

Лабораторная работа № 5

Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов.

Акопян И.А

7 октября 2023

Российский университет дружбы народов, Москва, Россия

- Акопян Изабелла Арменовна
- Бизнес-информатика
- Кафедра математического моделирования и ИИ
- Российский университет дружбы народов
- 1032203961@pfur.ru

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.

Создать программы, чтобы поработать с битами (SetUID, SetGID, Sticky-бит) и понять как дополнительные атрибуты влияют на работу с файлами.

Создали простую программу на языке C для чтения uid и gid. Проверили ее работу, сравнив с системной утилитой id

```
guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ nano simpleid.c
guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ gcc simpleid.c -o simpleid
guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ ./simpleid
uid=1003, gid=1003
guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ id
uid=1003(guest) gid=1003(guest) groups=1003(guest)
guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ nano simpleid.c
```

Use "fg" to return to nano.

```
[1]+  Stopped                  nano simpleid.c
guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ nano simpleid.c
guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ gcc simpleid.c -o simpleid2
guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ ./simpleid2
e_uid=1003, e_gid=1003
real_uid=1003, real_gid=1003
guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ ls -l simpleid
-rwxrwxr-x 1 guest guest 16792 окт  7 14:05 simpleid
guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ ls -l simpleid1
ls: cannot access 'simpleid1': No such file or directory
guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ ls -l simpleid2
-rwxrwxr-x 1 root guest 16880 окт  7 14:09 simpleid2
```

```
root@szafgvsgd-HP-Pavilion-17-Notebook-PC: /
Файл  Правка  Вид  Поиск  Терминал  Справка
iaakopyan@szafgvsgd-HP-Pavilion-17-Notebook-PC:~$ su -
Password:
root@szafgvsgd-HP-Pavilion-17-Notebook-PC:~# chown root:guest /home/guest/simpleid2
root@szafgvsgd-HP-Pavilion-17-Notebook-PC:~# chown u+s /home/guest/simpleid2
chown: invalid user: 'u+s'
root@szafgvsgd-HP-Pavilion-17-Notebook-PC:~# chmod u+s /home/guest/simpleid2
root@szafgvsgd-HP-Pavilion-17-Notebook-PC:~# su
```

Рис. 2: chown и chmod

Создали программу для чтения файлов.

```

guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ touch readfile.c
guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ nano readfile.c

Use "fg" to return to nano.

[2]+  Stopped                  nano readfile.c
guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ nano readfile.c

Use "fg" to return to nano.

[3]+  Stopped                  nano readfile.c
guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ fg
nano readfile.c
guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ gcc readfile.c -o readfile
readfile.c:7:15: error: expected ')' before 'char'
    7 |   main (int argc, char* argv[])
      |                   ^
guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ nano readfile.c
guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ gcc readfile.c -o readfile
guest@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ cat readfile.c
#include <fcntl.h>
#include <stdio.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <unistd.h>
int
main (int argc, char* argv[])
{
    unsigned char buffer[16];
    size_t bytes_read;
    int i;
    int fd = open (argv[1], O_RDONLY);
    do
    {
        bytes_read = read (fd, buffer, sizeof(buffer));
        for(i=0; i < bytes_read; ++i) printf("%c", buffer[i]);
    }
    while(bytes_read == sizeof(buffer));
    close (fd);
    return 0;
}

```

Рис. 3: readfile.c

```

root@szafgvsdg-HP-Pavilion-17-Notebook-PC:/home/guest# chown root:root readfile
root@szafgvsdg-HP-Pavilion-17-Notebook-PC:/home/guest# chmod o-r readfile.c
root@szafgvsdg-HP-Pavilion-17-Notebook-PC:/home/guest# chmod g-rw readfile.c
root@szafgvsdg-HP-Pavilion-17-Notebook-PC:/home/guest# chmod u+s readfile

