

Отчет по лабораторной работе №6

Основы информационной безопасности

Назармамадов Умед Джамshedович

Содержание

Цель работы	1
Теоретическое введение	1
Выполнение лабораторной работы.....	2
Выводы.....	6
Список литературы.....	6

Список таблиц

Элементы списка иллюстраций не найдены.

Цель работы

Развить навыки администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux1. Проверить работу SELinux на практике совместно с веб-сервером Apache.

Теоретическое введение

SELinux (Security-Enhanced Linux) обеспечивает усиление защиты путем внесения изменений как на уровне ядра, так и на уровне пространства пользователя, что превращает ее в действительно «непробиваемую» операционную систему. Впервые эта система появилась в четвертой версии CentOS, а в 5 и 6 версии реализация была существенно дополнена и улучшена. SELinux имеет три основных режим работы:

Enforcing: режим по умолчанию. При выборе этого режима все действия, которые каким-то образом нарушают текущую политику безопасности, будут блокироваться, а попытка нарушения будет зафиксирована в журнале.

Permissive: в случае использования этого режима, информация о всех действиях, которые нарушают текущую политику безопасности, будут зафиксированы в журнале, но сами действия не будут заблокированы.

Disabled: полное отключение системы принудительного контроля доступа.

Политика SELinux определяет доступ пользователей к ролям, доступ ролей к доменам и доступ доменов к типам. Контекст безопасности — все атрибуты SELinux — роли, типы и домены. Более подробно см. в [4].

Apache — это свободное программное обеспечение, с помощью которого можно создать веб-сервер. Данный продукт возник как доработанная версия другого HTTP-клиента от национального центра суперкомпьютерных приложений (NCSA). Для чего нужен Apache сервер:

чтобы открывать динамические PHP-страницы,

для распределения поступающей на сервер нагрузки,

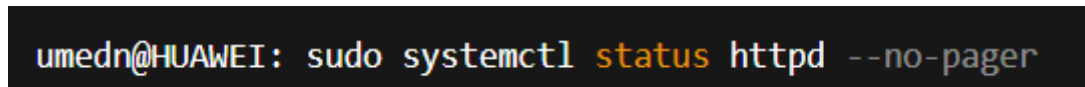
для обеспечения отказоустойчивости сервера,

чтобы потренироваться в настройке сервера и запуске PHP-скриптов.

Apache является кроссплатформенным ПО и поддерживает такие операционные системы, как Linux, BSD, MacOS, Microsoft, BeOS и другие.

Выполнение лабораторной работы

Базовая проверка httpd и контекстов. убеждаюсь, что httpd работает. (рис. [4:001]).

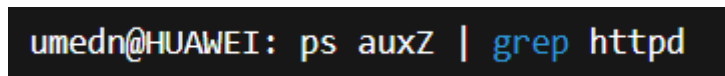


```
umedn@HUAWEI: sudo systemctl status httpd --no-pager
```

Название рисунка

Название рисунка

смотрю контекст процессов Apache.

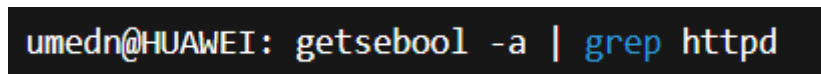


```
umedn@HUAWEI: ps auxZ | grep httpd
```

Название рисунка

Название рисунка

смотрю SELinux-переключатели для httpd.



```
umedn@HUAWEI: getsebool -a | grep httpd
```

Название рисунка

Название рисунка

вывожу статистику политики (пользователи/роли/типы).

```
umedn@HUAWEI: seinfo
```

Название рисунка

Название рисунка

Контексты в /var/www и создание test.html проверяю контексты /var/www и /var/www/html.

```
umedn@HUAWEI: ls -lZ /var/www
umedn@HUAWEI: ls -lZ /var/www/html
```

Название рисунка

Название рисунка

создаю файл /var/www/html/test.html от root.

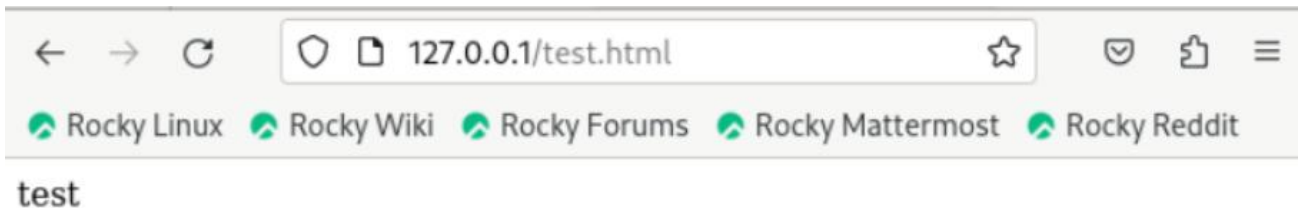
```
umedn@HUAWEI: echo "<html><body>test</body></html>" | sudo tee /var/www/html/test.html >/dev/null
```

Название рисунка

Название рисунка

смотрю контекст нового файла (по умолчанию должен быть httpd_sys_content_t). Что делаю: проверяю через браузер <http://127.0.0.1/test.html> — должен открыться «test».

```
umedn@HUAWEI: ls -lZ /var/www/html/test.html
```



