

Лабораторная работа №9

Отчет

Славинский Владислав Вадимович

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	24
4	Ответы на контрольные вопросы	25

Список иллюстраций

2.1	Просмотр информации о состоянии SELinux с помощью root прав .	7
2.2	Режим работы SELinux	7
2.3	Изменения режима работы	8
2.4	Изменение файла и перезапуск системы	8
2.5	Перезагрузки и проверка статуса SELinux	9
2.6	Попытка переключить режим работы SELinux	9
2.7	Редактирование файла и перезагрузка системы	10
2.8	Перезагрузка системы	11
2.9	Проверка статуса	12
2.10	Просмотр контекста безопасности файла	13
2.11	Копирование файла в домашний каталог	13
2.12	Перезапись файла из домашнего каталога	14
2.13	Исправление контекста безопасности	14
2.14	Массовое исправление контекста безопасности	15
2.15	Перемаркированные сообщения на перезагрузке	15
2.16	Установка ПО	16
2.17	Создание нового хранилища для файлов web-сервера	16
2.18	Вставка текста	17
2.19	Редактирование файла httpd.conf	17
2.20	Редактирование файла httpd.conf	18
2.21	Запуск веб-сервера и службы	18
2.22	Запуск	19
2.23	Применение новой метки контекста	19
2.24	Восстановление контекста безопасности	20
2.25	Проверка веб-сервера	20
2.26	Список переключателей SELinux	21
2.27	Список переключателей	21
2.28	Изменения значения переключателя	22
2.29	Просмотр списка переключателей	22
2.30	Изменение постоянного значения переключателя	23

Список таблиц

1 Цель работы

Получить навыки работы с контекстом безопасности и политиками SELinux.

2 Выполнение лабораторной работы

Запускаем терминал в режиме суперпользователя через su - и посмотрим текущую информацию о состоянии SELinux: `sestatus -v`. 1) SELinux status: enabled - это строка означает, что SELinux активирован. 2) SELinux mount - SELinux смонтирована в каталоге `/sys/fs/selinux`, SELinux root directory - корневой каталог конфигурации SELinux находится в `/etc/selinux`, 3) Loaded policy name - загружена политика безопасности типа `targeted`, которая защищает только определенные системные процессы, 4) Current mode: enforcing - принудительный режим работы, SELinux активно блокирует действия, нарушающие политику безопасности, 5) Mode from config file: enforcing - режим конфигурационного файла установлен как `enforcing`, что означает сохранение этого режима после перезагрузки, 6) Policy MLS status: enabled - поддержка многоуровневой безопасности активирована (Multi-Level-Security), 7) Policy deny_unknown status: allowed - неизвестные действия по умолчанию разрешены, 8) Memory protection checking: actual (secure) - проверка защиты памяти выполняется на безопасном уровне, 9) Max kernel policy version: 33 - максимальная поддерживаемая версия политики ядра 33, 10) Current context - Текущий процесс (терминал) работает в неограниченном контексте с высоким уровнем привелегий, 11) Init context - процесс `init` (родительский процесс системы) работает в соответствующем контексте, 12) `/usr/sbin/sshd` - SSH демон работает в правильном контексте безопасности для SSH службы, 13) `/etc/passwd`, `/etc/shadow` - имеют правильные контексты для файлов с паролями, 14) `/bin/bash`, `/bin/login` - имеют контексты исполняемых файлов оболочки и входа в систему, 15) `/sbin/agetty`, `/sbin/init` - имеют соответствующие контексты

для системных служб, 16) /usr/sbin/sshd - имеет правильный контекст для SSH демона. (рис. 2.1)

```
[slavinskiyvv@slavinskiyvv ~]$ su -
Password:
[root@slavinskiyvv ~]# sestatus -v
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

Process contexts:
Current context:               unconfined_u:unconfined_r:unconfined_t:s0-s0:c0
Init context:                  system_u:system_r:init_t:s0
/usr/sbin/sshd                 system_u:system_r:sshd_t:s0-s0:c0,c1023

File contexts:
Controlling terminal:          unconfined_u:object_r:user_devpts_t:s0
/etc/passwd                    system_u:object_r:passwd_file_t:s0
/etc/shadow                    system_u:object_r:shadow_t:s0
/bin/bash                     system_u:object_r:shell_exec_t:s0
/bin/login                     system_u:object_r:login_exec_t:s0
/bin/sh                        system_u:object_r:bin_t:s0 -> system_u:object_r
                               t:s0
/sbin/agetty                   system_u:object_r:getty_exec_t:s0
/sbin/init                     system_u:object_r:bin_t:s0 -> system_u:object_r
                               t:s0
/usr/sbin/sshd                 system_u:object_r:sshd_exec_t:s0
[root@slavinskiyvv ~]#
```

