Отчёт по лабораторной работе №6

Дисциплина: Информационная безопасность

Выполнил: Танрибергенов Эльдар

Содержание

Список иллюстраций

# 1 Цель работы

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

# 2 Задания

* Выполнить указания к работе
* Проверить работу *SELinux* на практике совместно с веб-сервером *Apache*

# 3 Указания к работе

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. Подготовка лабораторного стенда и методические рекомендации. 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. При необходимости администратор должен разбираться в работе *SELinux* и уметь как исправить конфигурационный файл */etc/selinux/config*, так и проверить используемый режим и политику. Необходимо, чтобы был установлен веб-сервер *Apache*.

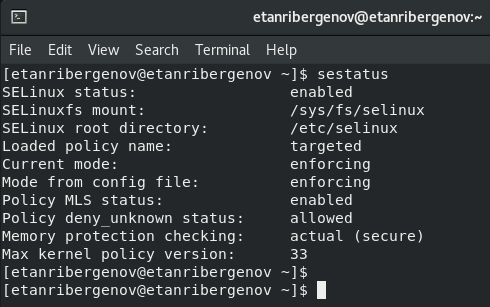


Рис. 1: Статус SELinux

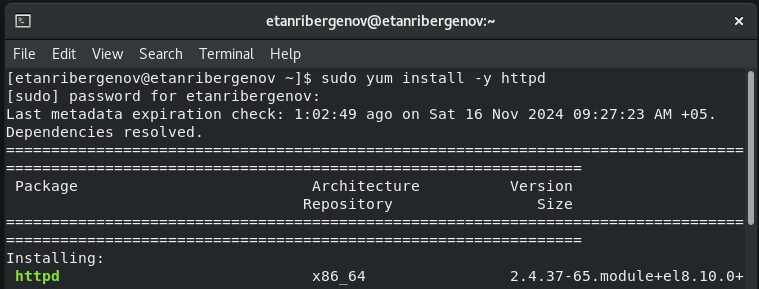


Рис. 2: Установка веб-сервера Apache

В конфигурационном файле */etc/httpd/conf/httpd.conf* необходимо задать параметр *ServerName*:

ServerName test.ru

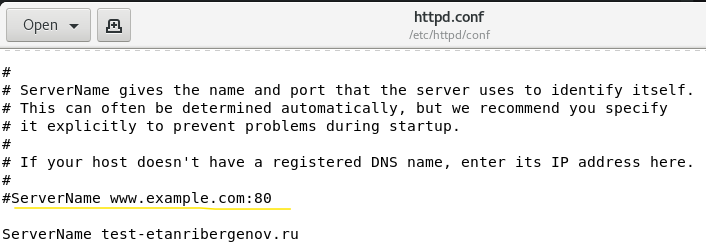


Рис. 3: Установка параметра в конф. файле веб-сервера Apache

чтобы при запуске веб-сервера не выдавались лишние сообщения об ошибках, не относящихся к лабораторной работе. Также необходимо проследить, чтобы пакетный фильтр был отключён или в своей рабочей конфигурации позволял подключаться к 80-у и 81-у портам протокола tcp. 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

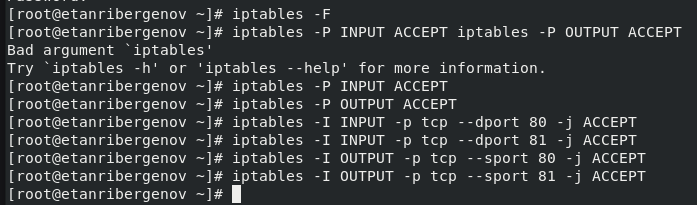


Рис. 4: Отключение пакетного фильтра и разрешение подключения к портам 80 и 81 портам протокола tcp

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

# 4 Выполнение работы

1. Вошёл в систему с полученными учётными данными и убедился, что *SELinux* работает в режиме *enforcing* политики *targeted* с помощью команд ***getenforce*** и ***sestatus***.

