

Отчёт по лабораторной работе №9

Управление SELinux

Турсунов Мухамметназар

Содержание

1	Цель работы	5
2	Выполнение	6
2.1	Управление режимами SELinux	6
2.2	Использование restorecon для восстановления контекста безопасности	10
3	Выполнение	12
3.1	Настройка контекста безопасности для нестандартного расположения файлов веб-сервера	12
3.2	Работа с переключателями SELinux	15
4	Контрольные вопросы	17
5	Заключение	19

Список иллюстраций

2.1	Вывод команды <code>sestatus -v</code>	6
2.2	Переключение режима SELinux на Permissive	7
2.3	Изменение параметра SELINUX=disabled	8
2.4	SELinux отключён — попытка включения невозможна	8
2.5	Включение режима enforcing в конфигурационном файле	9
2.6	Автоматическое восстановление меток SELinux при загрузке	9
2.7	SELinux снова включён, активен режим enforcing	10
2.8	Использование <code>restorecon</code> для восстановления контекста файла <code>hosts</code>	11
2.9	Автоматическое восстановление контекстов SELinux после создания / <code>autorelabel</code>	11
3.1	Создание каталога <code>/web</code> и файла <code>index.html</code>	12
3.2	Изменение параметров <code>DocumentRoot</code> и <code>Directory</code> в <code>httpd.conf</code>	13
3.3	Запуск службы <code>httpd</code> и её автозагрузка	13
3.4	Отображение стандартной страницы Rocky Linux	14
3.5	Присвоение контекста <code>httpd_sys_content_t</code> каталогу <code>/web</code>	14
3.6	Отображение пользовательской страницы веб-сервера	15
3.7	Просмотр и изменение состояния переключателя <code>ftpd_anon_write</code>	16

Список таблиц

1 Цель работы

Получить навыки работы с контекстом безопасности и политиками SELinux.

2 Выполнение

2.1 Управление режимами SELinux

1. Сначала был выполнен переход в режим суперпользователя с помощью команды **su -**.

Далее просмотрено текущее состояние SELinux командой **sestatus -v**, которая показывает параметры политики и контексты безопасности.

```
mtursunov@mtursunov:~$ su
Password:
root@mtursunov:/home/mtursunov# 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@mtursunov:/home/mtursunov#
```

Рис. 2.1: Вывод команды **sestatus -v**

В результате видно:

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

- **Current mode: enforcing** — активен режим принудительного контроля;
- **Loaded policy name: targeted** — применяется политика *targeted*, защищающая основные службы;
- **Policy MLS status: enabled** — многоуровневая защита включена;
- ниже отображаются контексты процессов и файлов, например, *system_u:object_r:passwd_file_t:s0* для */etc/passwd*.

2. Для определения текущего режима SELinux использовалась команда **getenforce**.

По умолчанию система находилась в состоянии **Enforcing** — политика безопасности применялась ко всем процессам.

Затем командой **setenforce 0** режим был временно изменён на **Permissive**, при котором нарушения фиксируются, но не блокируются.

Повторная проверка через **getenforce** подтвердила изменение.

```
root@mtursunov: /home/mtursunov#
root@mtursunov: /home/mtursunov# getenforce
Enforcing
root@mtursunov: /home/mtursunov# setenforce 0
root@mtursunov: /home/mtursunov# getenforce
Permissive
root@mtursunov: /home/mtursunov# █
```

Рис. 2.2: Переключение режима SELinux на Permissive

3. Далее был открыт файл */etc/sysconfig/selinux* с помощью текстового редактора **nano**.

В параметре **SELINUX** установлено значение *disabled*, что полностью отключает механизм SELinux после перезагрузки.