Рис. 2.1: Просмотр информации о состоянии SELinux с помощью root прав

Посмотрим, в каком режиме работает SELinux: `getenforce`. Видим, что работает в режиме `enforcing` (в режиме принудительного исполнения). (рис. 2.2)

```
[root@slavinskiyvv ~]# getenforce
Enforcing
[root@slavinskiyvv ~]#
```

Рис. 2.2: Режим работы SELinux

Изменим режим работы SELinux на разрешающий (Permissive): `setenforce 0`. Потом введем снова `getenforce`. (рис. 2.3)

```

/usr/sbin/sshd                                system_u:0
[root@slavinskiyvv ~]# getenforce
Enforcing
[root@slavinskiyvv ~]# setenforce 0
[root@slavinskiyvv ~]# getenforce
Permissive
[root@slavinskiyvv ~]# █

```

Рис. 2.3: Изменения режима работы

В файле `/etc/sysconfig/selinux` с помощью редактора установим `SELINUX=disabled` и перезапустим систему. (рис. 2.4)

```

# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - No SELinux policy is loaded.
# See also:
# https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/9
#
# NOTE: Up to RHEL 8 release included, SELINUX=disabled would also
# fully disable SELinux during boot. If you need a system with SELinux
# fully disabled instead of SELinux running with no policy loaded, you
# need to pass selinux=0 to the kernel command line. You can use grubby
# to persistently set the bootloader to boot with selinux=0:
#
#   grubby --update-kernel ALL --args selinux=0
#
# To revert back to SELinux enabled:
#
#   grubby --update-kernel ALL --remove-args selinux
#
SELINUX=disabled
# SELINUXTYPE= can take one of these three values:
#   targeted - Targeted processes are protected,
#   minimum - Modification of targeted policy. Only selected processes a
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted

```

Рис. 2.4: Изменение файла и перезапуск системы

После перезагрузки запустим терминал и получим полномочия администрато-

ра.Посмотрим статус SELinux: `getenforce`. Мы видим, что SELinux теперь отключён.
(рис. 2.5)

```
[slavinskiyvv@slavinskiyvv ~]$ su -  
Password:  
[root@slavinskiyvv ~]# getenforce  
Disabled  
[root@slavinskiyvv ~]#
```

Рис. 2.5: Перезагрузки и проверка статуса SELinux

Попробуем переключить режим работы SELinux: `setenforce 1`. Мы не можем переключаться между отключённым и принудительным режимом без перезагрузки системы.(рис. 2.6)

```
[root@slavinskiyvv ~]# getenforce  
Disabled  
[root@slavinskiyvv ~]# setenforce 1  
setenforce: SELinux is disabled  
[root@slavinskiyvv ~]#
```

Рис. 2.6: Попытка переключить режим работы SELinux

Откроем файл `/etc/sysconfig/selinux` с помощью редактора и установим:

SELINUX=enforcing. Затем перезагрузим систему. (рис. 2.7)

```
# disabled - no SELinux policy is loaded.
# See also:
# https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/SELinux_tutorial/sectors7_selinux_tutorial_disabling_selinux.html
#
# NOTE: Up to RHEL 8 release included, SELINUX=disabled would
# fully disable SELinux during boot. If you need a system with
# fully disabled instead of SELinux running with no policy loaded,
# need to pass selinux=0 to the kernel command line. You can
# to persistently set the bootloader to boot with selinux=0:
#
# grubby --update-kernel ALL --args selinux=0
#
# To revert back to SELinux enabled:
#
# grubby --update-kernel ALL --remove-args selinux
#
SELINUX=enforcing
# SELINUXTYPE= can take one of these three values:
#   targeted - Targeted processes are protected,
#   minimum - Modification of targeted policy. Only selected processes are protected.
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

Рис. 2.7: Редактирование файла и перезагрузка системы

Во время загрузки системы мы получаем предупреждающее сообщение (Relabeling could take a very long time) о необходимости восстановления меток SELinux, это занимает некоторое время, а также требуется дополнительная перезагрузка системы. (рис. 2.8)

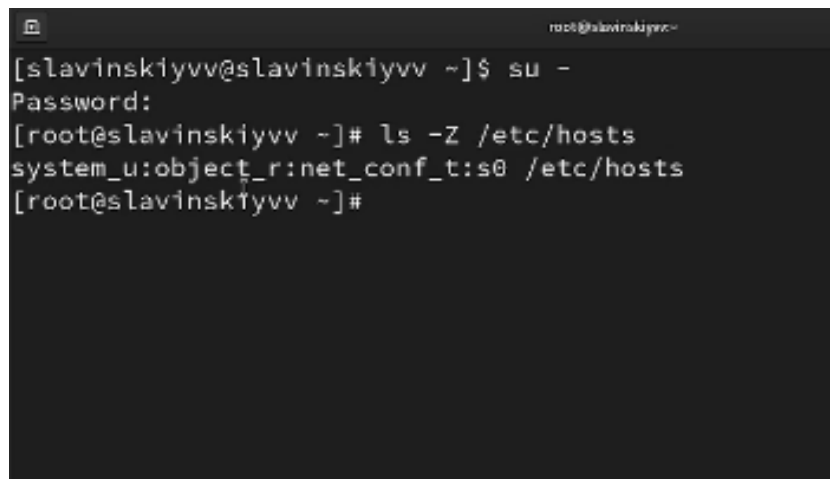

