

Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов

Тимур Акмурадов

30 марта, 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@timurakmuradov:~$ mkdir lab5
guest@timurakmuradov:~$ cd lab5/
guest@timurakmuradov:~/lab5$ touch simpleid.c
guest@timurakmuradov:~/lab5$ gcc simpleid.c
guest@timurakmuradov:~/lab5$ gcc simpleid.c -o simpleid
guest@timurakmuradov:~/lab5$ ./simpleid
uid=1001, gid=1001
guest@timurakmuradov:~/lab5$ id
uid=1001(guest) gid=1001(guest) rpyнны=1001(guest) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c
0.c1023
guest@timurakmuradov:~/lab5$ █
```

Рис. 1: результат программы simpleid

Программа simpleid2

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

Рис. 2: результат программы simpleid2

Программа readfile

```
guest@timurakmuradov:~/lab5$  
guest@timurakmuradov:~/lab5$ touch readfile.c  
guest@timurakmuradov:~/lab5$ gcc readfile.c  
readfile.c: В функции «main»:  
readfile.c:20:19: предупреждение: сравнение указателя и целого  
    20 | while (bytes_read == (buffer));  
        |                   ^~  
guest@timurakmuradov:~/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@timurakmuradov:~/lab5$ su  
Пароль:  
root@timurakmuradov:/home/guest/lab5# chown root:root readfile  
root@timurakmuradov:/home/guest/lab5# chmod -rwx readfile.c  
root@timurakmuradov:/home/guest/lab5# chmod u+s readfile  
root@timurakmuradov:/home/guest/lab5#  
exit  
guest@timurakmuradov:~/lab5$ cat readfile.c  
cat: readfile.c: Отказано в доступе  
guest@timurakmuradov:~/lab5$ ./readfile readfile.c  
#include <stdio.h>guest@timurakmuradov:~/lab5$  
guest@timurakmuradov:~/lab5$ ./readfile /etc/shadow  
root:$y$j9T$zLZFguest@timurakmuradov:~/lab5$  
guest@timurakmuradov:~/lab5$ █
```

Рис. 3: результат программы readfile

Исследование Sticky-бита

```
quest@timurakmuradov:~/lab$  
quest@timurakmuradov:~/lab$  
quest@timurakmuradov:~/lab$ cd /tmp  
quest@timurakmuradov:/tmp$ echo test >> file01.txt  
quest@timurakmuradov:/tmp$ chmod g+rw file01.txt  
quest@timurakmuradov:/tmp$ su guest2  
Пароль:  
quest2@timurakmuradov:/tmp$ cd /tmp  
quest2@timurakmuradov:/tmp$ cat file01.txt  
test  
quest2@timurakmuradov:/tmp$ echo test > file01.txt  
quest2@timurakmuradov:/tmp$ echo 123 >> file01.txt  
quest2@timurakmuradov:/tmp$ rm file01.txt  
rm: невозможно удалить 'file01.txt': Операция не позволена  
quest2@timurakmuradov:/tmp$ su  
Пароль:  
root@timurakmuradov:/tmp# chmod -t /tmp  
root@timurakmuradov:/tmp#  
exit  
quest2@timurakmuradov:/tmp$ rm file01.txt  
quest2@timurakmuradov:/tmp$ su  
Пароль:  
root@timurakmuradov:/tmp# chmod +t /tmp  
root@timurakmuradov:/tmp#  
exit  
quest2@timurakmuradov:/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.