# Цель работы

Целью данной лабораторной работы является развить навыки администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux1. Проверить работу SELinx на практике совместно с веб-сервером Apache.

# Подготовка лабораторного стенда и методические рекомендации

1. Установили веб-сервер Apache.
2. В конфигурационном файле /etc/httpd/httpd.conf задали параметр ServerName.
3. Отключаем пакетный фильтр.

# Выполнение лабораторной работы

1. Входим в систему с полученными учётными данными. Проверили, что SELinux работает в режиме enforcing политики targeted с помощью команд **getenforce** и **sestatus**. (@fig:004)

```
[yupopova@yupopova ~]$ getenforce
Enforcing
[yupopova@yupopova ~]$ 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
```

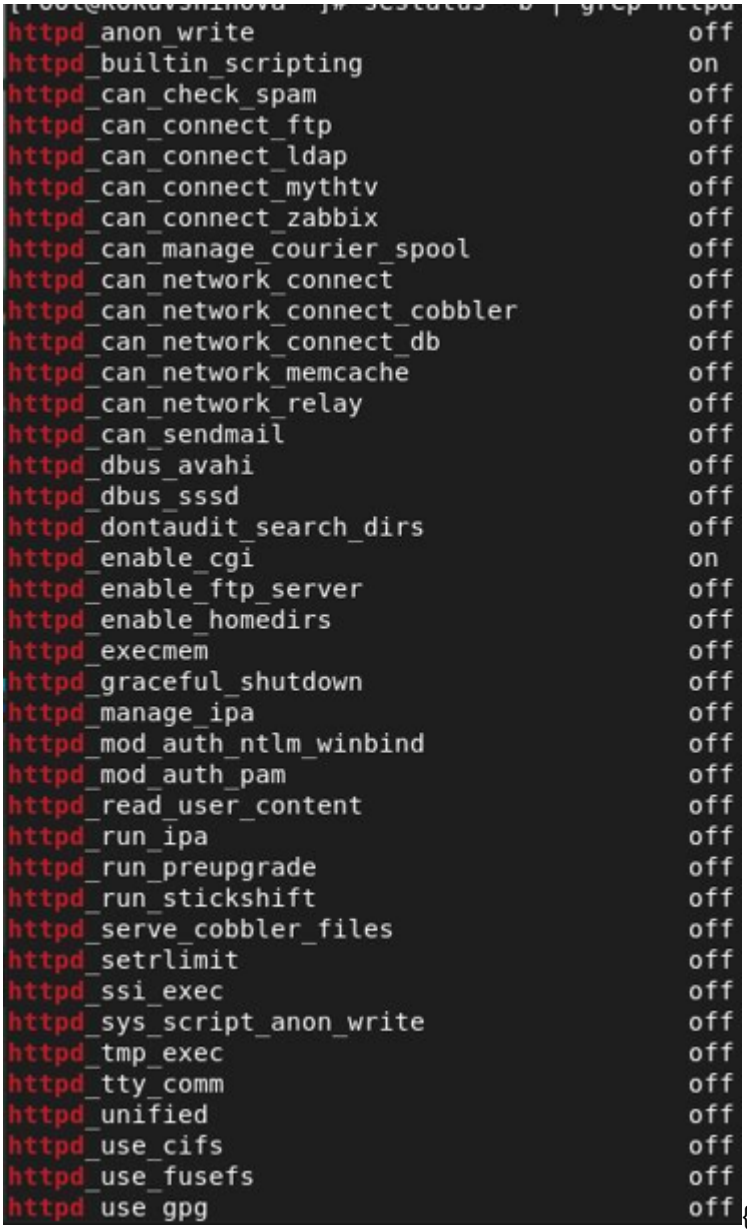
{#fig:004 width=100%}

2. Запустили веб-сервер и обратились к нему с помощью команды (@fig:005): service httpd status
3. Нашли веб-сервер Apache в списке процессов. Контекст безопасности - unconfined_u:unconfined_r:unconfined_t. (@fig:006)

```
system_u:system_r:httpd_t:s0    root       39619  0.0  0.5  20248 11704 ?        Ss   17:09
 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0    apache     39620  0.0  0.3  21572  7444 ?        S    17:09
 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0    apache     39624  0.0  0.5 1079376 11100 ?       Sl   17:09
 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0    apache     39625  0.0  0.6 1210512 13148 ?       Sl   17:09
 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0    apache     39626  0.0  0.5 1079376 11100 ?       Sl   17:09
 0:00 /usr/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 40041 0.0  0.1 221692 2292 pts/0 S
+ 17:30   0:00 grep --color=auto httpd
```

{#fig:006 width=100%}

4. Посмотрели текущее состояние переключателей SELinux для Apache с помощью команды **sestatus -b | grep httpd**. (@fig:007)



