Лабораторная работа №6

Дисциплина: Информационная безопасность

Губина Ольга Вячеслвовна

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.

# 2 Задание

* Изучить на практике работу SELinx и Apache.

# 3 Теоретическое введение

**SELinux** (SELinux) — это система принудительного контроля доступа, реализованная на уровне ядра. Впервые эта система появилась в четвертой версии CentOS, а в 5 и 6 версии реализация была существенно дополнена и улучшена. Эти улучшения позволили SELinux стать универсальной системой, способной эффективно решать массу актуальных задач. Стоит помнить, что классическая система прав Unix применяется первой, и управление перейдет к SELinux только в том случае, если эта первичная проверка будет успешно пройдена.

Для того, чтобы понять, в чем состоит практическая ценность SELinux, рассмотрим несколько примеров, когда стандартная система контроля доступа недостаточна. Если SELinux отключен, то вам доступна только классическая дискреционная система контроля доступа, которая включает в себя DAC (избирательное управление доступом) или ACL(списки контроля доступа). То есть речь идет о манипулировании правами на запись, чтение и исполнение на уровне пользователей и групп пользователей, чего в некоторых случаях может быть совершенно недостаточно. Например:

* Администратор не может в полной мере контролировать действия пользователя. Например, пользователь вполне способен дать всем остальным пользователям права на чтение собственных конфиденциальных файлов, таких как ключи SSH.
* Процессы могут изменять настройки безопасности. Например, файлы, содержащие в себе почту пользователя должны быть доступны для чтения только одному конкретному пользователю, но почтовый клиент вполне может изменить права доступа так, что эти файлы будут доступны для чтения всем.
* Процессы наследуют права пользователя, который их запустил. Например, зараженная трояном версия браузера Firefox в состоянии читать SSH-ключи пользователя, хотя не имеет для того никаких оснований.[1]

**Apache** – это свободное программное обеспечение для размещения веб-сервера. Он хорошо показывает себя в работе с масштабными проектами, поэтому заслуженно считается одним из самых популярных веб-серверов. Кроме того, Apache очень гибок в плане настройки, что даёт возможность реализовать все особенности размещаемого веб-ресурса.[2]

Установить веб-сервер Apache можно слеудющим образом. Откройте окно терминала и обновите списки пакетов репозитория, введя следующее: sudo yum update

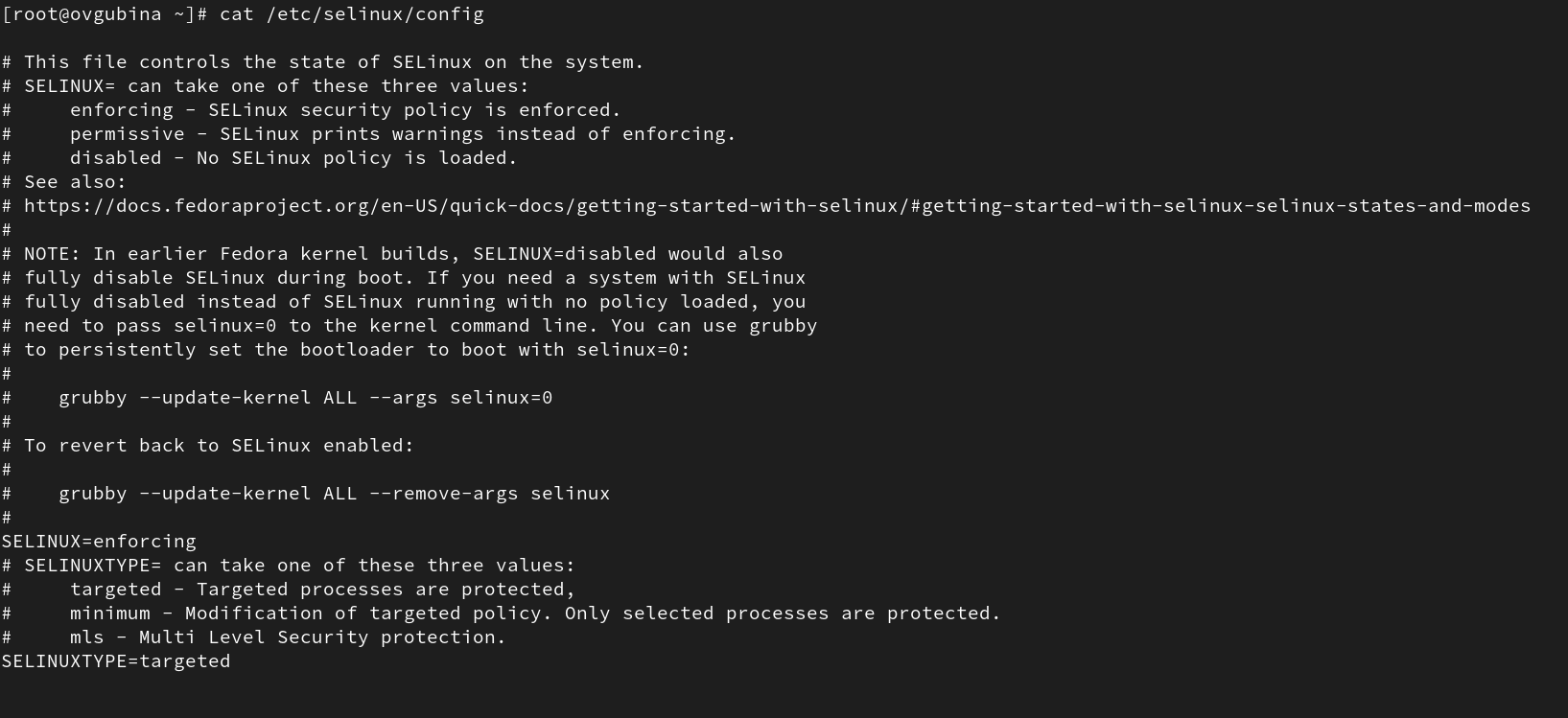
Теперь вы можете установить Apache с помощью команды: sudo yum –y install httpd

