

Информационная безопасность. Отчет по лабораторной работе №6

Мандатное разграничение прав в Linux

Горбунова Ярослава Михайловна

Содержание

1	Цель работы	5
2	Теоретическое введение	6
2.1	Организация и описание лабораторного стенда	6
2.2	Подготовка лабораторного стенда и методические	6
3	Выполнение лабораторной работы	9
4	Выводы	21
5	Список литературы	22

List of Figures

2.1	Подготовка лабораторного стенда. Пункт 4	7
2.2	Подготовка лабораторного стенда. Пункт 5	8
3.1	Пункт 1-2	10
3.2	Пункт 3	11
3.3	Пункт 4	12
3.4	Пункт 5	13
3.5	Пункт 6-8	14
3.6	Пункт 9-10	14
3.7	Пункт 11	15
3.8	Пункт 12	15
3.9	Пункт 13	16
3.10	Пункт 14	16
3.11	Пункт 15 (1)	17
3.12	Пункт 15 (2)	17
3.13	Пункт 15 (3)	18
3.14	Пункт 16	18
3.15	Пункт 17	19
3.16	Пункт 18	19
3.17	Пункт 19 (1)	19
3.18	Пункт 19 (2)	19
3.19	Пункт 21	20
3.20	Пункт 24	20

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 [1].

2 Теоретическое введение

2.1 Организация и описание лабораторного стенда

Для проведения указанной лабораторной работы на одно рабочее место требуется компьютер с установленной операционной системой Linux, поддерживающей технологию SELinux. Предполагается использовать стандартный дистрибутив Linux CentOS с включённой политикой SELinux targeted и режимом enforcing. Для выполнения заданий требуется наличие учётной записи администратора (root) и учётной записи обычного пользователя. Постоянно работать от учётной записи root неправильно с точки зрения безопасности.

2.2 Подготовка лабораторного стенда и методические

рекомендации

1. При подготовке стенда обратите внимание, что необходимая для работы и указанная выше политика targeted и режим enforcing используются в данном дистрибутиве по умолчанию, т.е. каких-то специальных настроек не требуется. При этом следует убедиться, что политика и режим включены, особенно когда работа будет проводиться повторно и велика вероятность изменений при предыдущем использовании системы.
2. При необходимости администратор должен разбираться в работе SELinux и уметь как исправить конфигурационный файл `/etc/selinux/config`, так и

проверить используемый режим и политику.

3. Необходимо, чтобы был установлен веб-сервер Apache. При установке системы в конфигурации «рабочая станция» указанный пакет не ставится.
4. В конфигурационном файле `/etc/httpd/httpd.conf` необходимо задать параметр `ServerName`: `ServerName test.ru` чтобы при запуске веб-сервера не выдавались лишние сообщения об ошибках, не относящихся к лабораторной работе (fig. 2.1).

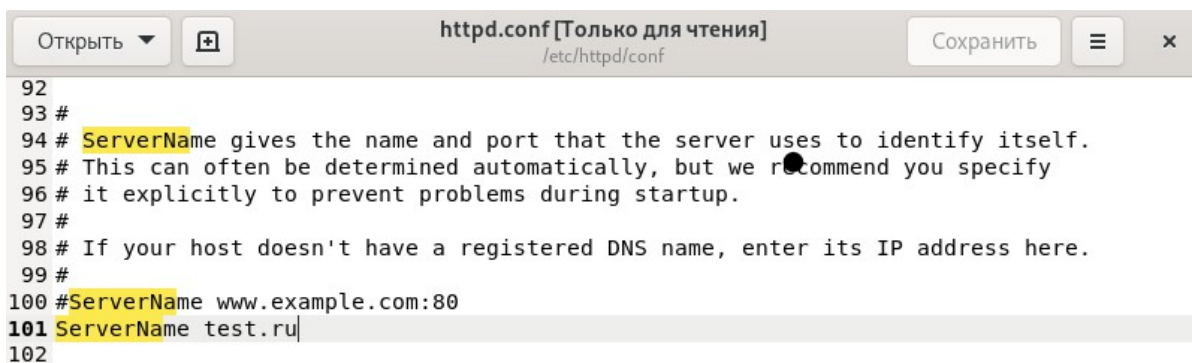


Figure 2.1: Подготовка лабораторного стенда. Пункт 4

5. Также необходимо проследить, чтобы пакетный фильтр был отключён или в своей рабочей конфигурации позволял подключаться к 80-у и 81-у портам протокола tcp (fig. 2.1).

