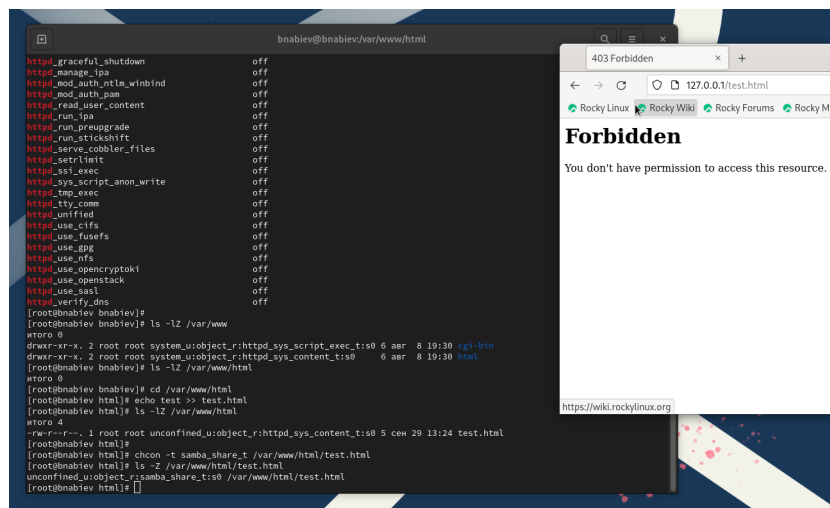


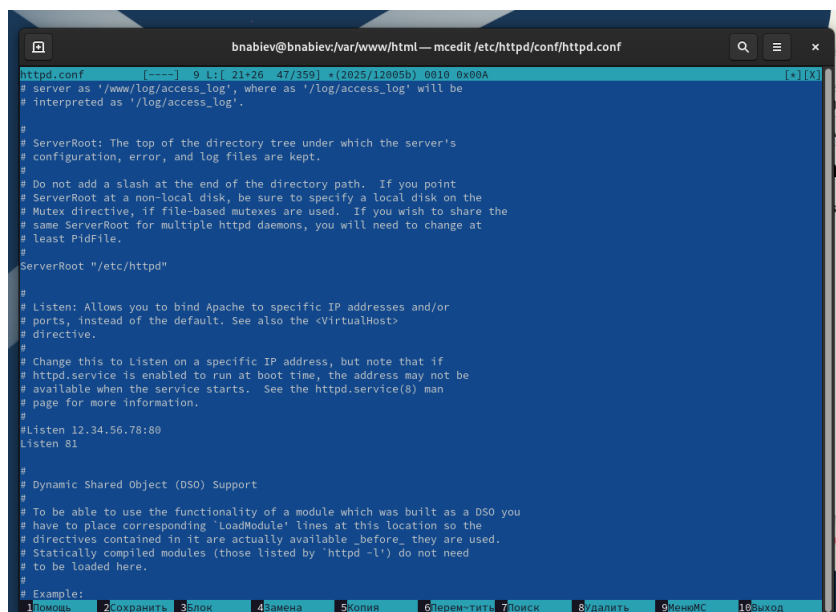
Figure 2.5: ошибка доступа после изменения контекста

