

Информационная безопасность

Лабораторная работа №5

Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов

Выполнила: Халфина Айсылу Зуфаровна

Группа: НПМбд-02-19

08.10.2022

Цель работы

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

1) Создание программы

Сперва создадим программу **simpleid.c**

```
#include <sys/types.h>
#include <unistd.h>
#include <stdio.h>
int
main ()
{
    uid_t uid = geteuid ();
    gid_t gid = getegid ();
    printf ("uid=%d, gid=%d\n", uid, gid);
    return 0;
}
```

Скомпилируем программу и убедимся, что файл программы создан. Выполним программу **simpleid** и системную программу **id**. Видим что программа отображает идентификаторы пользователя и

группы корректно.

```
guest@akhalfina:~  
File Edit View Search Terminal Help  
[guest@akhalfina ~]$ touch simpleid.c  
[guest@akhalfina ~]$ gcc simpleid.c -o simpleid  
[guest@akhalfina ~]$ ./simpleid  
uid=1001, gid=1001  
[guest@akhalfina ~]$ id  
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023  
[guest@akhalfina ~]$
```

Усложним программу, добавив вывод действительных идентификаторов. Назовём программу **simpleid2.c**

```
#include <sys/types.h>  
#include <unistd.h>  
#include <stdio.h>  
int  
main ()  
{  
    uid_t real_uid = getuid ();  
    uid_t e_uid = geteuid ();  
    gid_t real_gid = getgid ();  
    gid_t e_gid = getegid ();  
    printf ("e_uid=%d, e_gid=%d\n", e_uid, e_gid);  
    printf ("real_uid=%d, real_gid=%d\n", real_uid, real_gid);  
    return 0;  
}
```

Скомпилируем и запустим **simpleid2.c**

```
[guest@akhalfina ~]$ gcc simpleid2.c -o simpleid2  
[guest@akhalfina ~]$ ./simpleid2  
e_uid=1001, e_gid=1001  
real_uid=1001, real_gid=1001  
[guest@akhalfina ~]$
```

Сменим владельца файла и выставим ему SetU'D-бит. Проверим правильность смены атрибутов. Запустим программу и выполним команду **id**

```
guest@akhalfina:/home/guest
File Edit View Search Terminal Help
[guest@akhalfina ~]$ su
Password:
[root@akhalfina guest]# ls
Desktop Documents Music Public simpleid2 simpleid.c Videos
dir1 Downloads Pictures simpleid simpleid2.c Templates
[root@akhalfina guest]# chown root:guest /home/guest/simpleid2
[root@akhalfina guest]# chmod u+s /home/guest/simpleid2
[root@akhalfina guest]# ls -l simpleid2
-rwsrwxr-x. 1 root guest 8576 Oct  8 14:30 simpleid2
[root@akhalfina guest]# ./simpleid2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[root@akhalfina guest]# id
uid=0(root) gid=0(root) groups=0(root) context=unconfined_u:unconfined_r:unconfi
ned_t:s0-s0:c0.c1023
[root@akhalfina guest]#
```

2) Исследование Sticky-бита

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

```
guest@akhalfina:~
File Edit View Search Terminal Help
[guest@akhalfina ~]$ ls -l / | grep tmp
drwxrwxrwt. 30 root root 8192 Oct  8 14:35 tmp
[guest@akhalfina ~]$ echo "test" > /tmp/file01.txt
[guest@akhalfina ~]$ ls -l /tmp/file01.txt
-rw-rw-r--. 1 guest guest 5 Oct  8 14:54 /tmp/file01.txt
[guest@akhalfina ~]$ chmod o+rw /tmp/file01.txt
[guest@akhalfina ~]$ ls -l /tmp/file01.txt
-rw-rw-rw-. 1 guest guest 5 Oct  8 14:54 /tmp/file01.txt
[guest@akhalfina ~]$
```

От пользователя **guest2** (не являющегося владельцем) попробуем прочитав файл **/tmp/file01.txt**, дозаписать слово **test2**. Так же попробуем записать в файл слово **test3**, стерев при этом всю имеющуюся в файле информацию. Затем попробуем удалить файл. Видим что все операции, кроме удаления, выполнены успешно.

```
guest2@akhalfina:/home/guest
File Edit View Search Terminal Help
[guest@akhalfina ~]$ su guest2
Password:
[guest2@akhalfina guest]$ cat /tmp/file01.txt
test
[guest2@akhalfina guest]$ echo "test2" > /tmp/file01.txt
[guest2@akhalfina guest]$ cat /tmp/file01.txt
test2
[guest2@akhalfina guest]$ echo "test3" > /tmp/file01.txt
[guest2@akhalfina guest]$ cat /tmp/file01.txt
test3
[guest2@akhalfina guest]$ rm /tmp/file0l.txt
rm: cannot remove '/tmp/file0l.txt': No such file or directory
[guest2@akhalfina guest]$ rm /tmp/file01.txt
rm: cannot remove '/tmp/file01.txt': No such file or directory
[guest2@akhalfina guest]$ rm /tmp/file01.txt
rm: cannot remove '/tmp/file01.txt': Operation not permitted
[guest2@akhalfina guest]$
```

Повысим свои права до суперпользователя и снимем с директории атрибут **t**. Попробуем повторить все те же операции. Видим что на этот раз все они были выполнены успешно.

```
guest2@akhalfina:/home/guest
File Edit View Search Terminal Help
[guest2@akhalfina guest]$ cat /tmp/file01.txt
test3
[guest2@akhalfina guest]$ echo "test2" > /tmp/file01.txt
[guest2@akhalfina guest]$ cat /tmp/file01.txt
test2
[guest2@akhalfina guest]$ echo "test3" > /tmp/file01.txt
[guest2@akhalfina guest]$ cat /tmp/file01.txt
test3
[guest2@akhalfina guest]$ rm /tmp/file01.txt
[guest2@akhalfina guest]$ █
```

Выводы

В результате выполнения работы, мы изучили механизмы изменения идентификаторов, применения SetUID- и Sticky-битов, получили практические навыки работы в консоли с дополнительными атрибутами, рассмотрел работу механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.