

Управление SELinux

Часть 1

Славинский В.В.

1 ноября 2025

Российский университет дружбы народов, Москва, Россия Россия

Информация

..... {.columns align=center} ::: {.column width="70%"}

- Славинский Владислав Вадимович
- Студент
- Российский университет дружбы народов
- [1132246169@pfur.ru]

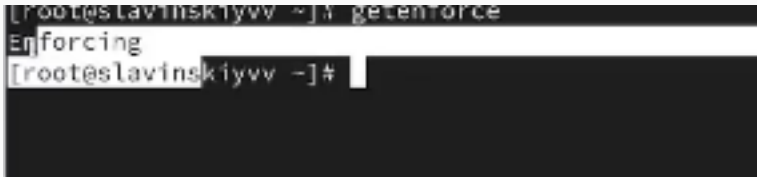
::: ::: {.column width="30%"}

Вводная часть

Просмотр информации о состоянии SELinux с помощью root прав

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)

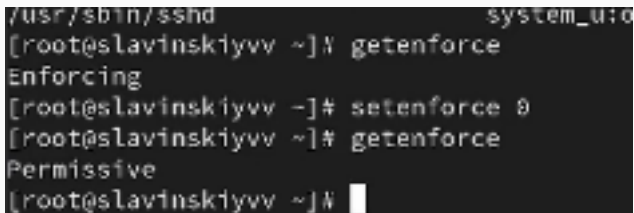
Посмотрим, в каком режиме работает SELinux: `getenforce`. Видим, что работает в режиме `enforcing` (в режиме принудительного исполнения).

A terminal window with a black background and white text. The prompt is [root@slavinskiyvv ~]. The command getenforce has been entered, and the output is Enforcing. The prompt has changed to [root@slavinskiyvv ~]#.

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

Рис. 2: sc2

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

A terminal window with a black background and white text. The prompt is `[root@slavinskiyvv ~]#`. The first command is `getenforce`, which outputs `Enforcing`. The second command is `setenforce 0`. The third command is `getenforce`, which outputs `Permissive`. The prompt is followed by a white cursor.

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

Рис. 3: sc3

Изменение файла и перезапуск системы

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

```
# 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 are
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted
```


Перезагрузки и проверка статуса SELinux

После перезагрузки запустим терминал и получим полномочия администратора. Посмотрим статус SELinux: `getenforce`. Мы видим, что SELinux теперь отключён.

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

Попытка переключить режим работы SELinux

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

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

Редактирование файла и перезагрузка системы

Откроем файл `/etc/sysconfig/selinux` с помощью редактора и установим: `SELINUX=enforcing`.
Затем перезагрузим систему.

```
# disabled - no SELinux policy is loaded.
# See also:
# https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/SELinux_guide/chapter-Getting_SELinux_workin
#
# 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
```

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.

```
[ 0.053425] RETNed: WARNING: Spectre v2 mitigation leaves CPU vulnerable to
RETNed attacks, data leaks possible!
[ 1.418990] Warning: Unmaintained driver is detected: e1000
[ 1.848960] vmwgfx 0000:00:02.0: [drm] *ERROR* vmwgfx seems to be running on
an unsupported hypervisor.
[ 1.848972] vmwgfx 0000:00:02.0: [drm] *ERROR* This configuration is likely b
roken.
[ 1.848974] vmwgfx 0000:00:02.0: [drm] *ERROR* Please switch to a supported g
raphics device to avoid problems.
[ 3.951816] selinux-autorelabel[17071]: *** Warning -- SELinux targeted policy relabel is required.
[ 3.951122] selinux-autorelabel[17071]: *** Relabeling could take a very long time, depending on file
[ 3.951161] selinux-autorelabel[17071]: *** system size and speed of hard drive.
[ 3.956610] selinux-autorelabel[17071]: Warning: /sbin/initfiles -T & restore
[ 10.746770] selinux-autorelabel[17931]: Warning: Skipping the following /etc filesystems:
[ 10.746915] selinux-autorelabel[17931]: /run/credentials/systemd-cryptsetup.service
[ 10.746962] selinux-autorelabel[17931]: /run/credentials/systemd-tmpfiles-setup-dev.service
[ 10.747003] selinux-autorelabel[17931]: /run/credentials/systemd-tmpfiles-setup.service
[ 10.747182] selinux-autorelabel[17931]: Relabeling / /boot /dev /dev/hugepages /dev/mqueue /dev/pts /dev/shm /run /sys /sys/fs/cgroup /sys/fs/pstore /sys/kernel
Ldshub /sys/kernel/tracing
-
```

Проверка статуса

После перезагрузки в терминале с полномочиями администратора посмотрим текущую информацию о состоянии SELinux: `sestatus -v`. Видим, что система работает в режиме enforcing.