Отключить фильтр можно командами

```
iptables -F
```

```
iptables -P INPUT ACCEPT iptables -P OUTPUT ACCEPT
```

либо добавить разрешающие правила:

```
iptables -I INPUT -p tcp -dport 80 -j ACCEPT
```

```
iptables -I INPUT -p tcp -dport 81 -j ACCEPT
```

```
iptables -I OUTPUT -p tcp -sport 80 -j ACCEPT
```

```
iptables -I OUTPUT -p tcp -sport 81 -j ACCEPT
```

```
[root@ymgorbunova ymgorbunova]# iptables -F  
[root@ymgorbunova ymgorbunova]# iptables -P INPUT ACCEPT  
[root@ymgorbunova ymgorbunova]# iptables -P OUTPUT ACCEPT  
[root@ymgorbunova ymgorbunova]#
```

Figure 2.2: Подготовка лабораторного стенда. Пункт 5

6. Обратите внимание, что данные правила не являются «точными» и рекомендуемыми на все случаи жизни, они лишь позволяют правильно организовать работу стенда.
7. В работе специально не делается акцент, каким браузером (или какой консольной программой) будет производиться подключение к вебсерверу. По желанию могут использоваться разные программы, такие как консольные `links`, `lynx`, `wget` и графические `konqueror`, `opera`, `firefox` или др.

3 Выполнение лабораторной работы

1. Войдите в систему с полученными учётными данными и убедитесь, что SELinux работает в режиме enforcing политики targeted с помощью команд `getenforce` и `sestatus` (fig. 3.1).
2. Обратитесь с помощью браузера к веб-серверу, запущенному на вашем компьютере, и убедитесь, что последний работает (fig. 3.1):

`service httpd status` или
`/etc/rc.d/init.d/httpd status`

Если не работает, запустите его так же, но с параметром `start`.

```

[ymgorbunova@ymgorbunova ~]$ getenforce
Enforcing
[ymgorbunova@ymgorbunova ~]$ 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
[ymgorbunova@ymgorbunova ~]$ 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 pre>
   Active: active (running) since Tue 2022-10-11 15:58:26 MSK; 47min ago
     Docs: man:httpd.service(8)
  Main PID: 943 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes>
      Tasks: 213 (limit: 12209)
     Memory: 19.2M
        CPU: 1.050s
    CGroup: /system.slice/httpd.service
            └─943 /usr/sbin/httpd -DFOREGROUND
              └─962 /usr/sbin/httpd -DFOREGROUND
                └─963 /usr/sbin/httpd -DFOREGROUND
                  └─964 /usr/sbin/httpd -DFOREGROUND
                    └─965 /usr/sbin/httpd -DFOREGROUND

окт 11 15:58:26 ymgorbunova.localdomain systemd[1]: Starting The Apache HTTP Se>
окт 11 15:58:26 ymgorbunova.localdomain systemd[1]: Started The Apache HTTP Ser>
окт 11 15:58:26 ymgorbunova.localdomain httpd[943]: Server configured, listenin>
[ymgorbunova@ymgorbunova ~]$

```

Figure 3.1: Пункт 1-2

3. Найдите веб-сервер Apache в списке процессов, определите его контекст безопасности и занесите эту информацию в отчёт. Например, можно использовать команду (fig. 3.2) `ps auxZ | grep httpd` или `ps -eZ | grep httpd`

```

[ymgorbunova@ymgorbunova ~]$ ps auxZ | grep httpd
system_u:system_r:httpd_t:s0      root          943  0.0  0.2 20248  5156 ?        Ss   15:57   0:00
/usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0      apache        962  0.0  0.1 21572  2992 ?        S    15:57   0:00
/usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0      apache        963  0.0  0.3 1210512 7304 ?        Sl   15:57   0:00
/usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0      apache        964  0.0  0.2 1079376 5328 ?        Sl   15:57   0:00
/usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0      apache        965  0.0  0.2 1079376 5364 ?        Sl   15:57   0:00
/usr/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 ymgorbu+ 4381 0.0  0.1 221824 2272 pts/1 S+ 1
6:46   0:00 grep --color=auto httpd
[ymgorbunova@ymgorbunova ~]$ ps -eZ | grep httpd
system_u:system_r:httpd_t:s0      943 ?          00:00:00 httpd
system_u:system_r:httpd_t:s0      962 ?          00:00:00 httpd
system_u:system_r:httpd_t:s0      963 ?          00:00:00 httpd
system_u:system_r:httpd_t:s0      964 ?          00:00:00 httpd
system_u:system_r:httpd_t:s0      965 ?          00:00:00 httpd
[ymgorbunova@ymgorbunova ~]$

