Информационная безопасность. Отчет по лабораторной работе №6

Мандатное разграничение прав в Linux

Горбунова Ярослава Михайловна

Содержание

# 1 Цель работы

Развить навыки администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux. Проверить работу SELinx на практике совместно с веб-сервером Apache [1].

# 2 Теоретическое введение

## 2.1 Организация и описание лабораторного стенда

Для проведения указанной лабораторной работы на одно рабочее место требуется компьютер с установленной операционной системой Linux, поддерживающей технологию SELinux. Предполагается использовать стандартный дистрибутив Linux CentOS с включённой политикой SELinux targeted и режимом enforcing. Для выполнения заданий требуется наличие учётной записи администратора (root) и учётной записи обычного пользователя. Постоянно работать от учётной записи root неправильно с точки зрения безопасности.

## 2.2 Подготовка лабораторного стенда и методические

рекомендации

1. При подготовке стенда обратите внимание, что необходимая для работы и указанная выше политика targeted и режим enforcing используются в данном дистрибутиве по умолчанию, т.е. каких-то специальных настроек не требуется. При этом следует убедиться, что политика и режим включены, особенно когда работа будет проводиться повторно и велика вероятность изменений при предыдущем использовании системы.
2. При необходимости администратор должен разбираться в работе SELinux и уметь как исправить конфигурационный файл /etc/selinux/config, так и проверить используемый режим и политику.
3. Необходимо, чтобы был установлен веб-сервер Apache. При установке системы в конфигурации «рабочая станция» указанный пакет не ставится.
4. В конфигурационном файле /etc/httpd/httpd.conf необходимо за дать параметр ServerName: ServerName test.ru чтобы при запуске веб-сервера не выдавались лишние сообщения об ошибках, не относящихся к лабораторной работе (fig. 1).



Figure 1: Подготовка лабораторного стенда. Пункт 4

1. Также необходимо проследить, чтобы пакетный фильтр был отключён или в своей рабочей конфигурации позволял подключаться к 80-у и 81-у портам протокола tcp (fig. 1).

Отключить фильтр можно командами

iptables -F

iptables -P INPUT ACCEPT iptables -P OUTPUT ACCEPT

либо добавить разрешающие правила:

iptables -I INPUT -p tcp –dport 80 -j ACCEPT

iptables -I INPUT -p tcp –dport 81 -j ACCEPT

iptables -I OUTPUT -p tcp –sport 80 -j ACCEPT

iptables -I OUTPUT -p tcp –sport 81 -j ACCEPT

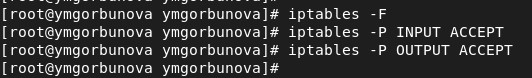


Figure 2: Подготовка лабораторного стенда. Пункт 5

1. Обратите внимание, что данные правила не являются «точными» и рекомендуемыми на все случаи жизни, они лишь позволяют правильно организовать работу стенда.
2. В работе специально не делается акцент, каким браузером (или какой консольной программой) будет производиться подключение к вебсерверу. По желанию могут использоваться разные программы, такие как консольные links, lynx, wget и графические konqueror, opera, firefoxили др.

# 3 Выполнение лабораторной работы

1. Войдите в систему с полученными учётными данными и убедитесь, что SELinux работает в режиме enforcing политики targeted с помощью команд getenforceи sestatus (fig. 3).
2. Обратитесь с помощью браузера к веб-серверу, запущенному на вашем компьютере, и убедитесь, что последний работает (fig. 3):

service httpd status или

/etc/rc.d/init.d/httpd status

Если не работает, запустите его так же, но с параметром start.