**httpd** - это имя службы Apache в CentOS. Опция –y автоматически отвечает да на запрос подтверждения.[3]

# 4 Выполнение лабораторной работы

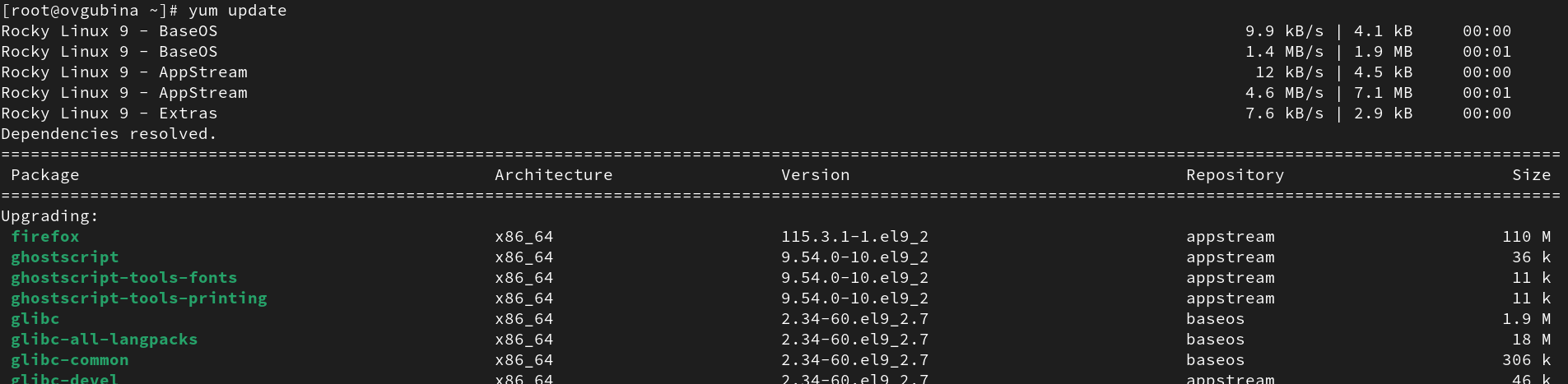
## 4.1 Подготовительный этап

Сперва проверим конфигурационный файл SELinux - видим, что политика targeted и режим enforcing используются в данном дистрибутиве по умолчанию, т.е. каких-то специальных настроек не требуется (рис. ??).

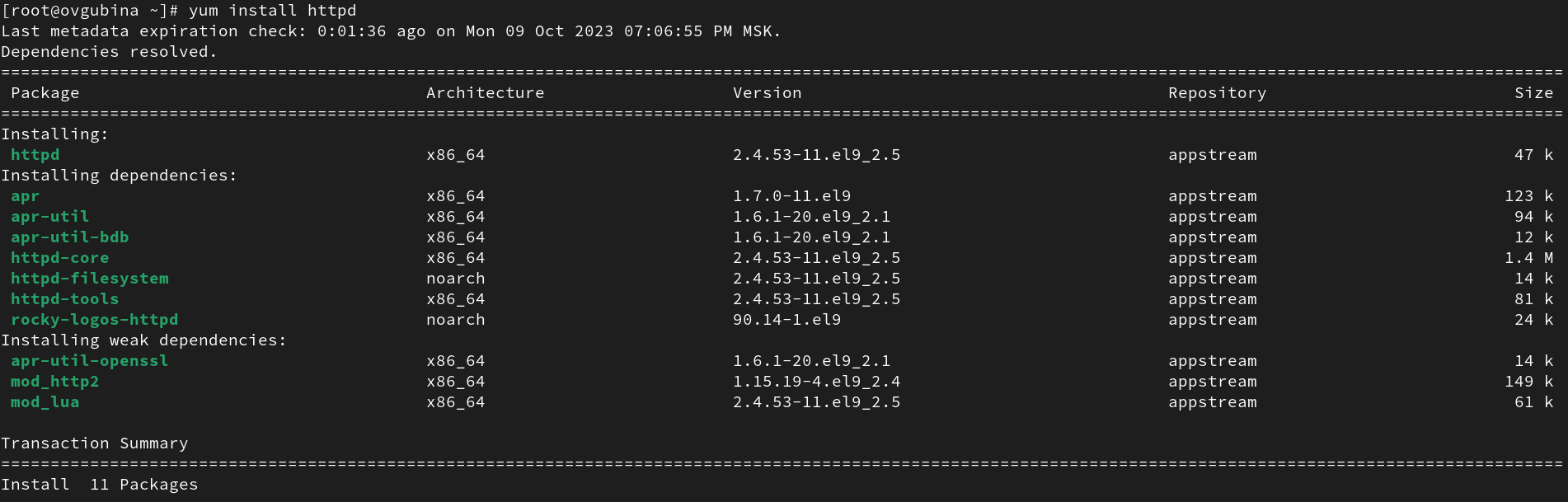


Проверка политики и режима

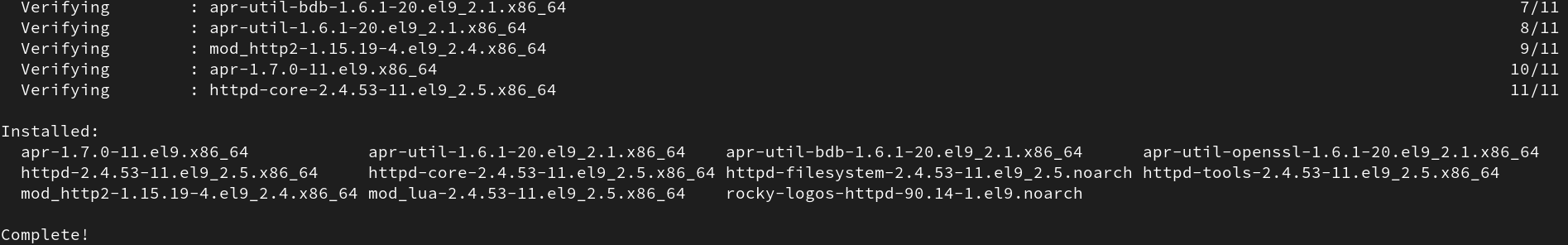
Дальше потребуется установить менеджер Apache, для этого предварительно обновим систему (рис. ??), только после этого устанавливаем Apache (httpd) (рис. ??-??).