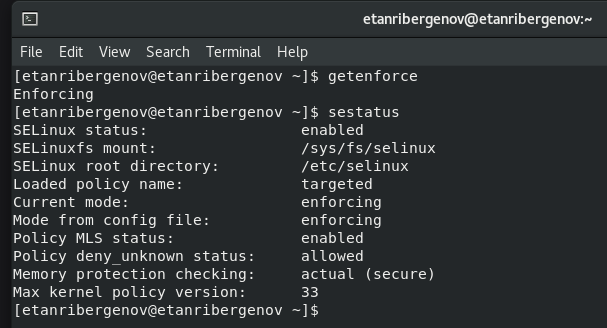


Рис. 5: Проверка того, что SeLinux работает в режиме enforcing политики targeted

1. Обратился с помощью браузера к веб-серверу, запущенному на компьютере - неуспешно. Проверил статус веб-сервера - он не включен.

service httpd status

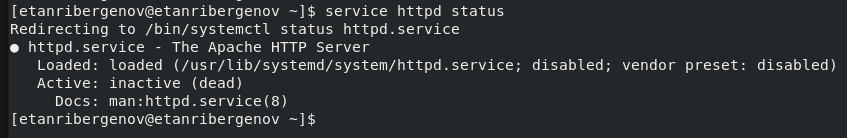


Рис. 6: Проверка статуса веб-сервера

Запустил его.

service httpd start

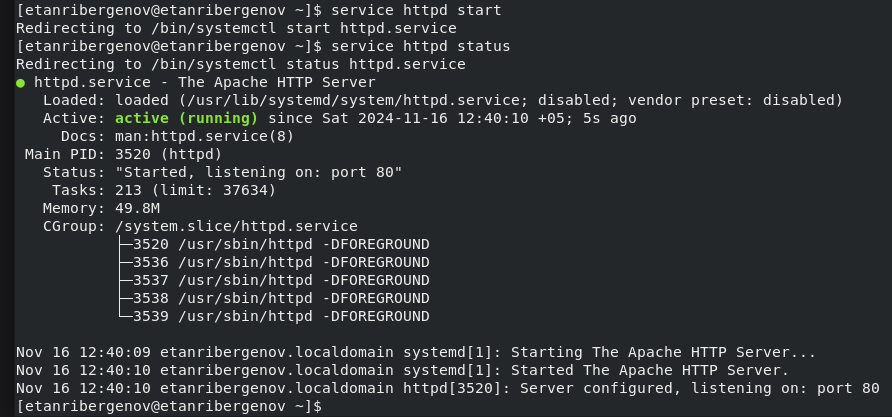


Рис. 7: Запуск веб-сервера

1. Нашёл веб-сервер *Apache* в списке процессов, определил его контекст безопасности - *httpd\_t*.

ps auxZ | grep httpd

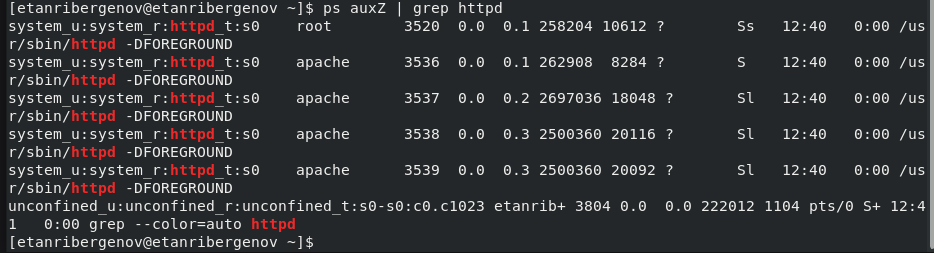


Рис. 8: Процесс веб-сервера Apache в списке процессов

1. Посмотрел текущее состояние переключателей *SELinux* для *Apache* с помощью команды

sestatus -b httpd

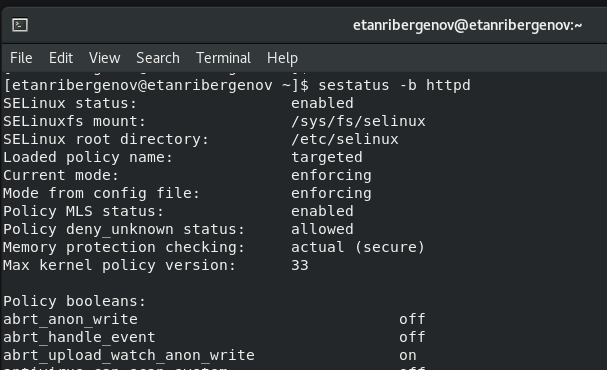


Рис. 9: Просмотр состояния переключателей SELinux для Apache

Многие из них находятся в положении «off».

1. Посмотрел статистику по политике с помощью команды *seinfo*, также определил множество пользователей, ролей, типов.