```

Figure 3.2: Пункт 3

4. Посмотрите текущее состояние переключателей SELinux для Apache с помощью команды (fig. 3.3) `sestatus -bigrep httpd` Обратите внимание, что многие из них находятся в положении «off».

```

[ymgorbunova@ymgorbunova ~]$ 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
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 on
httpd_use_cifs off
httpd_use_fusefs off
httpd_use_gpg off
httpd_use_nfs off
httpd_use_opencryptoki off
httpd_use_openstack off
httpd_use_sasl off
httpd_verify_dns off
[ymgorbunova@ymgorbunova ~]$

```

Figure 3.3: Пункт 4

5. Посмотрите статистику по политике с помощью команды `seinfo`, также определите множество пользователей, ролей, типов (fig. 3.4).

```
[ymgorbunova@ymgorbunova ~]$ seinfo
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:            4995     Attributes:         254
Users:            8        Roles:              14
Booleans:         347     Cond. Expr.:       382
Allow:            63727    Neverallow:         0
Auditallow:       163     Dontaudit:          8391
Type_trans:       251060   Type_change:        87
Type_member:       35     Range_trans:        5958
Role_allow:        38     Role_trans:         418
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

[ymgorbunova@ymgorbunova ~]$
```

Figure 3.4: Пункт 5

6. Определите тип файлов и поддиректорий, находящихся в директории `/var/www`, с помощью команды (fig. 3.5) `ls -lZ /var/www`
7. Определите тип файлов, находящихся в директории `/var/www/html` (fig. 3.5): `ls -lZ /var/www/html`
8. Определите круг пользователей, которым разрешено создание файлов в директории `/var/www/html` (fig. 3.5).

```
[ymgorbunova@ymgorbunova ~]$ ls -lZ /var/www
итого 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 мая 16 15:10 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0 6 мая 16 15:10 html
[ymgorbunova@ymgorbunova ~]$ ls -lZ /var/www/html
итого 0
```

Figure 3.5: Пункт 6-8

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 (fig. 3.6):
test
10. Проверьте контекст созданного вами файла. Занесите в отчёт контекст, присваиваемый по умолчанию вновь созданным файлам в директории /var/www/html (fig. 3.6).

```
[root@ymgorbunova ymgorbunova]# echo "<html>
<body>test</body>
</html>" > /var/www/html/test.html
[root@ymgorbunova ymgorbunova]# cat /var/www/html/test.html
<html>
<body>test</body>
</html>
[root@ymgorbunova ymgorbunova]# touch /var/www/html/my.html
[root@ymgorbunova ymgorbunova]# cat /var/www/html/my.html
[root@ymgorbunova ymgorbunova]#
```

Figure 3.6: Пункт 9-10

11. Обратитесь к файлу через веб-сервер, введя в браузере адрес <http://127.0.0.1/test.html>. Убедитесь, что файл был успешно отображён (fig. 3.7).