Обновление

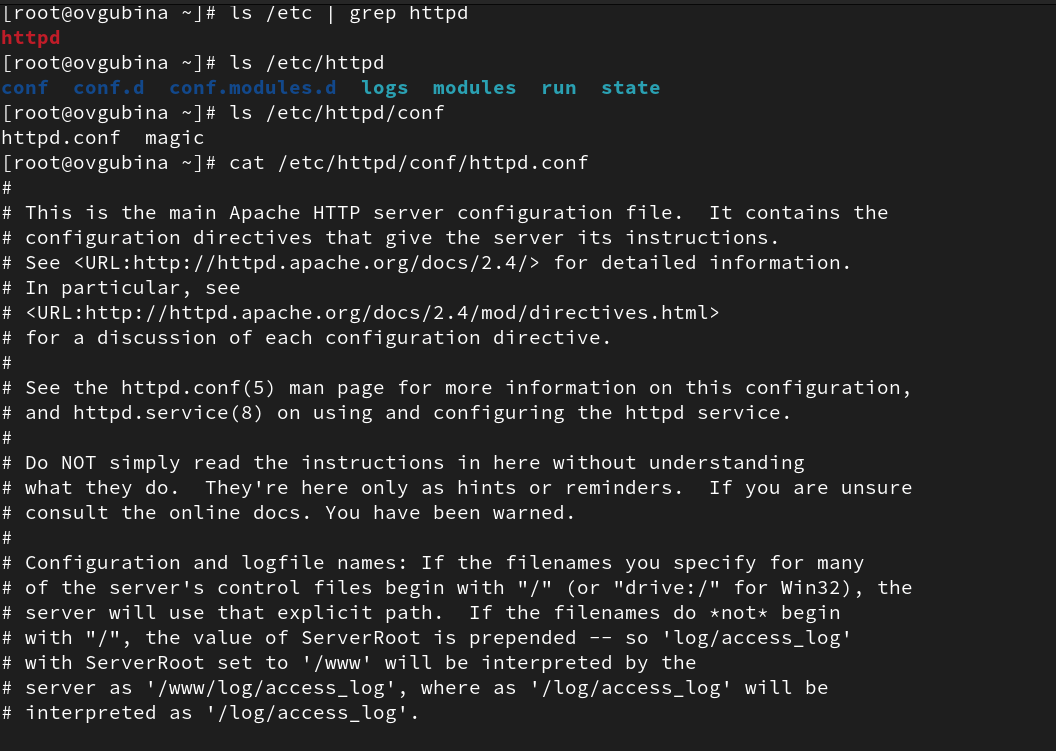


Загрузка Apache

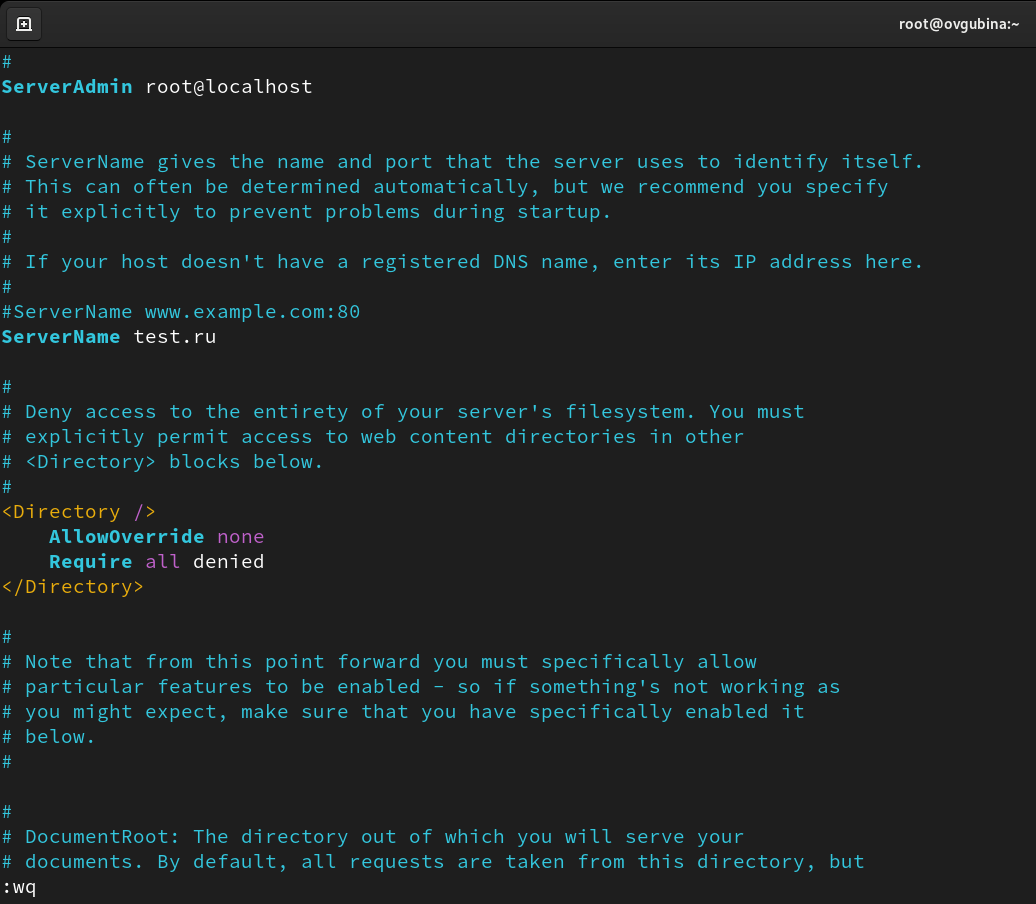


Конец загрузки Apache

Далее зададим ServerName test.ru в конфигурационной файле httpd (рис. ??), для этого сперва найдем, где он находитсЯ (рис. ??).

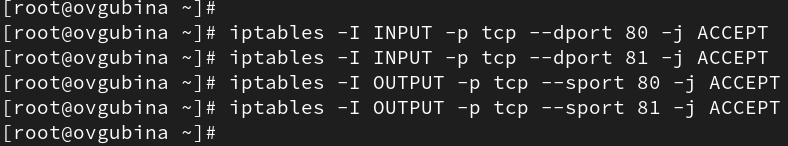


Расположение конфигурационного файла httpd



Задаем ServerName

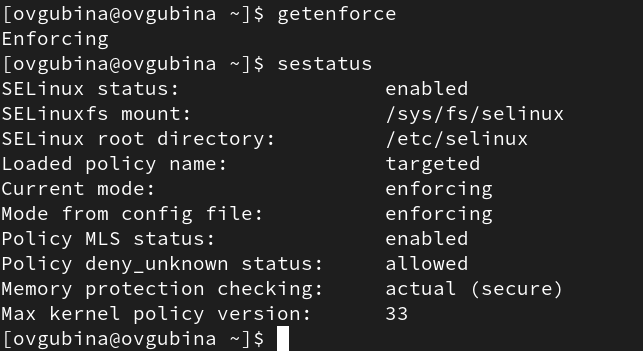
Чтобы пакетный фильтр в своей рабочей конфигурации позволял подключаться к 80-у и 81-у портам протокола tcp добавим разрешающие правила (рис. ??):