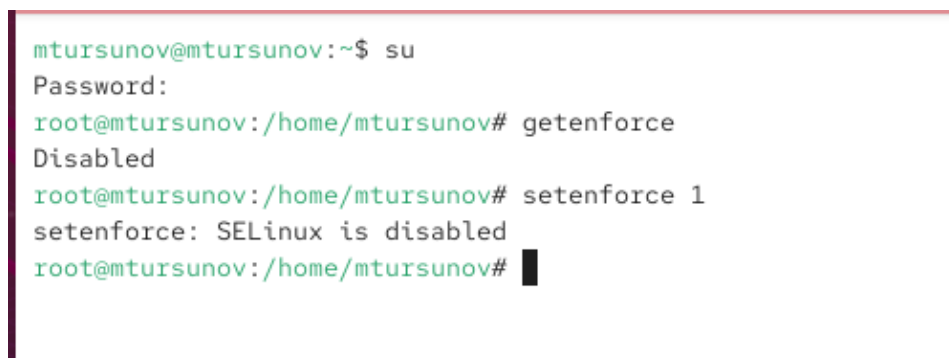
```
mtursunov@mtursunov: /home/mtursunov - nano /etc/sysconfig/selinux
GNU nano 8.1 /etc/sysconfig/selinux Modified
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - No SELinux policy is loaded.
# See also:
# https://docs.fedoraproject.org/en-US/quick-docs/getting-started-with-selinux/#getting-started-with-se
#
# NOTE: In earlier Fedora kernel builds, 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 are protected.
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted
^G Help      ^O Write Out ^F Where Is  ^K Cut       ^T Execute  ^C Location  M-U Undo
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify  ^_ Go To Line M-E Redo
```

Рис. 2.3: Изменение параметра SELINUX=disabled

4. После перезагрузки выполнена проверка текущего состояния.

getenforce показал значение **Disabled**, что подтверждает отключение SELinux.

Попытка активировать его командой **setenforce 1** завершилась сообщением *SELinux is disabled*, так как смена режима невозможна без перезапуска системы.

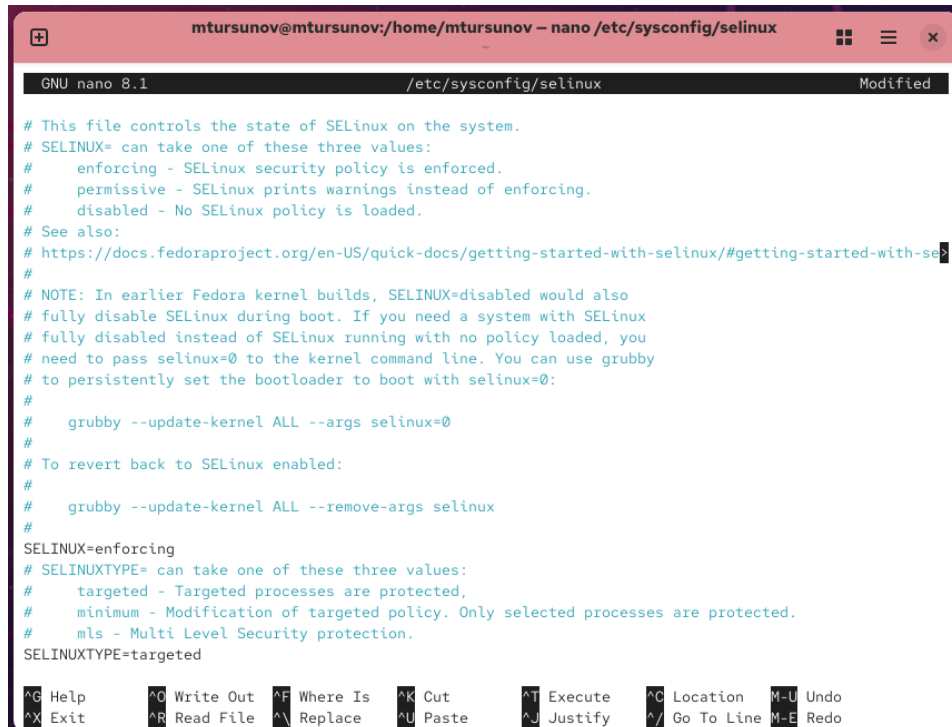


```
mtursunov@mtursunov:~$ su
Password:
root@mtursunov:/home/mtursunov# getenforce
Disabled
root@mtursunov:/home/mtursunov# setenforce 1
setenforce: SELinux is disabled
root@mtursunov:/home/mtursunov#
```

Рис. 2.4: SELinux отключён — попытка включения невозможна

5. Для повторного включения защиты в том же конфигурационном файле установлено значение *SELINUX=enforcing*.

