

Информационная безопасность. Отчет по лабораторной работе №6

Мандатное разграничение прав в Linux

Терентьев Егор Дмитриевич 1032192875

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	13
4	Список литературы	14

List of Figures

2.1	getenforce	6
2.2	service_httpd_status	7
2.3	ps_auxz_httpd	7
2.4	sestatus_httpd	8
2.5	seinfo	9
2.6	ls_lz	9
2.7	html_file	10
2.8	test_file_on_site	10
2.9	samba_share_t	10
2.10	error	11
2.11	logs	11
2.12	restart_log	11
2.13	semanage_port	12
2.14	httpd_sys_content	12
2.15	semanage_rm	12

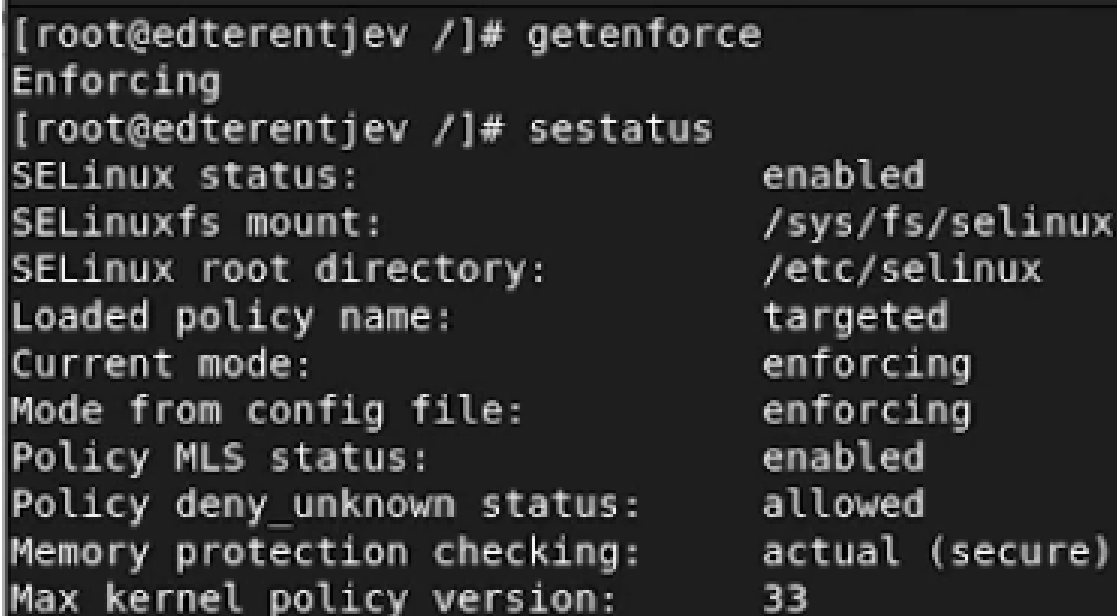
List of Tables

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

Войдите в систему с полученными учётными данными и убедитесь, что SELinux работает в режиме enforcing политики targeted с помощью команд `getenforce` и `sestatus`. fig. 2.1.

A terminal window with a dark background and light-colored text. The prompt is [root@edterentjev /]#. The first command is getenforce, which returns Enforcing. The second command is sestatus, which returns a detailed status report for SELinux.

```
[root@edterentjev /]# getenforce
Enforcing
[root@edterentjev /]# 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
```