{#fig:007 width=100%}

5. Посмотрели статистику по политике с помощью команды **seinfo**. Определили, что множество пользователей = 8; ролей = 14; типов = 5002. (@fig:008)

```
Statistics for policy file: /sys/fs/selinux/policy
Policy Version:             33 (MLS enabled)
Target Policy:              selinux
Handle unknown classes:     allow
  Classes:             133    Permissions:         454
  Sensitivities:         1    Categories:         1024
  Types:              5002    Attributes:          254
  Users:                 8    Roles:                14
  Booleans:            347    Cond. Expr.:         381
  Allow:             63996    Neverallow:            0
  Auditallow:          168    Dontaudit:          8417
  Type_trans:       258486    Type_change:          87
  Type_member:          35    Range_trans:        5960
  Role allow:           38    Role_trans:          420
  Constraints:          72    Validatetrans:         0
  MLS Constrain:        72    MLS Val. Tran:         0
  Permissives:           0    Polcap:                5
  Defaults:              7    Typebounds:            0
  Allowxperm:            0    Neverallowxperm:       0
  Auditallowxperm:       0    Dontauditxperm:        0
  Ibendportcon:          0    Ibpkeycon:             0
  Initial SIDs:         27    Fs_use:               33
  Genfscon:            106    Portcon:             651
  Netifcon:              0    Nodecon:               0
```

{#fig:008 width=100%}

6. Определили тип файлов и поддиректорий, находящихся в директории /var/www, с помощью команды **ls -lZ /var/www**. (@fig:009)

7. Необходимо было определить тип файлов, находящихся в директории /var/www/html, с помощью команды **ls -lZ /var/www/html**. Но в данной директории файлов не обнаружилось. (@fig:010)

8. Определим круг пользователей, которым разрешено создание файлов в директории /var/www/html - только uesr. (@fig:011)

```
итого 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0      6
```

{#fig:011 width=100%}

9. Создали от имени суперпользователя html-файл /var/www/html/test.html следующего содержания: (@fig:012)

```
1 <html>
2 <body> test </body>
3 </html>
```

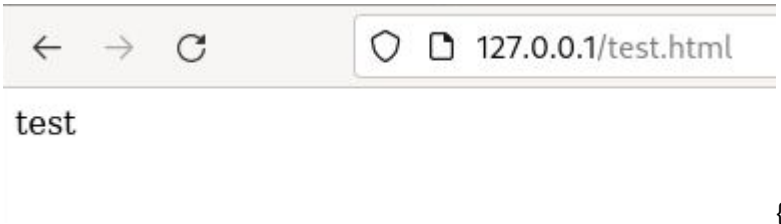
{#fig:012 width=100%}

10. Проверили контекст созданного файла - httpd_sys_content_t. (@fig:013)

```
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0
/html/test.html
```

{#fig:013 width=100%}

11. Обратитились к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html и убедились, что файл был успешно отображён. (@fig:014)



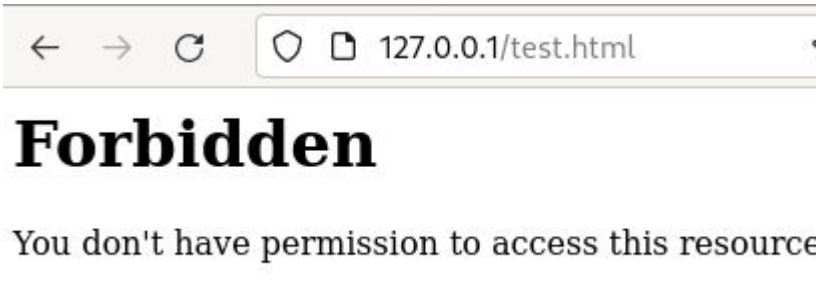

{#fig:014 width=100%}

12. Изучили справку man httpd_selinux. Тип файла test.html - контекст созданного файла - httpd_sys_content_t. (@fig:015)