```
slavinskiyvv@slavinskiyvv:~$ sestatus -v
SELinux status: enabled
SELinuxfs mount: /sys/fs/selinux
SELinux root directory: /etc/selinux
Loaded policy name: targeted
Current mode: enforcing
Mode from config file: enforcing
Policy MLS status: enabled
Policy deny_unknown status: allowed
Memory protection checking: actual (secure)
Max kernel policy version: 33

Process contexts:
Current context: unconfined_u:unconfined_r:unc
s0:c0.c1023
Init context: system_u:system_r:init_t:s0

File contexts:
Controlling terminal: unconfined_u:object_r:user_de
/etc/passwd system_u:object_r:passwd_file
/etc/shadow system_u:object_r:shadow_t:s0
/bin/bash system_u:object_r:shell_exec_
/bin/login system_u:object_r:login_exec_
/bin/sh system_u:object_r:bin_t:s0 ->
ect_r:shell_exec_t:s0
/sbin/agetty system_u:object_r:getty_exec_
/sbin/init system_u:object_r:bin_t:s0 ->
ect_r:init_exec_t:s0
/usr/sbin/sshd system_u:object_r:sshd_exec_t
[slavinskiyvv@slavinskiyvv ~]$
```

Рис. 2.9: Проверка статуса

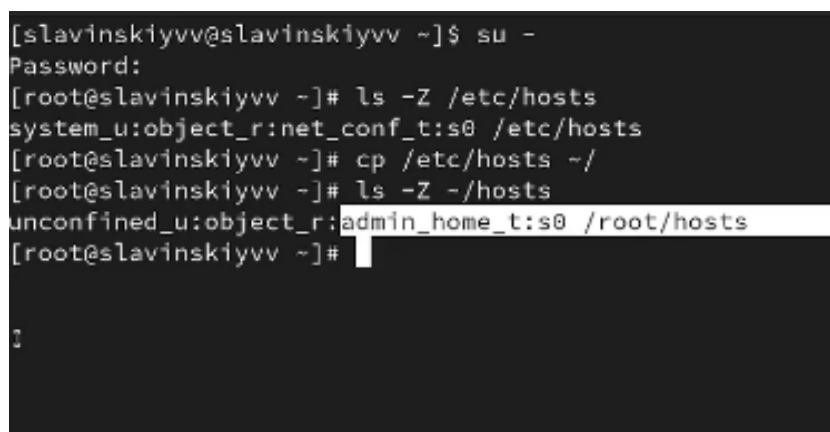
Запустим терминал и получим полномочия администратора. Затем посмотрим контекст безопасности файла /etc/hosts: `ls -Z /etc/hosts`. Видим, что присутствует метка `net_conf_t`, что указывает на тип файла сетевой конфигурации. (рис. 2.10)

A terminal window with a dark background. The prompt is [slavinskiyv@slavinskiyv ~]\$. The user enters 'su -' and provides a password. The prompt changes to [root@slavinskiyv ~]#. The user enters 'ls -Z /etc/hosts'. The output is 'system_u:object_r:net_conf_t:s0 /etc/hosts'. The prompt returns to [root@slavinskiyv ~]#.

