

Отчет по лабораторной работе № 6. Мандатное разграничение прав в Linux

дисциплина: Информационная безопасность

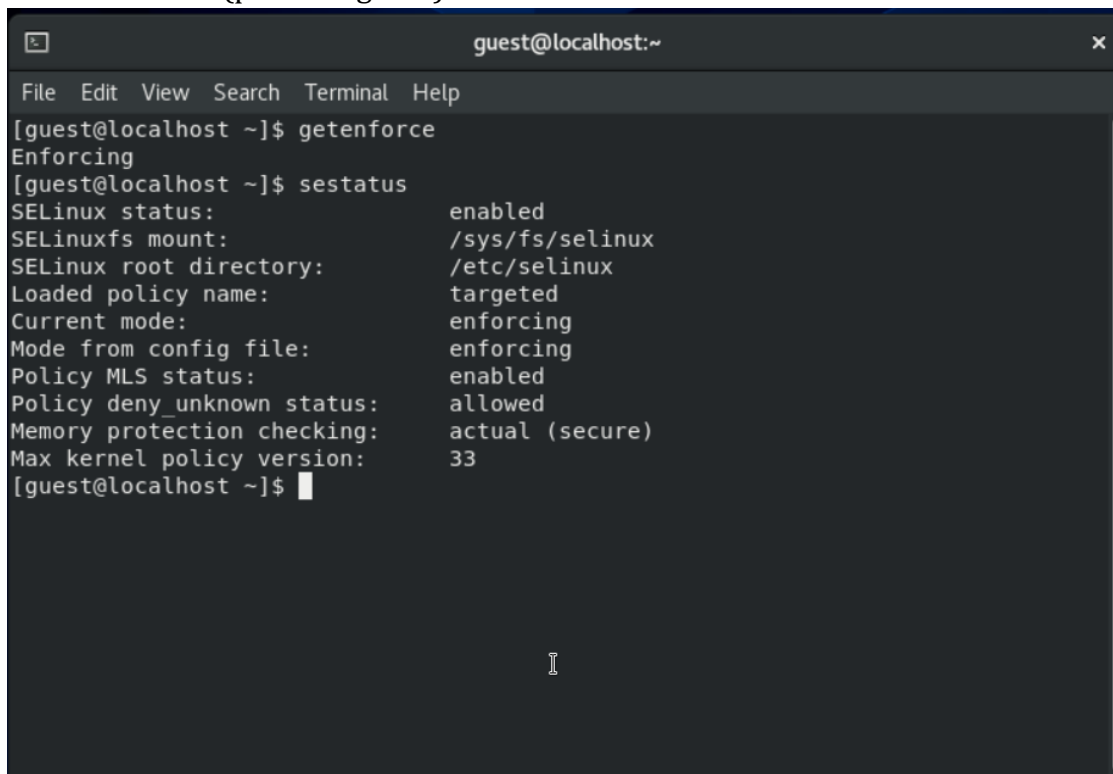
Смирнова Мария Александровна

Цель работы

Развитие навыков администрирования ОС Linux. Практическое освоение технологии SELinux. Проверка работы SELinux совместно с веб-сервером Apache.