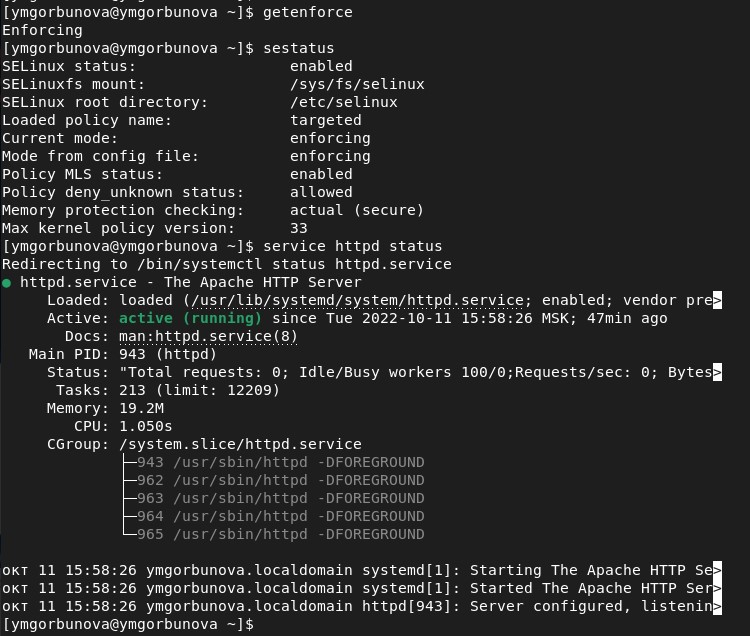


Figure 3: Пункт 1-2

1. Найдите веб-сервер Apache в списке процессов, определите его контекст безопасности и занесите эту информацию в отчёт. Например, можно использовать команду (fig. 4) ps auxZ | grep httpd или ps -eZ | grep httpd

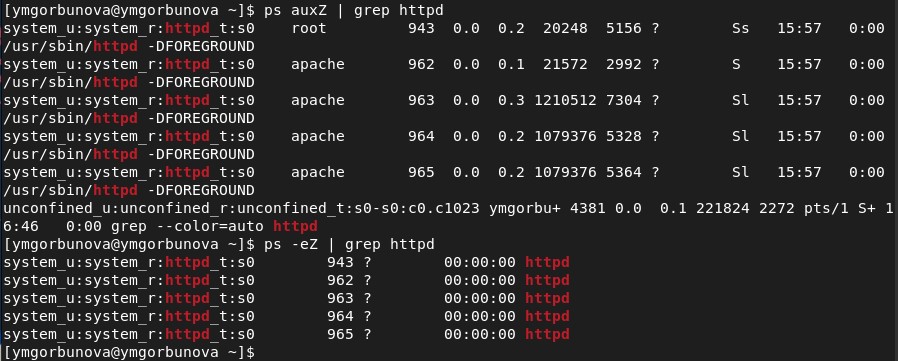


Figure 4: Пункт 3

1. Посмотрите текущее состояние переключателей SELinux для Apache с помощью команды (fig. 5) sestatus -bigrep httpd Обратите внимание, что многие из них находятся в положении «off».

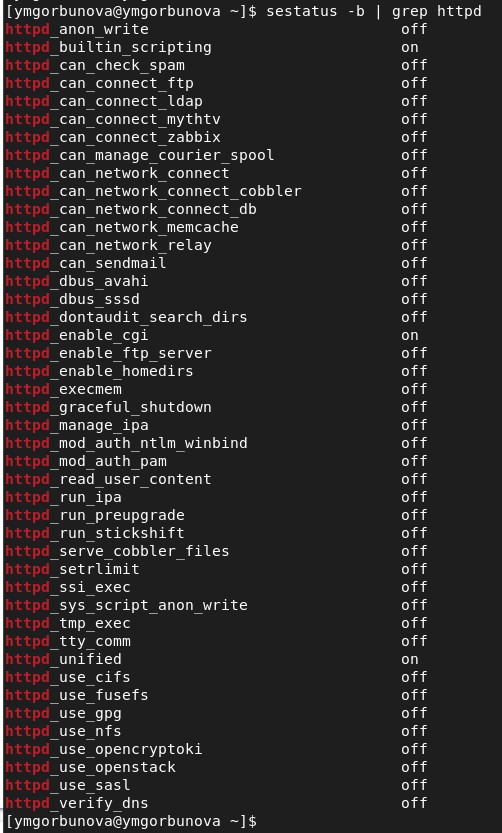


Figure 5: Пункт 4

1. Посмотрите статистику по политике с помощью команды seinfo, также определите множество пользователей, ролей, типов (fig. 6).