Задаем ServerName

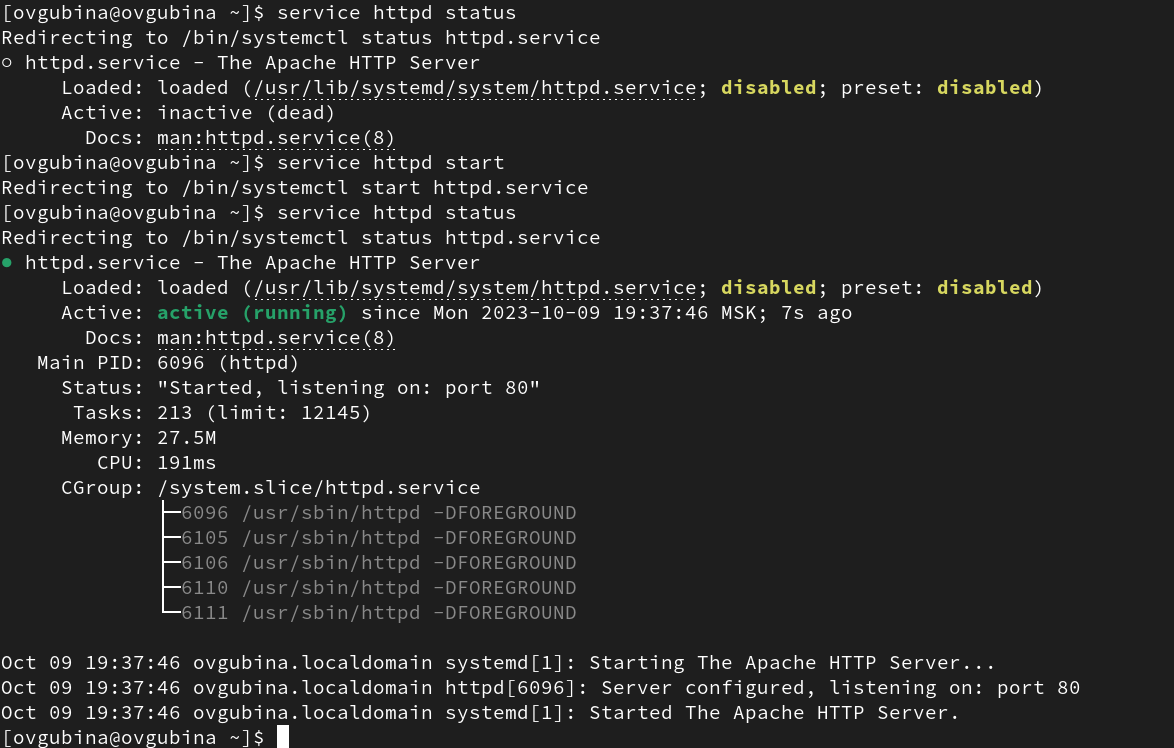
## 4.2 Порядок выполнения работы

1. Войдите в систему с полученными учётными данными и убедитесь, что SELinux работает в режиме enforcing политики targeted с помощью команд getenforce и sestatus (рис. ??).



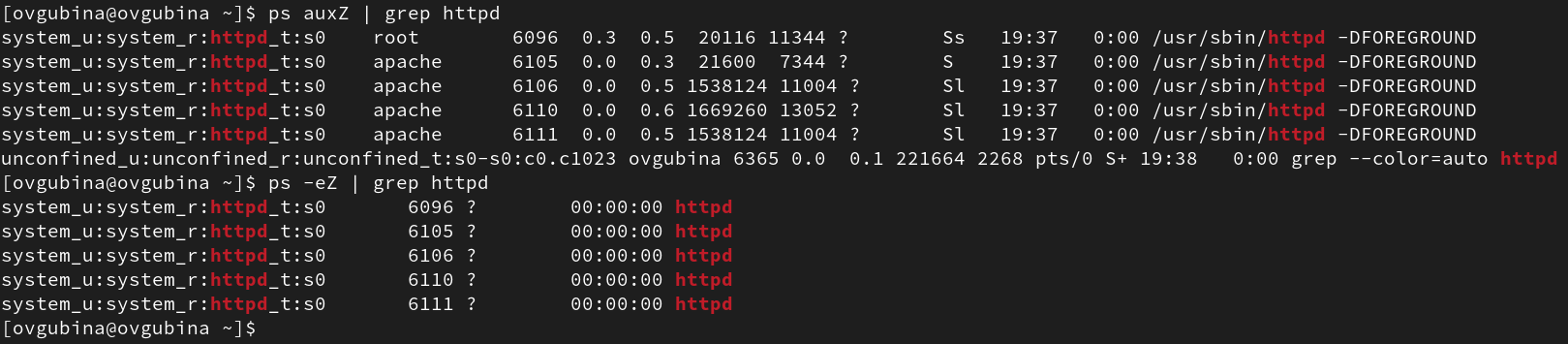
Режим enforcing политики targeted

1. Обратитесь с помощью браузера к веб-серверу, запущенному на вашем компьютере, и убедитесь, что последний работает: service httpd status. Видим, что он неактивен, поэтому запускаем его командой service httpd start, после чего снова проверяем, в этот раз серсвис активен (рис. ??).



Запуск

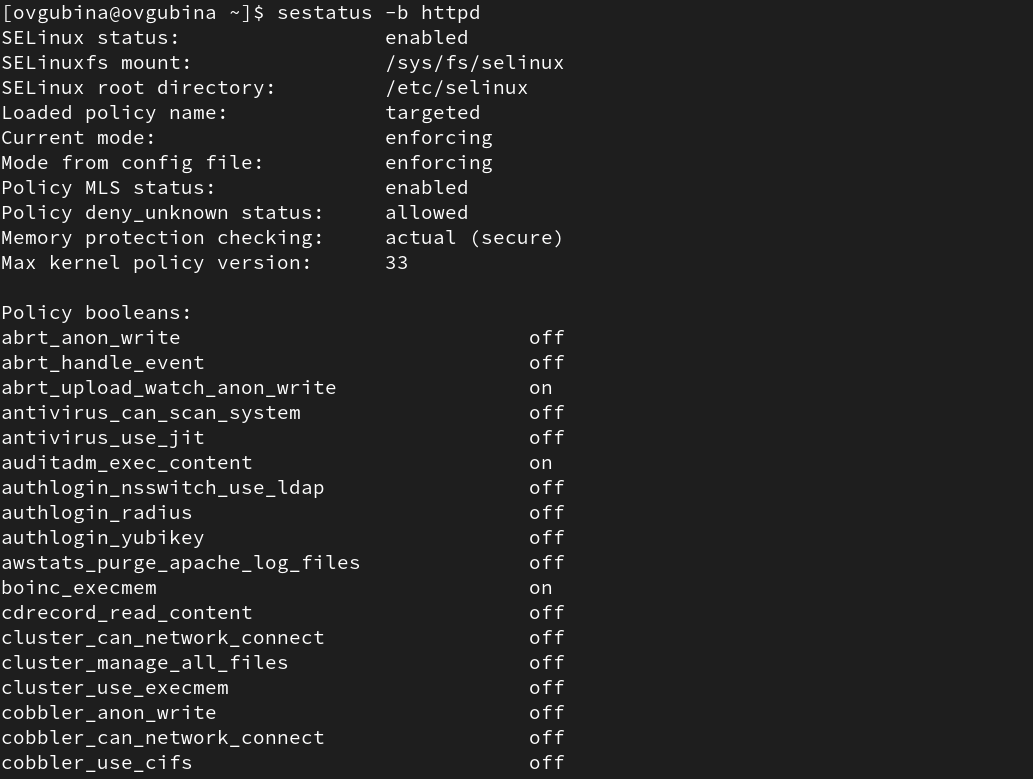
1. Найдите веб-сервер Apache в списке процессов, определите его контекст безопасности и занесите эту информацию в отчёт. Например, можно использовать команду ps auxZ | grep httpd (??).