Выполнение лабораторной работы

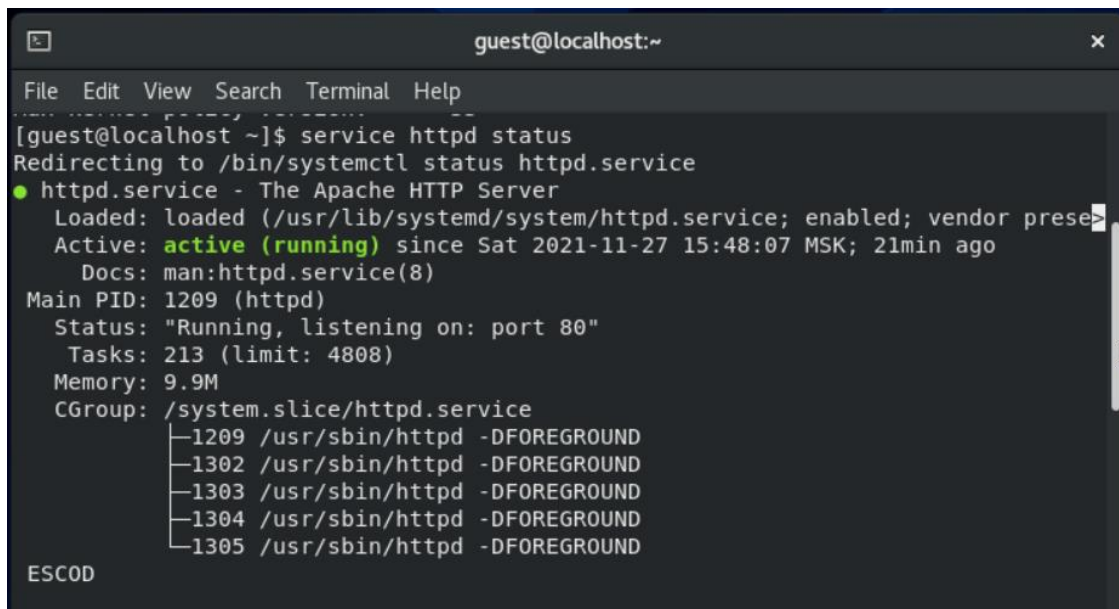
1. От имени пользователя guest войдем в систему. Проверим, что SELinux работает в режиме enforcing политики targeted с помощью команд `getenforce` и `sestatus` (рис. -@fig:001).

A screenshot of a terminal window titled 'guest@localhost:~'. The terminal shows the execution of two commands: 'getenforce' and 'sestatus'. The output of 'getenforce' is 'Enforcing'. The output of 'sestatus' is a multi-line status report. The terminal has a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'.

```
guest@localhost:~  
File Edit View Search Terminal Help  
[guest@localhost ~]$ getenforce  
Enforcing  
[guest@localhost ~]$ 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  
[guest@localhost ~]$
```

Проверка SELinux

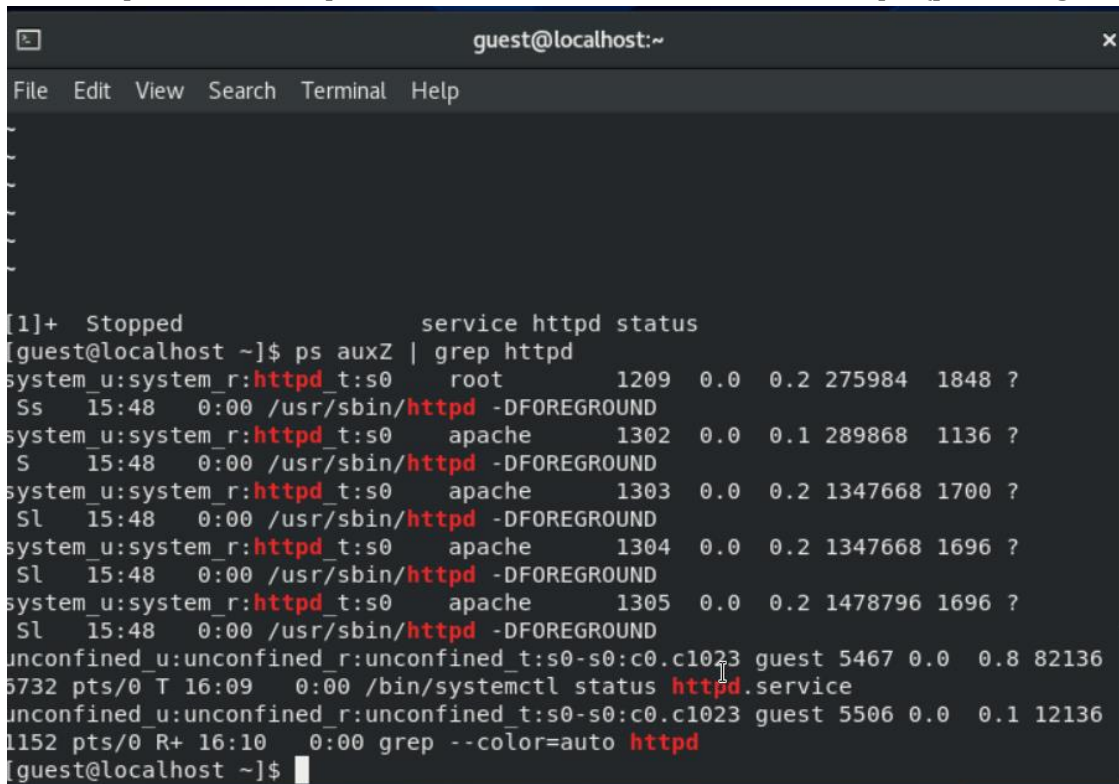
2. Обратимся к веб-серверу, запущенному на нашем устройстве и проверим, что он работает (рис. -@fig:002).



```
guest@localhost:~  
File Edit View Search Terminal Help  
[guest@localhost ~]$ service httpd status  
Redirecting to /bin/systemctl status httpd.service  
● httpd.service - The Apache HTTP Server  
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor prese  
   Active: active (running) since Sat 2021-11-27 15:48:07 MSK; 21min ago  
     Docs: man:httpd.service(8)  
  Main PID: 1209 (httpd)  
    Status: "Running, listening on: port 80"  
   Tasks: 213 (limit: 4808)  
  Memory: 9.9M  
   CGroup: /system.slice/httpd.service  
           └─1209 /usr/sbin/httpd -DFOREGROUND  
             └─1302 /usr/sbin/httpd -DFOREGROUND  
               └─1303 /usr/sbin/httpd -DFOREGROUND  
                 └─1304 /usr/sbin/httpd -DFOREGROUND  
                   └─1305 /usr/sbin/httpd -DFOREGROUND  
ESCOD
```

