

Отчёт

по лабораторной работе 6

Агеева Анастасия Борисовна

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	23

List of Figures

3.1	рис.1. Обновление Apache.	7
3.2	рис.2.	7
3.3	рис.3	8
3.4	рис.4	8
3.5	рис.5	8
3.6	рис.6	9
3.7	рис.7	10
3.8	рис.8	10
3.9	рис.9	11
3.10	рис.10	11
3.11	рис.11	12
3.12	рис.12	12
3.13	рис.13	13
3.14	рис.14	14
3.15	рис.15	14
3.16	рис.17	15
3.17	рис.18	15
3.18	рис.19	15
3.19	рис.20	15
3.20	рис.21	16
3.21	рис.22	16
3.22	рис.23	17
3.23	рис.24	17
3.24	рис.25	18
3.25	рис.26	18
3.26	рис.27	19
3.27	рис.28	19
3.28	рис.29	19
3.29	рис.32	20
3.30	рис.33	20
3.31	рис.34	20
3.32	рис.35	21
3.33	рис.36	21
3.34	рис.37	21
3.35	рис.38	22

List of Tables

1 Цель работы

Развить навыки администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux¹. Проверить работу SELinx на практике совместно с веб-сервером Apache.

2 Задание

Лабораторная работа подразумевает использование стандартного дистрибутива Linux CentOS с включённой политикой SELinux targeted и режимом enforcing

3 Выполнение лабораторной работы

1. Установим/обновим веб-сервер Apache (рис.1).

```
[abageeval@abageeval ~]$ su
Пароль:
[root@abageeval abageeval]# yum install httpd -y
Загружены модули: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: mirror.docker.ru
 * extras: mirror.reconn.ru
 * updates: mirror.s.datahouse.ru
Разрешение зависимостей
--> Проверка сценария
--> Пакет httpd.x86_64 0:2.4.6-97.el7.centos.2 помечен для установки
--> Обработка зависимостей: httpd-tools = 2.4.6-97.el7.centos.2 пакета: httpd-2.4.6-97.el7.centos.2.x86_64
--> Проверка сценария
--> Обработка зависимостей: /etc/mime.types пакета: httpd-2.4.6-97.el7.centos.2.x86_64
--> Пакет httpd-tools.x86_64 0:2.4.6-97.el7.centos.2 помечен для установки
--> Пакет mailcap.noarch 0:2.1.41-2.el7 помечен для установки
--> Проверка зависимостей окончена

Зависимости определены
```

Package	Архитектура	Версия	Репозиторий	Размер
Установка:				
httpd	x86_64	2.4.6-97.el7.centos.2	updates	2.7 М

Figure 3.1: рис.1. Обновление Apache.

2. В конфигурационном файле /etc/httpd/conf/httpd.conf зададим параметр ServerName (рис.2).

```
# virtual host being defined.
#
#
# ServerAdmin: Your address, where problems with the server should be
# e-mailed. This address appears on some server-generated pages, such
# as error documents. e.g. admin@your-domain.com
#
ServerAdmin root@localhost
#
# ServerName gives the name and port that the server uses to identify itself.
# This can often be determined automatically, but we recommend you specify
# it explicitly to prevent problems during startup.
#
# If your host doesn't have a registered DNS name, enter its IP address here.
#
#ServerName test.ru
#
# Deny access to the entirety of your server's filesystem. You must
```




Figure 3.2: рис.2.

3. Добавим разрешающие правила для подключения к 80-у и 81-у портам протокола TCP (рис.3-6).

```
[root@abageeval abageeval]# iptables -I INPUT -p tcp --dport 80 -j ACCEPT
[root@abageeval abageeval]# iptables -I INPUT -p tcp --dport 81 -j ACCEPT
[root@abageeval abageeval]# iptables -I OUTPUT -p tcp --sport 81 -j ACCEPT
[root@abageeval abageeval]# iptables -I OUTPUT -p tcp --sport 80 -j ACCEPT
[root@abageeval abageeval]#
```