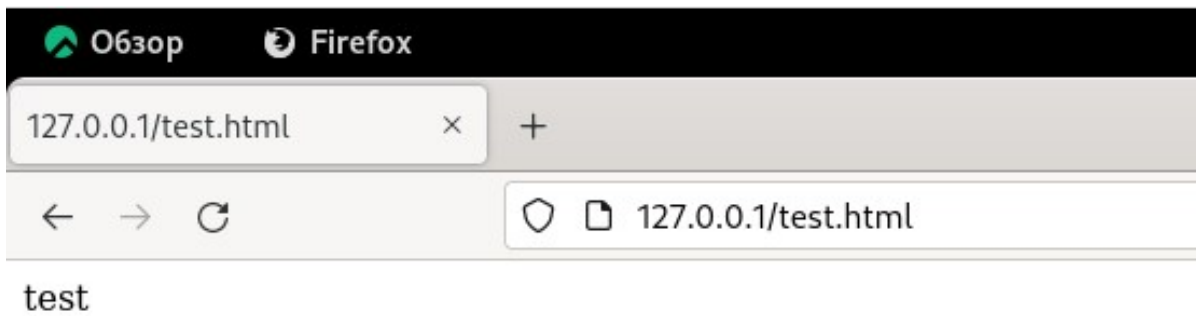


Figure 3.7: Пункт 11

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`

Рассмотрим полученный контекст детально. Обратите внимание, что так как по умолчанию пользователи CentOS являются свободными от типа (`unconfined` в переводе с англ. означает свободный), созданному нами файлу `test.html` был сопоставлен SELinux, пользователь `unconfined_u`. Это первая часть контекста. Далее политика ролевого разделения доступа RBAC используется процессами, но не файлами, поэтому роли не имеют никакого значения для файлов. Роль `object_r` используется по умолчанию для файлов на «постоянных» носителях и на сетевых файловых системах. (В директории `/ргос` файлы, относящиеся к процессам, могут иметь роль `system_r`. Если активна политика MLS, то могут использоваться и другие роли, например, `secadm_r`. Данный случай мы рассматривать не будем, как и предназначение `:s0`). Тип `httpd_sys_content_t` позволяет процессу `httpd` получить доступ к файлу. Благодаря наличию последнего типа мы получили доступ к файлу при обращении к нему через браузер (fig. 3.8).

```
[root@ymgorbunova ymgorbunova]# ls -Z /var/www/html/test.html
unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.html
[root@ymgorbunova ymgorbunova]#
```

Figure 3.8: Пункт 12

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 (fig. 3.9).

```
[root@ymgorbunova ymgorbunova]# chcon -t samba_share_t /var/www/html/test.html
[root@ymgorbunova ymgorbunova]# ls -Z /var/www/html/test.html
unconfined_u:object_r:samba_share_t:s0 /var/www/html/test.html
```

Figure 3.9: Пункт 13

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` (fig. 3.10).

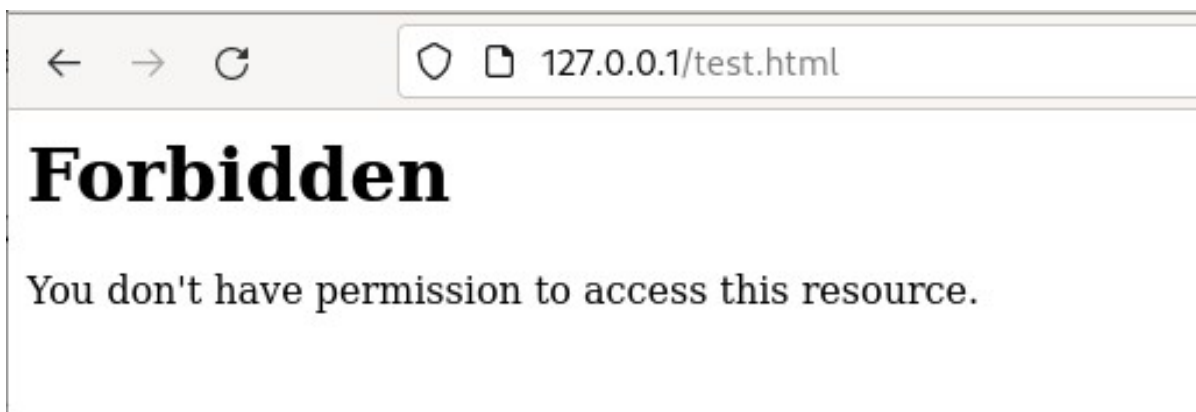


Figure 3.10: Пункт 14

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 (fig. 3.11-fig. 3.13).

```
[root@ymgorbunova ymgorbunova]# ls -l /var/www/html/test.html
-rw-r--r--. 1 root root 33 окт 11 17:11 /var/www/html/test.html
```

Figure 3.11: Пункт 15 (1)

