Отчёт по лабораторной работе №9

Управление SELinux

Сулейм Гамбердов

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

Получить навыки работы с контекстом безопасности и политиками SELinux.

2 Ход выполнения

2.1 Управление режимами SELinux

1. В терминале получены права администратора с помощью команды **su**.
   Это позволило выполнять системные команды, требующие привилегий *root*.

2. Проверено текущее состояние SELinux командой **sestatus -v**.
   Из вывода видно:

   - **SELinux status: enabled** — механизм безопасности SELinux включён.

   - **Loaded policy name: targeted** — используется политика *targeted*, защищающая только ключевые системные службы.

   - **Current mode: enforcing** — активен режим принудительного применения политик.

   - **Mode from config file: enforcing** — конфигурация также задаёт принудительный режим.

   - **Policy MLS status: enabled** — включена многоуровневая защита.

   - Разделы *Process contexts* и *File contexts* показывают текущие контексты безопасности процессов и файлов, например:

`/usr/sbin/sshd` — `system_u:system_r:sshd_t:s0-s0:c0.c1023`.

Это означает, что служба **sshd** выполняется в собственном домене безопасности.

```
root@sigamberdov:/home/sigamberdov# 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.c1023
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:shell_exec_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:init_exec_t:s0
/usr/sbin/sshd                  system_u:object_r:sshd_exec_t:s0
root@sigamberdov:/home/sigamberdov# getenforce
Enforcing
root@sigamberdov:/home/sigamberdov# setenforce 0
root@sigamberdov:/home/sigamberdov# getenforce
Permissive
root@sigamberdov:/home/sigamberdov#
```

Рис. 2.1: Вывод команды sestatus -v

3. Проверен режим SELinux с помощью **getenforce**.

   Вывод *Enforcing* подтвердил активный режим принудительного применения политик.

4. Режим SELinux изменён на разрешающий (**Permissive**) командой **setenforce 0**.

   Повторная проверка **getenforce** показала *Permissive*.

   В этом режиме нарушения политик только фиксируются в журнале, но не блокируются.