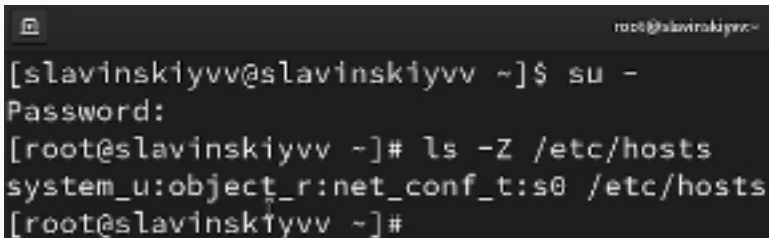
```
slavinskiyv@slavinskiyv:~$ sestatus -v
SELinux status: enabled
SELinuxfs mount: /sys/fs/selinux
SELinux root directory: /etc/selinux
Loaded policy name: targeted
Current mode: enforcing
Mode from config file: enforcing
Policy MLS status: enabled
Policy deny_unknown status: allowed
Memory protection 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
```

Просмотр контекста безопасности файла

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.

A terminal window with a dark background. The title bar shows a window icon and the text 'root@slavinskiyv ~'. The terminal content shows a user switching to root using 'su -', entering a password, and then running 'ls -Z /etc/hosts'. The output shows the file's security context as 'system_u:object_r:net_conf_t:s0'.

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

Копирование файла в домашний каталог

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`.

```
[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 ~]#
```

}

Перезапись файла из домашнего каталога

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`.

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

I

Исправление контекста безопасности

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

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

Массовое 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.

```
[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 ~]# touch /.autorelabel
[root@slavinskiyv ~]# reboot
```

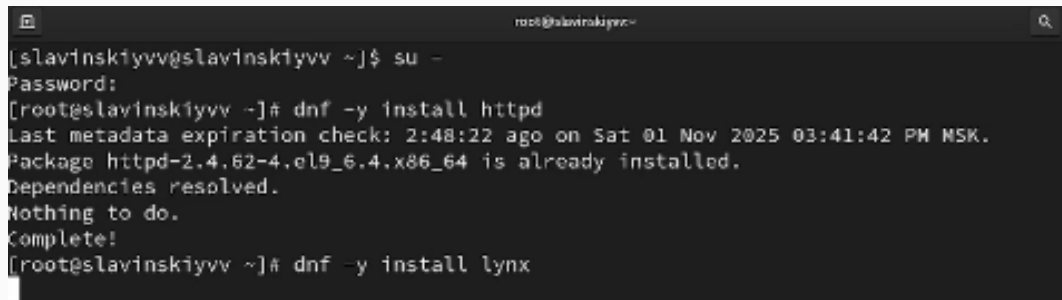
Перемаркированные сообщения на перезагрузке

Вот какие сообщения выводятся при перезагрузке.

```
Starting Create Static Device Nodes in /dev...
[ OK ] Mounted FUSE Control File System.
[ OK ] Mounted Kernel Configuration File System.
[ OK ] Finished Mounting of UIC wires, msp430 etc. using dmccatd in progress polling.
[ OK ] Finished Load/Save OS Random Seed.
[ OK ] Finished Coldplug 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 configs...
Starting Load Kernel Module fscm...
[ OK ] Finished Load Kernel Module configs.
[ OK ] Finished Load Kernel Module fscm.
[ OK ] Started /usr/sbin/udevadm settle --wait --autoactivation event v1.
t.1552351 selinux-autorelabel[7001]: *** Warning -- SELinux targeted policy relabel is required.
t.1554491 selinux-autorelabel[7001]: *** Relabeling could take a very long time, depending on file
t.1564031 selinux-autorelabel[7001]: *** system size and speed of hard drives.
t.2825911 selinux-autorelabel[7001]: Warning: /sbin/vxfs -T @ restore
```

Рис. 15: sc15

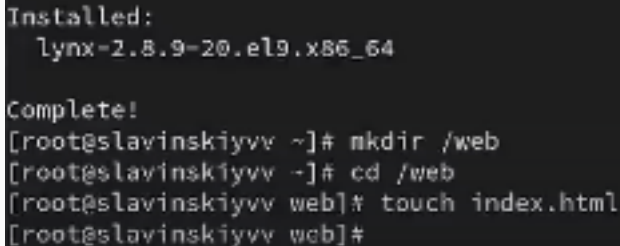
Запустим терминал в режиме администратора. Затем установим необходимое программное обеспечение: `dnf -y install httpd`, `dnf -y install lynx`.

A terminal window with a dark background and light-colored text. The window title is 'root@slavinskiyvv ~'. The user 'slavinskiyvv' is at the prompt and enters 'su -'. A 'Password:' prompt is shown. The user then enters 'dnf -y install httpd'. The terminal output shows the last metadata expiration check, that the package 'httpd-2.4.62-4.el9_6.4.x86_64' is already installed, dependencies are resolved, and nothing needs to be done. The user then enters 'dnf -y install lynx' and the terminal is partially obscured by a black rectangle.

```
root@slavinskiyvv ~  
[slavinskiyvv@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
```

Создание нового хранилища для файлов web-сервера

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

A terminal window with a dark background and light gray text. It shows the output of a Lynx installation, followed by a series of commands to create a web directory and file. The text is as follows:

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

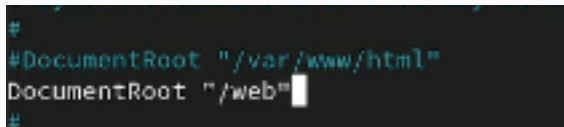
Complete!
[root@slavinskiyvv ~]# mkdir /web
[root@slavinskiyvv ~]# cd /web
[root@slavinskiyvv web]# touch index.html
[root@slavinskiyvv web]#
```

Рис. 17: sc17

Поместим в этот файл следующий текст: Welcome to my web-server.