Контекст безопасности

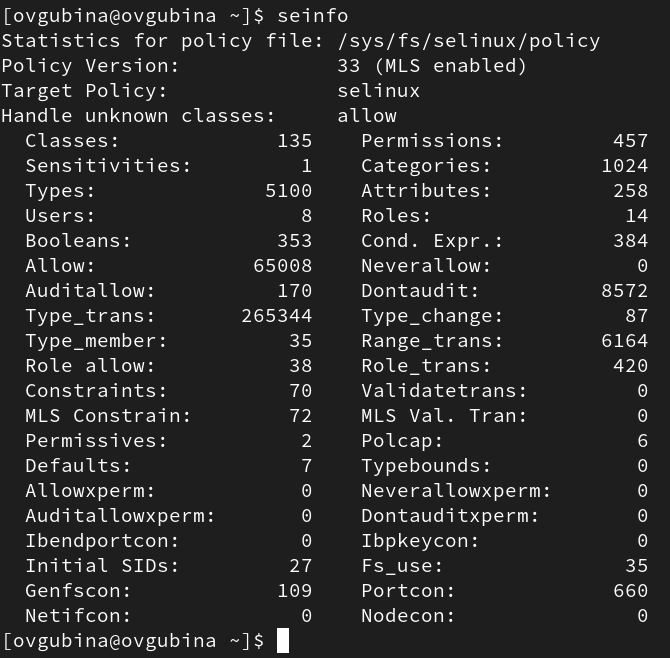
Видим, что веб-сервер имеет контекст безопасности httpd\_t.

1. Посмотрите текущее состояние переключателей SELinux для Apache с помощью команды sestatus -b httpd (рис. ??). Многие из них находятся в положении «off».



Переключатели SELinux для Apache

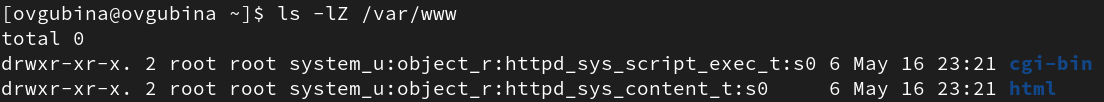
1. Посмотрите статистику по политике с помощью команды seinfo, также определите множество пользователей, ролей, типов (рис. ??).



Статистика по политике

Число типов = 5100, ролей = 14, пользователей = 8.

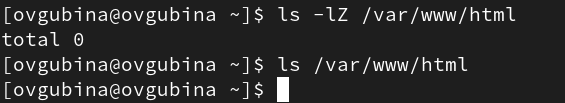
1. Определите тип файлов и поддиректорий, находящихся в директории /var/www, с помощью команды ls -lZ /var/www (рис. ??).



Типы файлов

В каталоге находятся только директории.

1. Определите тип файлов, находящихся в директории /var/www/html: ls -lZ /var/www/html (рис. ??).



Типы файлов

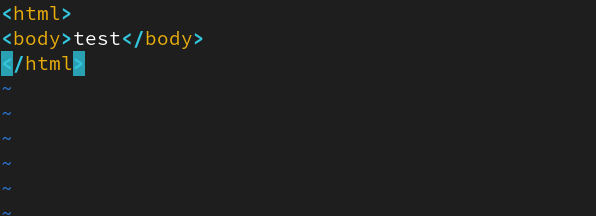
Директория пуста.

1. Определите круг пользователей, которым разрешено создание файлов в директории /var/www/html - это только пользователь root.
2. 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 (рис. ??-??):

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

Создание html файла

Создание html файла



Создание html файла

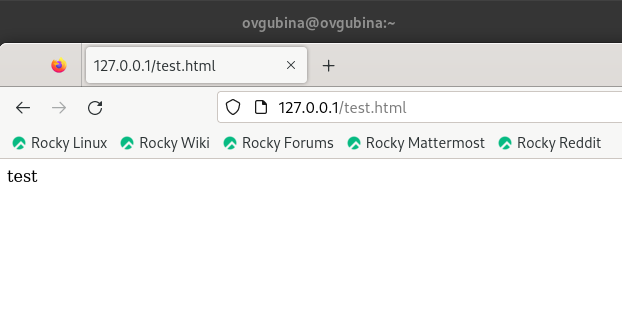
1. Проверьте контекст созданного вами файла. Занесите в отчёт контекст, присваиваемый по умолчанию вновь созданным файлам в директории /var/www/html (рис. ??).



Создание html файла

Контекст безопасности - **httpd\_sys\_content\_t**.

1. Обратитесь к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html (рис. ??). Файл был успешно отображён.



Обращение к файлу через веб-сервер

1. Изучите справку man httpd\_selinux и выясните, какие контексты файлов определены для httpd.

*Вызвать данную справку неудалось, контексты безопасновти httpd были просмотрены через интернет.*

Сопоставьте их с типом файла test.html. Проверить контекст файла можно командой ls -Z /var/www/html/test.html (рис. ??). Контекст безопасности файла - **httpd\_sys\_content\_t**. Данный контекст входит в перечень контекстов безопасности httpd.

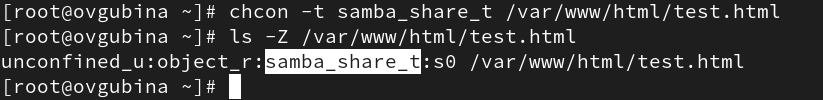
Роль **object\_r** используется по умолчанию для файлов на «постоянных» носителях и на сетевых файловых системах. Тип **httpd\_sys\_content\_t** позволяет процессу httpd получить доступ к файлу. Благодаря наличию последнего типа мы получили доступ к файлу при обращении к нему через браузер.



Контекст безропасности html файла

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 (рис. ??):

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

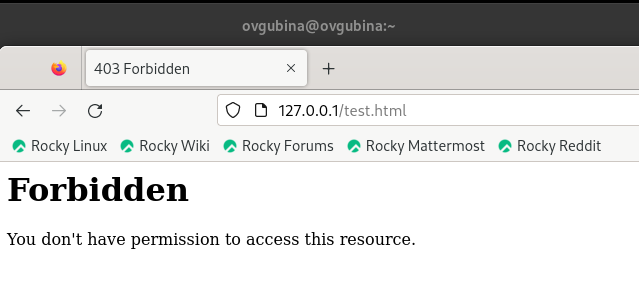


Смена контекста безропасности html файла

Видим, что контекст безопасности действительно изменился.

