

# Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов

---

Лукашов Никита НБИ-01-20

28 сентября, 2023, 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@nalukashov ~]$  
[guest@nalukashov ~]$ mkdir lab5  
[guest@nalukashov ~]$ cd lab5  
[guest@nalukashov lab5]$ touch simpleid.c  
[guest@nalukashov lab5]$  
[guest@nalukashov lab5]$ gcc simpleid.c  
[guest@nalukashov lab5]$ gcc simpleid.c -o simpleid  
[guest@nalukashov lab5]$ ./simpleid  
uid=1001, gid=1001  
[guest@nalukashov lab5]$ id  
uid=1001(guest) gid=1001(guest) группы=1001(guest),10(wheel) контекст=unconfined  
_u:unconfined_r:unconfined_t:s0-s0:c0.c1023  
[guest@nalukashov lab5]$
```

Figure 1: результат программы simpleid

# Программа simpleid2

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

Figure 2: результат программы simpleid2

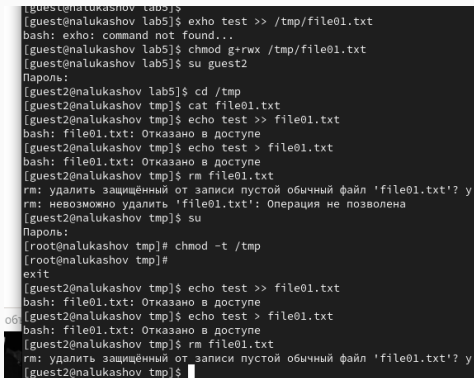
# Программа readfile

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

Figure 3: результат программы readfile



# Исследование Sticky-бита



```
[guest@nalukashov lab5]$  
[guest@nalukashov lab5]$ exho test >> /tmp/file01.txt  
bash: exho: command not found...  
[guest@nalukashov lab5]$ chmod g+rwX /tmp/file01.txt  
[guest@nalukashov lab5]$ su guest2  
Пароль:  
[guest2@nalukashov lab5]$ cd /tmp  
[guest2@nalukashov tmp]$ cat file01.txt  
[guest2@nalukashov tmp]$ echo test >> file01.txt  
bash: file01.txt: Отказано в доступе  
[guest2@nalukashov tmp]$ echo test > file01.txt  
bash: file01.txt: Отказано в доступе  
[guest2@nalukashov tmp]$ rm file01.txt  
rm: удалить защищенный от записи пустой обычный файл 'file01.txt'? y  
rm: невозможно удалить 'file01.txt': Операция не позволена  
[guest2@nalukashov tmp]$ su  
Пароль:  
[root@nalukashov tmp]# chmod -t /tmp  
[root@nalukashov tmp]#  
exit  
[guest2@nalukashov tmp]$ echo test >> file01.txt  
bash: file01.txt: Отказано в доступе  
[guest2@nalukashov tmp]$ echo test > file01.txt  
bash: file01.txt: Отказано в доступе  
[guest2@nalukashov tmp]$ rm file01.txt  
rm: удалить защищенный от записи пустой обычный файл 'file01.txt'? y  
[guest2@nalukashov 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.