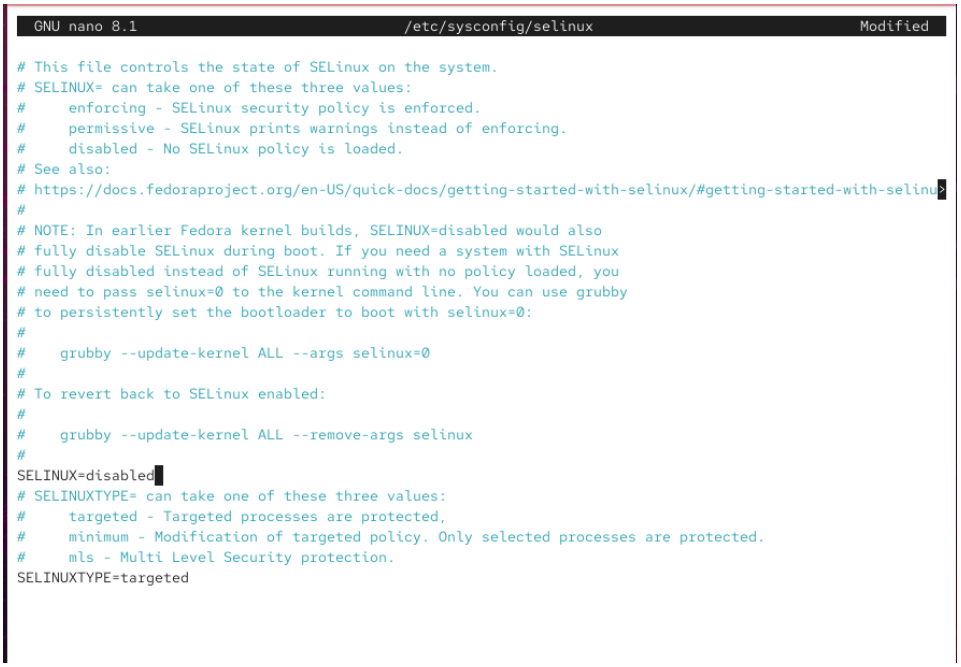

Рис. 2.2: Изменение режима SELinux на Permissive

5. Для полного отключения SELinux открыт файл **/etc/sysconfig/selinux** с помощью редактора *nano* и изменена строка **SELINUX=disabled**.
   После сохранения изменений система была перезагружена.

```
sigamberdov@sigamberdov:~$ su
Password:
root@sigamberdov:/home/sigamberdov# getenforce
Disabled
root@sigamberdov:/home/sigamberdov# setenforce 1
setenforce: SELinux is disabled
root@sigamberdov:/home/sigamberdov#
```

Рис. 2.3: Отключение SELinux в конфигурационном файле

6. После перезапуска и входа под пользователем root команда **getenforce** показала *Disabled*, что означает полное отключение SELinux.

Рис. 2.4: SELinux отключён

7. Попытка включить SELinux командой **setenforce 1** завершилась сообщением *SELinux is disabled*.

   Это подтверждает, что между состояниями *disabled* и *enforcing* невозможно переключаться без перезагрузки системы.

8. Для возврата к принудительному режиму SELinux снова открыт файл **/etc/sysconfig/selinux** и установлено значение **SELINUX=enforcing**.

   После этого система была перезагружена.

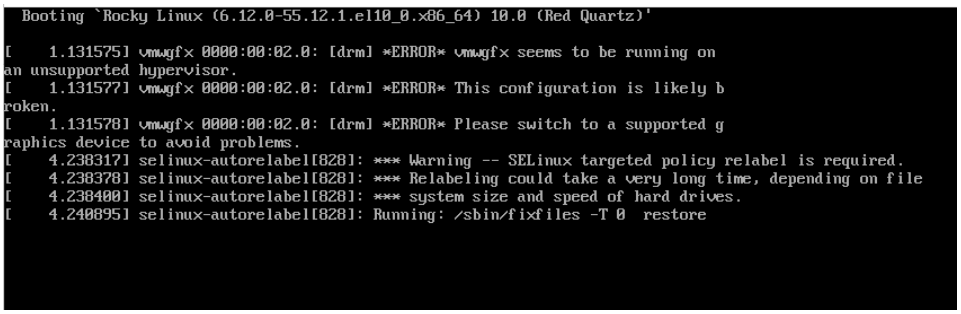

Рис. 2.5: Включение enforcing-режима в конфигурации

9. Во время загрузки появилось предупреждение о необходимости восстановления меток SELinux.

   Сообщение **“SELinux targeted policy relabel is required”** информировало, что выполняется автоматическая перемаркировка файлов, что может занять некоторое время.