```
Welcome to my web-server
```

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



```
#  
#DocumentRoot "/var/www/html"  
DocumentRoot "/web"  
#
```

Рис. 19: sc19

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`.

```
#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>
```

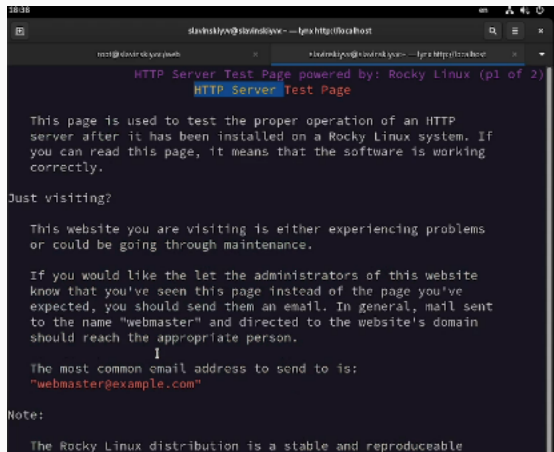

Запуск веб-сервера и службы

Запустим веб-сервер и службу httpd: `systemctl start httpd`, `systemctl enable httpd`.

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

Рис. 21: sc21

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`.



```
slavinskiy@slavinskiy:~$ lynx http://localhost
HTTP Server Test Page powered by: Rocky Linux (p1 of 2)
HTTP Server Test Page

This page is used to test the proper operation of an HTTP
server after it has been installed on a Rocky Linux system. If
you can read this page, it means that the software is working
correctly.

Just visiting?

This website you are visiting is either experiencing problems
or could be going through maintenance.

If you would like the let the administrators of this website
know that you've seen this page instead of the page you've
expected, you should send them an email. In general, mail sent
to the name "webmaster" and directed to the website's domain
should reach the appropriate person.

!
The most common email address to send to is:
"webmaster@example.com"

Note:

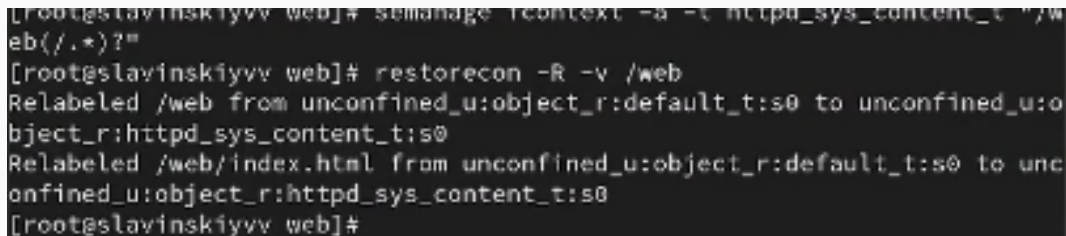
The Rocky Linux distribution is a stable and reproduceable
```

Применение новой метки контекста

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

```
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]# semanage fcontext -a -t httpd_sys_content_t "/web(/.*)"?"  
[root@slavinskiyvv web]#
```

Восстановим контекст безопасности: `restorecon -R -v /web`.

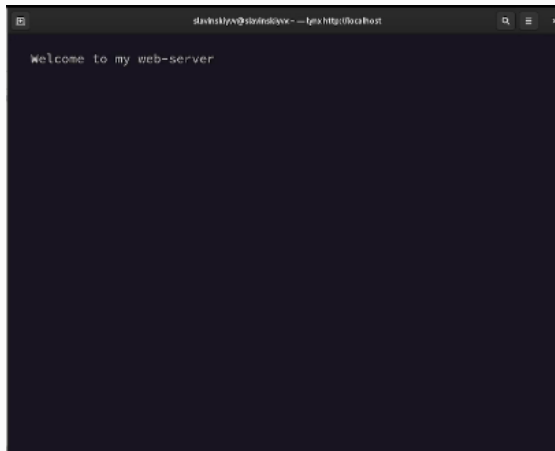


```
[root@slavinskiyvv web]# semanage recontext -a -t httpd_sys_content_t "/web(/.*)?"  
[root@slavinskiyvv web]# restorecon -R -v /web  
Relabeled /web from unconfined_u:object_r:default_t:s0 to unconfined_u:object_r:httpd_sys_content_t:s0  
Relabeled /web/index.html from unconfined_u:object_r:default_t:s0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

Рис. 24: sc24

Проверка веб-сервера

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.



Список переключателей SELinux

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

```
[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 ~]#
```

Список переключателей

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.

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

Изменения значения переключателя

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`.

```
ftpd_anon_write --> off
[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 ~]#
```


Просмотр списка переключателей

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

```
[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 ~]#
```

Изменение постоянного значения переключателя

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.

```
ftpd_anon_write (off, 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 ~]#
```