1. Попробуйте ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html (рис. ??).

Мы получили сообщение об ошибке: **Forbidden You don’t have permission to access /test.html on this server.**

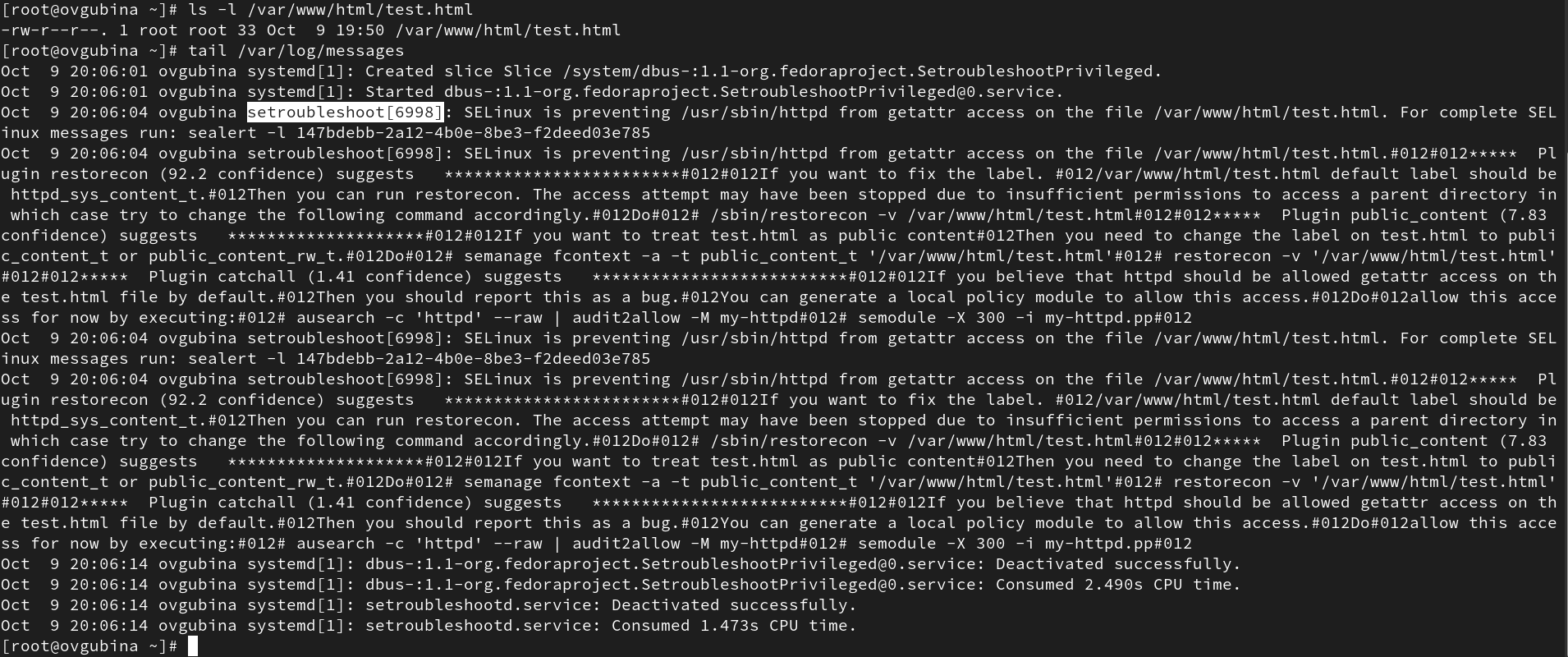


Попытка получения доступа к файлу через веб-сервер

1. Проанализируйте ситуацию. Почему файл не был отображён, если права доступа позволяют читать этот файл любому пользователю? ls -l /var/www/html/test.html (рис. ??) - нет доступа к файлу из-за недопустимого контекста безопасности для httpd.

Просмотрите log-файлы веб-сервера Apache. Также просмотрите системный лог-файл: tail /var/log/messages (рис. ??).

Если в системе окажутся запущенными процессы setroubleshootd и audtd (рис. ??), то вы также сможете увидеть ошибки, аналогичные указанным выше, в файле /var/log/audit/audit.log (рис. ??).

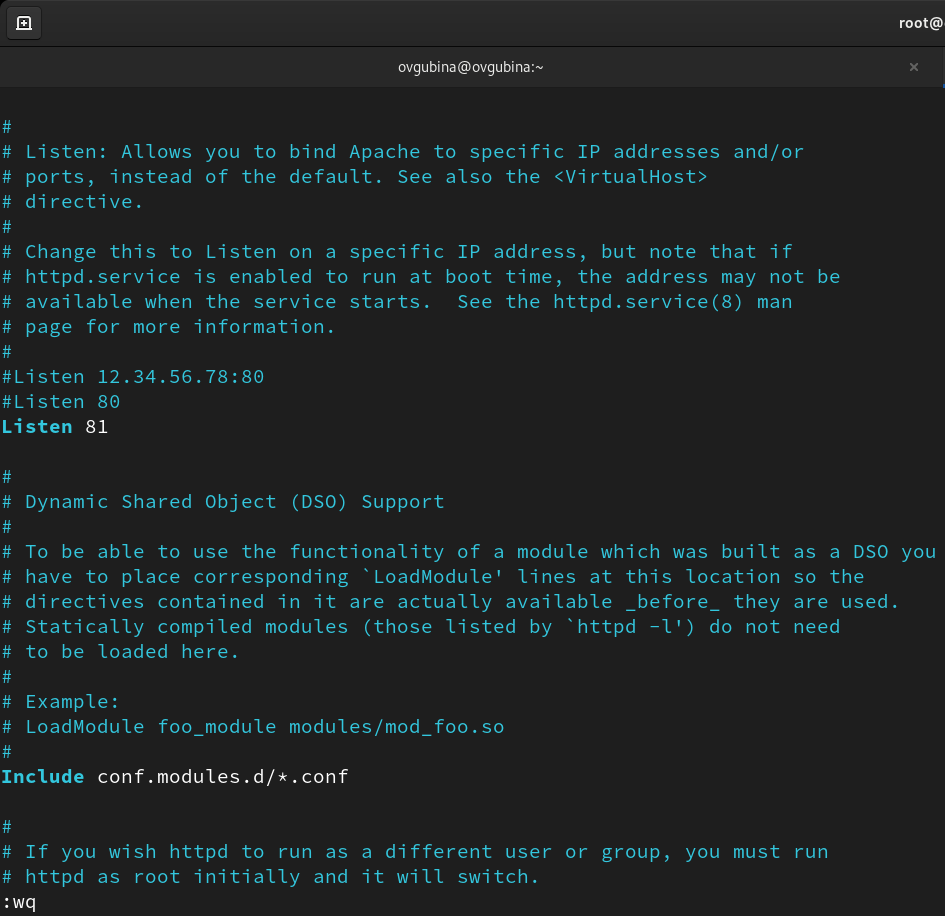


tail /var/log/messages



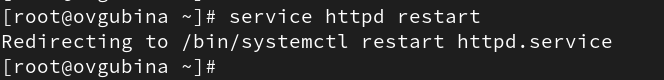
/var/log/audit/audit.log

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.conf найдите строчку Listen 80 и замените её на Listen 81 (рис. ??).