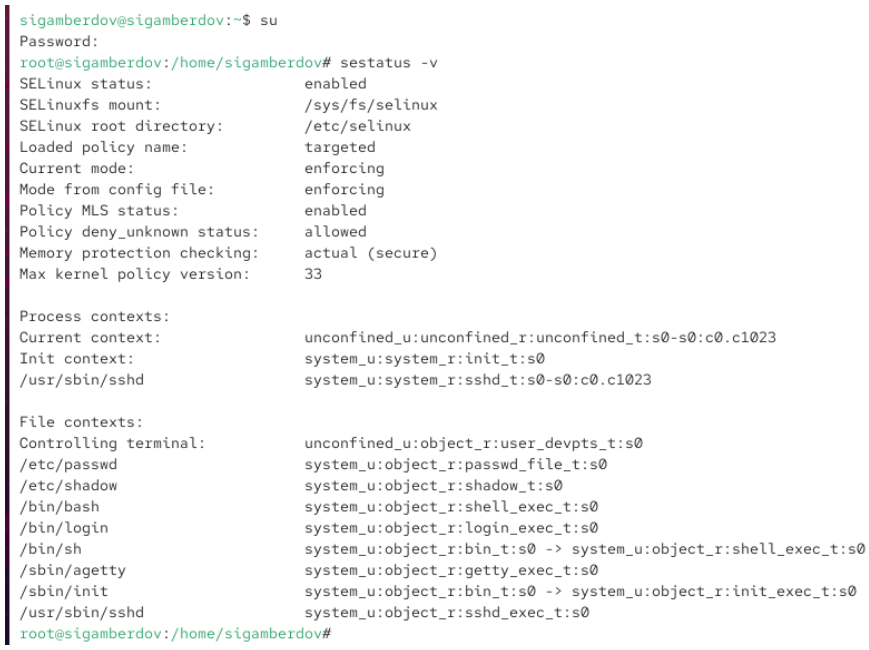

Рис. 2.6: Автоматическое восстановление меток SELinux при загрузке

10. После завершения загрузки команда **sestatus -v** показала, что система снова работает в режиме *enforcing*.

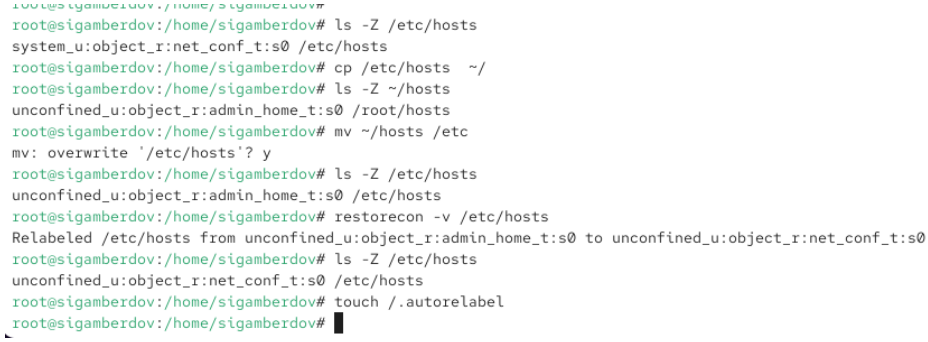

Рис. 2.7: Проверка состояния SELinux после восстановления

2.2 Использование restorecon для восстановления контекста безопасности

1. Получены права администратора и просмотрен контекст файла **/etc/hosts** с помощью **ls -Z /etc/hosts**.
   У файла установлен контекст **net_conf_t**, что соответствует категории сетевых конфигураций.

2. Файл **/etc/hosts** скопирован в домашний каталог (**cp /etc/hosts ~/**).
   Проверка **ls -Z ~/hosts** показала контекст **admin_home_t**, присвоенный файлам пользователя.

3. При перемещении копии обратно в каталог **/etc** (**mv ~/hosts /etc**) контекст остался **admin_home_t**, что является некорректным для системного файла.

4. Для исправления контекста выполнена команда **restorecon -v /etc/hosts**.
   Контекст изменён с **admin_home_t** на **net_conf_t**, что подтверждает восстановление корректной метки безопасности.

5. Повторная проверка **ls -Z /etc/hosts** показала правильный контекст **net_conf_t**.

6. Для массового восстановления контекстов на файловой системе создан файл **/.autorelabel** с помощью **touch /.autorelabel**.
   После перезагрузки система автоматически перемаркировала файлы, о чём сообщалось при старте системы.