Figure 3.3: рис.3

4. Убедимся, что SELinux работает в режиме enforcing политики targeted с помощью команд `getenforce` и `sestatus` (Рис. 4)
5. Обратимся к веб-серверу, запущенному на нашем стенде, и убедимся, что он работает. (рис.4).

```
[root@abageeval abageeval]# 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
Max kernel policy version:     31
[root@abageeval abageeval]# getenforce
Enforcing
[root@abageeval abageeval]# 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 preset: disabled)
   Active: inactive (dead)
     Docs: man:httd(8)
           man:apachectl(8)
[root@abageeval abageeval]#
```

Figure 3.4: рис.4

6. Найдем Apache в списке процессов, определим его контекст безопасности. В нашем случае контекст безопасности `unconfined_u:system_r:httpd_t` (Рис. 5)

```
[root@abageeval abageeval]# getenforce
Enforcing
[root@abageeval abageeval]# 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 preset: disabled)
   Active: inactive (dead)
     Docs: man:httd(8)
           man:apachectl(8)
[root@abageeval abageeval]# ps auxZ | grep httpd
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 6622 0.0  0.0 112732 980 pts
/0 R+ 18:53  0:00 grep --color=auto httpd
[root@abageeval abageeval]#
```

Figure 3.5: рис.5

7. Посмотрим текущее состояние переключателей SELinux для Apache. Многие из переключателей в положении “off” (рис.6-14).

```
[root@abageeval abageeval]# 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
Max kernel policy version:     31

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
awstats_purge_apache_log_files off
boinc_execmem                  on
cdrecord_read_content          off
cluster_can_network_connect    off
```

Figure 3.6: рис.6

```

virt_use_usb                on
virt_use_xserver            off
webadm_manage_user_files   off
webadm_read_user_files     off
wine_mmap_zero_ignore      off
xdm_bind_vnc_tcp_port      off
xdm_exec_bootloader        off
xdm_sysadm_login           off
xdm_write_home             off
xen_use_nfs                off
xend_run_blktp             on
xend_run_qemu              on
xguest_connect_network     on
xguest_exec_content        on
xguest_mount_media         on
xguest_use_bluetooth       on
xserver_clients_write_xshm off
xserver_execmem            off
xserver_object_manager     off
zabbix_can_network         off
zabbix_run_sudo            off
zarafe_setrlimit           off
zebra_write_config         off
zoneminder_anon_write     off
zoneminder_run_sudo       off
[root@abageeval abageeval]# █

```

Figure 3.7: рис.7

```

cluster_manage_all_files   off
cluster_use_execmem        off
cobbler_anon_write         off
cobbler_can_network_connect off
cobbler_use_cifs           off
cobbler_use_nfs            off
collectd_tcp_network_connect off
condor_tcp_network_connect off
conman_can_network         off
conman_use_nfs             off
container_connect_any      off
cron_can_relabel           off
cron_system_cronjob_use_shares off
cron_userdomain_transition on
cups_execmem               off
cvs_read_shadow            off
daemons_dump_core         off
daemons_enable_cluster_mode off
daemons_use_tcp_wrapper   off
daemons_use_tty           off
dbadm_exec_content         on
dbadm_manage_user_files    off
dbadm_read_user_files      off
deny_execmem               off
deny_ptrace                off
dhcpc_exec_intables        off

```

Figure 3.8: рис.8

domain_can_mmap_files	on
domain_can_write_kmsg	off
domain_fd_use	on
domain_kernel_load_modules	off
entropyd_use_audio	on
exim_can_connect_db	off
exim_manage_user_files	off
exim_read_user_files	off
fcron_cron	off
fenced_can_network_connect	off
fenced_can_ssh	off
fips_mode	on
ftpd_anon_write	off
ftpd_connect_all_unreserved	off
ftpd_connect_db	off
ftpd_full_access	off
ftpd_use_cifs	off
ftpd_use_fusefs	off
ftpd_use_nfs	off
ftpd_use_passive_mode	off
git_cgi_enable_homedirs	off
git_cgi_use_cifs	off
git_cgi_use_nfs	off
git_session_bind_all_unreserved_ports	off
git_session_users	off
git_system_enable_homedirs	off
git_system_use_cifs	off

Figure 3.9: рис.9

gluster_export_all_ro	off
gluster_export_all_rw	on
gluster_use_execmem	off
gpg_web_anon_write	off
gssd_read_tmp	on
guest_exec_content	on
haproxy_connect_any	off
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_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_sys_vfs	off