Httpd status

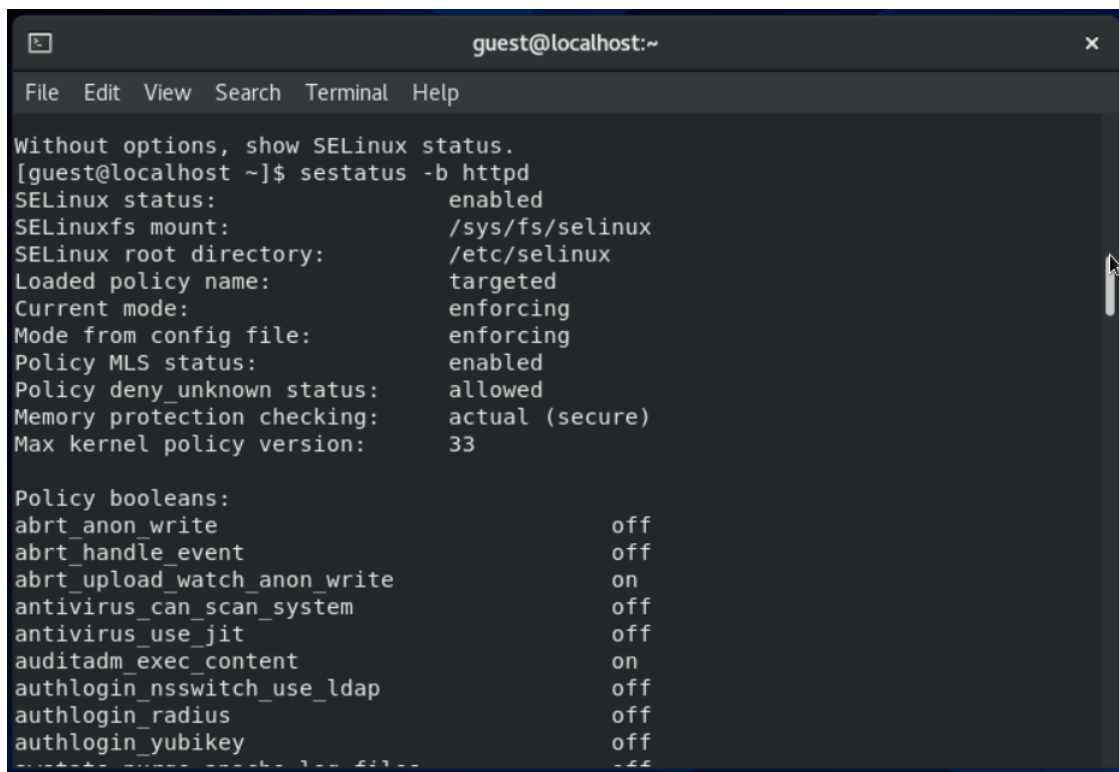
3. Посмотрим командой `ps auxZ | grep httpd` список процессов и найдем в нем Apache. Посмотрим на его контекст безопасности: httpd (рис. -@fig:003).



```
guest@localhost:~  
File Edit View Search Terminal Help  
[1]+  Stopped                  service httpd status  
[guest@localhost ~]$ ps auxZ | grep httpd  
system_u:system_r:httpd_t:s0    root      1209  0.0  0.2 275984  1848 ?  
Ss  15:48   0:00 /usr/sbin/httpd -DFOREGROUND  
system_u:system_r:httpd_t:s0    apache    1302  0.0  0.1 289868  1136 ?  
S   15:48   0:00 /usr/sbin/httpd -DFOREGROUND  
system_u:system_r:httpd_t:s0    apache    1303  0.0  0.2 1347668 1700 ?  
Sl  15:48   0:00 /usr/sbin/httpd -DFOREGROUND  
system_u:system_r:httpd_t:s0    apache    1304  0.0  0.2 1347668 1696 ?  
Sl  15:48   0:00 /usr/sbin/httpd -DFOREGROUND  
system_u:system_r:httpd_t:s0    apache    1305  0.0  0.2 1478796 1696 ?  
Sl  15:48   0:00 /usr/sbin/httpd -DFOREGROUND  
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 guest 5467 0.0  0.8 82136  
5732 pts/0 T 16:09   0:00 /bin/systemctl status httpd.service  
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 guest 5506 0.0  0.1 12136  
1152 pts/0 R+ 16:10   0:00 grep --color=auto httpd  
[guest@localhost ~]$
```

Контекст безопасности Apache

4. Посмотрим на текущее состояние переключателей, большинство из них находятся в положении off (рис. -@fig:004).

A terminal window titled 'guest@localhost:~' with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal output shows the command 'sestatus -b httpd' and its results. The first section shows general SELinux status, and the second section shows policy booleans.

```
Without options, show SELinux status.  
[guest@localhost ~]$ sestatus -b httpd  
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  
  
Policy booleans:  
abrt_anon_write                off  
abrt_handle_event              off  
abrt_upload_watch_anon_write   on  
antivirus_can_scan_system      off  
antivirus_use_jit              off  
auditadm_exec_content          on  
authlogin_nsswitch_use_ldap    off  
authlogin_radius               off  
authlogin_yubikey              off  
autotune_cpu_ticks             off
```

Переключатели

5. Посмотрим статистику по политике с помощью команды seinfo. У нас есть 8 пользователей, 38 ролей и 35 типов (рис. -@fig:005).

```
guest@localhost:~  
File Edit View Search Terminal Help  
Policy Version: 31 (MLS enabled)  
Target Policy: selinux  
Handle unknown classes: allow  
Classes: 132 Permissions: 464  
Sensitivities: 1 Categories: 1024  
Types: 4961 Attributes: 255  
Users: 8 Roles: 14  
Booleans: 338 Cond. Expr.: 386  
Allow: 112594 Neverallow: 0  
Auditallow: 166 Dontaudit: 10358  
Type_trans: 252747 Type_change: 87  
Type_member: 35 Range_trans: 5781  
Role allow: 38 Role_trans: 421  
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: 34  
Genfscon: 107 Portcon: 642  
Netifcon: 0 Nodecon: 0  
[guest@localhost ~]$
```

