

# Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов

---

Максим Платонов

27 февраля, 2024, 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@mplatonov ~]$  
[guest@mplatonov ~]$  
[guest@mplatonov ~]$ cd  
[guest@mplatonov ~]$ mkdir lab5  
[guest@mplatonov ~]$ cd lab5  
[guest@mplatonov lab5]$ touch simpleid.c  
[guest@mplatonov lab5]$ touch simpleid2.c  
[guest@mplatonov lab5]$ gedit simpleid.c  
[guest@mplatonov lab5]$  
[guest@mplatonov lab5]$ gcc simpleid.c  
[guest@mplatonov lab5]$ gcc simpleid.c -o simpleid  
[guest@mplatonov lab5]$ ./simpleid  
uid=1001, gid=1001  
[guest@mplatonov lab5]$ id  
uid=1001(guest) gid=1001(guest) группы=1001(guest),10(wheel)  
d_r:unconfined_t:s0-s0:c0.c1023  
[guest@mplatonov lab5]$
```

**Figure 1:** результат программы simpleid

# Программа simpleid2

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

Figure 2: результат программы simpleid2

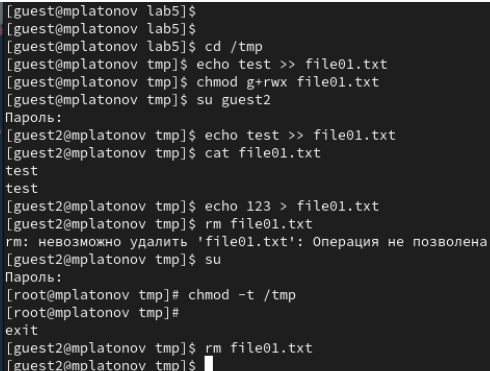
# Программа readfile

```
[guest@mplatonov lab5]$  
[guest@mplatonov lab5]$ touch readfile.c  
[guest@mplatonov lab5]$ gedit readfile.c  
[guest@mplatonov lab5]$ gcc readfile.c  
readfile.c: В функции «main»:  
readfile.c:20:19: предупреждение: сравнение указателя и целого  
20 | while (bytes_read == (buffer));  
    |                   ^~  
[guest@mplatonov lab5]$ gcc readfile.c -o readfile  
readfile.c: В функции «main»:  
readfile.c:20:19: предупреждение: сравнение указателя и целого  
20 | while (bytes_read == (buffer));  
    |                   ^~  
[guest@mplatonov lab5]$ su  
Пароль:  
[root@mplatonov lab5]# chown root:root readfile  
[root@mplatonov lab5]# chmod -rwx readfile.c  
[root@mplatonov lab5]# chmod u+s readfile  
[root@mplatonov lab5]#  
exit  
[guest@mplatonov lab5]$ cat readfile.c  
cat: readfile.c: Отказано в доступе  
[guest@mplatonov lab5]$ ./readfile readfile.c  
#include <stdio.h>[guest@mplatonov lab5]$  
[guest@mplatonov lab5]$ ./readfile /etc/shadow  
root:$6$0mJpkg1j[guest@mplatonov lab5]$  
[guest@mplatonov lab5]$
```

Figure 3: результат программы readfile



# Исследование Sticky-бита

A terminal window with a dark background and light-colored text. The text shows a sequence of commands and their outputs in a Linux environment. The user starts as 'guest' in a directory 'mplatonov lab5', moves to '/tmp', creates a file 'file01.txt', and sets permissions 'g+rx'. Then, they switch to user 'guest2' and successfully append 'test' to the file. After switching back to 'guest', they attempt to remove the file but are denied permission because of the sticky bit. Finally, the root user switches back to 'guest2' and successfully removes the file.

```
[guest@mplatonov lab5]$  
[guest@mplatonov lab5]$  
[guest@mplatonov lab5]$ cd /tmp  
[guest@mplatonov tmp]$ echo test >> file01.txt  
[guest@mplatonov tmp]$ chmod g+rx file01.txt  
[guest@mplatonov tmp]$ su guest2  
Пароль:  
[guest2@mplatonov tmp]$ echo test >> file01.txt  
[guest2@mplatonov tmp]$ cat file01.txt  
test  
test  
[guest2@mplatonov tmp]$ echo 123 > file01.txt  
[guest2@mplatonov tmp]$ rm file01.txt  
rm: невозможно удалить 'file01.txt': Операция не позволена  
[guest2@mplatonov tmp]$ su  
Пароль:  
[root@mplatonov tmp]# chmod -t /tmp  
[root@mplatonov tmp]#  
exit  
[guest2@mplatonov tmp]$ rm file01.txt  
[guest2@mplatonov 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.