

# **Лабораторная работа №5.**

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

Стариков Данила Андреевич

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>3</b>
<b>2</b>	<b>Порядок выполнения работы</b>	<b>4</b>
2.1	Создание программы . . . . .	4
2.2	Исследование Sticky-бита . . . . .	9
<b>3</b>	<b>Вывод</b>	<b>14</b>
	<b>Список литературы</b>	<b>15</b>

# 1 Цель 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.

## 2 Порядок выполнения работы

### 2.1 Создание программы

1. Вошли в систему от имени пользователя guest.
2. Создали программу simpleid.c (Листинг 2.1):

---

**Листинг 2.1** Текст программы simpleid.c

---

```
#include<sys/types.h>
#include<unistd.h>
#include<stdio.h>

int main(){
    uid_t uid = geteuid();
    gid_t gid = getegid();
    printf("uid=%d, gid=%d\n", uid, gid);
    return 0;
}
```

---

3. Скомпилировали программу и убедились, что файл программы создан:

```
gcc simpleid.c -o simpleid
```

4. Выполнили программу simpleid (Рис. 2.1):

```
./simpleid
```

5. Выполнили системную программу id (Рис. 2.1):

id

```
[guest@dastarikov ~]$ ./simpleid
uid=1001, gid=1001
[guest@dastarikov ~]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@dastarikov ~]$
```

Рис. 2.1: Сравнение выводов программ id и simpleid.

6. Усложнили программу, добавив вывод действительных идентификаторов (Листинг 2.2):

---

**Листинг 2.2** Текст программы simpleid2.c

---

```
#include<sys/types.h>
#include<unistd.h>
#include<stdio.h>

int main(){
    uid_t real_uid = getuid();
    uid_t e_uid = geteuid();
    gid_t real_gid = getgid();
    gid_t e_gid = getegid();
    printf("e_uid=%d, e_gid=%d\n", e_uid, e_gid);
    printf("real_uid=%d, real_gid=%d\n", real_uid, real_gid);
    return 0;
}
```

---

Подробнее об отличиях geteuid, getuid, getegid и getgid можно прочитать в разделе 10.7 [1].

7. Скомпилировали и запустили simpleid2.c:

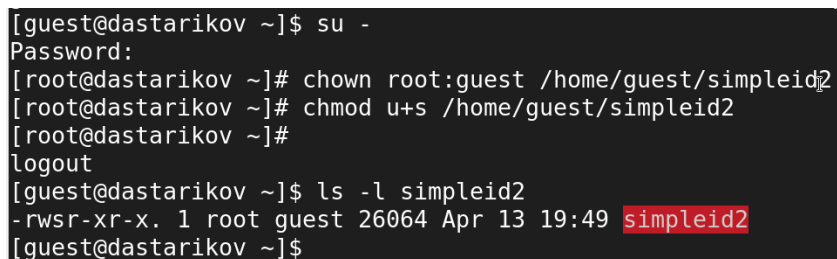
```
gcc simpleid2.c -o simpleid2
./simpleid2
```

8. Получили права суперпользователя и выполнили команды (Рис. 2.2):

```
su -  
chown root:guest /home/guest/simpleid2  
chmod u+s /home/guest/simpleid2
```

9. Выполнили проверку правильности установки новых атрибутов и смены владельца файла simpleid2 (Рис. 2.2):

```
ls -l simpleid2
```

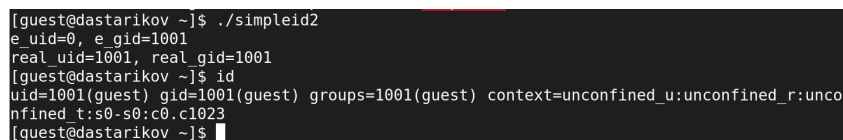


```
[guest@dastarikov ~]$ su -  
Password:  
[root@dastarikov ~]# chown root:guest /home/guest/simpleid2  
[root@dastarikov ~]# chmod u+s /home/guest/simpleid2  
[root@dastarikov ~]#  
logout  
[guest@dastarikov ~]$ ls -l simpleid2  
-rwsr-xr-x. 1 root guest 26064 Apr 13 19:49 simpleid2  
[guest@dastarikov ~]$
```

Рис. 2.2: Изменение владельца файла simpleid2 и добавление SUID-бита.

10. Запустили simpleid2 и id (Рис. 2.3):

```
./simpleid2  
id
```



```
[guest@dastarikov ~]$ ./simpleid2  
e_uid=0, e_gid=1001  
real_uid=1001, real_gid=1001  
[guest@dastarikov ~]$ id  
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023  
[guest@dastarikov ~]$
```

Рис. 2.3: Сравнение выводов программ id и simpleid2.