```
[slavinskiyv@slavinskiyv ~]$ su -
Password:
[root@slavinskiyv ~]# ls -Z /etc/hosts
system_u:object_r:net_conf_t:s0 /etc/hosts
[root@slavinskiyv ~]#
```

Рис. 2.10: Просмотр контекста безопасности файла

Скопируем файл /etc/hosts в домашний каталог: `cp /etc/hosts ~/`, проверим контекст файла ~/hosts: `ls -Z ~/hosts`. Поскольку копирование считается созданием нового файла, то параметр контекста в файле ~/hosts, расположенном в домашнем каталоге, станет `admin_home_t`. (рис. 2.11)

A terminal window with a dark background. The prompt is [slavinskiyv@slavinskiyv ~]\$. The user enters 'su -' and provides a password. The prompt changes to [root@slavinskiyv ~]#. The user enters 'ls -Z /etc/hosts'. The output is 'system_u:object_r:net_conf_t:s0 /etc/hosts'. The user enters 'cp /etc/hosts ~/'. The prompt returns to [root@slavinskiyv ~]#. The user enters 'ls -Z ~/hosts'. The output is 'unconfined_u:object_r:admin_home_t:s0 /root/hosts'. The prompt returns to [root@slavinskiyv ~]#.

```
[slavinskiyv@slavinskiyv ~]$ su -
Password:
[root@slavinskiyv ~]# ls -Z /etc/hosts
system_u:object_r:net_conf_t:s0 /etc/hosts
[root@slavinskiyv ~]# cp /etc/hosts ~/
[root@slavinskiyv ~]# ls -Z ~/hosts
unconfined_u:object_r:admin_home_t:s0 /root/hosts
[root@slavinskiyv ~]#
```

Рис. 2.11: Копирование файла в домашний каталог

Попытаемся перезаписать существующий файл hosts из домашнего каталога в каталог /etc: `mv ~/hosts /etc`. И убедимся, что тип контекста по-прежнему установлен на `admin_home_t`: `ls -Z /etc/hosts`. (рис. 2.12)

```

[slavinskiyv@slavinskiyv ~]$ su -
Password:
[root@slavinskiyv ~]# ls -Z /etc/hosts
system_u:object_r:net_conf_t:s0 /etc/hosts
[root@slavinskiyv ~]# cp /etc/hosts ~/
[root@slavinskiyv ~]# ls -Z ~/hosts
unconfined_u:object_r:admin_home_t:s0 /root/hosts
[root@slavinskiyv ~]# mv ~/hosts /etc
mv: overwrite '/etc/hosts'? y
[root@slavinskiyv ~]# ls -Z /etc/hosts
unconfined_u:object_r:admin_home_t:s0 /etc/hosts
[root@slavinskiyv ~]#

```

Рис. 2.12: Перезапись файла из домашнего каталога

Исправим контекст безопасности: `restorecon -v /etc/hosts`. Опция `-v` покажет нам процесс изменения. И проверим, что тип контекста изменился: `ls -Z /etc/hosts`. (рис. 2.13)

```

[root@slavinskiyv ~]# cp /etc/hosts ~/
[root@slavinskiyv ~]# ls -Z ~/hosts
unconfined_u:object_r:admin_home_t:s0 /root/hosts
[root@slavinskiyv ~]# mv ~/hosts /etc
mv: overwrite '/etc/hosts'? y
[root@slavinskiyv ~]# ls -Z /etc/hosts
unconfined_u:object_r:admin_home_t:s0 /etc/hosts
[root@slavinskiyv ~]# restorecon -v /etc/hosts
Relabeled /etc/hosts from unconfined_u:object_r:admin_home_t:s0 to unconfined_u:object_r:net_conf_t:s0
[root@slavinskiyv ~]# ls -Z /etc/hosts
unconfined_u:object_r:net_conf_t:s0 /etc/hosts
[root@slavinskiyv ~]#

```

Рис. 2.13: Исправление контекста безопасности

Для массового исправления контекста безопасности на файловой системе введем `touch /.autorelabel` и перезагрузим систему. Во время перезапуска нажмем клавишу `esc` чтобы мы увидели загрузочные сообщения. (рис. 2.14)

```
[root@slavinskiyvv ~]# mv ~/hosts /etc
mv: overwrite '/etc/hosts'? y
[root@slavinskiyvv ~]# ls -Z /etc/hosts
unconfined_u:object_r:admin_home_t:s0 /etc/hosts
[root@slavinskiyvv ~]# restorecon -v /etc/hosts
Relabeled /etc/hosts from unconfined_u:object_r:admin_home_t:s0 to
unconfined_u:object_r:net_conf_t:s0
[root@slavinskiyvv ~]# ls -Z /etc/hosts
unconfined_u:object_r:net_conf_t:s0 /etc/hosts
[root@slavinskiyvv ~]# touch /.autorelabel
[root@slavinskiyvv ~]# reboot
```

Рис. 2.14: Массовое исправление контекста безопасности

Вот какие сообщения выводятся при перезагрузке.(рис. 2.15)