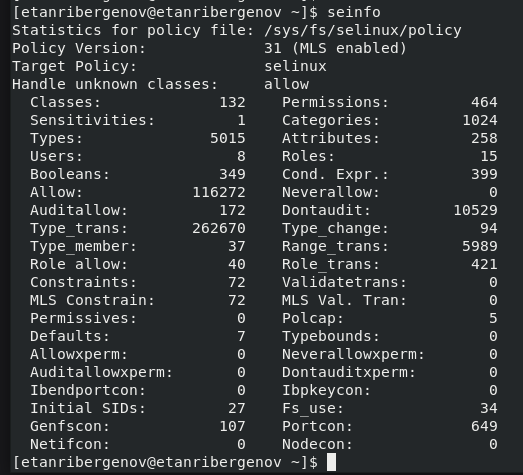


Рис. 10: Статистика по политике

1. Определил тип файлов и поддиректорий, находящихся в директории */var/www*, с помощью команды

ls -lZ /var/www

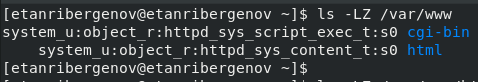


Рис. 11: Просмотр типов файлов и поддиректорий

1. Определил тип файлов, находящихся в директории */var/www/html*:

ls -lZ /var/www/html

Рис. 12: Просмотр типов файлов, находящихся в директории /var/www/html

Рис. 12: Просмотр типов файлов, находящихся в директории /var/www/html

1. Определил круг пользователей, которым разрешено создание файлов в директории */var/www/html*.

ls -l var/www | grep html

Рис. 13: Просмотр атрибутов директории для определения круга пользователей, которым разрешено создание файлов в директории

Рис. 13: Просмотр атрибутов директории для определения круга пользователей, которым разрешено создание файлов в директории

Только у суперпользователя есть разрешение на запись в директорию.

1. Создал от имени суперпользователя html-файл */var/www/html/test-etanribergenov.html* следующего содержания:

<html>  
<body>test</body>  
</html>

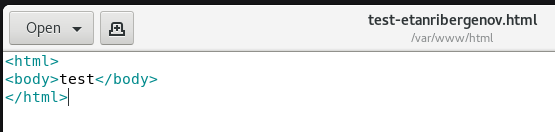


Рис. 14: Содержимое HTML-файла test-etanribergenov

1. Проверил контекст созданного файла.

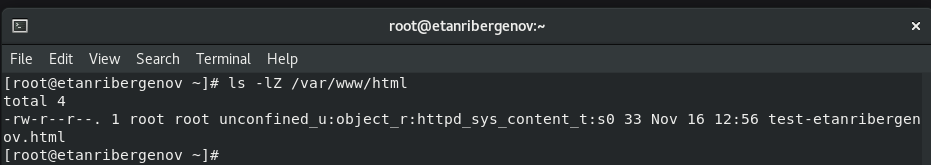


Рис. 15: Проверка контекста безопасности созданного файла

Контекст, присваиваемый по умолчанию вновь созданным файлам в директории */var/www/html* - *httpd\_sys\_content\_t*.

1. Обратился к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test-etanribergenov.html. Убедился, что файл был успешно отображён.

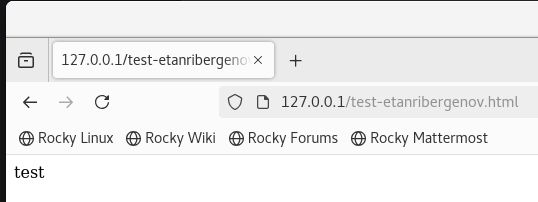


Рис. 16: Обращение к файлу через веб-сервер в браузере

1. Изучил справку *man httpd\_selinux* и выяснил, какие контексты файлов определены для httpd.