11. Проделайте тоже самое относительно SGID-бита (Рис. 2.4):

```
su -  
chown root:root /home/guest/simpleid2  
chmod u-s /home/guest/simpleid2  
chmod g+s /home/guest/simpleid2  
ls -l simpleid2
```

```
[guest@dastarikov ~]$ su -
Password:
[root@dastarikov ~]# chmod u-s /home/guest/simpleid2
[root@dastarikov ~]# chmod g+s /home/guest/simpleid2
[root@dastarikov ~]#
logout
[guest@dastarikov ~]$ ls -l simpleid2
-rwxr-sr-x. 1 root guest 26064 Apr 13 19:49 simpleid2
[guest@dastarikov ~]$
```

Рис. 2.4: Изменение группы-владельца файла simpleid2 и добавление GUID-бита.

12. Запустили simpleid2 и id (Рис. 2.5):

```
./simpleid2
```

```
id
```

```
[root@dastarikov ~]# chmod g+s /home/guest/simpleid2
[root@dastarikov ~]# ls -l /home/guest/simpleid2
-rwxr-sr-x. 1 root root 26064 Apr 13 19:49 /home/guest/simpleid2
[root@dastarikov ~]#
logout
[guest@dastarikov ~]$ ./simpleid2
e_uid=1001, e_gid=0
real_uid=1001, real_gid=1001
[guest@dastarikov ~]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@dastarikov ~]$
```

Рис. 2.5: Сравнение выводов программ id и simpleid2 для SGID-бита.

13. Создали программу readfile.c (Листинг 2.2):

14. Откомпилировали её.

```
gcc readfile.c -o readfile
```

15. Сменили владельца у файла readfile.c на root и изменили права так, чтобы только суперпользователь мог прочитать его, а guest не мог (Рис. 2.6).

16. Проверили, что пользователь guest не может прочитать файл readfile.c (Рис. 2.5).

---

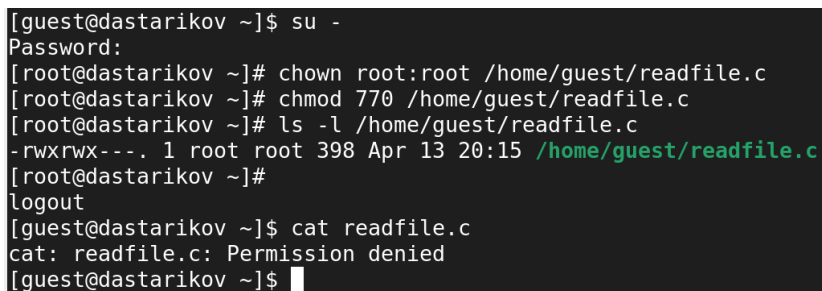
**Листинг 2.3** Текст программы readfile.c

---

```
#include<fcntl.h>
#include<stdio.h>
#include<sys/stat.h>
#include<sys/types.h>
#include<unistd.h>

int main(int argc, char* argv[]){
    unsigned char buffer[16];
    size_t bytes_read;
    int i;
    int fd = open(argv[1], O_RDONLY);
    do{
        bytes_read = read(fd, buffer, sizeof (buffer));
        for (i=0;i<bytes_read;++i) printf("%c", buffer[i]);
    } while (bytes_read == sizeof(buffer));
    close(fd);
    return 0;
}
```

---



```
[guest@dastarikov ~]$ su -
Password:
[root@dastarikov ~]# chown root:root /home/guest/readfile.c
[root@dastarikov ~]# chmod 770 /home/guest/readfile.c
[root@dastarikov ~]# ls -l /home/guest/readfile.c
-rwxrwx---. 1 root root 398 Apr 13 20:15 /home/guest/readfile.c
[root@dastarikov ~]#
logout
[guest@dastarikov ~]$ cat readfile.c
cat: readfile.c: Permission denied
[guest@dastarikov ~]$
```

Рис. 2.6: Изменение прав файла readfile.c и проверка изменений.

