

# **Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов**

---

Петров Артем Евгеньевич

15 декабря, 2025, 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@aepetrov: ~/lab5
guest@aepetrov:~/lab5$ gcc simpleid.c
guest@aepetrov:~/lab5$ gcc simpleid.c -o simpleid
guest@aepetrov:~/lab5$ ./simpleid
uid=1001, gid=1001
guest@aepetrov:~/lab5$ id
uid=1001(guest) gid=1001(guest) группы=1001(guest) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
guest@aepetrov:~/lab5$
```

Рис. 1: результат программы simpleid

# Программа simpleid2

```
-----  
guest@aepetrov:~/lab5$ gcc simpleid2.c  
guest@aepetrov:~/lab5$ gcc simpleid2.c -o simpleid2  
guest@aepetrov:~/lab5$ ./simpleid2  
e_uid=1001, e_gid=1001  
real_uid=1001, real_gid=1001  
guest@aepetrov:~/lab5$ su  
Пароль:  
root@aepetrov:/home/guest/lab5# chown root:guest simpleid2  
root@aepetrov:/home/guest/lab5# chmod u+s simpleid2  
root@aepetrov:/home/guest/lab5# ./simpleid2  
e_uid=0, e_gid=0  
real_uid=0, real_gid=0  
root@aepetrov:/home/guest/lab5# id  
uid=0(root) gid=0(root) группы=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:s  
0-s0:c0.c1023  
root@aepetrov:/home/guest/lab5# chmod g+s simpleid2  
root@aepetrov:/home/guest/lab5# ./simpleid2  
e_uid=0, e_gid=1001  
real_uid=0, real_gid=0  
root@aepetrov:/home/guest/lab5#  
exit  
guest@aepetrov:~/lab5$ ./simpleid2  
e_uid=0, e_gid=1001  
real_uid=1001, real_gid=1001  
guest@aepetrov:~/lab5$
```

Рис. 2: результат программы simpleid2

```
guest@aepetrov:~/lab5$  
guest@aepetrov:~/lab5$ cat readfile.c  
cat: readfile.c: Отказано в доступе  
guest@aepetrov:~/lab5$ ./readfile readfile.c  
#include <stdio.h>  
guest@aepetrov:~/lab5$ ./readfile /etc/shadow  
root:$y$j9T$z1ZFguest@aepetrov:~/lab5$  
guest@aepetrov:~/lab5$
```

**Рис. 3:** результат программы readfile

# Исследование Sticky-бита

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

Рис. 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.