```
Starting Create Static Device Nodes in /dev...
[ OK ] Mounted POSIX Character File System.
[ OK ] Mounted Kernel Configuration File System.
[ OK ] Finished Waiting for UMS wireset, macports etc. using dbusctl or program polling.
[ OK ] Finished Load/Save OS Hardwired.
[ OK ] Finished Enabling All udev Devices.
[ OK ] Finished Apply Kernel Variables.
Starting Wait for udev To Complete Device Initialization...
[ OK ] Finished Flush Journal to Persistent Storage.
[ OK ] Finished Create Static Device Nodes in /dev.
Starting Rule-based Manager for Device Events and Files...
[ OK ] Started Rule-based Manager for Device Events and Files.
Starting Load Kernel Module configfs...
Starting Load Kernel Module fuse...
[ OK ] Finished Load Kernel Module configfs.
[ OK ] Finished Load Kernel Module fuse.
[ OK ] Started udevd.service udevd -- udevd activation event 1.
[ 4.155235] selinux: autorelabel[7001]: *** Warning -- SELinux targeted policy reload is required.
[ 4.156491] selinux: autorelabel[7001]: *** Relabeling could take a very long time, depending on file
[ 4.156493] selinux: autorelabel[7001]: *** system size and speed of hard drives.
[ 4.282591] selinux: autorelabel[7001]: Running: /sbin/tidfiles -T 0 restore
```

Рис. 2.15: Перемаркированные сообщения на перезагрузке

Запустим терминал в режиме администратора. Затем установим необходимое программное обеспечение: `dnf -y install httpd`, `dnf -y install lynx`.(рис. 2.16)

```
root@slavinskiyvv:~# su -
Password:
[root@slavinskiyvv ~]# dnf -y install httpd
Last metadata expiration check: 2:48:22 ago on Sat 01 Nov 2025 03:41:42 PM MSK.
Package httpd-2.4.62-4.el9_6.4.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@slavinskiyvv ~]# dnf -y install lynx
```

Рис. 2.16: Установка ПО

Создадим новое хранилище для файлов web-сервера: `mkdir /web`. Создадим файл `index.html` в каталоге с контентом веб-сервера: `cd /web, touch index.html`. (рис. 2.17)

```
Installed:
  lynx-2.8.9-20.el9.x86_64

Complete!
[root@slavinskiyvv ~]# mkdir /web
[root@slavinskiyvv ~]# cd /web
[root@slavinskiyvv web]# touch index.html
[root@slavinskiyvv web]#
```

Рис. 2.17: Создание нового хранилища для файлов web-сервера

Поместим в этот файл следующий текст: `Welcome to my web-server`. (рис. 2.18)

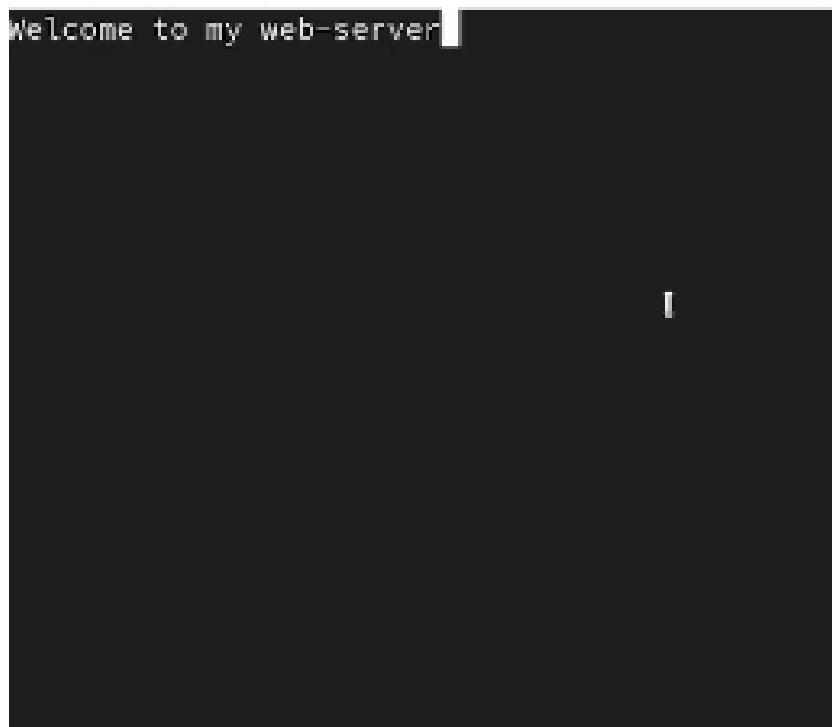


Рис. 2.18: Вставка текста

В файле `/etc/httpd/conf/httpd.conf` закомментируем строку `DocumentRoot "/var/www/html"` и ниже добавим строку `DocumentRoot "/web"`. (рис. 2.19)

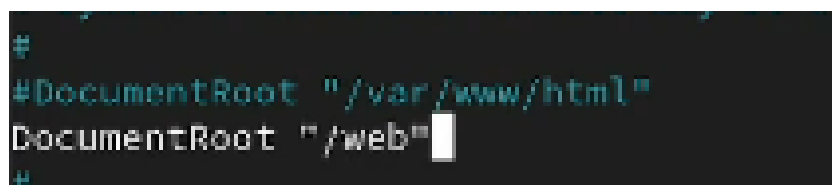


Рис. 2.19: Редактирование файла `httpd.conf`

Затем в этом же файле ниже закомментируем раздел `<Directory "/var/www">` `AllowOverride None` `Require all granted` и добавим следующий раздел, определяющий правила доступа: `<Directory "/web">` `AllowOverride None` `Require all granted`. (рис. 2.20)