Seinfo

6. Определим тип файлов в директории `/var/www` и `/var/www/html`. Круг пользователей, которым разрешено создание файлов в последней директории - root (рис. -@fig:006)

```
guest@localhost:~  
File Edit View Search Terminal Help  
Allow: 112594 Neverallow: 0  
Auditallow: 166 Dontaudit: 10358  
Type_trans: 252747 Type_change: 87  
Type_member: 35 Range_trans: 5781  
Role_allow: 38 Role_trans: 421  
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 Ibkeycon: 0  
Initial SIDs: 27 Fs_use: 34  
Genfscon: 107 Portcon: 642  
Netifcon: 0 Nodecon: 0  
[guest@localhost ~]$ ls -lZ /var/www  
total 0  
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 Nov 12 07  
:58 cgi-bin  
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0 6 Nov 12 07  
:58 html  
[guest@localhost ~]$ ls -lZ /var/www/html  
total 0  
[guest@localhost ~]$
```

Директория /var/www/html

7. Создадим от имени суперпользователя html-файл с содержанием из задания (рис. -@fig:007)

```
guest@localhost:/home/guest  
File Edit View Search Terminal Help  
GNU nano 2.9.8 /var/www/html/test.html Modified  
<html>  
<body>test</body>  
</html>
```

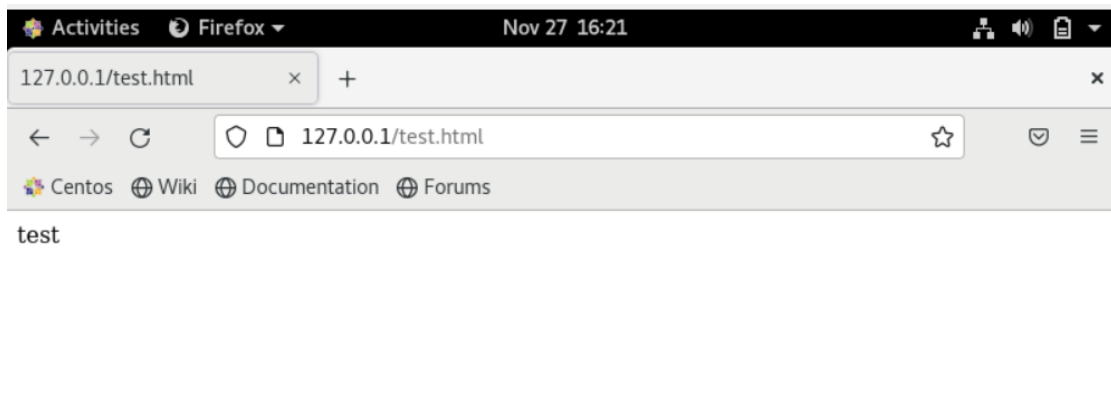
test.html

8. Проверим контекст созданного файла - httpd_sys_content_t. Это контекст по умолчанию (рис. -@fig:008)

```
[root@localhost guest]# ls -lZ /var/www/html  
total 4  
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 33 Nov 27 1  
6:17 test.html  
[root@localhost guest]#
```

Контекст test.html

9. Обратимся к созданному файлу через веб-сервер, введя в браузере `http://127.0.0.1/test.html`. Увидим, что файл успешно отображен (рис. -@fig:009).



http://127.0.0.1/test.html

10. Выясним, какие контексты файлов определены для `httpd` и сопоставим их с контекстом нашего файла `test` (рис. -@fig:010)

```
[guest@localhost ~]$ ls -Z /var/www/html/test.html
unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.html
[guest@localhost ~]$
```