Figure 3.10: рис.10

mcelog_client	off
mcelog_exec_scripts	on
mcelog_foreground	off
mcelog_server	off
minidlna_read_generic_user_content	off
mmap_low_allowed	off
mock_enable_homedirs	off
mount_anyfile	on
mozilla_plugin_bind_unreserved_ports	off
mozilla_plugin_can_network_connect	off
mozilla_plugin_use_bluejeans	off
mozilla_plugin_use_gps	off
mozilla_plugin_use_spice	off
mozilla_read_content	off
mpd_enable_homedirs	off
mpd_use_cifs	off
mpd_use_nfs	off
mplayer_execstack	off
mysql_connect_any	off
nagios_run_pnp4nagios	off
nagios_run_sudo	off
nagios_use_nfs	off
named_tcp_bind_http_port	off
named_write_master_zones	off
neutron_can_network	off
nfs export all ro	on

Figure 3.11: рис.11

rsync_anon_write	off
rsync_client	off
rsync_export_all_ro	off
rsync_full_access	off
samba_create_home_dirs	off
samba_domain_controller	off
samba_enable_home_dirs	off
samba_export_all_ro	off
samba_export_all_rw	off
samba_load_libgfapi	off
samba_portmapper	off
samba_run_unconfined	off
samba_share_fusefs	off
samba_share_nfs	off
sanlock_enable_home_dirs	off
sanlock_use_fusefs	off
sanlock_use_nfs	off
sanlock_use_samba	off
saslauthd_read_shadow	off
secadm_exec_content	on
secure_mode	off
secure_mode_insmode	off
secure_mode_policyload	off
selinuxuser_direct_dri_enabled	on
selinuxuser_execheap	off
selinuxuser_execmod	on

Figure 3.12: рис.12

selinuxuser_direct_dri_enabled	on
selinuxuser_execheap	off
selinuxuser_execmod	on
selinuxuser_execstack	on
selinuxuser_mysql_connect_enabled	off
selinuxuser_ping	on
selinuxuser_postgresql_connect_enabled	off
selinuxuser_rw_noexattrfile	on
selinuxuser_share_music	off
selinuxuser_tcp_server	off
selinuxuser_udp_server	off
selinuxuser_use_ssh_chroot	off
sge_domain_can_network_connect	off
sge_use_nfs	off
smartmon_3ware	off
smbd_anon_write	off
spamassassin_can_network	off
spamd_enable_home_dirs	on
spamd_update_can_network	off
squid_connect_any	on
squid_use_tproxy	off
ssh_chroot_rw_homedirs	off
ssh_keysign	off
ssh_sysadm_login	off
staff_exec_content	on
staff_use_svirt	off

Figure 3.13: рис.13

```

staff_use_svirt off
swift_can_network off
sysadm_exec_content on
telepathy_connect_all_ports off
telepathy_tcp_connect_generic_network_ports on
tftp_anon_write off
tftp_home_dir off
tmpreaper_use_cifs off
tmpreaper_use_nfs off
tmpreaper_use_samba off
tomcat_can_network_connect_db off
tomcat_read_rpm_db off
tomcat_use_execmem off
tor_bind_all_unreserved_ports off
tor_can_network_relay off
unconfined_chrome_sandbox_transition on
unconfined_login on
unconfined_mozilla_plugin_transition on
unprivuser_use_svirt off
use_ecryptfs_home_dirs off
use_fusefs_home_dirs off
use_lpd_server off
use_nfs_home_dirs off
use_samba_home_dirs off
user_exec_content on
varnishd_connect_any off

```

Figure 3.14: рис.14

8. Посмотрим статистику по политике с помощью команды `seinfo`, также определим множество пользователей, ролей и типов. Пользователей: 9 Ролей: 12 Типов: 3920 (рис.15).
9. Определим тип файлов и поддиректории, находящихся в директории `/var/www` (рис.15).
10. Определим тип файлов, находящихся в директории `/var/www/html` (рис.15).

```

[root@abageeval abageeval]# seinfo
bash: seinfo: команда не найдена...
[root@abageeval abageeval]# seinfo
bash: seinfo: команда не найдена...
[root@abageeval abageeval]# ls -lZ /var/www
drwxr-xr-x. root root system_u:object_r:httpd_sys_script_exec_t:s0 cgi-bin
drwxr-xr-x. root root system_u:object_r:httpd_sys_content_t:s0 html
[root@abageeval abageeval]# ls -lZ /var/www/html
[root@abageeval abageeval]#