Figure 2.1: `getenforce`

Обратитесь с помощью браузера к веб-серверу, запущенному на вашем компьютере, и убедитесь, что последний работает fig. 2.2.

```

[root@edterentjev /]# service httpd status
Redirecting to /bin/systemctl status httpd.service
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor pr>
   Active: active (running) since Wed 2022-10-12 15:34:11 MSK; 10min ago
     Docs: man:httpd.service(8)
   Main PID: 40253 (httpd)
    Status: "Total requests: 4; Idle/Busy workers 100/0;Requests/sec: 0.00646;>
      Tasks: 213 (limit: 24684)
     Memory: 27.1M
        CPU: 316ms
    CGroup: /system.slice/httpd.service
            └─40253 /usr/sbin/httpd -DFOREGROUND
              └─40254 /usr/sbin/httpd -DFOREGROUND
                └─40259 /usr/sbin/httpd -DFOREGROUND
                  └─40260 /usr/sbin/httpd -DFOREGROUND
                    └─40261 /usr/sbin/httpd -DFOREGROUND

Oct 12 15:34:11 edterentjev.localdomain systemd[1]: Starting The Apache HTTP Se>
Oct 12 15:34:11 edterentjev.localdomain systemd[1]: Started The Apache HTTP Ser>
Oct 12 15:34:11 edterentjev.localdomain httpd[40253]: Server configured, listen>
[root@edterentjev /]#

```

Figure 2.2: service_httpd_status

Найдите веб-сервер Apache в списке процессов, определите его контекст безопасности и занесите эту информацию в отчёт: system_u:system_r:httpd_t:s0 fig. 2.3.

```

[root@edterentjev /]# ps -eZ | grep httpd
system_u:system_r:httpd_t:s0      40253 ?        00:00:00 httpd
system_u:system_r:httpd_t:s0     40254 ?        00:00:00 httpd
system_u:system_r:httpd_t:s0     40259 ?        00:00:00 httpd
system_u:system_r:httpd_t:s0     40260 ?        00:00:00 httpd
system_u:system_r:httpd_t:s0     40261 ?        00:00:00 httpd

```

Figure 2.3: ps_auxz_httpd

Посмотрите текущее состояние переключателей SELinux для Apache fig. 2.4

```
Without options, show SELinux status.  
[root@edterentjev ~]# sestatus -b | grep httpd  
httpd_anon_write off  
httpd_builtin_scripting on  
httpd_can_check_spam off  
httpd_can_connect_ftp off  
httpd_can_connect_ldap off  
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
```

Figure 2.4: sestatus_httpd

Посмотрите статистику по политике с помощью команды `seinfo`, также определите множество пользователей, ролей, типов. fig. 2.5

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 Constrains:	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

Figure 2.5: seinfo

Определите тип файлов и поддиректорий, находящихся в директории /var/www, с помощью команды `ls -lZ /var/www` Определите тип файлов, находящихся в директории /var/www/html: `ls -lZ /var/www/html` fig. 2.6.

```
[root@edterentjev /]# ls -lZ /var/www
total 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 May 16 15
:10 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0 6 May 16 15
:10 html
[root@edterentjev /]# ls -lZ /var/www/html
total 0
```

Figure 2.6: ls_lz

Создайте от имени суперпользователя (так как в дистрибутиве после установки только ему разрешена запись в директорию) html-файл fig. 2.7.

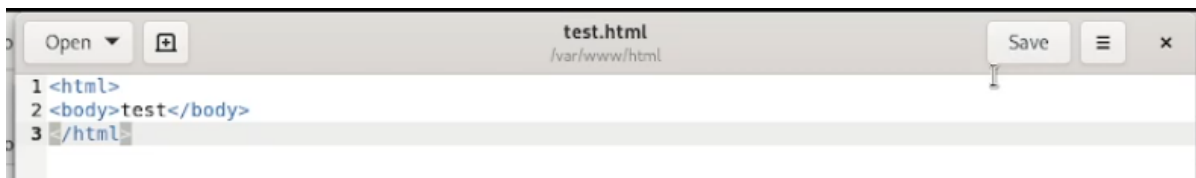


Figure 2.7: html_file

Обратитесь к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1/test.html`. Убедитесь, что файл был успешно отображён. fig. 2.8