17. Сменили у программы readfile владельца и установите SetUID-бит (Рис. 2.7).
18. Проверили, что программа readfile прочитает файл readfile.c (Рис. 2.7):



```
[guest@dastarikov ~]$ su -
Password:
[root@dastarikov ~]# chown root /home/guest/readfile
[root@dastarikov ~]# chmod u+s /home/guest/readfile
[root@dastarikov ~]# ls -l /home/guest/readfile
-rwsr-xr-x. 1 root guest 26008 Apr 13 20:15 /home/guest/readfile
[root@dastarikov ~]#
logout
[guest@dastarikov ~]$ ./readfile readfile.c
#include<fcntl.h>
#include<stdio.h>
#include<sys/stat.h>
#include<sys/types.h>
#include<unistd.h>

int main(int argc, char* argv[]){
    unsigned char buffer[16];
    size_t bytes_read;
    int i;

    int fd = open (argv[1], O_RDONLY);
    do{
        bytes_read = read(fd, buffer, sizeof(buffer));
        for(i=0; i<bytes_read;++i) printf("%c", buffer[i]);
    }while (bytes_read == sizeof(buffer));
    close(fd);
    return 0;
}
[guest@dastarikov ~]$
```

Рис. 2.7: Сравнение выводов программ id и simpleid2 для SGID-бита.

19. Проверили, что программа readfile прочитает файл /etc/shadow (Рис. 2.8)?

```
[guest@dastarikov ~]$ ./readfile /etc/shadow
root:$6$dz/KFZWdA5GU.9ho$LJyGhWIh9Bpg4LNacTwZr7seaDuS1fkUBuYkn5ohDnMz59jZIBw5/1rLGkYE7fEu
7oLMQqZqpK0PUZeKfRpb00::0:99999:7:::
bin:!:19469:0:99999:7:::
daemon:!:19469:0:99999:7:::
adm:!:19469:0:99999:7:::
lp:!:19469:0:99999:7:::
sync:!:19469:0:99999:7:::
shutdown:!:19469:0:99999:7:::
halt:!:19469:0:99999:7:::
```

Рис. 2.8: Сравнение выводов программ id и simpleid2 для SGID-бита.

## 2.2 Исследование Sticky-бита

1. Выяснили, что установлен атрибут Sticky на директории /tmp (Рис. 2.9), выполнив команду

```
ls -l / | grep tmp
```

```
[guest@dastarikov ~]$ ls -l / | grep tmp
drwxrwxrwt. 15 root root 4096 Apr 13 20:24 tmp
[guest@dastarikov ~]$
```

Рис. 2.9: Проверка атрибута Sticky на /tmp.

2. От имени пользователя guest создали файл file01.txt в директории /tmp со словом test (Рис. 2.10):

```
echo "test" > /tmp/file01.txt
```

3. Просмотрели атрибуты у только что созданного файла и разрешили чтение и запись для категории пользователей «все остальные» (Рис. 2.10):

```
ls -l /tmp/file01.txt
chmod o+rw /tmp/file01.txt
ls -l /tmp/file01.txt
```

```
[guest@dastarikov ~]$ ls -l /tmp/file01.txt
-rw-r--r--. 1 guest guest 5 Apr 13 20:27 /tmp/file01.txt
[guest@dastarikov ~]$ echo "test" > /tmp/file01.txt
[guest@dastarikov ~]$ ls -l /tmp/file01.txt
-rw-r--r--. 1 guest guest 5 Apr 13 20:27 /tmp/file01.txt
[guest@dastarikov ~]$ chmod o+rw /tmp/file01.txt
[guest@dastarikov ~]$ ls -l /tmp/file01.txt
-rw-r--rw-. 1 guest guest 5 Apr 13 20:27 /tmp/file01.txt
```

Рис. 2.10: Изменение атрибутов файла file01.txt.

4. От пользователя guest2 (не являющегося владельцем) попробовали прочитать файл /tmp/file01.txt (Рис. 2.11):

```
su guest2
cat /tmp/file01.txt
```

5. От пользователя guest2 попробовали дозаписать в файл /tmp/file01.txt слово test2 командой (Рис. 2.11)

```
echo "test2" >> /tmp/file01.txt
```

6. Проверили содержимое файла командой (Рис. 2.11)

```
cat /tmp/file01.txt
```

7. От пользователя guest2 попробовали записать в файл /tmp/file01.txt слово test3, стерев при этом всю имеющуюся в файле информацию командой (Рис. 2.11)

```
echo "test3" > /tmp/file01.txt
```