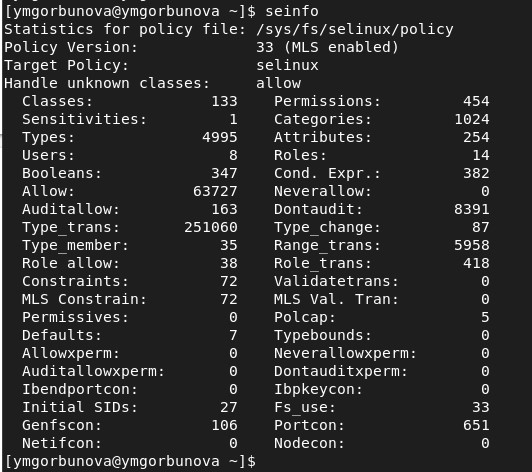


Figure 6: Пункт 5

1. Определите тип файлов и поддиректорий, находящихся в директории /var/www, с помощью команды (fig. 7) ls -lZ /var/www
2. Определите тип файлов, находящихся в директории /var/www/html (fig. 7): ls -lZ /var/www/html
3. Определите круг пользователей, которым разрешено создание файлов в директории /var/www/html (fig. 7).

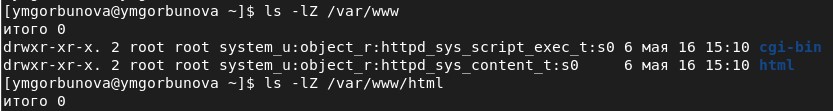


Figure 7: Пункт 6-8

1. Создайте от имени суперпользователя (так как в дистрибутиве после установки только ему разрешена запись в директорию) html-файл /var/www/html/test.html следующего содержания (fig. 8):

* test

1. Проверьте контекст созданного вами файла. Занесите в отчёт контекст, присваиваемый по умолчанию вновь созданным файлам в директории /var/www/html (fig. 8).

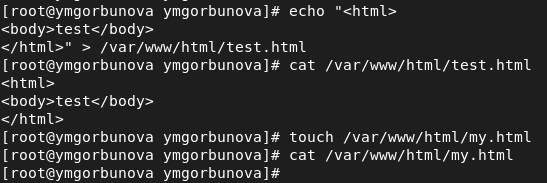


Figure 8: Пункт 9-10

1. Обратитесь к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html. Убедитесь, что файл был успешно отображён (fig. 9).

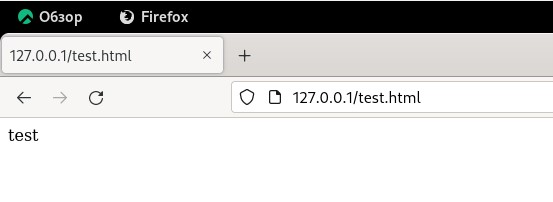


Figure 9: Пункт 11

1. Изучите справку man httpd\_selinux и выясните, какие контексты файлов определены для httpd. Сопоставьте их с типом файла test.html. Проверить контекст файла можно командой ls -Z. ls -Z /var/www/html/test.html

Рассмотрим полученный контекст детально. Обратите внимание, что так как по умолчанию пользователи CentOS являются свободными от типа (unconfined в переводе с англ. означает свободный), созданному нами файлу test.htmlбыл 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. (В директории /ргос файлы, относящиеся к процессам, могут иметь роль system\_r. Если активна политика MLS, то могут использоваться и другие роли, например, secadm\_r. Данный случай мы рассматривать не будем, как и предназначение :s0). 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 (fig. 10).

Figure 10: Пункт 12

Figure 10: Пункт 12

1. Измените контекст файла /var/www/html/test.html с httpd\_sys\_content\_t на любой другой, к которому процесс httpd не должен иметь доступа, например, на samba\_share\_t: chcon -t samba\_share\_t /var/www/html/test.html ls -Z /var/www/html/test.html После этого проверьте, что контекст поменялся (fig. 11).

Figure 11: Пункт 13

Figure 11: Пункт 13

1. Попробуйте ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html. Вы должны получить сообщение об ошибке: Forbidden You don’t have permission to access /test.html on this server (fig. 12).

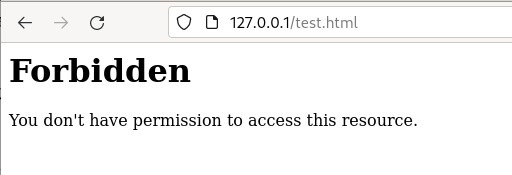


Figure 12: Пункт 14

1. Проанализируйте ситуацию. Почему файл не был отображён, если права доступа позволяют читать этот файл любому пользователю? ls -l /var/www/html/test.html Просмотрите log-файлы веб-сервера Apache. Также просмотрите системный лог-файл: tail /var/log/messages Если в системе окажутся запущенными процессы setroubleshootd и audtd, то вы также сможете увидеть ошибки, аналогичные указанным выше, в файле /var/log/audit/audit.log. Проверьте это утверждение самостоятельно (fig. 13-fig. 15).

