

# Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов

---

Комрон Баротов

23 сентября, 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@kbarotov ~]$ cd
[guest@kbarotov ~]$ mkdir lab5
[guest@kbarotov ~]$ cd lab5
[guest@kbarotov lab5]$ touch simpleid.c
[guest@kbarotov lab5]$ gedit simpleid.c
[guest@kbarotov lab5]$ gcc simpleid.c
[guest@kbarotov lab5]$ gcc simpleid.c -o simpleid
[guest@kbarotov lab5]$ ./simpleid
uid=1001, gid=1001
[guest@kbarotov lab5]$ id
uid=1001(guest) gid=1001(guest) группы=1001(guest),10(wheel) контекст=unconfined_u:unconfined_r:u
nconfined_t:s0-s0:c0.c1023
[guest@kbarotov lab5]$
```

Figure 1: результат программы simpleid

# Программа simpleid2

```
[guest@kbarotov lab5]$  
[guest@kbarotov lab5]$ touch simpleid2.c  
[guest@kbarotov lab5]$ gedit simpleid2.c  
[guest@kbarotov lab5]$ gcc simpleid2.c  
[guest@kbarotov lab5]$ gcc simpleid2.c -o simpleid2  
[guest@kbarotov lab5]$ ./simpleid2  
e_uid=1001, e_gid=1001  
real_uid=1001, real_gid=1001  
[guest@kbarotov lab5]$ su  
Пароль:  
[root@kbarotov lab5]# chown root:guest simpleid2  
[root@kbarotov lab5]# chmod u+s simpleid2  
[root@kbarotov lab5]# ./simpleid2  
e_uid=0, e_gid=0  
real_uid=0, real_gid=0  
[root@kbarotov lab5]# id  
uid=0(root) gid=0(root) rpynmw=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c  
1023  
[root@kbarotov lab5]# ./simpleid2  
e_uid=0, e_gid=0  
real_uid=0, real_gid=0  
[root@kbarotov lab5]# chmod g+s simpleid2  
[root@kbarotov lab5]# ./simpleid2  
e_uid=0, e_gid=1001  
real_uid=0, real_gid=0  
[root@kbarotov lab5]#  
exit  
[guest@kbarotov lab5]$ chmod g+s simpleid2  
chmod: изменение прав доступа для 'simpleid2': Операция не позволена  
[guest@kbarotov lab5]$ ./simpleid2  
e_uid=0, e_gid=1001  
real_uid=1001, real_gid=1001  
[guest@kbarotov lab5]$
```

Figure 2: результат программы simpleid2

# Программа readfile

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

Figure 3: результат программы readfile



# Исследование Sticky-бита

```
[guest@kbarotov lab5]$  
[guest@kbarotov lab5]$ echo "test" >> /tmp/file01.txt  
[guest@kbarotov lab5]$ chmod g+rwX /tmp/file01.txt  
[guest@kbarotov lab5]$ su guest2  
Пароль:  
[guest2@kbarotov lab5]$ cd /tmp  
[guest2@kbarotov tmp]$ cat file01.txt  
test  
[guest2@kbarotov tmp]$ echo "test2" >> /tmp/file01.txt  
[guest2@kbarotov tmp]$ cat file01.txt  
test  
test2  
[guest2@kbarotov tmp]$ echo "test3" > /tmp/file01.txt  
[guest2@kbarotov tmp]$ rm file01.txt  
rm: невозможно удалить 'file01.txt': Операция не позволена  
[guest2@kbarotov tmp]$ su  
Пароль:  
[root@kbarotov tmp]# chmod -t /tmp  
[root@kbarotov tmp]# exit  
exit  
[guest2@kbarotov tmp]$ cd /tmp  
[guest2@kbarotov tmp]$ echo "test2" >> /tmp/file01.txt  
[guest2@kbarotov tmp]$ rm file01.txt  
[guest2@kbarotov tmp]$ su  
Пароль:  
[root@kbarotov tmp]# chmod +t /tmp  
[root@kbarotov 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.