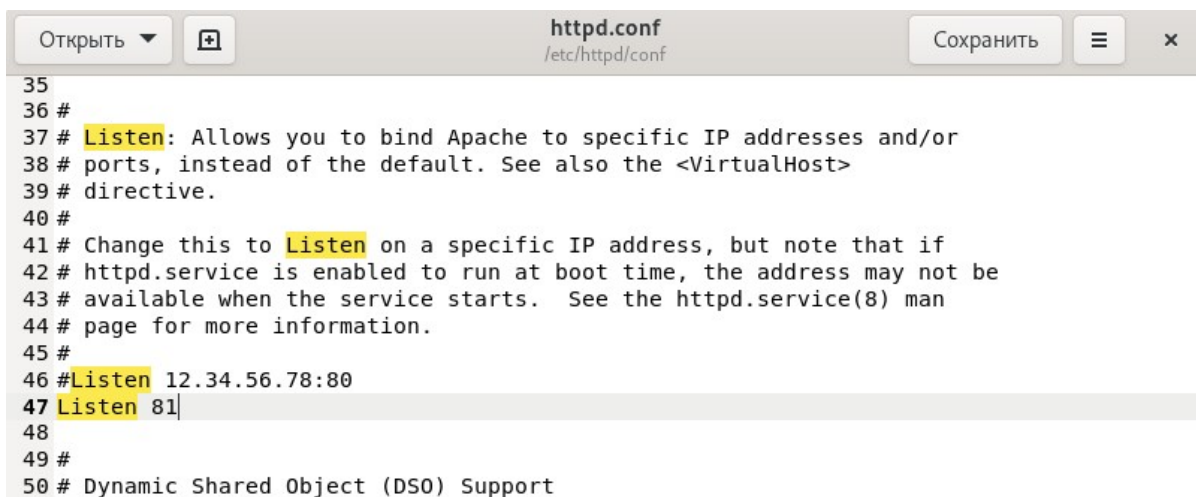
```
[root@ymgorbunova ymgorbunova]# tail /var/log/messages
Oct 11 17:30:15 ymgorbunova setroubleshoot[5129]: failed to retrieve rpm info for /var/www/html/test.htm
l
Oct 11 17:30:15 ymgorbunova setroubleshoot[5129]: SELinux запрещает /usr/sbin/httpd доступ getattr к фай
лу /var/www/html/test.html. Для выполнения всех сообщений SELinux: sealert -l 21b3fad3-a74f-4ebe-a9a6-ac8
d253ee43b
Oct 11 17:30:15 ymgorbunova setroubleshoot[5129]: SELinux запрещает /usr/sbin/httpd доступ getattr к фай
лу /var/www/html/test.html.#012#012***** Модуль restorecon предлагает (точность 92.2) *****
*****#012#012Если вы хотите исправить метку.$TARGETЗнак _PATH по умолчанию должен быть httpd_sys_cont
ent_t#012То вы можете запустить restorecon. Возможно, попытка доступа была остановлена из-за недостаточн
ых разрешений для доступа к родительскому каталогу, и в этом случае попробуйте соответствующим образом
изменить следующую команду.#012Сделать#012# /sbin/restorecon -v /var/www/html/test.html#012#012***** Мо
дуль public_content предлагает (точность 7.83) *****#012#012Если вы хотите лечить test.
html как общедоступный контент#012То необходимо изменить метку test.html с public_content_t на public_co
ntent_rw_t.#012Сделать#012# semanage fcontext -a -t public_content_t '/var/www/html/test.html'#012# rest
orecon -v '/var/www/html/test.html'#012#012***** Модуль catchall предлагает (точность 1.41) *****
*****#012#012Если вы считаете, что httpd должно быть разрешено getattr доступ к test.html fi
le по умолчанию.#012То рекомендуется создать отчет об ошибке.#012Чтобы разрешить доступ, можно создать л
окальный модуль политики.#012Сделать#012# зрешить этот доступ сейчас, выполнив:#012# ausearch -c 'httpd'
--raw | audit2allow -M my-httpd#012# semodule -X 300 -i my-httpd.pp#012
Oct 11 17:30:25 ymgorbunova systemd[1]: dbus-:1.10-org.fedoraproject.SetroubleshootPrivileged@0.service:
Main process exited, code=killed, status=14/ALRM
Oct 11 17:30:25 ymgorbunova systemd[1]: dbus-:1.10-org.fedoraproject.SetroubleshootPrivileged@0.service:
Failed with result 'signal'.
Oct 11 17:30:25 ymgorbunova systemd[1]: dbus-:1.10-org.fedoraproject.Setroubleshootd@0.service: Main pro
cess exited, code=killed, status=14/ALRM
Oct 11 17:30:25 ymgorbunova systemd[1]: dbus-:1.10-org.fedoraproject.Setroubleshootd@0.service: Failed w
ith result 'signal'.
Oct 11 17:32:11 ymgorbunova systemd[1]: Starting Fingerprint Authentication Daemon...
Oct 11 17:32:11 ymgorbunova systemd[1]: Started Fingerprint Authentication Daemon.
Oct 11 17:32:15 ymgorbunova su[5219]: (to root) ymgorbunova on pts/1
[root@ymgorbunova ymgorbunova]#
```