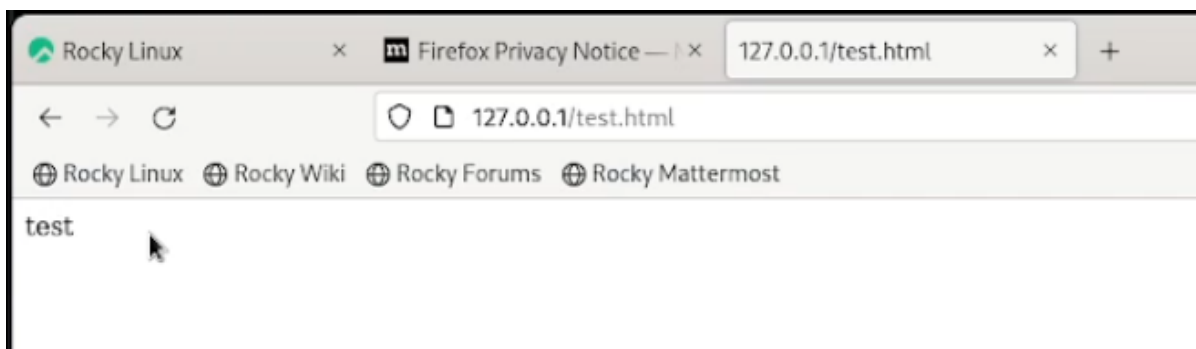


Figure 2.8: test_file_on_site

Проверить контекст файла. 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` fig. 2.9.

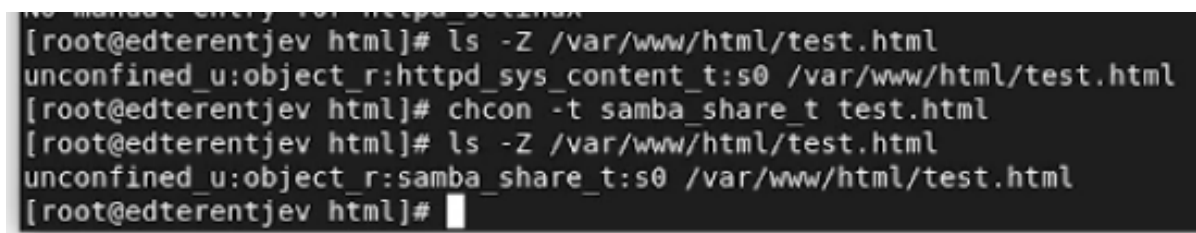


Figure 2.9: samba_share_t

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 fig. 2.10.