Название рисунка

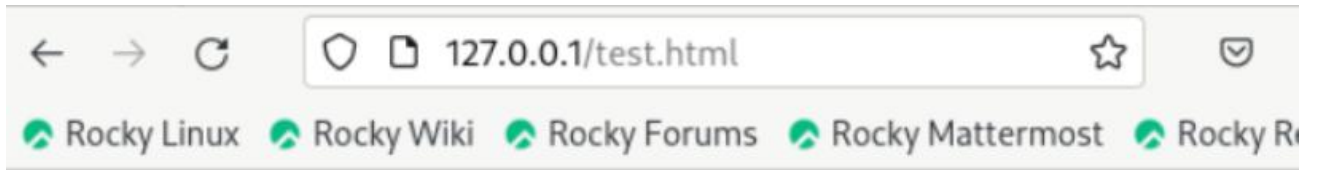
Название рисунка

Название рисунка

Название рисунка

Ломаем доступ контекстом и анализируем. нарочно меняю тип на «чужой» (например, samba_share_t). снова открываю <http://127.0.0.1/test.html> — ожидаю 403 Forbidden.

```
umedn@HUAWEI: sudo chcon -t samba_share_t /var/www/html/test.html
umedn@HUAWEI: ls -Z /var/www/html/test.html
```



Forbidden

You don't have permission to access this resource.

Название рисунка

Название рисунка

Название рисунка

Название рисунка

смотрю права и анализирую логи (messages/audit/httpd).

```
umedn@HUAWEI: ls -l /var/www/html/test.html
umedn@HUAWEI: sudo tail -n 50 /var/log/messages
umedn@HUAWEI: sudo tail -n 50 /var/log/audit/audit.log
```

Название рисунка

Название рисунка

Перевод Apache на порт 81 и разрешение его в SELinux меняю порт в конфиге httpd, Listen 80 → Listen 81.

```
umedn@HUAWEI: sudo sed -i 's/^Listen 80/Listen 81/' /etc/httpd/conf/httpd.conf
```

Название рисунка

Название рисунка

разрешаю порт 81 для типа http_port_t и проверяю список

```
umedn@HUAWEI: sudo systemctl restart httpd || echo "restart failed (expected)"
```

```
umedn@HUAWEI: sudo semanage port -a -t http_port_t -p tcp 81
umedn@HUAWEI: sudo semanage port -l | grep http_port_t
```

Название рисунка

Название рисунка

Название рисунка

Название рисунка

снова стартую httpd и проверяю доступ к <http://127.0.0.1:81/test.html>.

```
umedn@HUAWEI: sudo systemctl restart httpd
```

 Rocky Linux  Rocky Wiki  Rocky Forums  Rocky Mattermost  Rocky Reddit

Попытка соединения не удалась

Firefox не может установить соединение с сервером 127.0.0.1.

- Возможно, сайт временно недоступен или перегружен запросами. Подождите некоторое время и попробуйте снова.
- Если вы не можете загрузить ни одну страницу – проверьте настройки соединения с Интернетом.
- Если ваш компьютер или сеть защищены межсетевым экраном или прокси-сервером – убедитесь, что Firefox разрешён выход в Интернет.

Попробовать снова

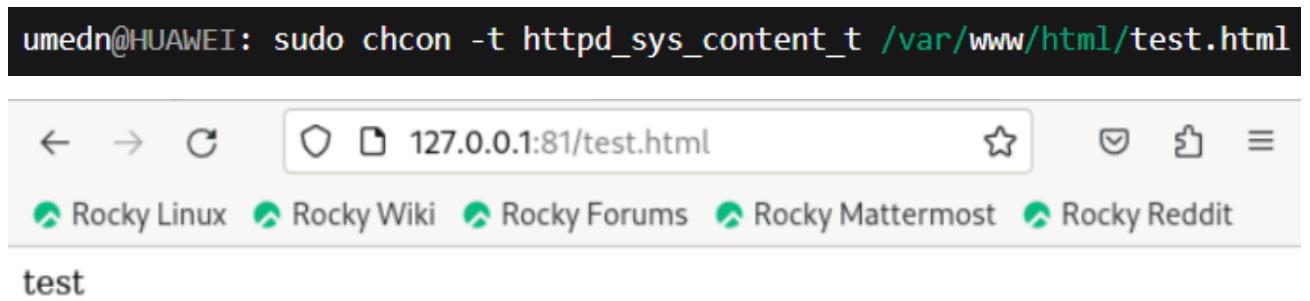
Название рисунка

Название рисунка

Название рисунка

Название рисунка

Возврат рабочего состояния возвращаю корректный тип файлу (httpd_sys_content_t), проверяю доступ на :81.



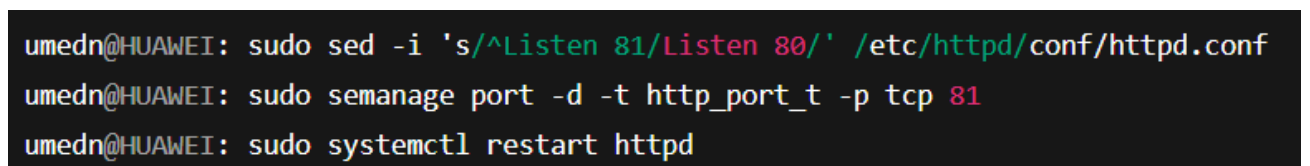
Название рисунка

Название рисунка

Название рисунка

Название рисунка

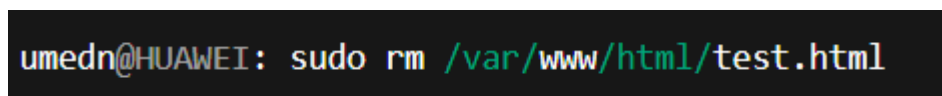
возвращаю порт 80 и очищаю добавленный порт 81 в SELinux.



Название рисунка

Название рисунка

удаляю тестовый файл.



Название рисунка

Название рисунка

Выводы

В ходе выполнения данной лабораторной работы были развиты навыки администрирования ОС Linux, получено первое практическое знакомство с технологией SELinux и проверена работа SELinux на практике совместно с веб-сервером Apache.

Список литературы