```

#DocumentRoot "/var/www/html"
DocumentRoot "/web"
#
# Relax access to content within /var/www.
#
#<Directory "/var/www">
#     AllowOverride None
#     # Allow open access:
#     Require all granted
#</Directory>

<Directory "/web">
AllowOverride None
Require all granted
</Directory>

# Further relax access to the default document root:
<Directory "/var/www/html">
#
# Possible values for the Options directive are
# or any combination of:

```

Рис. 2.20: Редактирование файла httpd.conf

Запустим веб-сервер и службу httpd: `systemctl start httpd`, `systemctl enable httpd`. (рис. 2.21)

```

complete!
root@slavinskiyvv ~]# mkdir /web
root@slavinskiyvv ~]# cd /web
root@slavinskiyvv web]# touch index.html
root@slavinskiyvv web]# nano index.html
root@slavinskiyvv web]# nano /etc/httpd/conf/httpd.conf
root@slavinskiyvv web]# systemctl start httpd
root@slavinskiyvv web]# systemctl enable httpd
root@slavinskiyvv web]#

```

Рис. 2.21: Запуск веб-сервера и службы

В терминале под учётной записью своего пользователя при обращении к веб-серверу в текстовом браузере lynx введем: `lynx http://localhost`. Мы увидим веб-страницу Red Hat по умолчанию, а не содержимое только что созданного файла `index.html`. (рис. 2.22)

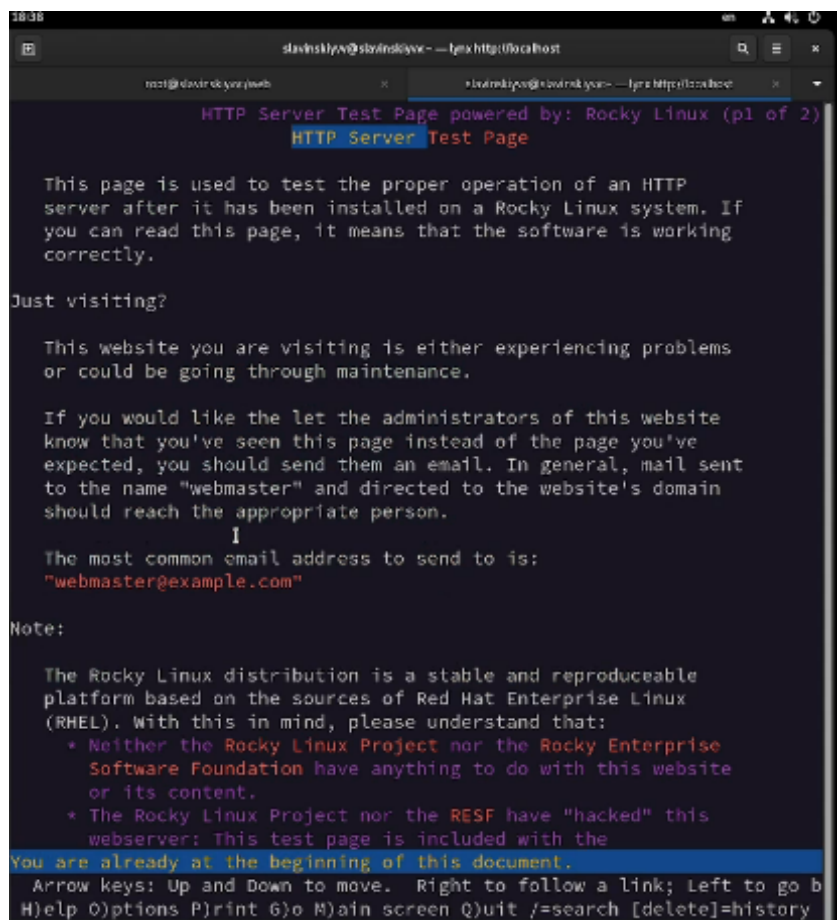


Рис. 2.22: Запуск

В терминале с полномочиями администратора применим новую метку контекста к /web: `semanage fcontext -a -t httpd_sys_content_t "/web(/.*)?"`. (рис. 2.23)