```

Figure 3.15: рис.15

11. Определим круг пользователей, которым разрешено создание файлов в директории `/var/www/html` (рис.17).

```
[root@abageeval abageeval]# echo "test" > /var/www/html/test.txt
[root@abageeval abageeval]# exit
exit
[abageeval@abageeval ~]$ echo "test" > /var/www/html/test.txt
bash: /var/www/html/test.txt: Отказано в доступе
[abageeval@abageeval ~]$
```

Figure 3.16: рис.17

12. По рисунку видно, что только root может создать файл в данной директории.
13. В следствие этого создадим от имени суперпользователя html-файл /var/www/html/test.html следующего содержания (рис.18).

```
GNU nano 2.3.1      Файл: /var/www/html/test.html
<html>
<body>test</body>
</html>

]
```

Figure 3.17: рис.18

14. Проверим контекст созданного файла. В нашем случае контекст unconfined_u:object_r:httpd_sys_content_t (рис.19).

```
[root@abageeval abageeval]# nano /var/www/html/test.html
[root@abageeval abageeval]# ls -lZ /var/www/html
-rw-r--r--. root root unconfined_u:object_r:httpd_sys_content_t:s0 test.html
-rw-r--r--. root root unconfined_u:object_r:httpd_sys_content_t:s0 test.txt
[root@abageeval abageeval]# ls -l /var/www/html
итого 8
-rw-r--r--. 1 root root 33 ноя 25 19:07 test.html
-rw-r--r--. 1 root root  5 ноя 25 19:03 test.txt
[root@abageeval abageeval]#
```

Figure 3.18: рис.19

15. Обратимся к файлу через веб-сервер, введя в firefox адрес <http://127.0.0.1/test.html> Убедимся, что файл был успешно отображен (рис.20).

```
test
```

Figure 3.19: рис.20

16. Проверим контекст файла. Т.к. по умолчанию пользователи CentOS являются unconfined от типа, 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. 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. (рис.21).
17. Изменим контекст файла /var/www/html/test.html с httpd_sys_content_t на samba_shate_t. Как видно из рисунка, контекст успешно сменился. (рис.21).

```
{root@abageeval abageeval}# ls -Z /var/www/html/test.html
-rw-r--r--. root root unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.h
tml
{root@abageeval abageeval}# chcon -t samba_share_t /var/www/html/test.html
{root@abageeval abageeval}# ls -Z /var/www/html/test.html
-rw-r--r--. root root unconfined_u:object_r:samba_share_t:s0 /var/www/html/test.html
{root@abageeval abageeval}#
```

Figure 3.20: рис.21

18. Попробуем еще раз получить доступ к файлу через веб-сервер, введя в firefox адрес http://127.0.0.1/test.html. Как видно из рисунка, мы получили сообщение об ошибке. (рис.22).