Смена прослушиваемого порта

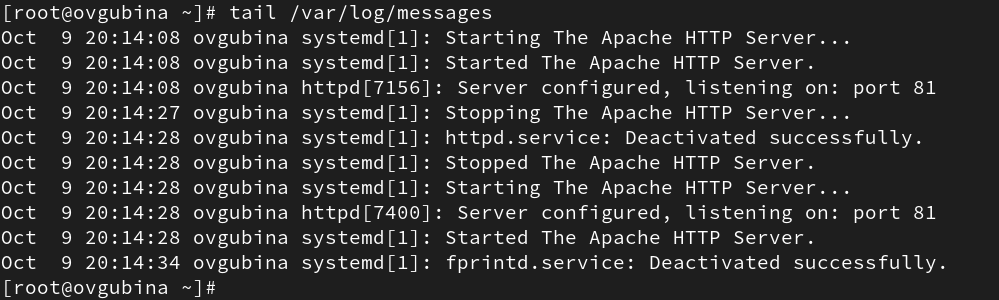
1. Выполните перезапуск веб-сервера Apache (рис. ??). Произошёл сбой? Поясните почему?



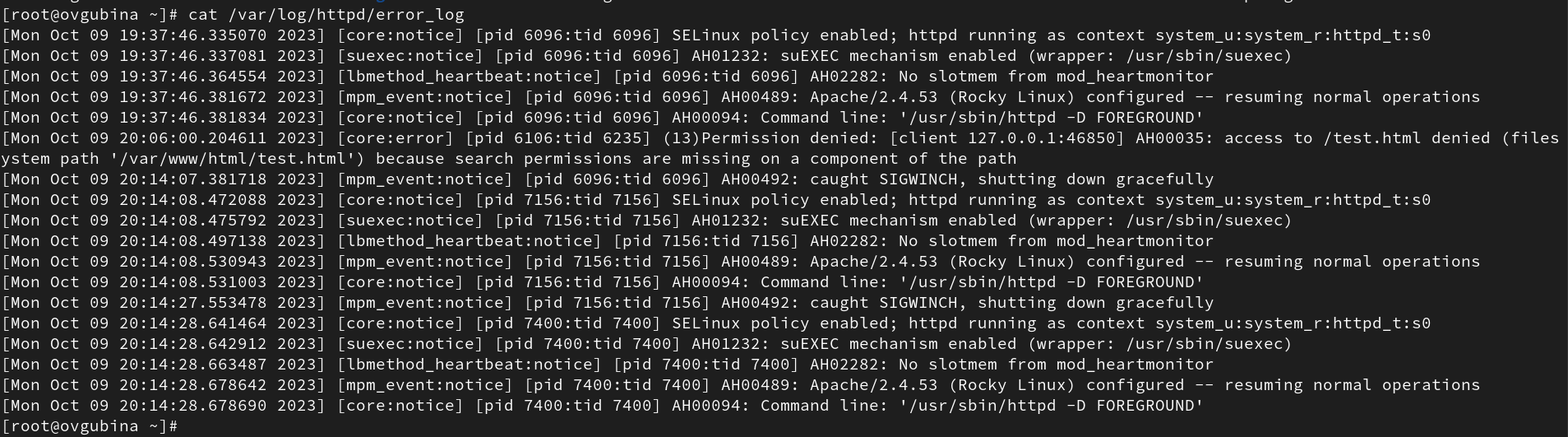
Перезапуск Apache

Никакой ошибки не возникает, поскольку в изначальных настройках системы порт 81 уже был прописан в рекомендациях системы.

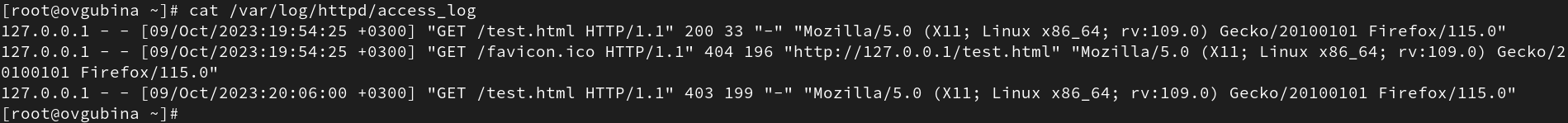
1. Проанализируйте лог-файлы: tail -nl /var/log/messages - нет никаких ошибок (рис. ??). Просмотрите файлы /var/log/http/error\_log (рис. ??), /var/log/http/access\_log (рис. ??) и /var/log/audit/audit.log (рис. ??) и выясните, в каких файлах появились записи - нет записей об ошибках, т.к. нет ошибок.