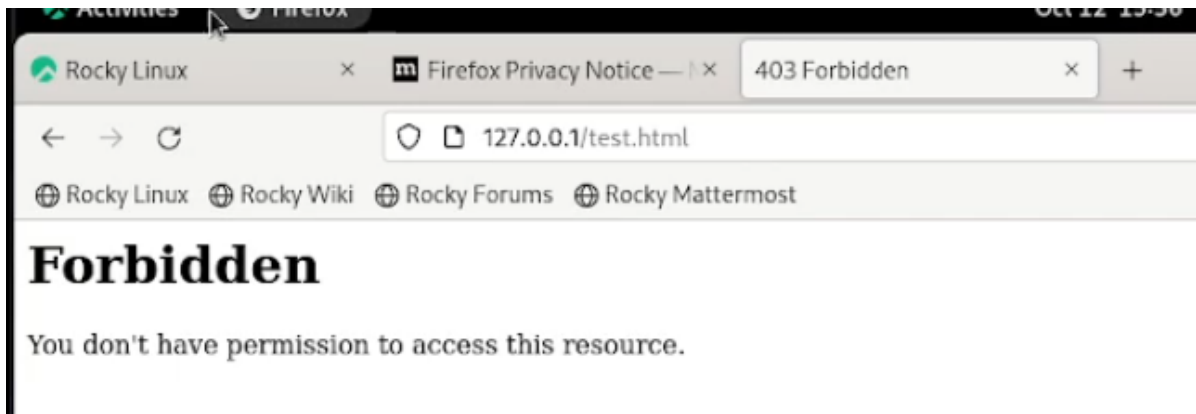


Figure 2.10: error

ls -l /var/www/html/test.html Просмотрите log-файлы веб-сервера Apache. Также просмотрите системный лог-файл: tail /var/log/messages fig. 2.11.

```
[root@edterentjev html]# ls -l /var/www/html/test.html
-rw-r--r--. 1 root root 33 Oct 12 15:52 /var/www/html/test.html
[root@edterentjev html]# tail /var/log/messages
Oct 12 15:56:15 edterentjev systemd[1]: Started dbus-:1.10-org.fedoraproject.SetroubleshootPrivileged@0.service.
Oct 12 15:56:16 edterentjev setroubleshoot[41356]: SELinux is preventing /usr/sbin/httpd from getattr access on the file /var/www/html/test.html. For complete SELinux messages run: sealert -l 9525684d-2200-4be7-99a8-53f4782e54ee
Oct 12 15:56:16 edterentjev setroubleshoot[41356]: SELinux is preventing /usr/sbin/httpd from getattr access on the file /var/www/html/test.html.#012#012****
```

Figure 2.11: logs

Попробуйте запустить веб-сервер Apache на прослушивание TCP-порта 81. Выполните перезапуск веб-сервера Apache. Произошёл сбой? Проанализируйте лог-файлы. fig. 2.12

```
[root@edterentjev html]# systemctl restart httpd
[root@edterentjev html]# su - edterentjev
[edterentjev@edterentjev ~]$ tail -n1 /var/log/messages
tail: cannot open '/var/log/messages' for reading: Permission denied
[edterentjev@edterentjev ~]$ sudo tail -n1 /var/log/messages
[sudo] password for edterentjev:
Oct 12 16:01:39 edterentjev systemd[1]: Started Fingerprint Authentication Daemon.
[edterentjev@edterentjev ~]$ sudo tail -n1 /var/log/http/error_log
tail: cannot open '/var/log/http/error_log' for reading: No such file or directory
[edterentjev@edterentjev ~]$
```

Figure 2.12: restart_log

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. fig. 2.13

```
[edterentjev@edterentjev ~]$ sudo semanage port -a -t http_port_t -p tcp 81
ValueError: Port tcp/81 already defined
[edterentjev@edterentjev ~]$ semanage port -l | grep http_port_t
ValueError: SELinux policy is not managed or store cannot be accessed.
[edterentjev@edterentjev ~]$ sudo semanage port -l | grep http_port_t
http_port_t          tcp      80, 81, 443, 488, 8008, 8009, 8443, 9000
pegasus_http_port_t  tcp      5988
```

Figure 2.13: semanage_port

Верните контекст `httpd_sys_content_t` к файлу `/var/www/html/test.html` fig. 2.14

```
[edterentjev@edterentjev ~]$ sudo chcon -t httpd_sys_content_t /var/www/html/test.html
```

Figure 2.14: httpd_sys_content

Удалите привязку `http_port_t` к 81 порту и проверьте, что порт 81 удалён. (Удалить привязку невозможно) Удалите файл `/var/www/html/test.html` fig. 2.15

```
[edterentjev@edterentjev conf]$ sudo semanage port -d -t http_port_t -p tcp 81
ValueError: Port tcp/81 is defined in policy, cannot be deleted
[edterentjev@edterentjev conf]$ sudo semanage port -a -p tcp 81 -t http_port_t
ValueError: Port tcp/81 already defined
[edterentjev@edterentjev conf]$ sudo semanage port -d -p tcp 81 -t http_port_t
ValueError: Port tcp/81 is defined in policy, cannot be deleted
[edterentjev@edterentjev conf]$ rm /var/www/html/test.html
rm: remove write-protected regular file '/var/www/html/test.html'? y
rm: cannot remove '/var/www/html/test.html': Permission denied
[edterentjev@edterentjev conf]$
```

Figure 2.15: semanage_rm

3 Выводы

В результате выполнения работы я развитл навыки администрирования ОС Linux. Получил первое практическое знакомство с технологией SELinux, а также проверил работу SELinx на практике совместно с веб-сервером Apache.

4 Список литературы

1. Методические материалы курса