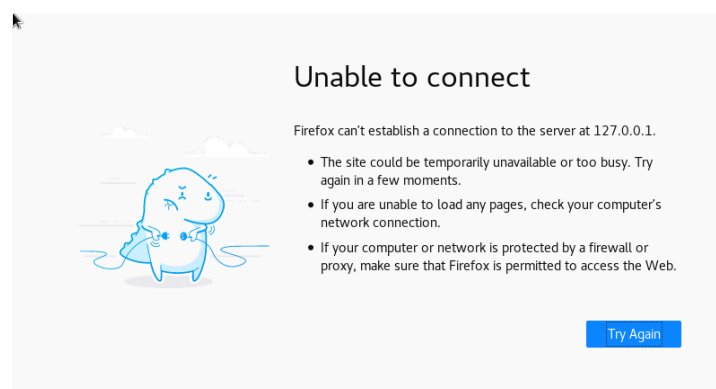


Figure 3.21: рис.22

19. Проанализируем ситуацию, просмотрев log-файлы веб-сервера Apache, системный log-файл и audit.log при условии уже запущенных процессов setroubleshootd и audtd. Исходя из log-файлов, мы можем заметить, что проблема в измененном контексте на шаге 17, т.к. процесс httpd не имеет доступа на samba_share_t. (рис.23-25).

```
[root@abageeval abageeval]# ls -l /var/www/html/test.html
-rw-r--r--. 1 root root 33 ноя 25 19:07 /var/www/html/test.html
[root@abageeval abageeval]# tail /var/log/messages
Nov 25 19:06:01 abageeval dbus[2751]: [system] Activating service name='org.freedesktop
.problems' (using servicehelper)
Nov 25 19:06:01 abageeval dbus[2751]: [system] Successfully activated service 'org.free
desktop.problems'
Nov 25 19:10:01 abageeval systemd: Created slice User Slice of root.
Nov 25 19:10:01 abageeval systemd: Started Session 6 of user root.
Nov 25 19:10:02 abageeval systemd: Removed slice User Slice of root.
Nov 25 19:14:27 abageeval journal: g_simple_action_set_enabled: assertion 'G_IS_SIMPLE_
ACTION (simple)' failed
Nov 25 19:15:48 abageeval journal: g_simple_action_set_enabled: assertion 'G_IS_SIMPLE_
ACTION (simple)' failed
Nov 25 19:20:01 abageeval systemd: Created slice User Slice of root.
Nov 25 19:20:01 abageeval systemd: Started Session 7 of user root.
Nov 25 19:20:01 abageeval systemd: Removed slice User Slice of root.
[root@abageeval abageeval]#
```

Figure 3.22: рис.23

```
[root@abageeval abageeval]# tail /var/log/audit/audit.log
type=CRED_DISP msg=audit(1637856602.157:264): pid=7038 uid=0 auid=0 ses=6 subj=system_u
:system_r:crond_t:s0-s0:c0.c1023 msg='op=PAM:setcred grantors=pam_env,pam_unix acct="ro
ot" exe="/usr/sbin/crond" hostname=? addr=? terminal=cron res=success'
type=USER_END msg=audit(1637856602.165:265): pid=7038 uid=0 auid=0 ses=6 subj=system_u:
system_r:crond_t:s0-s0:c0.c1023 msg='op=PAM:session_close grantors=pam_loginuid,pam_ke
yinit,pam_limits,pam_systemd acct="root" exe="/usr/sbin/crond" hostname=? addr=? termina
l=cron res=success'
type=USER_ACCT msg=audit(1637857201.302:266): pid=7273 uid=0 auid=4294967295 ses=429496
7295 subj=system_u:system_r:crond_t:s0-s0:c0.c1023 msg='op=PAM:accounting grantors=pam_
access,pam_unix,pam_localuser acct="root" exe="/usr/sbin/crond" hostname=? addr=? termi
nal=cron res=success'
type=CRED_ACQ msg=audit(1637857201.310:267): pid=7273 uid=0 auid=4294967295 ses=4294967
295 subj=system_u:system_r:crond_t:s0-s0:c0.c1023 msg='op=PAM:setcred grantors=pam_env,
pam_unix acct="root" exe="/usr/sbin/crond" hostname=? addr=? terminal=cron res=success'
type=LOGIN msg=audit(1637857201.311:268): pid=7273 uid=0 subj=system_u:system_r:crond_t
:s0-s0:c0.c1023 old-auid=4294967295 auid=0 tty=(none) old-ses=4294967295 ses=7 res=1
type=USER_START msg=audit(1637857201.541:269): pid=7273 uid=0 auid=0 ses=7 subj=system
_u:system_r:crond_t:s0-s0:c0.c1023 msg='op=PAM:session_open grantors=pam_loginuid,pam_ke
yinit,pam_limits,pam_systemd acct="root" exe="/usr/sbin/crond" hostname=? addr=? termin
al=cron res=success'
type=CRED_REFR msg=audit(1637857201.546:270): pid=7273 uid=0 auid=0 ses=7 subj=system_u
:system_r:crond_t:s0-s0:c0.c1023 msg='op=PAM:setcred grantors=pam_env,pam_unix acct="ro
ot" exe="/usr/sbin/crond" hostname=? addr=? terminal=cron res=success'
type=CRED_DISP msg=audit(1637857201.630:271): pid=7273 uid=0 auid=0 ses=7 subj=system_u
:system_r:crond_t:s0-s0:c0.c1023 msg='op=PAM:setcred grantors=pam_env,pam_unix acct="ro
```