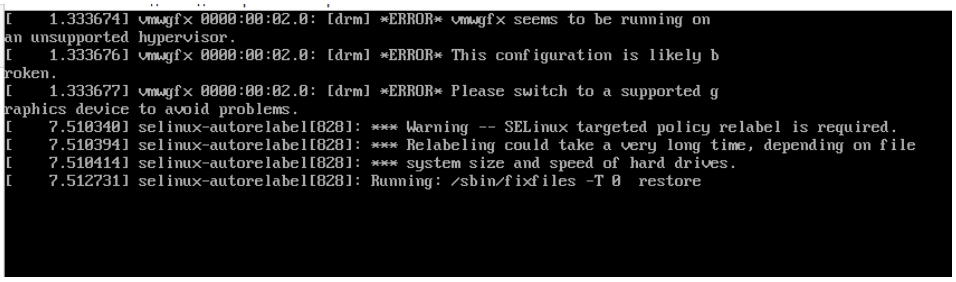

Рис. 2.8: Использование restorecon и autorelabel для восстановления контекста

2.3 Настройка контекста безопасности для нестандартного расположения файлов веб-сервера

1. В терминале получены полномочия администратора.
   Это необходимо для установки и настройки веб-сервера Apache.

2. Установлены необходимые пакеты — веб-сервер **httpd** и текстовый браузер **lynx**, позволяющий тестировать локальные веб-страницы из терминала.
   После завершения установки появилось сообщение *Complete!*, подтверждающее успешное выполнение операции.

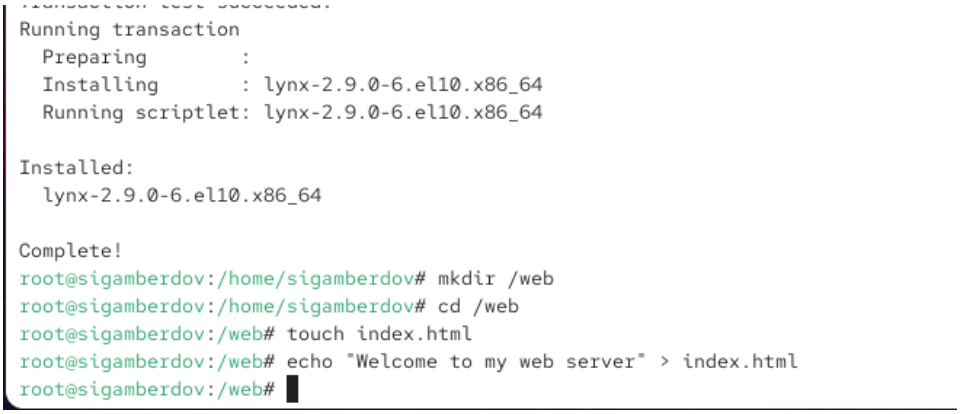

Рис. 2.9: Установка lynx и подготовка каталога /web

3. Создан новый каталог **/web** для размещения пользовательского контента веб-сервера.
   Далее в этот каталог добавлен файл **index.html** с текстом *Welcome to my web server*.
   Это будет стартовая страница, доступная при обращении к серверу.

4. Открыт файл конфигурации **/etc/httpd/conf/httpd.conf**.
   В нём закомментирована строка
   `DocumentRoot "/var/www/html"`

и добавлено новое значение
`DocumentRoot "/web"`.

Эти изменения определяют новый путь для контента веб-сервера и задают разрешения на доступ к нему.