tail -nl /var/log/messages



/var/log/http/error\_log



/var/log/http/access\_log



/var/log/audit/audit.log

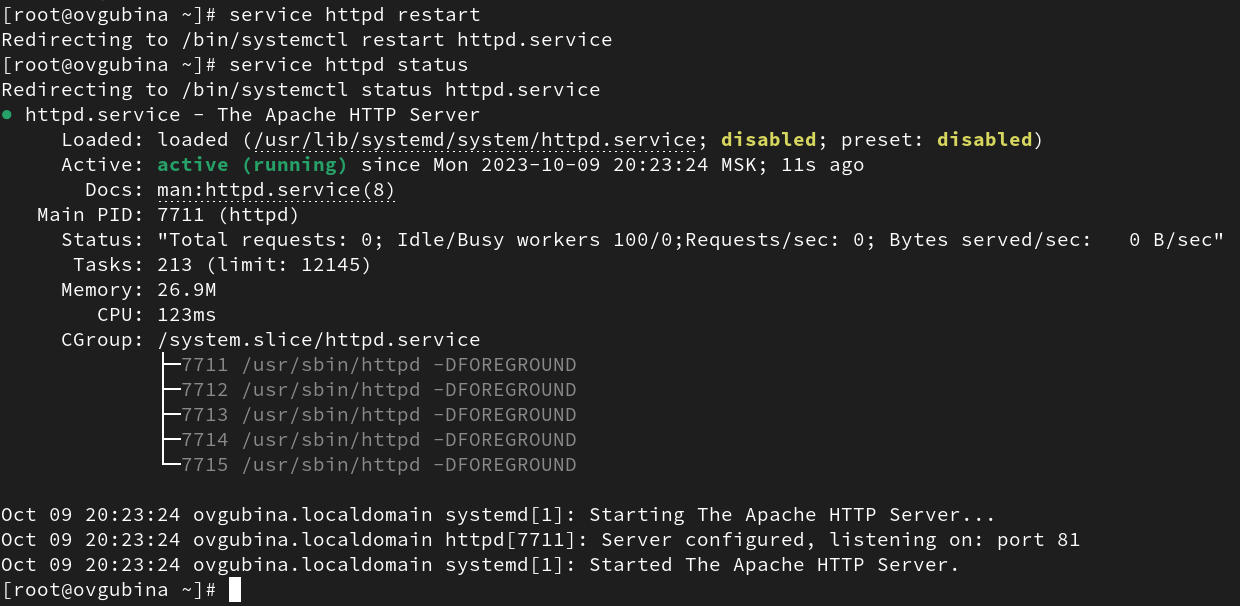
1. Выполните команду semanage port -a -t http\_port\_t -р tcp 81 После этого проверьте список портов командой semanage port -l | grep http\_port\_t (рис. ??). Добавление порта не производим, т.к. нам известно, что он и так уже добвлен - сразу смотрим список.

**Порт 81 есть в списке.**



Список портов

1. Попробуйте запустить веб-сервер Apache ещё раз. Поняли ли вы, почему он сейчас запустился, а в предыдущем случае не смог?



Запуск веб-сервера

Сервер запустился также успешно, как и в тот раз, посколько оба раза порт 81 был в списке портов.

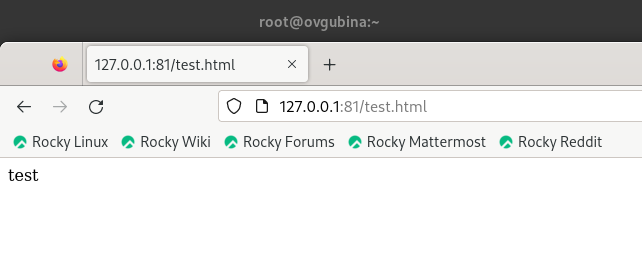
1. Верните контекст httpd\_sys\_cоntent\_\_t к файлу /var/www/html/ test.html: chcon -t httpd\_sys\_content\_t /var/www/html/test.html (рис. ??).

После этого попробуйте получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1:81/test.html (рис. ??).

Втидим содержимое файла — слово «test».

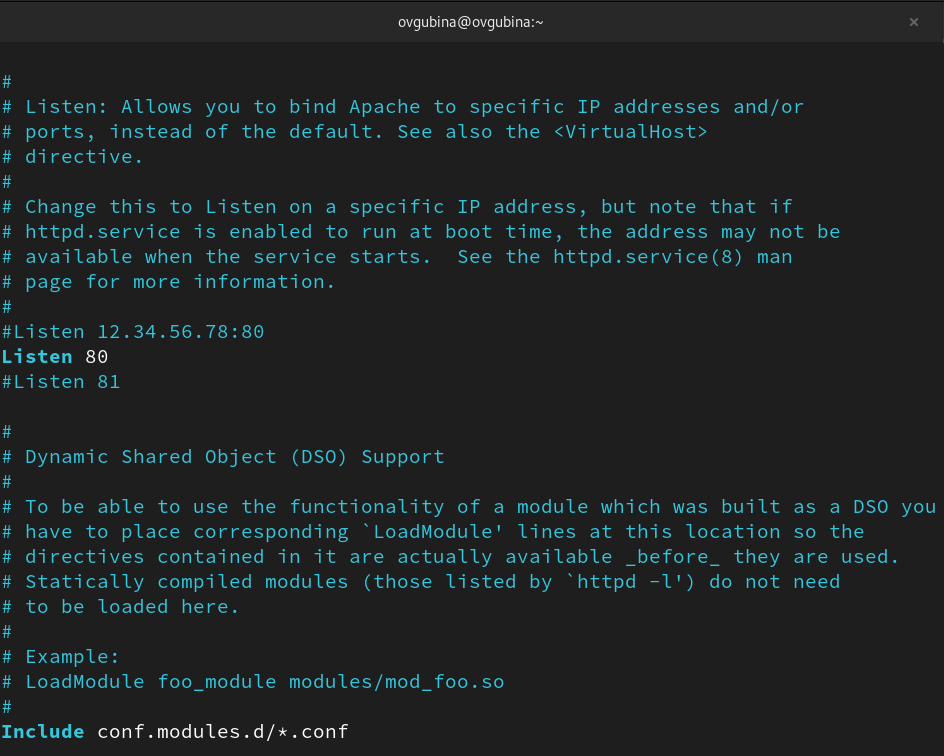


Возвращаем контекст безопасности



Получение доступа к файлу через веб-сервер

1. Исправьте обратно конфигурационный файл apache, вернув Listen 80 (рис. ??).

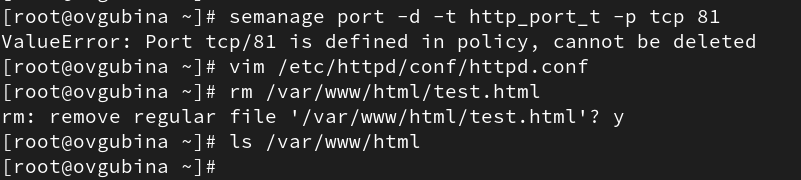


Изменение прослушиваемого порта

1. Удалите привязку http\_port\_t к 81 порту: semanage port -d -t http\_port\_t -p tcp 81 и проверьте, что порт 81 удалён.

Мы не можем этого сделать, поскольку это была изначальная настройка системы, работает - не трогай, поэтому мы пока не станем этого делать (рис. ??).

1. Удалите файл /var/www/html/test.html (рис. ??): rm /var/www/html/test.html



Удаление файла

# 5 Выводы

Развила навыки администрирования ОС Linux. Получила первое практическое знакомство с технологией SELinux1. Проверила работу SELinx на практике совместно с веб-сервером Apache.

# Список литературы

1. SELinux – описание и особенности работы с системой [Электронный ресурс]. 2023. URL: <https://habr.com/ru/companies/kingservers/articles/209644/>.

2. Что такое Apache [Электронный ресурс]. 2023. URL: <https://eternalhost.net/blog/hosting/web-server-apache>.

3. Установка веб-сервера Apache на Linux [Электронный ресурс]. 2023. URL: <https://wiki.merionet.ru/articles/ustanovka-apache-v-linux>.