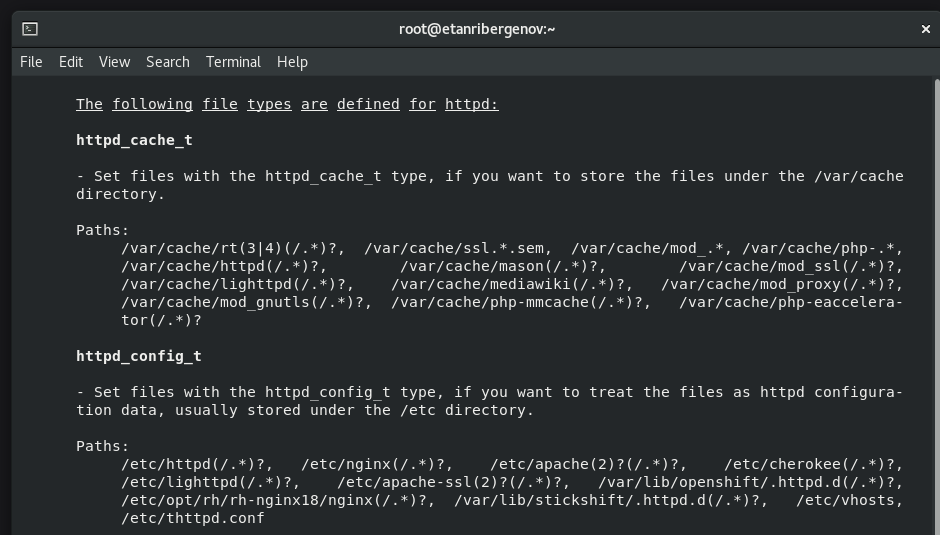


Рис. 17: Информация о контекстах безопасности SELinux файлов определённых для httpd

Сопоставил их с типом файла *test-etanribergenov.html*. Да, тип контекста этого файла есть в справке.

ls -Z /var/www/html/test-etanribergenov.html

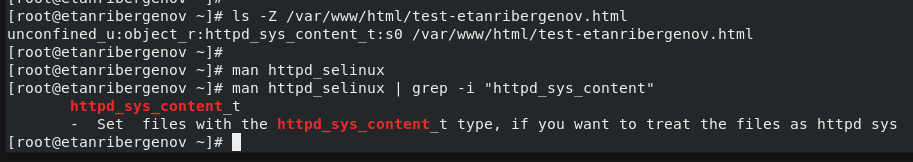


Рис. 18: Сопоставление типа файла с типами, описанными в справке

Рассмотрим полученный контекст детально. Так как по умолчанию пользователи CentOS являются свободными от типа (unconfined в переводе с англ. означает свободный), созданному файлу *test-etanribergenov.html* был 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. 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.

1. Изменил контекст файла */var/www/html/test-etanribergenov.html* с *httpd\_sys\_content\_t* на *samba\_share\_t*, к которому процесс *httpd* не имеет доступа. После этого проверил, что контекст поменялся.

chcon -t samba\_share\_t /var/www/html/test-etanribergenov.html  
ls -Z /var/www/html/test-etanribergenov.html



Рис. 19: Изменение контекста SELinux для файла

1. Попробовал ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test-etanribergenov.html. Получил сообщение об ошибке:

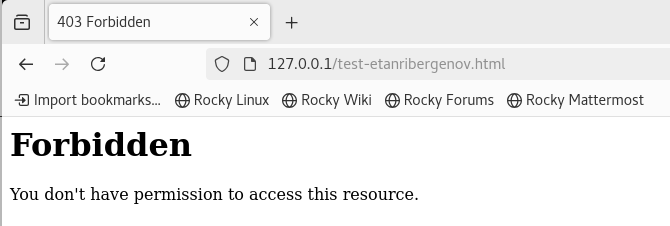


Рис. 20: Попытка получения доступа к файлу через веб-сервер в браузере

1. Файл не был отображён, несмотря на то, что права доступа позволяют читать этот файл любому пользователю, потому что контекст безопасности изменился и *httpd* больше не имеет доступа к файлу.

Просмотрел системный лог-файл:

tail /var/log/messages

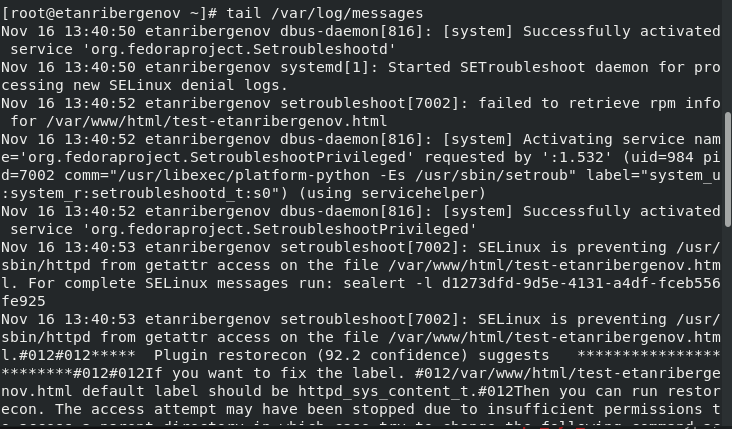


Рис. 21: Сообщения в системном лог-файле

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*.