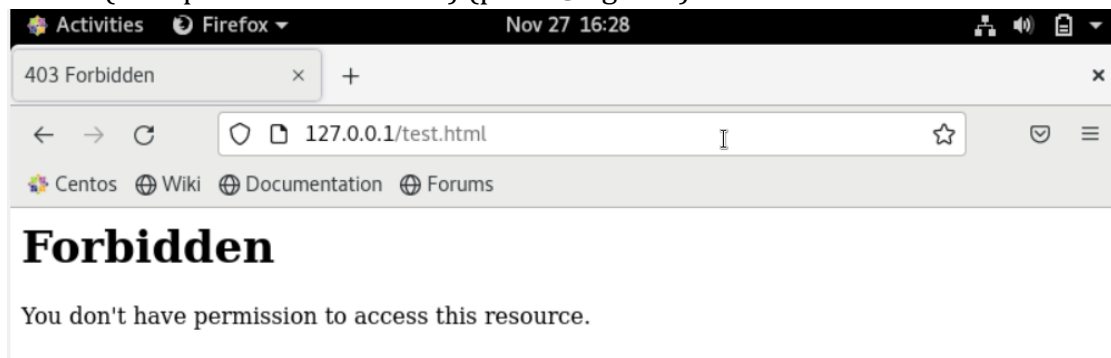
Контексты файлов httpd

11. Изменим контекст файла с `httpd_sys_content_t` на `samba_share_t` (рис. -@fig:011)

```
[guest@localhost ~]$ su
Password:
[root@localhost guest]# chcon -t samba_share_t /var/www/html/test.html
[root@localhost guest]# ls -Z /var/www/html/test.html
unconfined_u:object_r:samba_share_t:s0 /var/www/html/test.html
[root@localhost guest]#
```

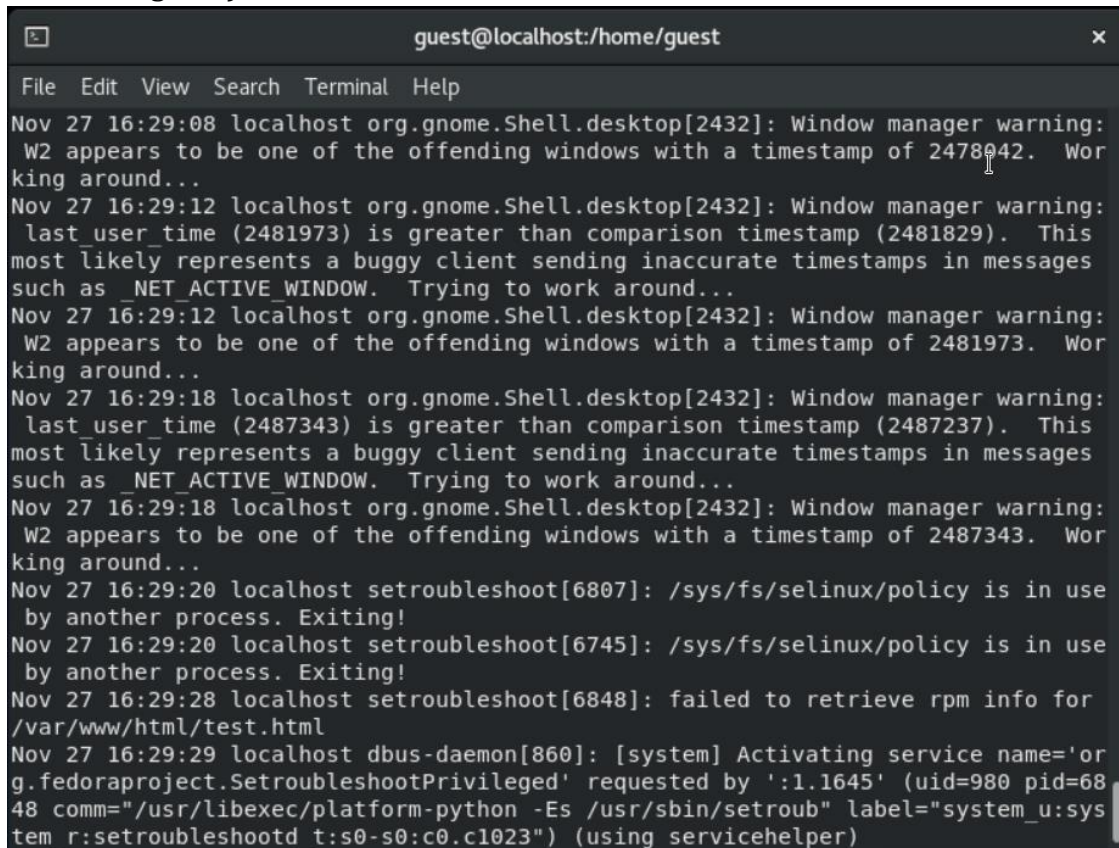
Изменение контекста test

12. Попробуем еще раз получить доступ к файлу через браузер. Получим сообщение об ошибке. Файл не был отображен из-за неправильного контекста (который мы поменяли) (рис. -@fig:012)