Figure 13: Пункт 15 (1)

Figure 13: Пункт 15 (1)

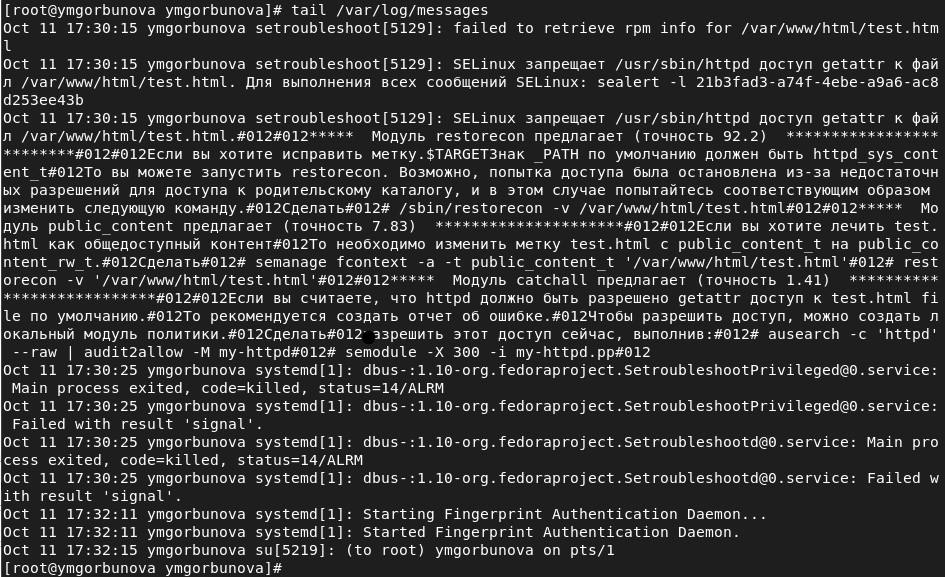


Figure 14: Пункт 15 (2)

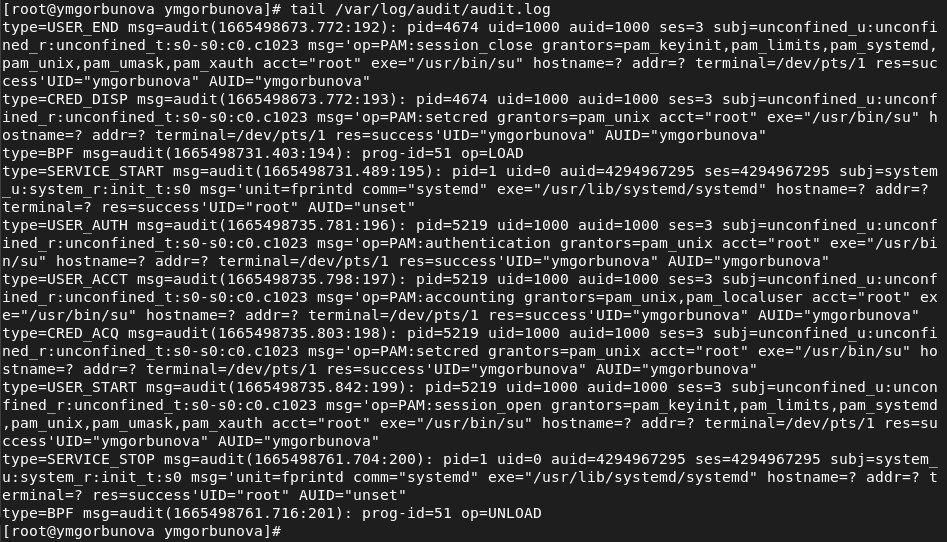


Figure 15: Пункт 15 (3)

1. Попробуйте запустить веб-сервер Apache на прослушивание ТСР-порта 81 (а не 80, как рекомендует IANA и прописано в /etc/services). Для этого в файле /etc/httpd/httpd.confнайдите строчку Listen 80и замените её на Listen 81 (fig. 16).