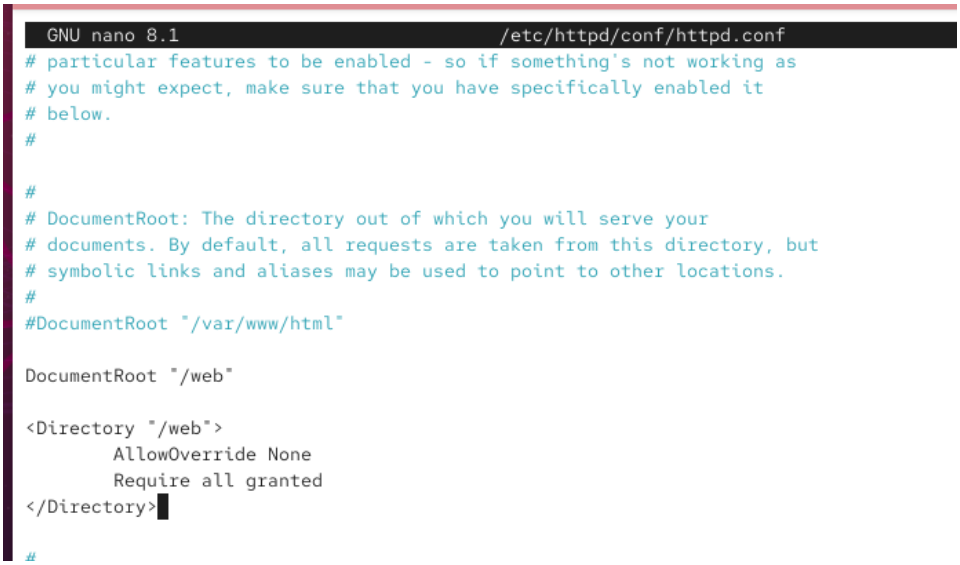

Рис. 2.10: Изменение конфигурации DocumentRoot

5. После сохранения изменений веб-сервер **httpd** был запущен и добавлен в автозагрузку.
   При тестовом обращении через браузер **lynx http://localhost** отобразилась стандартная страница Rocky Linux, что означает — сервер работает, но ещё не использует новый каталог /web.

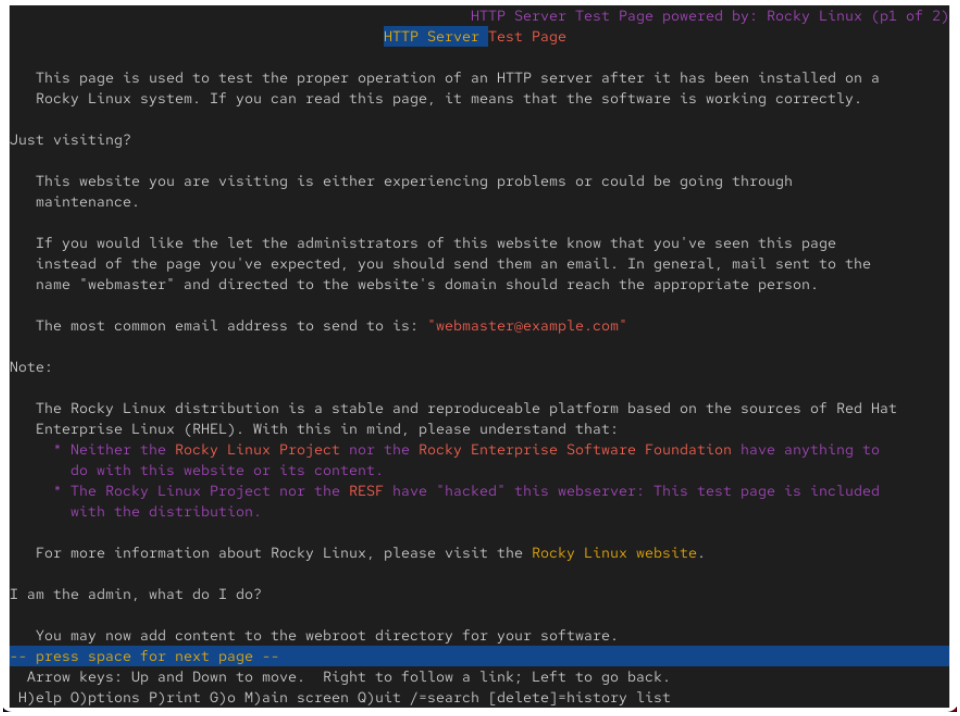

Рис. 2.11: Проверка работы веб-сервера через lynx — страница по умолчанию

6. Для корректного доступа веб-сервера к каталогу **/web** необходимо было изменить контекст безопасности SELinux.