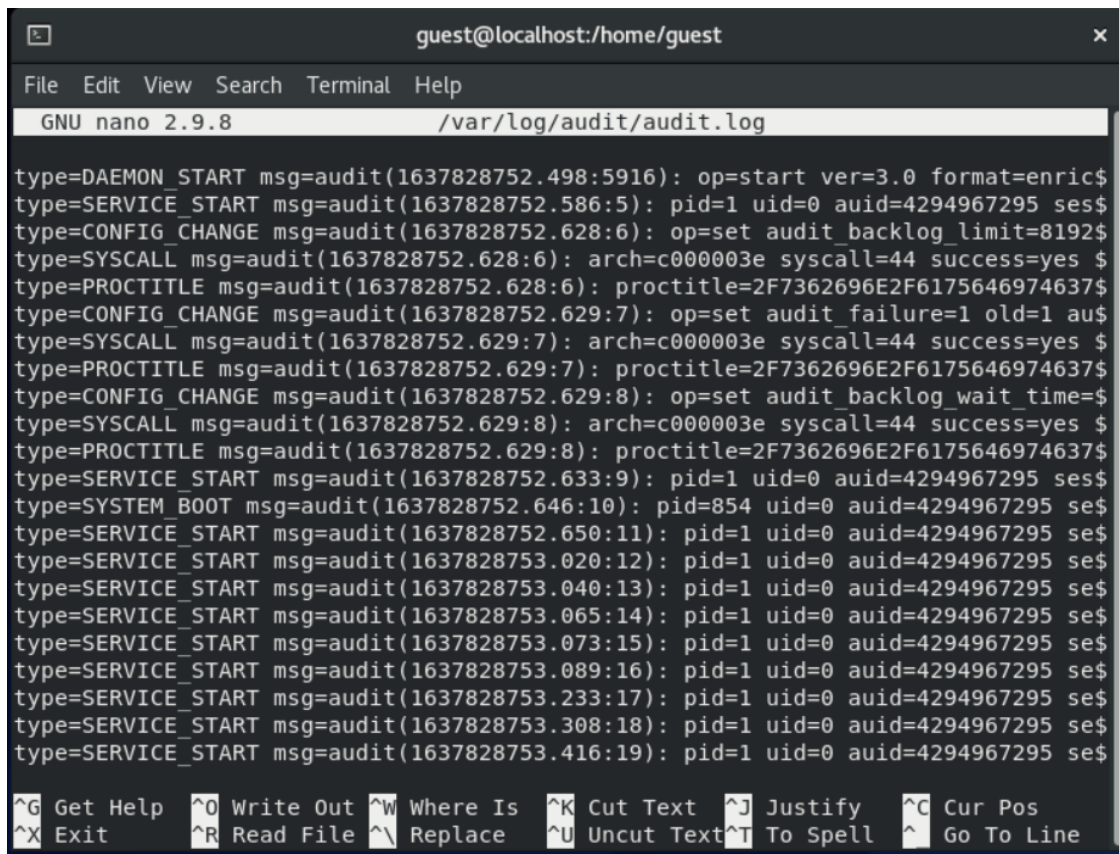
Сообщение об ошибке

13. Посмотрим log-файлы веб-сервера Apache и системный log-файл (рис. - @fig:013)

A screenshot of a terminal window titled 'guest@localhost:/home/guest'. The terminal displays a series of system log messages. The first three messages are warnings from the window manager (org.gnome.Shell.desktop[2432]) about W2 appearing as an offending window with timestamps 2478042, 2481829, and 2481973. The next two messages are from setroubleshootd (6807 and 6745) stating that /sys/fs/selinux/policy is in use by another process and exiting. The following message is from setroubleshootd (6848) reporting a failure to retrieve rpm info for /var/www/html/test.html. The final message is from dbus-daemon (860) indicating the activation of the service 'org.fedoraproject.SetroubleshootPrivileged' requested by user 1.1645 (uid=980, pid=6848) using the command '/usr/libexec/platform-python -Es /usr/sbin/setroub' with label 'system_u:system_r:setroubleshootd_t:s0-s0:c0.c1023'.