Поле **SELINUXTYPE** оставлено как *targeted*.



```
mtursunov@mtursunov:/home/mtursunov - nano /etc/sysconfig/selinux
GNU nano 8.1 /etc/sysconfig/selinux Modified

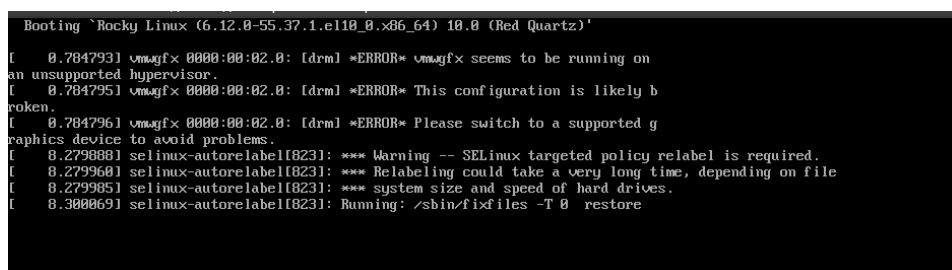
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - No SELinux policy is loaded.
# See also:
# https://docs.fedoraproject.org/en-US/quick-docs/getting-started-with-selinux/#getting-started-with-se
#
# NOTE: In earlier Fedora kernel builds, 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=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

^G Help      ^O Write Out ^F Where Is  ^K Cut       ^T Execute   ^C Location  M-U Undo
^X Exit      ^R Read File ^N Replace   ^U Paste     ^J Justify   ^_ Go To Line M-E Redo
```

Рис. 2.5: Включение режима enforcing в конфигурационном файле

6. При следующей загрузке системы появилось предупреждение о необходимости восстановления меток SELinux (relabeling).

Процесс выполнялся автоматически и мог занять продолжительное время в зависимости от объёма файловой системы.



```
Booting 'Rocky Linux (6.12.0-55.37.1.el10_0.x86_64) 10.0 (Red Quartz)'
```

```
[ 0.784793] vmwgfx 0000:00:02.0: [drm] *ERROR* vmwgfx seems to be running on
an unsupported hypervisor.
[ 0.784795] vmwgfx 0000:00:02.0: [drm] *ERROR* This configuration is likely b
roken.
[ 0.784796] vmwgfx 0000:00:02.0: [drm] *ERROR* Please switch to a supported g
raphics device to avoid problems.
[ 8.279888] selinux-autorelabel[8231]: *** Warning -- SELinux targeted policy relabel is required.
[ 8.279960] selinux-autorelabel[8231]: *** Relabeling could take a very long time, depending on file
[ 8.279985] selinux-autorelabel[8231]: *** system size and speed of hard drives.
[ 8.300069] selinux-autorelabel[8231]: Running: /sbin/fixfiles -T 0 restore
```

Рис. 2.6: Автоматическое восстановление меток SELinux при загрузке

7. После загрузки команда **sestatus -v** вновь показала, что SELinux включён и работает в режиме **enforcing**, а политика — *targeted*.

```
root@mtursunov: /home/mtursunov# 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@mtursunov: /home/mtursunov#
```

Рис. 2.7: SELinux снова включён, активен режим enforcing

2.2 Использование restorecon для восстановления контекста безопасности

1. Проверен текущий контекст безопасности файла */etc/hosts* командой **ls -Z /etc/hosts**.

Тип контекста — *net_conf_t*.

2. Файл был скопирован в домашний каталог с помощью команды **cp /etc/hosts ~/.**

После этого контекст нового файла *~/hosts* изменился на *admin_home_t*, что характерно для пользовательских файлов.

3. Файл из домашнего каталога был перемещён обратно в */etc*, после чего контекст остался *admin_home_t*, что не соответствует системным требованиям.

4. Для восстановления корректного контекста безопасности использовалась команда **restorecon -v /etc/hosts**.