```
Complete!
[root@slavinskiy ~]# mkdir /web
[root@slavinskiy ~]# cd /web
[root@slavinskiy web]# touch index.html
[root@slavinskiy web]# nano index.html
[root@slavinskiy web]# nano /etc/httpd/conf/httpd.conf
[root@slavinskiy web]# systemctl start httpd
[root@slavinskiy web]# systemctl enable httpd
[root@slavinskiy web]# semanage fcontext -a -t httpd_sys_content_t "/web(/.*)?"
[root@slavinskiy web]#
```

Рис. 2.23: Применение новой метки контекста

Восстановим контекст безопасности: `restorecon -R -v /web`. (рис. 2.24)

```

[root@slavinskiyvv web]# semanage fcontext -a -t httpd_sys_content_t "/web(/.*)?"
[root@slavinskiyvv web]# restorecon -R -v /web
Relabeled /web from unconfined_u:object_r:default_t:s0 to unconfined_u:object_r:httpd_sys_content_t:s0
Relabeled /web/index.html from unconfined_u:object_r:default_t:s0 to unconfined_u:object_r:httpd_sys_content_t:s0
[root@slavinskiyvv web]#

```

Рис. 2.24: Восстановление контекста безопасности

В терминале под учётной записью своего пользователя снова обратимся к веб-серверу: `lynx http://localhost`. У нас ничего не произошло. Значит перезапускаем систему и опять обращаемся к веб-серверу. Как видим, у нас все получилось. (рис. 2.25)

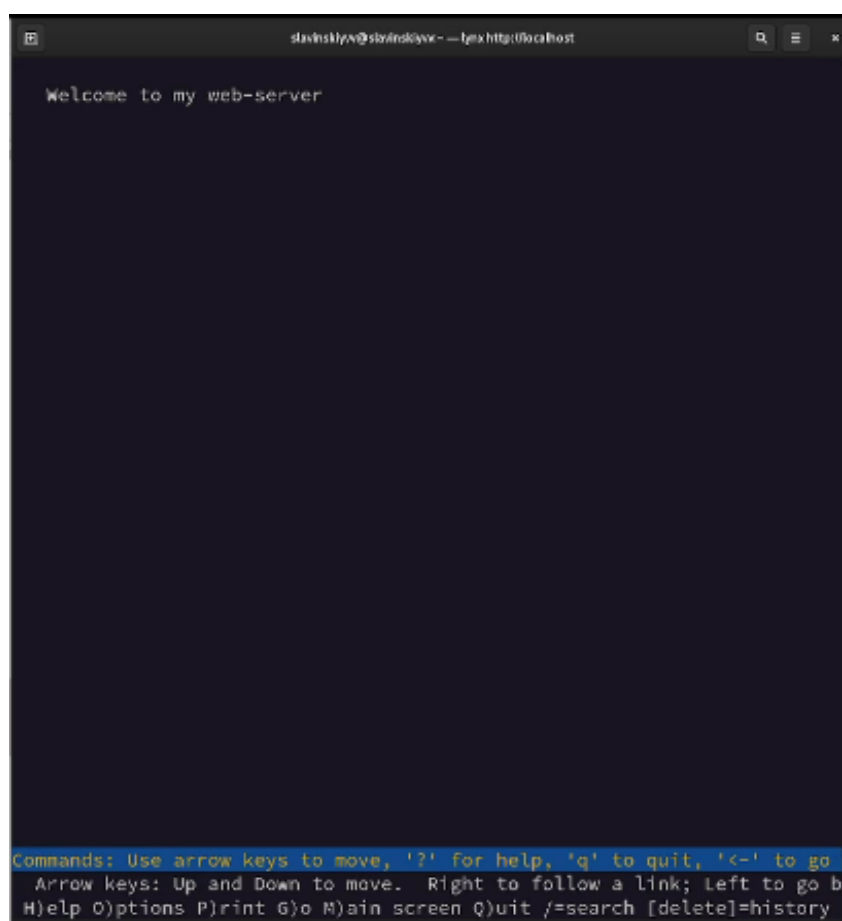


Рис. 2.25: Проверка веб-сервера

Через полномочия администратора посмотрим список переключателей SELinux для службы ftp: `getsebool -a | grep ftp` (рис. 2.26)

