Работа с SELinux и веб-сервером Apache

LAB №6

Чилеше Лупупа

29 сентября 2022 г.

Российский университет дружбы народов, Москва, Россия

Информация

Докладчик

:::::::::::::: {.columns align=center} ::: {.column width="70%"}

- Чилеше Лупупа
- Студент
- Российский университет дружбы народов

SELinux и веб-сервер Apache: управление доступом и контексты безопасности

Цель лабораторной работы

- Изучить SELinux в режиме enforcing с политикой targeted
- Понять, как SELinux управляет доступом к ресурсам Apache
- Научиться проверять, изменять и восстанавливать контексты безопасности
- Опробовать настройку политик SELinux для портов

Что такое SELinux?

- Модуль безопасности Linux с системой обязательного контроля доступа (MAC)
- Использует метки (контексты безопасности) для ограничения доступа
- Защищает службы даже при наличии разрешений POSIX

Начальная настройка

- Проверили статус SELinux командами getenforce, sestatus
- Проверили статус Apache: service httpd status
- Убедились, что процесс httpd работает с типом httpd_t

```
[lchileshe@lchileshe ~]$ getenforce
Enforcing
[lchileshe@lchileshe ~]$ sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:             targeted
Current mode:                   enforcing
Mode from config file:          enforcing
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Memory protection checking:     actual (secure)
Max kernel policy version:      33
[lchileshe@lchileshe ~]$
```

Работа с файловыми контекстами

- Проверили типы файлов в /var/www и /var/www/html
- Создали файл test.html с контекстом httpd_sys_content_t
- Убедились в доступности файла через браузер по адресу http://127.0.0.1/test.html

типы файлов в /var/www и /var/www/html

```
[lchileshe@lchileshe ~]$ ls -lZ /var/www
total 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 Jan 22
03:25 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0     6 Jan 22
03:25 html
```

Разбор контекста SELinux

- Пример: unconfined_u:object_r:httpd_sys_content_t:s0
- Пользователь: unconfined_u
- Роль: object_r (не влияет на файлы)
- Тип: httpd_sys_content_t — разрешает доступ Apache

##контекста SELinux

```
[root@lchileshe ~]# ls -Z /var/www/html/test.html
unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.html
[root@lchileshe ~]#
```

Эмуляция ошибки доступа

- Изменили тип контекста на samba_share_t: chcon -t samba_share_t test.html
- Попытка открыть файл → ошибка 403 Forbidden
- Причина найдена в логах: SELinux заблокировал доступ

Forbidden

You don't have permission to access this resource.

Настройка порта

Изменили порт Apache с 80 на 81 в httpd.conf - Перезапуск Apache завершился с ошибкой - Разрешили порт 81 для SELinux:

```
semanage port -a -t http_port_t -p tcp 81
```

Apache успешно запустился и обслуживает файл на порту 81

Очистка и восстановление

- Восстановили контекст файла: httpd_sys_content_t
- Вернули конфигурацию Apache на порт 80
- Удалили файл и удалили политику для порта 81

Выводы

- SELinux ограничивает доступ даже при разрешениях файловой системы
- Для Apache важен корректный тип файлового контекста
- Доступ к нестандартным портам нужно явно разрешать
- Лог-файлы SELinux (особенно audit.log) — важный инструмент отладки