Утилита изменила метку на *net_conf_t*, что подтверждено повторной проверкой **ls -Z /etc/hosts**.

```
root@mtursunov:/home/mtursunov# ls -Z /etc/hosts
system_u:object_r:net_conf_t:s0 /etc/hosts
root@mtursunov:/home/mtursunov# cp /etc/hosts ~/
root@mtursunov:/home/mtursunov# ls -Z ~/hosts
unconfined_u:object_r:admin_home_t:s0 /root/hosts
root@mtursunov:/home/mtursunov# mv ~/hosts /etc
mv: overwrite '/etc/hosts'? y
root@mtursunov:/home/mtursunov# ls -Z /etc/hosts
unconfined_u:object_r:admin_home_t:s0 /etc/hosts
root@mtursunov:/home/mtursunov# 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@mtursunov:/home/mtursunov# ls -Z /etc/hosts
unconfined_u:object_r:net_conf_t:s0 /etc/hosts
root@mtursunov:/home/mtursunov# touch /.autorelabel
root@mtursunov:/home/mtursunov#
```

Рис. 2.8: Использование restorecon для восстановления контекста файла hosts

5. Для массового восстановления контекстов на всей файловой системе создан файл */.autorelabel* с помощью команды **touch /.autorelabel**.

После перезагрузки система автоматически перемаркировала все файлы, что сопровождалось сообщениями о выполнении relabel.

```
[ 1.394272] vmwgfx 0000:00:02.0: [drm] *ERROR* vmwgfx seems to be running on
an unsupported hypervisor.
[ 1.394274] vmwgfx 0000:00:02.0: [drm] *ERROR* This configuration is likely b
roken.
[ 1.394275] vmwgfx 0000:00:02.0: [drm] *ERROR* Please switch to a supported g
raphics device to avoid problems.
[ 7.704382] selinux-autorelabel[821]: *** Warning -- SELinux targeted policy relabel is required.
[ 7.705344] selinux-autorelabel[821]: *** Relabeling could take a very long time, depending on file
[ 7.705410] selinux-autorelabel[821]: *** system size and speed of hard drives.
[ 7.714706] selinux-autorelabel[821]: Running: /sbin/fixfiles -T 0 restore
```

Рис. 2.9: Автоматическое восстановление контекстов SELinux после создания */.autorelabel*

3 Выполнение

3.1 Настройка контекста безопасности для нестандартного расположения файлов веб-сервера

1. После получения полномочий администратора было установлено необходимое программное обеспечение для веб-сервера и текстового браузера: **httpd** и **lynx**.

Затем создан новый каталог для хранения веб-контента — */web*, в котором размещён файл *index.html* с тестовой строкой *Welcome to my web server*.

```
Installed:
  lynx-2.9.0-6.el10.x86_64

Complete!
root@mtursunov:/home/mtursunov# mkdir /web
root@mtursunov:/home/mtursunov# cd /web
root@mtursunov:/web# touch index.html
root@mtursunov:/web# echo "Welcome to my web server" > index.html
root@mtursunov:/web# nano /etc/httpd/conf/httpd.conf
root@mtursunov:/web# systemctl start httpd
root@mtursunov:/web# systemctl enable httpd
root@mtursunov:/web# █
```

Рис. 3.1: Создание каталога */web* и файла *index.html*

2. В конфигурационном файле */etc/httpd/conf/httpd.conf* была закомментирована стандартная строка

DocumentRoot "/var/www/html" и добавлена новая — *DocumentRoot "/web"*.

Также внесён соответствующий раздел **Directory** для нового каталога, разрешающий доступ к файлам.

```
#
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
#DocumentRoot "/var/www/html"

DocumentRoot "/web"

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

Рис. 3.2: Изменение параметров DocumentRoot и Directory в httpd.conf

3. После внесения изменений запущена служба **httpd** и настроен её автоматический запуск при старте системы.

