Отчет по лабораторной работе №6

Основы информационной безопасности

Назармамадов Умед Джамшедович

Содержание

[Цель работы 1](#_Toc209057570)

[Теоретическое введение 1](#_Toc209057571)

[Выполнение лабораторной работы 2](#_Toc209057572)

[Выводы 6](#_Toc209057573)

[Список литературы 6](#_Toc209057574)

Список таблиц

**Элементы списка иллюстраций не найдены.**

# Цель работы

Развить навыки администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux1. Проверить работу SELinx 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 [@f].

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. [-@fig:001]).



|  |
| --- |
| Название рисунка |

Название рисунка

смотрю контекст процессов Apache.



|  |
| --- |
| Название рисунка |

Название рисунка

смотрю SELinux-переключатели для httpd.



|  |
| --- |
| Название рисунка |

Название рисунка

вывожу статистику политики (пользователи/роли/типы).



|  |
| --- |
| Название рисунка |

Название рисунка

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

Изображение выглядит как текст, Шрифт, снимок экрана, Графика

Контент, сгенерированный ИИ, может содержать ошибки.

|  |
| --- |
| Название рисунка |

Название рисунка

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



|  |
| --- |
| 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».



Изображение выглядит как текст, Шрифт, линия, снимок экрана

Контент, сгенерированный ИИ, может содержать ошибки.

|  |
| --- |
| 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.

Изображение выглядит как снимок экрана, Шрифт, текст, Графика

Контент, сгенерированный ИИ, может содержать ошибки.

Изображение выглядит как текст, Шрифт, линия, число

Контент, сгенерированный ИИ, может содержать ошибки.

|  |
| --- |
| Название рисунка |

Название рисунка

|  |
| --- |
| Название рисунка |

Название рисунка

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

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Контент, сгенерированный ИИ, может содержать ошибки.

|  |
| --- |
| Название рисунка |

Название рисунка

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



|  |
| --- |
| Название рисунка |

Название рисунка

разрешаю порт 81 для типа http\_port\_t и проверяю список



Изображение выглядит как Шрифт, снимок экрана, текст, Графика

Контент, сгенерированный ИИ, может содержать ошибки.

|  |
| --- |
| Название рисунка |

Название рисунка

|  |
| --- |
| Название рисунка |

Название рисунка

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



Изображение выглядит как текст, снимок экрана, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

|  |
| --- |
| Название рисунка |

Название рисунка

|  |
| --- |
| Название рисунка |

Название рисунка

Возврат рабочего состояния возвращаю корректный тип файлу (httpd\_sys\_content\_t), проверяю доступ на :81.



Изображение выглядит как текст, Шрифт, линия, число

Контент, сгенерированный ИИ, может содержать ошибки.

|  |
| --- |
| Название рисунка |

Название рисунка

|  |
| --- |
| Название рисунка |

Название рисунка

возвращаю порт 80 и очищаю добавленный порт 81 в SELinux.

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Контент, сгенерированный ИИ, может содержать ошибки.

|  |
| --- |
| 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