Figure 3.23: рис.24

```

7295 subj=system u:system r:cron d:t:s0-s0:c0.c1023 msg='op=PAM:accounting grantors=pam
access,pam_unix,pam_localuser acct="root" exe="/usr/sbin/crond" hostname=? addr=? termi
nal=cron res=success'
type=CRED_ACQ msg=audit(1637857201.310:267): pid=7273 uid=0 auid=4294967295 ses=4294967
295 subj=system u:system r:cron d:t:s0-s0:c0.c1023 msg='op=PAM:setcred grantors=pam_env,
pam_unix acct="root" exe="/usr/sbin/crond" hostname=? addr=? terminal=cron res=success'
type=LOGIN msg=audit(1637857201.311:268): pid=7273 uid=0 subj=system u:system r:cron d:t
:s0-s0:c0.c1023 old-auid=4294967295 auid=0 tty=(none) old-ses=4294967295 ses=7 res=1
type=USER_START msg=audit(1637857201.541:269): pid=7273 uid=0 auid=0 ses=7 subj=system
u:system r:cron d:t:s0-s0:c0.c1023 msg='op=PAM:session_open grantors=pam_loginuid,pam_ke
yinit,pam_limits,pam_systemd acct="root" exe="/usr/sbin/crond" hostname=? addr=? termin
al=cron res=success'
type=CRED_REFR msg=audit(1637857201.546:270): pid=7273 uid=0 auid=0 ses=7 subj=system u
:system r:cron d:t:s0-s0:c0.c1023 msg='op=PAM:setcred grantors=pam_env,pam_unix acct="ro
ot" exe="/usr/sbin/crond" hostname=? addr=? terminal=cron res=success'
type=CRED_DISP msg=audit(1637857201.630:271): pid=7273 uid=0 auid=0 ses=7 subj=system u
:system r:cron d:t:s0-s0:c0.c1023 msg='op=PAM:setcred grantors=pam_env,pam_unix acct="ro
ot" exe="/usr/sbin/crond" hostname=? addr=? terminal=cron res=success'
type=USER_END msg=audit(1637857201.642:272): pid=7273 uid=0 auid=0 ses=7 subj=system u:
system r:cron d:t:s0-s0:c0.c1023 msg='op=PAM:session_close grantors=pam_loginuid,pam_key
init,pam_limits,pam_systemd acct="root" exe="/usr/sbin/crond" hostname=? addr=? termina
l=cron res=success'
type=SERVICE_START msg=audit(1637857401.456:273): pid=1 uid=0 auid=4294967295 ses=42949
67295 subj=system u:system r:init t:s0 msg='unit=httpd comm="systemd" exe="/usr/lib/sys
temd/systemd" hostname=? addr=? terminal=? res=success'
[root@abageeval abageeval]#

```

Figure 3.24: рис.25

20. Попробуем запустить Apache на прослушивание TCP-порта 81, заменив в файле /etc/httpd/conf/httpd.conf строчку Listen 80 на Listen 81. (рис.26).

```

#Listen 12.34.56.78:80
Listen 81

```

Figure 3.25: рис.26

21. Перезапустим Apache и попробуем обратиться к файлу через веб-сервер, введя в firefox адрес <http://127.0.0.1/test.html>. Из этого можно сделать предположение, что в списках портов, работающих с веб-сервером Apache, отсутствует порт 81. (рис.27-28).

```
[root@abageeval abageeval]# service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[root@abageeval abageeval]# 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 preset: disabled)
   Active: active (running) since 2021-11-25 19:23:21 MSK; 13s ago
     Docs: man:httpd(8)
           man:apachectl(8)
   Main PID: 7360 (httpd)
   Status: "Total requests: 0; Current requests/sec: 0; Current traffic: 0 B/sec"
     Tasks: 6
    CGroup: /system.slice/httpd.service
            └─7360 /usr/sbin/httpd -DFOREGROUND
              └─7364 /usr/sbin/httpd -DFOREGROUND
                └─7365 /usr/sbin/httpd -DFOREGROUND
                  └─7366 /usr/sbin/httpd -DFOREGROUND
                    └─7367 /usr/sbin/httpd -DFOREGROUND
                      └─7370 /usr/sbin/httpd -DFOREGROUND

ноя 25 19:23:20 abageeval.localdomain systemd[1]: Starting The Apache HTTP Server...
ноя 25 19:23:21 abageeval.localdomain httpd[7360]: AH00558: http: Could not reliably determine the server's fully qualified domain name, using abageeval.localdomain. Set the 'ServerName' directive globally to suppress this message
ноя 25 19:23:21 abageeval.localdomain systemd[1]: Started The Apache HTTP Server.
Hint: Some lines were ellipsized, use -l to show in full.
[root@abageeval abageeval]#
```