```
[slavinskiyv@slavinskiyv ~]$
[slavinskiyv@slavinskiyv ~]$ su -
Password:
[root@slavinskiyv ~]# getsebool -a | grep ftp
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
httpd_can_connect_ftp --> off
httpd_enable_ftp_server --> off
tftp_anon_write --> off
tftp_home_dir --> off
[root@slavinskiyv ~]#
```

Рис. 2.26: Список переключателей SELinux

Для службы ftpd_anon посмотрим список переключателей: `semanage boolean -l | grep ftpd_anon`. Первое значение off - текущее состояние выполнения времени, второе значение off - постоянное состояние. Ftpd_anon_write разрешает или запрещает анонимным пользователям FTP выполнять операции записи. (рис. 2.27)

```
ftpd_use_passive_mode --> off
httpd_can_connect_ftp --> off
httpd_enable_ftp_server --> off
tftp_anon_write --> off
tftp_home_dir --> off
[root@slavinskiyv ~]# semanage boolean -l | grep ftpd_anon
ftpd_anon_write (off , off) Allow ftpd to anon write
[root@slavinskiyv ~]#
```

Рис. 2.27: Список переключателей

Изменим текущее значение переключателя для службы ftpd_anon_write с off на on: setsebool ftpd_anon_write on. Повторно посмотрим список переключателей SELinux для службы ftpd_anon_write: getsebool ftpd_anon_write.(рис. 2.28)

```
[root@slavinskiyvv ~]# semanage boolean -l | grep ftpd_anon
ftpd_anon_write (off , off) Allow ftpd to an
[root@slavinskiyvv ~]# setsebool ftpd_anon_write on
[root@slavinskiyvv ~]# getsebool ftpd_anon_write
ftpd_anon_write --> on
[root@slavinskiyvv ~]#
```

Рис. 2.28: Изменения значения переключателя

Посмотрим список переключателей: semanage boolean -l | grep ftpd_anon. Видим, что настройка времени выполнения включена, но постоянная настройка выключена(рис. 2.29)

```
[root@slavinskiyvv ~]# setsebool ftpd_anon_write on
[root@slavinskiyvv ~]# getsebool ftpd_anon_write
ftpd_anon_write --> on
[root@slavinskiyvv ~]# semanage boolean -l | grep ftpd_anon
ftpd_anon_write 1 (on , off) Allow ftpd to anon write
[root@slavinskiyvv ~]#
```

Рис. 2.29: Просмотр списка переключателей

Изменим постоянное значение переключателя для службы ftpd_anon_write с off на on: setsebool -P ftpd_anon_write on. Посмотрим список переключателей: semanage boolean -l | grep ftpd_anon. Теперь у нас ftpd_anon_write полностью включен. Оба значения установлены на on: 1) включены состояние во время выполнения и постоянное состояние после перезагрузки. Теперь анонимные пользователи FTP могут выполнять операции записи на сервер. (рис. 2.30)

```

ftpd_anon_write (on , off) Allow ftpd to anon write
[root@slavinskiyvv ~]# setsebool ftpd_anon_write on
[root@slavinskiyvv ~]# getsebool ftpd_anon_write
ftpd_anon_write --> on
[root@slavinskiyvv ~]# semanage boolean -l | grep ftpd_anon
ftpd_anon_write (on , off) Allow ftpd to anon write
[root@slavinskiyvv ~]# setsebool -P ftpd_anon_write on
[root@slavinskiyvv ~]# semanage boolean -l | grep ftpd_anon
ftpd_anon_write (on , on) Allow ftpd to anon write
[root@slavinskiyvv ~]#

```

Рис. 2.30: Изменение постоянного значения переключателя

3 Выводы

В ходе выполнения лабораторной работы были получены навыки работы с контекстом безопасности и политиками SELinux.

4 Ответы на контрольные вопросы

1. `setenforce 0`
2. `getsebool -a`
3. `setroubleshoot`
4. `semanage fcontext -a -t , restorecon -R -v`
5. `/etc/sysconfig/selinux`
6. `/var/log/messages`
7. `seinfo -t | grep ftp`
8. `setenforce 0`, если в режиме `permissive` проблема исчезает, то она связана с политиками SELinux. Если проблема остается, то причина в другой конфигурации