   Команда

   **semanage fcontext -a -t httpd_sys_content_t “/web(/.*)?“

   добавила правило, указывающее, что каталог и его содержимое принадлежат типу *httpd_sys_content_t*.

   Далее команда restorecon -R -v /web** применила изменения, обновив метки контекста безопасности.

   В результате метка каталога и файла *index.html* была изменена с *default_t* на *httpd_sys_content_t*, что позволило службе Apache безопасно их обслуживать.

```
root@sigamberdov:/web#
root@sigamberdov:/web# semanage fcontext -a -t httpd_sys_content_t "/web(/.*)?"
root@sigamberdov:/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@sigamberdov:/web#
```

Рис. 2.12: Назначение контекста безопасности для каталога /web

7. После применения новых меток и повторного обращения к серверу командой **lynx http://localhost** веб-браузер отобразил созданную пользователем страницу с текстом
   Welcome to my web server.
   Это подтвердило, что конфигурация выполнена корректно, а доступ к каталогу /web разрешён политикой SELinux.

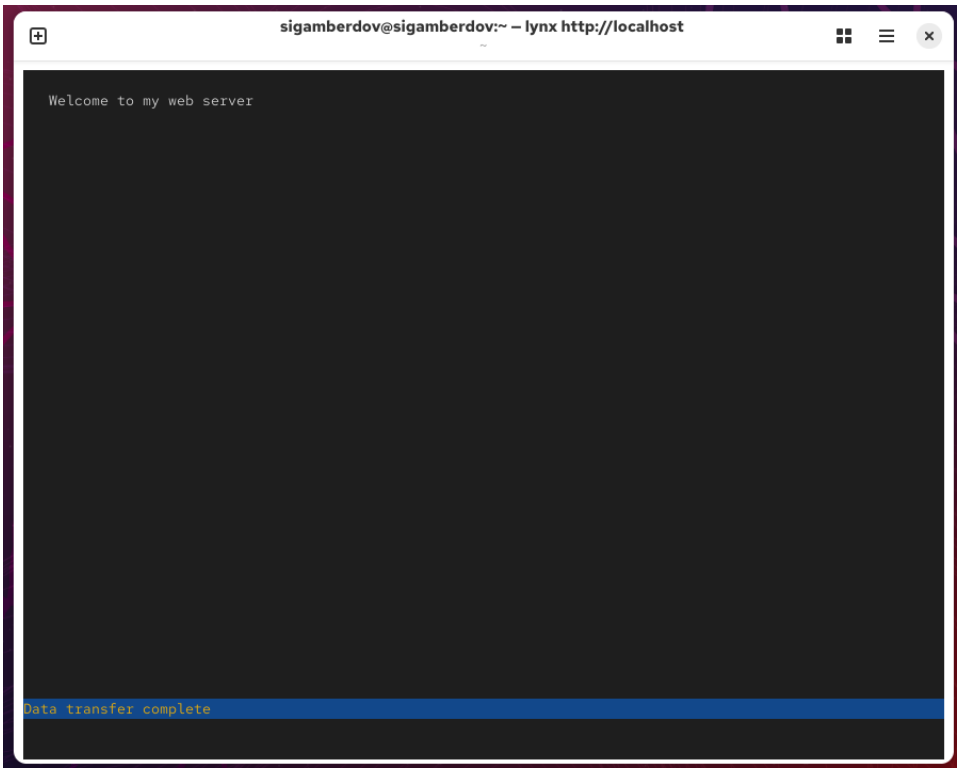

Рис. 2.13: Отображение пользовательской страницы веб-сервера

2.4 Дополнительная настройка SELinux для FTP-сервера

1. Для проверки доступных булевых параметров SELinux, связанных с FTP, использована команда **getsebool -a | grep ftp**.
   Из вывода видно, что все параметры были отключены.

2. Далее с помощью **semanage boolean -l | grep ftpd_anon** проверено состояние переменной **ftpd_anon_write**, отвечающей за разрешение анонимной записи в FTP.
   Первоначально её значение было *(off, off)*.