```
Installed:
  lynx-2.9.0-6.el10.x86_64

Complete!
root@mtursunov:/home/mtursunov# mkdir /web
root@mtursunov:/home/mtursunov# cd /web
root@mtursunov:/web# touch index.html
root@mtursunov:/web# echo "Welcome to my web server" > index.html
root@mtursunov:/web# nano /etc/httpd/conf/httpd.conf
root@mtursunov:/web# systemctl start httpd
root@mtursunov:/web# systemctl enable httpd
root@mtursunov:/web#
```

Рис. 3.3: Запуск службы httpd и её автозагрузка

4. При обращении к локальному веб-серверу через текстовый браузер **lynx** отобразилась стандартная страница Rocky Linux, что говорит о том, что SELinux не разрешил использовать новый каталог */web*.

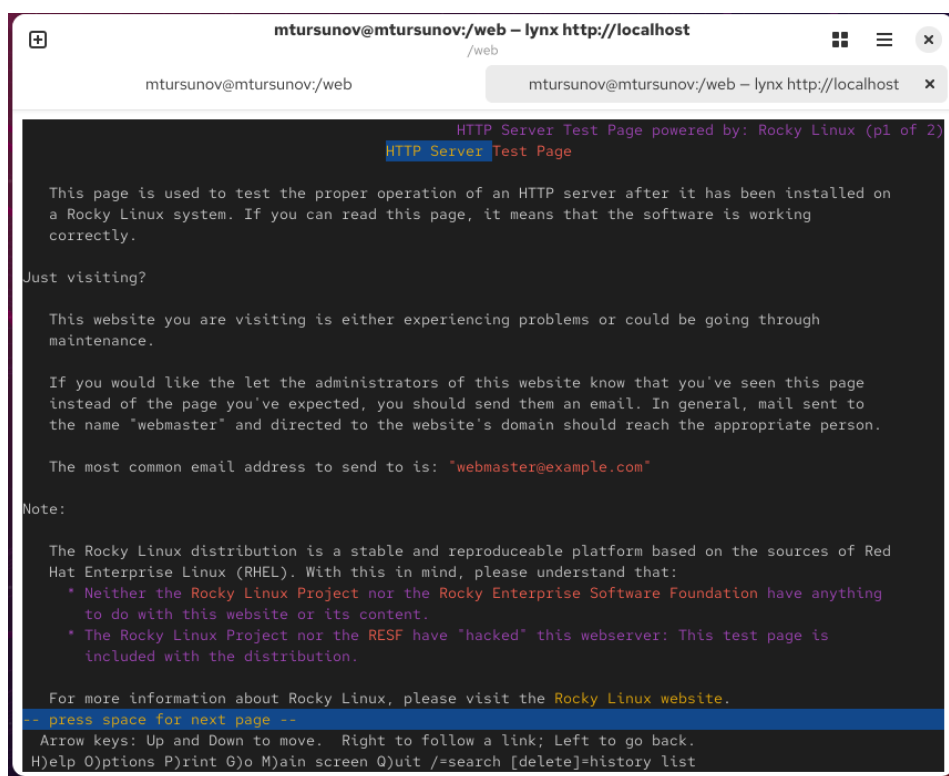


Рис. 3.4: Отображение стандартной страницы Rocky Linux

5. Для решения проблемы был назначен корректный контекст безопасности для каталога `/web` с помощью команды


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

, а затем выполнено **восстановление контекста** `restorecon -R -v /web`.
В результате файлам и каталогу были присвоены метки безопасности, разрешающие доступ службе httpd.**

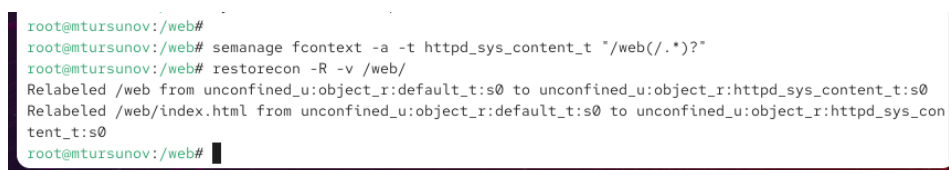


Рис. 3.5: Присвоение контекста `httpd_sys_content_t` каталогу `/web`

6. Повторное обращение к серверу через **lynx http://localhost** показало корректную загрузку пользовательской страницы с текстом *Welcome to my web*

server.

Это подтверждает, что настройка контекста SELinux для каталога /web выполнена успешно.

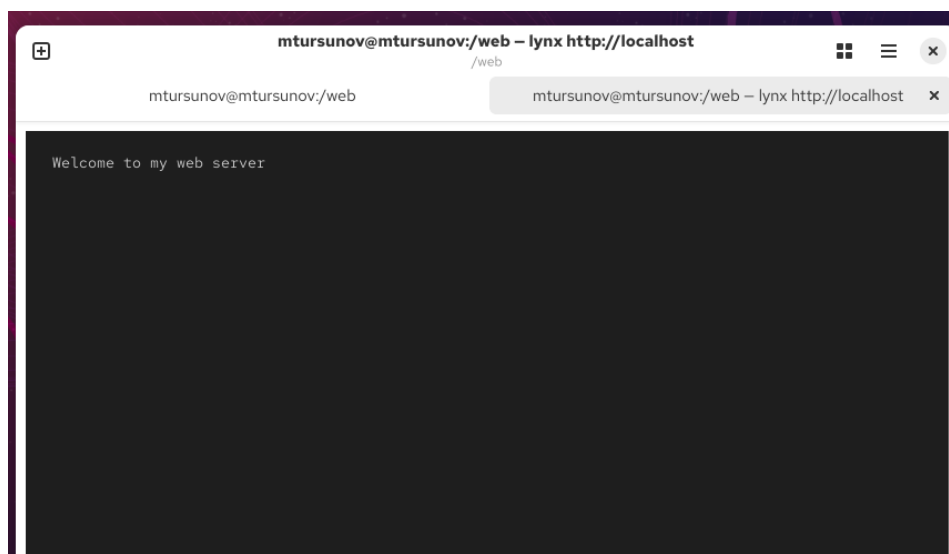


Рис. 3.6: Отображение пользовательской страницы веб-сервера

3.2 Работа с переключателями SELinux

1. Был выполнен просмотр всех переключателей SELinux, связанных с FTP-сервисом, командой **getsebool -a | grep ftp**.

Из вывода видно, что параметр **ftpd_anon_write** по умолчанию имеет состояние *off*.

2. Далее получен список переключателей для службы **ftpd_anon** с пояснениями с помощью команды **semanage boolean -l | grep ftpd_anon**.

Параметр **ftpd_anon_write** отвечает за разрешение анонимной записи в FTP.

3. Переключатель **ftpd_anon_write** был временно активирован командой **setsebool ftpd_anon_write on**, после чего проверено его состояние — значение изменилось на *on*.