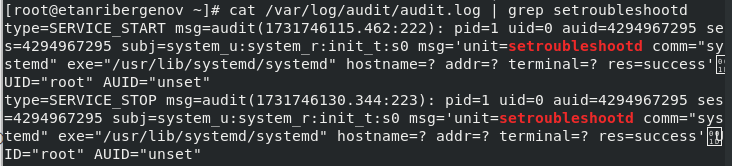


Рис. 22: Сообщения в лог-файле процесса audtd

1. Попробовал 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.conf* нашёл строчку ***Listen 80*** и заменил её на ***Listen 81***.

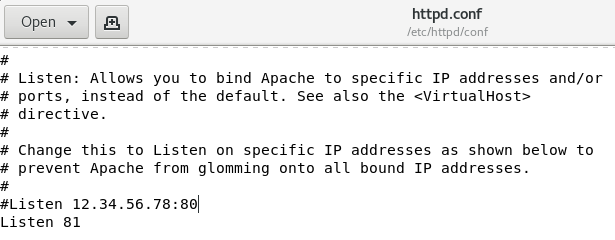


Рис. 23: Изменение порта прослушивания веб-сервера Apache

1. Выполнил перезапуск веб-сервера *Apache*. Сбой не произошёл. Это потому, что в разделе подготовки лабораторного стенда были приведены команды, разрешающие веб-серверу *Apache* прослушивать *tcp* порт 81.

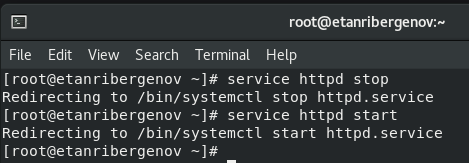


Рис. 24: Выполнение перезапуска веб-сервера

1. Проанализируйте лог-файлы:

tail -nl /var/log/messages

Рис. 25: Просмотр системного лог-файла

Рис. 25: Просмотр системного лог-файла

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.

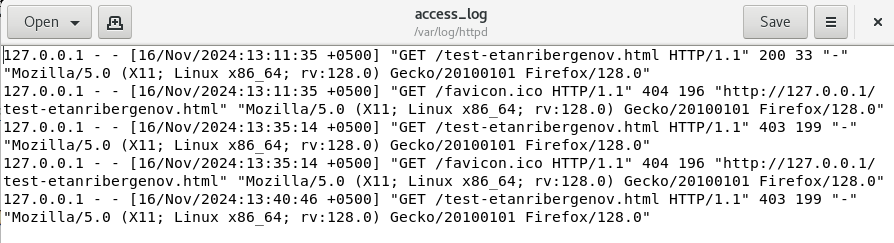


Рис. 26: Просмотр лог-файла access\_log

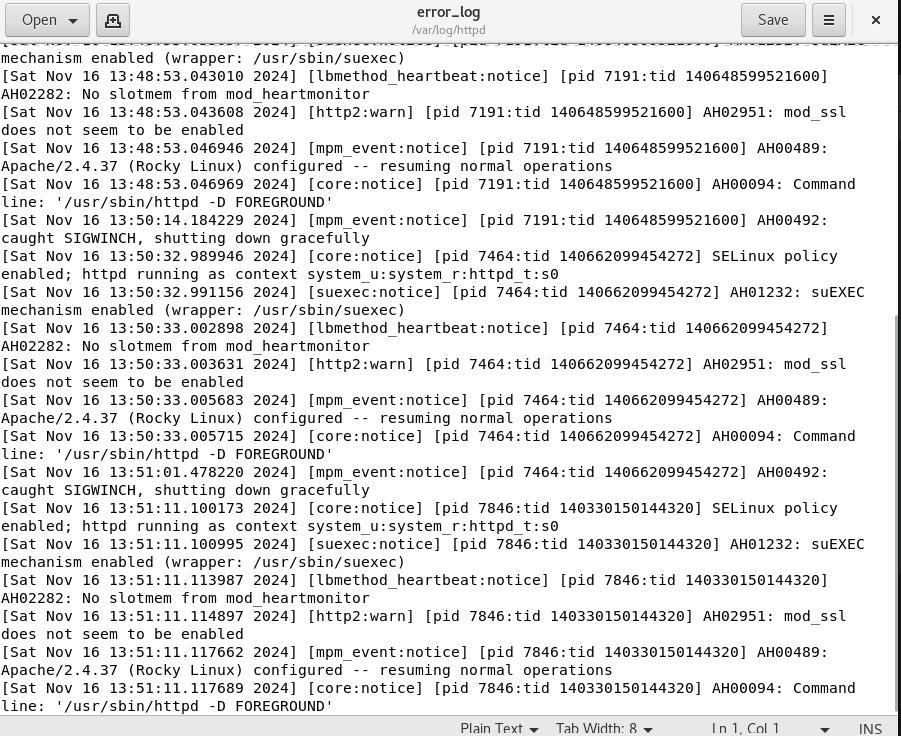


Рис. 27: Просмотр лог-файла error\_log

Сообщений об ошибке нет.