Figure 3.12: Пункт 15 (2)

```
[root@ymgorbunova ymgorbunova]# tail /var/log/audit/audit.log
type=USER_END msg=audit(1665498673.772:192): pid=4674 uid=1000 auid=1000 ses=3 subj=unconfined_u:unconfi
ned_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:session_close grantors=pam keyinit,pam_limits,pam_systemd,
pam_unix,pam_umask,pam_xauth acct="root" exe="/usr/bin/su" hostname=? addr=? terminal=/dev/pts/1 res=suc
cess'UID="ymgorbunova" AUID="ymgorbunova"
type=CRED_DISP msg=audit(1665498673.772:193): pid=4674 uid=1000 auid=1000 ses=3 subj=unconfined_u:unconfi
ned_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:setcred grantors=pam_unix acct="root" exe="/usr/bin/su" h
ostname=? addr=? terminal=/dev/pts/1 res=success'UID="ymgorbunova" AUID="ymgorbunova"
type=BPF msg=audit(1665498731.403:194): prog-id=51 op=LOAD
type=SERVICE_START msg=audit(1665498731.489:195): pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system
_u:system_r:init_t:s0 msg='unit=fprintd comm="systemd" exe="/usr/lib/systemd/systemd" hostname=? addr=?
terminal=? res=success'UID="root" AUID="unset"
type=USER_AUTH msg=audit(1665498735.781:196): pid=5219 uid=1000 auid=1000 ses=3 subj=unconfined_u:unconfi
ned_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:authentication grantors=pam_unix acct="root" exe="/usr/bi
n/su" hostname=? addr=? terminal=/dev/pts/1 res=success'UID="ymgorbunova" AUID="ymgorbunova"
type=USER_ACCT msg=audit(1665498735.798:197): pid=5219 uid=1000 auid=1000 ses=3 subj=unconfined_u:unconfi
ned_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:accounting grantors=pam_unix,pam_localuser acct="root" ex
e="/usr/bin/su" hostname=? addr=? terminal=/dev/pts/1 res=success'UID="ymgorbunova" AUID="ymgorbunova"
type=CRED_ACQ msg=audit(1665498735.803:198): pid=5219 uid=1000 auid=1000 ses=3 subj=unconfined_u:unconfi
ned_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:setcred grantors=pam_unix acct="root" exe="/usr/bin/su" ho
stname=? addr=? terminal=/dev/pts/1 res=success'UID="ymgorbunova" AUID="ymgorbunova"
type=USER_START msg=audit(1665498735.842:199): pid=5219 uid=1000 auid=1000 ses=3 subj=unconfined_u:unconfi
ned_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:session_open grantors=pam keyinit,pam_limits,pam_systemd
,pam_unix,pam_umask,pam_xauth acct="root" exe="/usr/bin/su" hostname=? addr=? terminal=/dev/pts/1 res=su
ccess'UID="ymgorbunova" AUID="ymgorbunova"
type=SERVICE_STOP msg=audit(1665498761.704:200): pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system
_u:system_r:init_t:s0 msg='unit=fprintd comm="systemd" exe="/usr/lib/systemd/systemd" hostname=? addr=? t
erminal=? res=success'UID="root" AUID="unset"
type=BPF msg=audit(1665498761.716:201): prog-id=51 op=UNLOAD
[root@ymgorbunova ymgorbunova]#
```

Figure 3.13: Пункт 15 (3)