4. Для сохранения параметра между перезагрузками он был включён постоянно с помощью команды **setsebool -P ftpd_anon_write on**.

Повторная проверка через **semanage boolean -l | grep ftpd_anon** показала, что оба состояния (*runtime* и *persistent*) установлены в *on*.

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

Рис. 3.7: Просмотр и изменение состояния переключателя ftpd_anon_write

4 Контрольные вопросы

1. **Вы хотите временно поставить SELinux в разрешающем режиме. Какую команду вы используете?**

`setenforce 0`

2. **Вам нужен список всех доступных переключателей SELinux. Какую команду вы используете?**

`getsebool -a`

3. **Каково имя пакета, который требуется установить для получения легко читаемых сообщений журнала SELinux в журнале аудита?**

`setroubleshoot`

4. **Какие команды вам нужно выполнить, чтобы применить тип контекста `httpd_sys_content_t` к каталогу `/web`?**

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

`restorecon -R -v /web`

5. **Какой файл вам нужно изменить, если вы хотите полностью отключить SELinux?**

`/etc/sysconfig/selinux`

6. **Где SELinux регистрирует все свои сообщения?**

`/var/log/audit/audit.log`

7. **Вы не знаете, какие типы контекстов доступны для службы `ftp`. Какая команда позволяет получить более конкретную информацию?**

```
semanage fcontext -l | grep ftp
```

8. **Ваш сервис работает не так, как ожидалось, и вы хотите узнать, связано ли это с SELinux или чем-то ещё. Какой самый простой способ узнать?**

```
setenforce 0
```

(Временное переключение в разрешающий режим для проверки влияния SELinux)

5 Заключение

В ходе работы были изучены принципы управления системой безопасности **SELinux** в операционной системе Linux.

Были рассмотрены режимы работы SELinux — **enforcing**, **permissive** и **disabled**, а также способы их временного и постоянного изменения.

Проведена настройка контекста безопасности для нестандартного каталога веб-сервера */web*, что обеспечило корректный доступ службы **httpd** к его содержимому.

С помощью инструментов **semanage** и **restorecon** освоены методы управления и восстановления меток безопасности.

Также изучена работа с переключателями SELinux (**booleans**) на примере параметра **ftpd_anon_write**, который был изменён и закреплён на постоянной основе.