Презентация по лабораторной работе №5

Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов

Ибатулина Д.Э.

13 апреля 2024

Российский университет дружбы народов, Москва, Россия

Информация

Докладчик

- Ибатулина Дарья Эдуардовна
- студентка группы НКАбд-01-22
- факультет физико-математических и естественных наук
- Российский университет дружбы народов
- deibatulina.github.io
- https://github.com/deibatulina

Вводная часть

Актуальность

Навыки работы с атрибутами (их установка и снятие), а также навык компиляции программных файлов и их исполнения - неотъемлемое умение специалиста по информационной безопасности.

Цели и задачи

Изучение механизмов изменения идентификаторов, применения SetUID- и Sticky-битов.

Получение практических навыков работы в консоли с дополнительными атрибутами.

Рассмотрение работы механизма смены идентификатора процессов пользователей, а также влияние бита *Sticky* на запись и удаление файлов.

Выполнение

Подготовка лабораторного стенда

Проверяем, установлен ли gcc.

```
[guest@deibatulina ~]$ gcc -v
Используются внутренние спецификации.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=/usr/libexec/gcc/x86_64-redhat-linux/11/lto-wrapper
OFFLOAD_TARGET_NAMES=nvptx-none
OFFLOAD_TARGET_DEFAULT=1
Целевая архитектура: x86_64-redhat-linux
Параметры конфигурации: ../configure --enable-bootstrap --enable-host-pie --enab
le-host-bind-now --enable-languages=c,c++,fortran,lto --prefix=/usr --mandir=/us
r/share/man --infodir=/usr/share/info --with-bugurl=https://bugs.rockylinux.org/
 --enable-shared --enable-threads=posix --enable-checking=release --with-system-
zlib --enable-__cxa_atexit --disable-libunwind-exceptions --enable-gnu-unique-ob
ject --enable-linker-build-id --with-gcc-major-version-only --enable-plugin --en
able-initfini-array --without-isl --enable-multilib --with-linker-hash-style=gnu
 --enable-offload-targets=nvptx-none --without-cuda-driver --enable-gnu-indirect
-function --enable-cet --with-tune=generic --with-arch_64=x86-64-v2 --with-arch_
32=x86-64 --build=x86_64-redhat-linux --with-build-config=bootstrap-lto --enable
-link-serialization=1
Модель многопоточности: posix
```

Создание программного файла

Переключаемся на учетную запись администратора и создаем файл `simpleid.c`, заполняем его предложенной программой.

```
[root@deibatulina ~]# su - guest
[guest@deibatulina ~]$ touch simpleid.c
```

Компиляция и исполнение файла

Компилирую и привожу файл в исполнение.

```
[guest@deibatulina ~]$ gcc simpleid.c -o simpleid
[guest@deibatulina ~]$ ls
dir1  simpleid  simpleid.c
[guest@deibatulina ~]$ ./simpleid
uid=1001, gid=1001
[guest@deibatulina ~]$ id
uid=1001(guest) gid=1001(guest) группы=1001(guest) контекст=unconfined_u:unconfi
ned_r:unconfined_t:s0-s0:c0.c1023
[guest@deibatulina ~]$
```

Смена пользователя файла на root и установка SetUID-бита

Теперь проделаю то же самое, но под учётной записью администратора.

```
[root@deibatulina ~]# sudo -i
[root@deibatulina ~]# chown root:guest /home/guest/simpleid2
[root@deibatulina ~]# chmod u+s /home/guest/simpleid2
[root@deibatulina ~]#
```

Действия с файлом

Создаём файл `readfile.c` и изменяем его владельца так, чтобы только суперпользователь мог прочитать его, а guest не мог.

```
[guest@deibatulina ~]$ su
Пароль:
[root@deibatulina guest]# chown root:guest readfile
[root@deibatulina guest]# chmod 700 readfile
[root@deibatulina guest]# chown root:guest readfile
[root@deibatulina guest]# chmod -r readfile.c
[root@deibatulina guest]# chmod u+c readfile
chmod: неверный режим: «u+c»
По команде «chmod --help» можно получить дополнительную информацию.
[root@deibatulina guest]# chmod u+s readfile
[root@deibatulina guest]#
```

Исследование Sticky-бита

Выясним, установлен ли атрибут Sticky на директории /tmp, от имени пользователя guest создадим файл file01.txt в директории /tmp со словом test, и просмотрим атрибуты у только что созданного файла и разрешим чтение и запись для категории пользователей «все остальные».

```
[deibatulina@deibatulina ~]$ ls -l / | grep tmp
drwxrwxrwt.  15 root root 4096 мар 13 19:58 tmp
[deibatulina@deibatulina ~]$ echo "test" > /tmp/file01.txt
[deibatulina@deibatulina ~]$ ls -l /tmp/file01.txt
-rw-r--r--. 1 deibatulina deibatulina 5 мар 13 20:02 /tmp/file01.txt
[deibatulina@deibatulina ~]$ chmod o+rw /tmp/file01.txt
[deibatulina@deibatulina ~]$ ls -l /tmp/file01.txt
-rw-r--rw-. 1 deibatulina deibatulina 5 мар 13 20:02 /tmp/file01.txt
[deibatulina@deibatulina ~]$
```

Изменение атрибутов файла

Повысим свои права до суперпользователя и выполним после этого команду, снимающую атрибут t (Sticky-бит) с директории /tmp, а затем покинем режим суперпользователя. От пользователя guest2 проверим, что атрибута t у директории /tmp нет. Повторив предыдущие шаги от имени других пользователей, я могу заметить, что всё получается. Отвечая на вопрос: Удалось ли вам удалить файл от имени пользователя, не являющегося его владельцем? я могу сказать, что удалось. Повысим свои права до суперпользователя и вернём атрибут t на директорию /tmp

```
[guest2@deibatulina ~]$ su -
Пароль:
[root@deibatulina ~]# chmod -t /tmp
[root@deibatulina ~]# exit
выход
[guest2@deibatulina ~]$ ls -l / | grep tmp
drwxrwxrwx.  14 root root 4096 мар 13 20:08 tmp
[guest2@deibatulina ~]$ su -
Пароль:
[root@deibatulina ~]# chmod +t /tmp
```

Результаты

Мною были изучены механизмы изменения идентификаторов и применения SetUID- и Sticky-битов. Получены практические навыки работы в консоли с дополнительными атрибутами. Были рассмотрены работа механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.