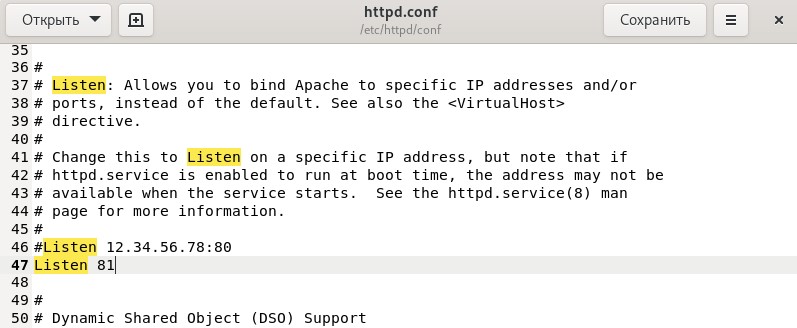


Figure 16: Пункт 16

1. Выполните перезапуск веб-сервера Apache. Произошёл сбой? Поясните почему? (fig. 17)

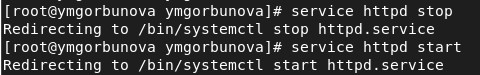


Figure 17: Пункт 17

1. Проанализируйте лог-файлы: tail -nl /var/log/messages Просмотрите файлы /var/log/http/error\_log, /var/log/http/access\_log и /var/log/audit/audit.log и выясните, в каких файлах появились записи (fig. 18).

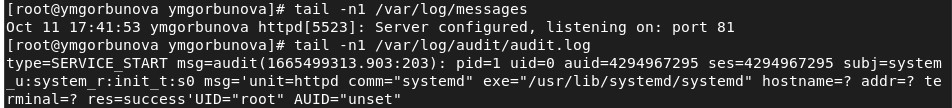


Figure 18: Пункт 18

1. Выполните команду semanage port -a -t http\_port\_t -р tcp 81 После этого проверьте список портов командой semanage port -l | grep http\_port\_t Убедитесь, что порт 81 появился в списке (fig. 19-fig. 20).

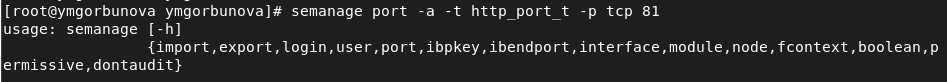


Figure 19: Пункт 19 (1)

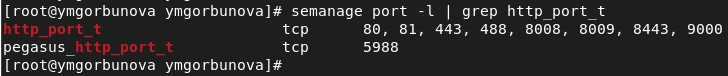


Figure 20: Пункт 19 (2)

1. Попробуйте запустить веб-сервер Apache ещё раз. Поняли ли вы, почему он сейчас запустился, а в предыдущем случае не смог? – В данном случае сервер запустился в обоих случаях, потому что была выполнена предварительная посготовка лабораторного стенда (см. Подготовка лабораторного стенда. Пункт 5).
2. Верните контекст httpd\_sys\_cоntent\_\_t к файлу /var/www/html/ test.html: chcon -t httpd\_sys\_content\_t /var/www/html/test.html После этого попробуйте получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1:81/test.html. Вы должны увидеть содержимое файла — слово «test» (fig. 21).

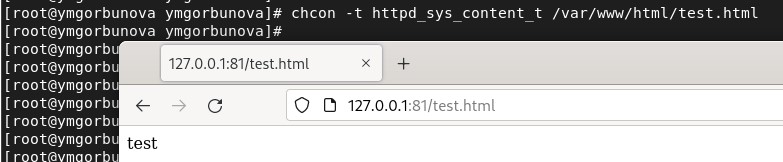


Figure 21: Пункт 21

1. Исправьте обратно конфигурационный файл apache, вернув Listen 80.
2. Удалите привязку http\_port\_t к 81 порту: semanage port -d -t http\_port\_t -p tcp 81 и проверьте, что порт 81 удалён.
3. Удалите файл /var/www/html/test.html: rm /var/www/html/test.html (fig. 22).

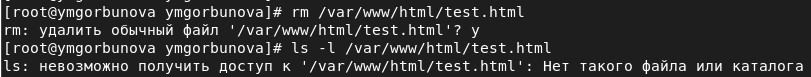


Figure 22: Пункт 24

# 4 Выводы

Развиты навыки администрирования ОС Linux. Получено первое практическое знакомство с технологией SELinux. Проверена работа SELinx на практике совместно с веб-сервером Apache.

# 5 Список литературы

1. Методические материалы курса