{#fig:015 width=100%}

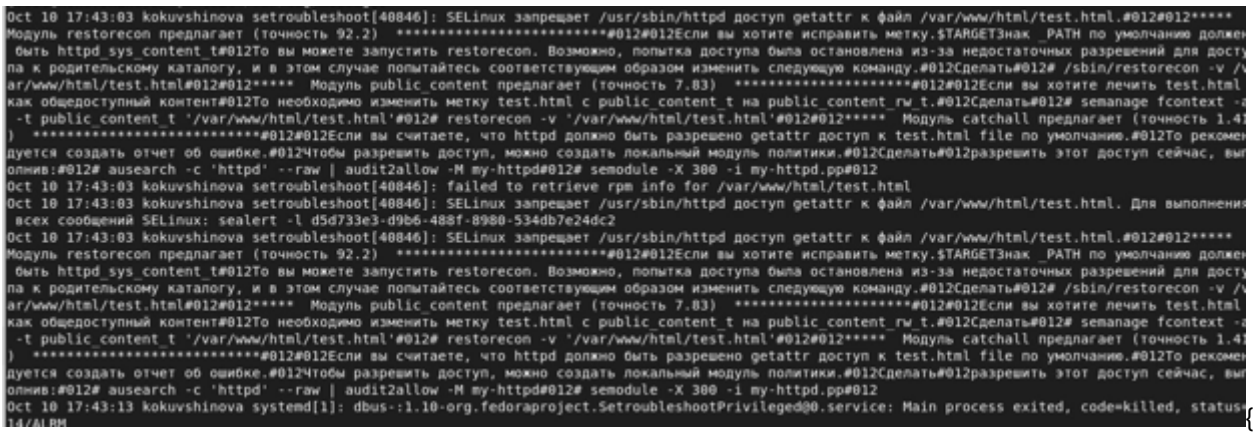
13. Изменили контекст файла /var/www/html/test.html с httpd_sys_content_t на samba_share_t: chcon -t samba_share_t /var/www/html/test.html ls -Z /var/www/html/test.html И проверили, что контекст поменялся. (@fig:016)

14. Пробуем ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html. В результате получили ошибку. (@fig:017)




{#fig:017 width=100%}

15. Проанализируем ситуацию. Почему файл не был отображён, если права доступа позволяют читать этот файл любому пользователю? ls -l /var/www/html/test.html Просмотрим log-файлы веб-сервера Apache и системный лог-файл: tail /var/log/messages В системе оказались запущенны процессы **setroubleshootd** и **audtd**. (@fig:018)




{#fig:018 width=100%}

16. Попробуем запустить веб-сервер Apache на прослушивание TCP-порта 81. Для этого в файле /etc/httpd/httpd.conf находим строчку Listen 80 и заменяем её на Listen 81. (@fig:019)

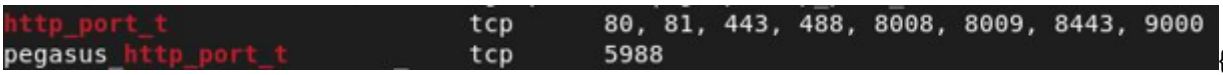
```
#Listen 12.34.56.78:80
Listen 81
```

{#fig:019 width=100%}

17. Выполним перезапуск веб-сервера Apache. Сбой не произошел.

18. Проанализируем лог-файлы: tail -nl /var/log/messages Просмотрим файлы /var/log/http/error_log, /var/log/http/access_log и /var/log/audit/audit.log. (@fig:020)

19. Выполним команду **semanage port -a -t http_port_t -p tcp 81**. Вылетает ValueError в связи с тем, что порт уже определен. После этого проверим список портов командой **semanage port -l | grep http_port_t** и убедились, что порт 81 появился в списке. (@fig:021)

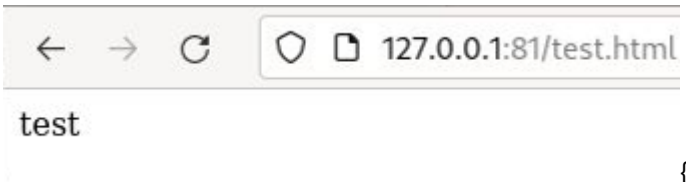


{#fig:021 width=100%}

20. Попробуем запустить веб-сервер Apache ещё раз. (@fig:022)

21. Вернули контекст httpd_sys_content__t к файлу /var/www/html/test.html: **chcon -t httpd_sys_content_t /var/www/html/test.html** (@fig:023)

После этого пробуем получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1:81/test.html. В результате увидели содержимое файла — слово «test». (@fig:024)



{#fig:024 width=100%}

22. Исправим обратно конфигурационный файл apache, вернув Listen 80. (@fig:025)

{#fig:025 width=100%}

23. Удалим привязку http_port_t к 81 порту: **semanage port -d -t http_port_t -p tcp 81** и проверим, что порт 81 удалён. Данная команда не была выполнена. (@fig:026)

{#fig:026 width=100%}

24. Удалим файл /var/www/html/test.html: **rm /var/www/html/test.html**. (@fig:027)

{#fig:027 width=100%}

# Вывод

В ходе выполнения лабораторной работы мы развили навыки администрирования ОС Linux. Получили первое практическое знакомство с технологией SELinux1. Проверили работу SELinx на практике совместно с веб-сервером Apache.

# Библиография

1. Кулябов Д. С., Королькова А. В., Геворкян М. Н. Мандатное разграничение прав в Linux / Кулябов Д. С., Королькова А. В., Геворкян М. Н. - Москва: - 5 с.: Мандатное разграничение прав в Linux.