```


Далее поработали с Sticky-битом

```
root@szafgvsdg-HP-Pavilion-17-Notebook-PC:/# ls
bin  cdrom  etc  lab02  lib32  libx32  media  opt  root  sbin  swapfile  tmp  var
boot  dev  home  lib  lib64  lost+found  mnt  proc  run  srv  sys  usr
root@szafgvsdg-HP-Pavilion-17-Notebook-PC:/# ls -l / | grep tmp
drwxrwxrwt  21 root root      4096 окт  7 16:10 tmp
root@szafgvsdg-HP-Pavilion-17-Notebook-PC:/# echo "test" > /tmp/file01.txt
root@szafgvsdg-HP-Pavilion-17-Notebook-PC:/# ls -l /tmp/file01.txt
-rw-r--r--  1 root root  5 окт  7 16:14 /tmp/file01.txt
root@szafgvsdg-HP-Pavilion-17-Notebook-PC:/# chmod o+rw /tmp/file01.txt
root@szafgvsdg-HP-Pavilion-17-Notebook-PC:/# ls -l /tmp/file01.txt
-rw-r--rw-  1 root root  5 окт  7 16:14 /tmp/file01.txt
root@szafgvsdg-HP-Pavilion-17-Notebook-PC:/#
```

Рис. 7: Проверка функциональности Sticky-бита на примере файла, созданного в каталоге /tmp

```

iaakopyan@szafgvsdg-HP-Pavilion-17-Notebook-PC:~$ su guest2
Password:
$ echo "test2" > tmp/file01.txt
sh: 1: cannot create tmp/file01.txt: Directory nonexistent
$ cat tmp/file01.txt
cat: tmp/file01.txt: No such file or directory
$ echo "test3" > /tmp/file01.txt
$ cat tmp/file01.txt
cat: tmp/file01.txt: No such file or directory
$ rm /tmp/file01.txt
rm: cannot remove '/tmp/file01.txt': Operation not permitted
$ su -
Password:
root@szafgvsdg-HP-Pavilion-17-Notebook-PC:~# chmod -t /tmp
root@szafgvsdg-HP-Pavilion-17-Notebook-PC:~# exit
logout
$ ls -l / | grep tmp
drwxrwxrwx  21 root root      4096 окт  7 16:14 tmp
$ echo "test2"> /tmp/file01.txt
sh: 8: echotest2: not found
$ echo "test2" > /tmp/file01.txt
$ cat tmp/file01.txt
cat: tmp/file01.txt: No such file or directory
$ rm /tmp/file01.txt
$ ls
13                                '2023-09-30 15-13-40.mkv'
1314.zip                         '2023-09-30 18-35-15.mkv'
13.zip                           '2023-09-30 18-38-25.mkv'
14                                '2023-10-07 14-01-18.mkv'
14.zip                           '2023-10-07 15-34-09.mkv'
'2023-09-05 13-56-42.mkv'         4
'2023-09-05 14-29-45.mkv'         4.sage
'2023-09-07 16-03-30.mkv'         5
'2023-09-07 16-05-23.mkv'         6
'2023-09-07 16-12-25.mkv'         6.zip
'2023-09-07 18-01-22.mkv'         7
'2023-09-08 15-02-40.mkv'         8
'2023-09-08 17-00-38.mkv'         cat.jpeg
'2023-09-08 17-17-17.mkv'         euler1.sage
'2023-09-09 18-23-56.mkv'         euler2.sage
'2023-09-09 18-28-06.mkv'         euler4.sage
'2023-09-13 12-07-20.mkv'         euler.sage
'2023-09-16 12-48-09.mkv'         file.txt
'2023-09-16 14-06-49.mkv'         filw

```

Рис. 8: атрибут t убран

Вернула все на место (см. рис. @fig:009)

```
$ su -  
Password:  
root@szafigvsgd-HP-Pavilion-17-Notebook-PC:~# chmod +t /tmp  
root@szafigvsgd-HP-Pavilion-17-Notebook-PC:~# exit  
logout  
$ ls -l / | grep tmp  
drwxrwxrwt 21 root root          4096 окт  7 16:22 tmp  
+ □
```

Рис. 9: атрибут t добавлен

Изучили теорию механизмов изменения идентификаторов, применения SetUID, SetGID и Sticky-битов. Рассмотрела работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.