Figure 3.26: рис.27

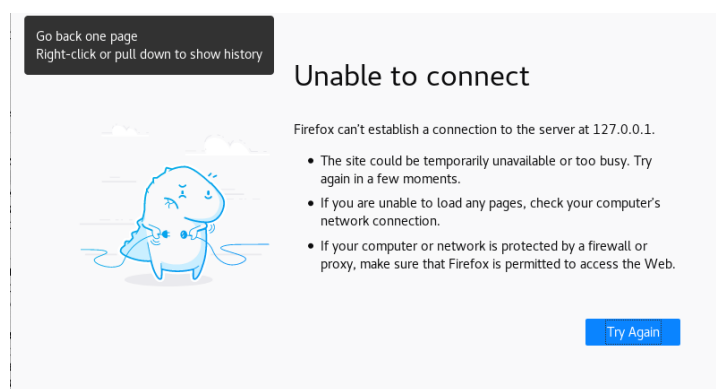


Figure 3.27: рис.28

22. Подтвердим свои догадки, просмотрев log-файлы. Во всех log-файлах появились записи, кроме /var/log/messages. (рис.29).

```
[root@abageeval abageeval]# tail -n1 /var/log/messages
Nov 25 19:23:21 abageeval systemd: Started The Apache HTTP Server.
[root@abageeval abageeval]# tail /var/log/httpd/error_log
[Thu Nov 25 19:23:21.370813 2021] [core:notice] [pid 7360] SELinux policy enabled; httpd running as context system_u:system_r:httpd_t:s0
[Thu Nov 25 19:23:21.378095 2021] [suexec:notice] [pid 7360] AH01232: suEXEC mechanism enabled (wrapper: /usr/sbin/suexec)
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using abageeval.localdomain. Set the 'ServerName' directive globally to suppress this message
[Thu Nov 25 19:23:21.427936 2021] [lbmethod_heartbeat:notice] [pid 7360] AH02282: No slotmem from mod_heartbeat
[Thu Nov 25 19:23:21.466048 2021] [mpm_prefork:notice] [pid 7360] AH00163: Apache/2.4.6 (CentOS) configured -- resuming normal operations
[Thu Nov 25 19:23:21.466090 2021] [core:notice] [pid 7360] AH00094: Command line: '/usr/sbin/httpd -D FOREGROUND'
[root@abageeval abageeval]# tail /var/log/httpd/access_log
[root@abageeval abageeval]#
```

Figure 3.28: рис.29

23. Чтобы подключиться к веб-серверу через порт 81, добавим его с помощью

команды `semanage port -a -t http_port_t -p tcp 81` После этого проверим список портов `semanage port -l | grep http_port_t`. (рис.32).

```
[root@abageeval abageeval]# semanage port -a -t http_port_t -p tcp 81
ValueError: Port tcp/81 уже определен
[root@abageeval abageeval]# 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@abageeval abageeval]#
```