1. Выполнил команду

semanage port -a -t http\_port\_t -р tcp 81

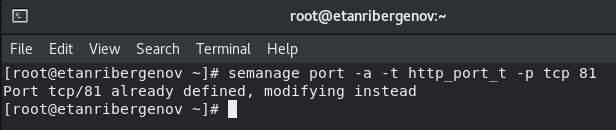


Рис. 28: Добавление tcp-порта 81 в список портов SELinux для веб-сервера

После этого проверил список портов командой

semanage port -l | grep http\_port\_t

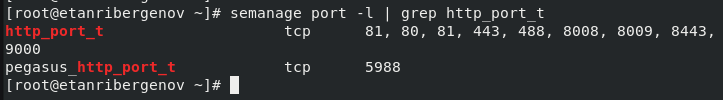


Рис. 29: Просмотр списка портов SELinux для веб-сервера

Убедился, что порт 81 появился в списке.

1. Попробовал запустить веб-сервер *Apache* ещё раз. Он вновь запустился без проблем, как и в прошлый раз.

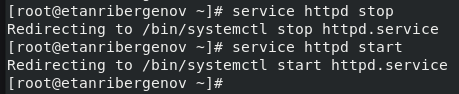


Рис. 30: Перезапуск веб-сервера Apache

1. Вернул контекст \*httpd\_sys\_cоntent\_\_t\* к файлу */var/www/html/test-etanribergenov.html*:

chcon -t httpd\_sys\_content\_t /var/www/html/test-etanribergenov.html

Рис. 31: Возвращение контекста SELinux к файлу

Рис. 31: Возвращение контекста SELinux к файлу

После этого попробовал получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1:81/test-etanribergenov.html. Увидел содержимое файла — слово «test».

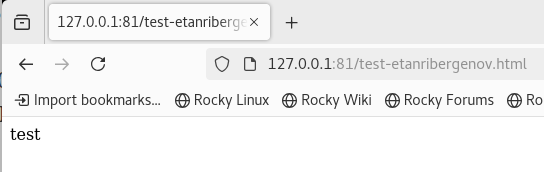


Рис. 32: Получение доступа к файлу через веб-сервер в браузере

1. Исправил обратно конфигурационный файл *apache*, вернув ***Listen 80***.

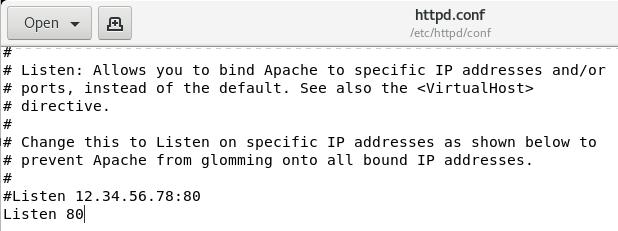


Рис. 33: Изменение конф. файла веб-сервера Apache

1. Удалил привязку *http\_port\_t* к 81 порту и проверил, что порт 81 удалён.

semanage port -d -t http\_port\_t -p tcp 81  
semanage port -l | grep http\_port\_t

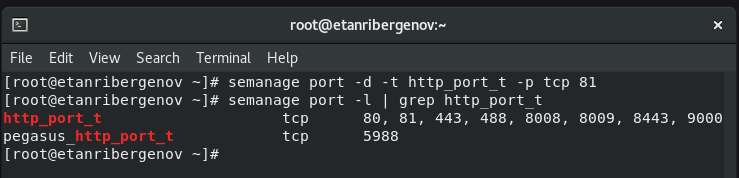


Рис. 34: Удаление привязки http\_port\_t к 81 порту и проверка выполнения действия

1. Удалил файл */var/www/html/test-etanribergenov.html*

rm /var/www/html/test-etanribergenov.html

Рис. 35: Удаление файла

Рис. 35: Удаление файла

# 5 Выводы

В результате выполнения работы я развил навыки администрирования ОС *Linux*. Получил первое практическое знакомство с технологией *SELinux*. Проверил работу *SELinux* на практике совместно с веб-сервером *Apache*.