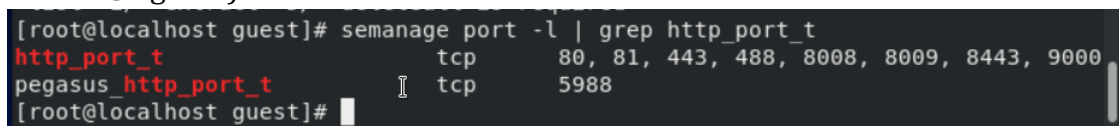
log-файл

14. Увидим ошибки, аналогичные указанным в файле audit.log (рис. -@fig:014)



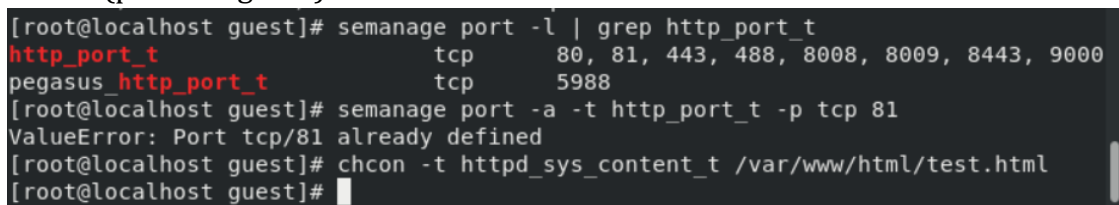
audit.log

15. В моем случае tcp port 81 уже есть по умолчанию в списке портов (рис. - @fig:015)



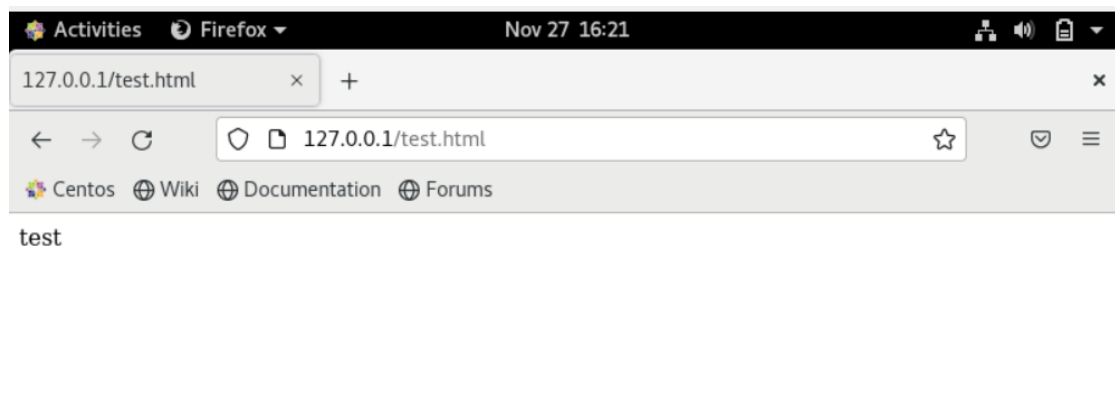
Порт 81

16. Мы также не можем его удалить. Заменим контекст обратно на корректный (рис. -@fig:016)



Контекст файла test

17. Наш файл через браузер снова открывается корректно. Удалим файл test.html (рис. -@fig:016)



Контекст файла test

Выводы

В процессе выполнения лабораторной работы мы развили навыков администрирования ОС Linux, а также освоили технологии SELinux. Мы проверили работу SELinux совместно с веб-сервером Apache.