3. Команда **setsebool ftpd_anon_write on** временно включила возможность записи для анонимных пользователей, что подтвердилось последующей проверкой **getsebool ftpd_anon_write**.

4. Для постоянного сохранения изменений параметр был активирован командой **setsebool -P ftpd_anon_write on**.
   Повторная проверка **semanage boolean -l | grep ftpd_anon** показала статус *(on, on)*, что означает успешное сохранение настройки.

```
root@sigamberdov:/home/sigamberdov# 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@sigamberdov:/home/sigamberdov# semanage boolean -l | grep ftpd_anon
ftpd_anon_write                (off  ,  off)  Allow ftpd to anon write
root@sigamberdov:/home/sigamberdov# setsebool ftpd_anon_write on
root@sigamberdov:/home/sigamberdov# getsebool ftpd_anon_write
ftpd_anon_write --> on
root@sigamberdov:/home/sigamberdov# semanage boolean -l | grep ftpd_anon
ftpd_anon_write                (on   ,  off)  Allow ftpd to anon write
root@sigamberdov:/home/sigamberdov# setsebool -P ftpd_anon_write on
root@sigamberdov:/home/sigamberdov# semanage boolean -l | grep ftpd_anon
ftpd_anon_write                (on   ,   on)  Allow ftpd to anon write
root@sigamberdov:/home/sigamberdov#
```

Рис. 2.14: Настройка булевой переменной SELinux для FTP

3 Контрольные вопросы

1. Чтобы временно перевести SELinux в разрешающий режим (**Permissive**), используется команда:

   - **setenforce 0**

     Она отключает блокировку действий, нарушающих политику SELinux, но продолжает их регистрировать в журнале.

2. Для вывода списка всех доступных переключателей (булевых параметров) SELinux применяется команда:

   - **getsebool -a**

     Она отображает все активные и неактивные флаги, регулирующие поведение SELinux для различных сервисов.

3. Для получения легко читаемых сообщений SELinux из журнала аудита необходимо установить пакет:

   - **setroubleshoot-server**

     Этот пакет позволяет интерпретировать сообщения SELinux и выводить рекомендации по устранению проблем.

4. Чтобы применить тип контекста **httpd_sys_content_t** к каталогу **/web**, необходимо выполнить следующие команды:

   - **semanage fcontext -a -t httpd_sys_content_t “/web(/.*)?“** — добавить правило для каталога и его содержимого;

- **restorecon -R -v /web** — применить изменения и пересоздать корректные метки безопасности.

5. Для полного отключения SELinux требуется изменить конфигурационный файл:

   - **/etc/sysconfig/selinux**

     В нём необходимо установить параметр **SELINUX=disabled** и затем перезагрузить систему.

6. Все сообщения SELinux регистрируются в журнале:

   - **/var/log/audit/audit.log**

     В нём фиксируются все события, связанные с политиками безопасности, контекстами и нарушениями.

7. Чтобы узнать, какие типы контекстов доступны для службы **ftp**, используется команда:

   - **semanage fcontext -l | grep ftp**

     Она показывает все пути и типы контекстов, связанные с FTP-сервисом.

8. Если служба работает некорректно и есть подозрение, что причина в SELinux, самый простой способ проверки — временно перевести SELinux в разрешающий режим:

   - **setenforce 0**

     Если после этого сервис начинает работать, значит, проблема связана с политиками SELinux.

4 Заключение

В ходе лабораторной работы были изучены методы управления механизмом безопасности **SELinux**: проверка текущего состояния системы, изменение режимов работы (*enforcing*, *permissive*, *disabled*), редактирование конфигурационного файла и восстановление контекстов безопасности.
Были отработаны практические приёмы настройки SELinux для нестандартных каталогов веб-сервера, а также использование команд **semanage** и **restorecon** для назначения правильных меток безопасности.