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.

```
[root@bnabiev html]#
[root@bnabiev html]# tail /var/log/messages
Sep 29 13:24:42 bnabiev gnome-shell[1759]: Window manager warning: W3 appears to be one of the offending windows with a times
tamp of 421041. Working around...
Sep 29 13:25:06 bnabiev systemd[1]: Starting SETroubleshoot daemon for processing new SELinux denial logs...
Sep 29 13:25:07 bnabiev systemd[1]: Started SETroubleshoot daemon for processing new SELinux denial logs.
Sep 29 13:25:07 bnabiev setroubleshoot[39628]: failed to retrieve rpm info for path '/var/www/html/test.html':
Sep 29 13:25:07 bnabiev systemd[1]: Started dbus-1.1-org.fedoraproject.SetroubleshootPrivileged0.service.
Sep 29 13:25:08 bnabiev setroubleshoot[39628]: SELinux запрещает /usr/sbin/httpd доступ getattr к файлу /var/www/html/test.htm
l. Для выполнения всех сообщений SELinux: sealert -l eb267be5-e845-4749-bc8d-ba31b87d3eef
Sep 29 13:25:08 bnabiev setroubleshoot[39628]: SELinux запрещает /usr/sbin/httpd доступ getattr к файлу /var/www/html/test.htm
l.#012#012***** Модуль restorecon предлагает (точность 92.2) *****#012#012Если вы хотите исправить метк
y.$TARGET3знак _PATH по умолчанию должен быть httpd_sys_content_t#012То вы можете запустить restorecon. Возможно, попытка дост
упа была остановлена из-за недостаточных разрешений для доступа к родительскому каталогу, и в этом случае попытайтесь соответ
ствующим образом изменить следующую команду.#012Сделать#012# /sbin/restorecon -v /var/www/html/test.html#012#012***** Модуль
public_content предлагает (точность 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.#012Сделать#012# semanage fcontext
-a -t public_content_t '/var/www/html/test.html'#012# restorecon -v '/var/www/html/test.html'#012#012***** Модуль catchall п
редлагает (точность 1.41) *****#012#012Если вы считаете, что httpd должно быть разрешено getattr досту
п к test.html file по умолчанию.#012То рекомендуется создать отчет об ошибке.#012Чтобы разрешить доступ, можно создать локаль
ный модуль политики.#012Сделать#012#разрешить этот доступ сейчас, выполнив:#012# ausearch -c 'httpd' --raw | audit2allow -M my
-httpd#012# semodule -X 380 -i my-httpd.pp#012
[root@bnabiev 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.



```
bnabiev@bnabiev:/var/www/html — mcedit /etc/httpd/conf/httpd.conf
httpd.conf  [----]  9 L: [ 21+26  47/359] + (2025/12005b) 0010 0x00A  [.*] [X]
# server as '/www/log/access_log', where as '/log/access_log' will be
# interpreted as '/log/access_log'.
#
# ServerRoot: The top of the directory tree under which the server's
# configuration, error, and log files are kept.
#
# Do not add a slash at the end of the directory path.  If you point
# ServerRoot at a non-local disk, be sure to specify a local disk on the
# Muxon directive, if file-based mutexes are used.  If you wish to share the
# 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 so_module modules/mod_so.so
```

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».

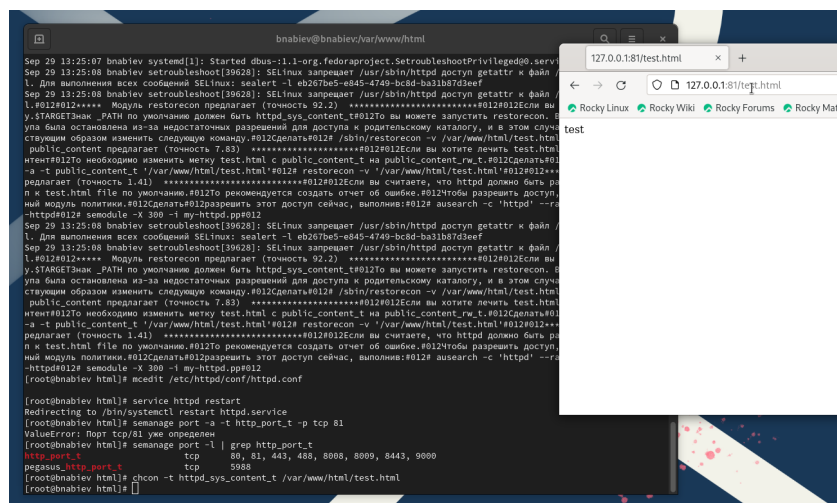


Figure 2.8: доступ по http на 81 порт

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