

Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов

Иванов Даниил НБИ-01-19

3 октября, 2022, Москва, Россия

Российский Университет Дружбы Народов

Цели и задачи

- SUID - разрешение на установку идентификатора пользователя. Это бит разрешения, который позволяет пользователю запускать исполняемый файл с правами владельца этого файла.
- SGID - разрешение на установку идентификатора группы. Принцип работы очень похож на SUID с отличием, что файл будет запускаться пользователем от имени группы, которая владеет файлом.

Цель лабораторной 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 лабораторной работы

Программа simpleid

```
[guest@ivanovdo ~]$ gcc simpleid.c
Permissive
[guest@ivanovdo ~]$
[guest@ivanovdo ~]$ mkdir lab5
[guest@ivanovdo ~]$ cd lab5/
[guest@ivanovdo lab5]$ touch simpleid.c
[guest@ivanovdo lab5]$ touch simpleid2.c
[guest@ivanovdo lab5]$ touch readfile.c
[guest@ivanovdo lab5]$ gedit simpleid.c
[guest@ivanovdo lab5]$ gcc simpleid.c
[guest@ivanovdo lab5]$ gcc simpleid.c -o simpleid
[guest@ivanovdo lab5]$ ./simpleid
uid=1001, gid=1001
[guest@ivanovdo lab5]$ id
uid=1001(guest) gid=1001(guest) группы=1001(guest) контекст=unconfined_u:unconfi
ned_r:unconfined_t:s0-s0:c0.c1023
[guest@ivanovdo lab5]$
```

Figure 1: результат программы simpleid

Программа simpleid2

```
ned_r:unconfined_t:s0-s0:c0.c1023
[guest@ivanovdo lab5]$ gedit simpleid2.c
[guest@ivanovdo lab5]$
[guest@ivanovdo lab5]$ gcc simpleid2.c
[guest@ivanovdo lab5]$ gcc simpleid2.c -o simpleid2
[guest@ivanovdo lab5]$ ./simpleid2
e_uid=1001, e_gid=1001
real_uid=1001, real_gid=1001
[guest@ivanovdo lab5]$ su
Пароль:
[root@ivanovdo lab5]# chown root:guest simpleid2
[root@ivanovdo lab5]# chmod u+s simpleid2
[root@ivanovdo lab5]# ./simpleid2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[root@ivanovdo lab5]# id
uid=0(root) gid=0(root) rpyны=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[root@ivanovdo lab5]# chmod g+s simpleid2
[root@ivanovdo lab5]# ./simpleid2
e_uid=0, e_gid=1001
real_uid=0, real_gid=0
[root@ivanovdo lab5]#
```

Figure 2: результат программы simpleid2

Программа readfile

```
[guest@ivanovdo lab5]$  
[guest@ivanovdo lab5]$  
[guest@ivanovdo lab5]$ cat readfile.c  
cat: readfile.c: Отказано в доступе  
[guest@ivanovdo lab5]$ ./readfile readfile.c  
#include <stdio.h>[guest@ivanovdo lab5]$ ./readfile /etc/shadow  
root:$6$CVAze4kP[guest@ivanovdo lab5]$
```

Figure 3: результат программы readfile

Исследование Sticky-бита

```
[guest@ivanovdo lab5]$  
[guest@ivanovdo lab5]$ cd /tmp  
[guest@ivanovdo tmp]$ echo "test" >> file01.txt  
[guest@ivanovdo tmp]$ chmod o+rx file01.txt  
[guest@ivanovdo tmp]$ ls -l file01.txt  
-rw-rw-r-x. 1 guest guest 5 окт  4 17:40 file01.txt  
[guest@ivanovdo tmp]$ su guest2  
Пароль:  
[guest2@ivanovdo tmp]$ cat file01.txt  
test  
[guest2@ivanovdo tmp]$ echo "test" >> file01.txt  
[guest2@ivanovdo tmp]$ cat file01.txt  
test  
test  
[guest2@ivanovdo tmp]$ echo "test" > file01.txt  
[guest2@ivanovdo tmp]$ rm file01.txt  
rm: невозможно удалить «file01.txt»: Операция не позволена  
[guest2@ivanovdo tmp]$ su  
Пароль:  
[root@ivanovdo tmp]# chmod -t /tmp  
[root@ivanovdo tmp]# exit  
exit  
[guest2@ivanovdo tmp]$ rm file01.txt  
[guest2@ivanovdo tmp]$ su  
Пароль:  
[root@ivanovdo tmp]# chmod +t /tmp  
[root@ivanovdo tmp]# exit  
exit  
[guest2@ivanovdo tmp]$ █
```

Figure 4: исследование Sticky-бита

Выводы

Изучили механизмы изменения идентификаторов, применения SetUID- и Sticky-битов. Получили практические навыки работы в консоли с дополнительными атрибутами. Также мы рассмотрели работу механизма смены идентификатора процессов пользователей и влияние бита Sticky на запись и удаление файлов.