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 (fig. 3.14).



```
35
36 #
37 # Listen: Allows you to bind Apache to specific IP addresses and/or
38 # ports, instead of the default. See also the <VirtualHost>
39 # directive.
40 #
41 # Change this to Listen on a specific IP address, but note that if
42 # httpd.service is enabled to run at boot time, the address may not be
43 # available when the service starts. See the httpd.service(8) man
44 # page for more information.
45 #
46 #Listen 12.34.56.78:80
47 Listen 81
48
49 #
50 # Dynamic Shared Object (DSO) Support
```

Figure 3.14: Пункт 16

17. Выполните перезапуск веб-сервера Apache. Произошёл сбой? Поясните почему? (fig. 3.15)

```
[root@ymgorbunova ymgorbunova]# service httpd stop
Redirecting to /bin/systemctl stop httpd.service
[root@ymgorbunova ymgorbunova]# service httpd start
Redirecting to /bin/systemctl start httpd.service
```

Figure 3.15: Пункт 17

18. Проанализируйте лог-файлы: `tail -n1 /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 (fig. 3.16).

```
[root@ymgorbunova ymgorbunova]# tail -n1 /var/log/messages
Oct 11 17:41:53 ymgorbunova httpd[5523]: Server configured, listening on: port 81
[root@ymgorbunova ymgorbunova]# tail -n1 /var/log/audit/audit.log
type=SERVICE_START msg=audit(1665499313.903:203): pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:init t:s0 msg='unit=httpd comm="systemd" exe="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success' UID="root" AUID="unset"
```

Figure 3.16: Пункт 18

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 (fig. 3.17-fig. 3.18).

```
[root@ymgorbunova ymgorbunova]# semanage port -a -t http_port_t -p tcp 81
usage: semanage [-h]
               {import,export,login,user,port,ibpkey,ibendport,interface,module,node,fcontext,boolean,permissive,dontaudit}
```

Figure 3.17: Пункт 19 (1)

```
[root@ymgorbunova ymgorbunova]# 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
[root@ymgorbunova ymgorbunova]#
```

Figure 3.18: Пункт 19 (2)

20. Попробуйте запустить веб-сервер Apache ещё раз. Поняли ли вы, почему он сейчас запустился, а в предыдущем случае не смог? – В данном случае сервер запустился в обоих случаях, потому что была выполнена предварительная подготовка лабораторного стенда (см. Подготовка лабораторного стенда. Пункт 5).
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» (fig. 3.19).

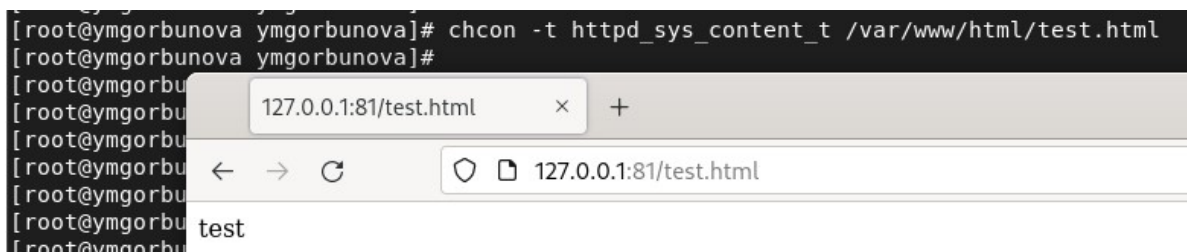


Figure 3.19: Пункт 21

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` (fig. 3.20).

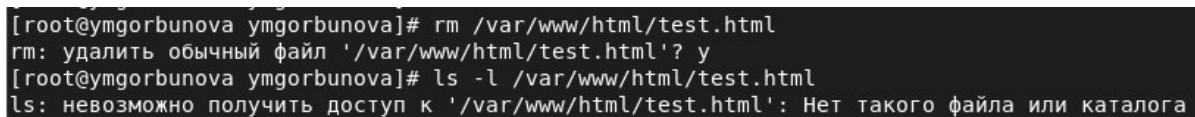


Figure 3.20: Пункт 24

4 Выводы

Развиты навыки администрирования ОС Linux. Получено первое практическое знакомство с технологией SELinux. Проверена работа SELinux на практике совместно с веб-сервером Apache.

5 Список литературы

1. Методические материалы курса