Figure 3.29: рис.32

24. Попробуем теперь запустить веб-сервер еще раз. Добавив порт 81 в систему, Apache смог прослушать данный порт, в следствие чего получилось обратиться к файлу `test.html`. (рис.33-34).

```
[root@abageeval abageeval]# service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[root@abageeval abageeval]# 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 preset: disabled)
   Active: active (running) since Чт 2021-11-25 19:30:14 MSK; 22s ago
     Docs: man:httpd(8)
           man:apachectl(8)
  Process: 7556 ExecStop=/bin/kill -WINCH ${MAINPID} (code=exited, status=0/SUCCESS)
 Main PID: 7564 (httpd)
   Status: "Total requests: 0; Current requests/sec: 0; Current traffic: 0 B/sec"
    Tasks: 6
   CGroup: /system.slice/httpd.service
           └─7564 /usr/sbin/httpd -DFOREGROUND
             └─7565 /usr/sbin/httpd -DFOREGROUND
               └─7566 /usr/sbin/httpd -DFOREGROUND
                 └─7567 /usr/sbin/httpd -DFOREGROUND
                   └─7568 /usr/sbin/httpd -DFOREGROUND
                     └─7569 /usr/sbin/httpd -DFOREGROUND

ноя 25 19:30:14 abageeval.localdomain systemd[1]: Stopped The Apache HTTP Server.
ноя 25 19:30:14 abageeval.localdomain systemd[1]: Starting The Apache HTTP Server...
ноя 25 19:30:14 abageeval.localdomain httpd[7564]: AH00558: httpd: Could not rel...
```

Figure 3.30: рис.33

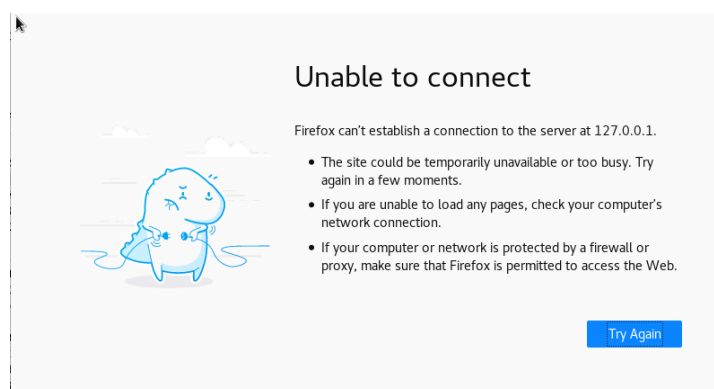


Figure 3.31: рис.34

25. Вернем обратно контекст `httpd_sys_content_t` к файлу `/var/www/html/test.html`. После этого вновь попробуем получить доступ к файлу через веб-сервер, введя в `firefox` адрес `http://127.0.0.1:81/test.html`. (рис.35-36).

```
[root@abageeva1 abageeva1]# chcon -t httpd_sys_content_t /var/www/html/test.html
[root@abageeva1 abageeva1]#
```

Figure 3.32: рис.35

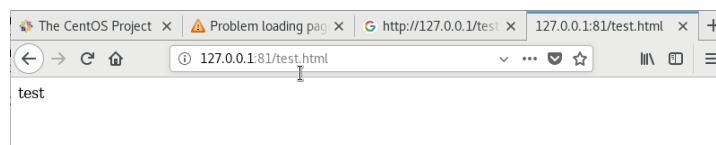


Figure 3.33: рис.36

26. Исправим обратно конфигурационный файл Apache, вернув `Listen 80` (рис.37).

```
GNU nano 2.3.1 Файл: /etc/httpd/conf/httpd.conf Изменён

# 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 specific IP addresses as shown below to
# prevent Apache from glomming onto all bound IP addresses.
#
#Listen 12.34.56.78:80
Listen 80

#
# 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 foo_module modules/mod_foo.so

^G Помощь ^O Записать ^R ЧитФайл ^Y ПредСтр ^K Вырезать ^C ТекПозиц
^X Выход ^_ Выровнять ^W Поиск ^V СледСтр ^U ОтмВырезк ^T Словарь
```

Figure 3.34: рис.37

27. Удалим привязку `http_port_t` к 81 порту (рис.38).
28. Проверим, что порт 81 удален. (рис.38).
29. Удалим файл `/var/www/html/test.html` (рис.38).

```

[root@abageeval abageeval]# semanage port -d -t http_port_t -p tcp 81
^[[AValueError: Порт tcp/81 определен на уровне политики и не может быть удален
[root@abageeval abageeval]# 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@abageeval abageeval]# rm /var/www/html/test.html
rm: удалить обычный файл «/var/www/html/test.html»? y
[root@abageeval abageeval]# █

```

Figure 3.35: рис.38

4 Выводы

Я развила навыки администрирования ОС Linux, получила первое практическое знакомство с технологией SELinux, а также проверила работу SELinux на практике совместно с веб-сервером Apache.