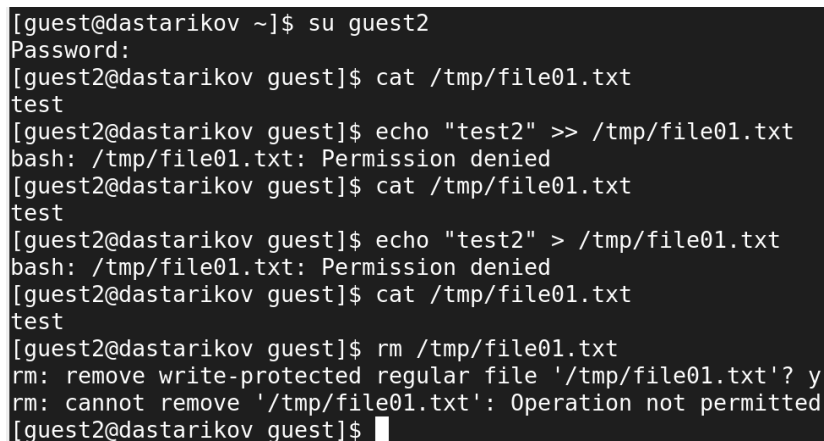
8. Проверили содержимое файла командой

```
cat /tmp/file01.txt
```

9. От пользователя guest2 попробуйте удалить файл /tmp/file01.txt командой

```
rm /tmp/file01.txt
```

При установленном атрибуте Sticky все вышеперечисленные операции запрещены.

A terminal window showing a series of commands and their outputs. The user is 'guest2' on a system named 'dastarikov'. The commands and outputs are: 'su guest2' (Password:), 'cat /tmp/file01.txt' (test), 'echo "test2" >> /tmp/file01.txt' (bash: /tmp/file01.txt: Permission denied), 'cat /tmp/file01.txt' (test), 'echo "test2" > /tmp/file01.txt' (bash: /tmp/file01.txt: Permission denied), 'cat /tmp/file01.txt' (test), and 'rm /tmp/file01.txt' (rm: remove write-protected regular file '/tmp/file01.txt'? y, rm: cannot remove '/tmp/file01.txt': Operation not permitted).

```
[guest@dastarikov ~]$ su guest2
Password:
[guest2@dastarikov guest]$ cat /tmp/file01.txt
test
[guest2@dastarikov guest]$ echo "test2" >> /tmp/file01.txt
bash: /tmp/file01.txt: Permission denied
[guest2@dastarikov guest]$ cat /tmp/file01.txt
test
[guest2@dastarikov guest]$ echo "test2" > /tmp/file01.txt
bash: /tmp/file01.txt: Permission denied
[guest2@dastarikov guest]$ cat /tmp/file01.txt
test
[guest2@dastarikov guest]$ rm /tmp/file01.txt
rm: remove write-protected regular file '/tmp/file01.txt'? y
rm: cannot remove '/tmp/file01.txt': Operation not permitted
[guest2@dastarikov guest]$
```

Рис. 2.11: Проверка прав работы с file01.txt у невладельца файла (guest2).

10. Повысили свои права до суперпользователя следующей командой

```
su -
```

и выполнили после этого команду, снимающую атрибут `t` (Sticky-бит) с директории `/tmp` (Рис. 2.12):

```
chmod -t /tmp
```

```
[guest2@dastarikov guest]$ su -  
Password:  
[root@dastarikov ~]# chmod -t /tmp  
[root@dastarikov ~]#  
logout  
[guest2@dastarikov guest]$ ls -l | grep tmp  
[guest2@dastarikov guest]$ ls -l | grep /tmp  
[guest2@dastarikov guest]$ ls -l / | grep tmp  
drwxrwxrwx. 15 root root 4096 Apr 13 20:32 tmp  
[guest2@dastarikov guest]$
```

Рис. 2.12: Снятие Sticky-бита с `/tmp`.

11. Покинули режим суперпользователя комбинацией `Ctrl+D`.
12. От пользователя `guest2` проверили, что атрибута `t` у директории `/tmp` нет (Рис. 2.12):

```
ls -l / | grep tmp
```

13. Повторили предыдущие шаги. После снятия Sticky-бита с файла мы также не могли записывать и перезаписывать данные файла, но смогли его удалить (Рис. 2.13).

```
[guest2@dastarikov guest]$ echo "test2" >> /tmp/file01.txt  
bash: /tmp/file01.txt: Permission denied  
[guest2@dastarikov guest]$ cat /tmp/file01.txt  
test  
[guest2@dastarikov guest]$ echo "test3" > /tmp/file01.txt  
bash: /tmp/file01.txt: Permission denied  
[guest2@dastarikov guest]$ cat /tmp/file01.txt  
test  
[guest2@dastarikov guest]$ rm /tmp/file01.txt  
rm: remove write-protected regular file '/tmp/file01.txt'? y  
[guest2@dastarikov guest]$ cat /tmp/file01.txt  
cat: /tmp/file01.txt: No such file or directory
```

Рис. 2.13: Проверка прав работы с `file01.txt` у невладельца файла (`guest2`) без Sticky-бита.

15. Повысили свои права до суперпользователя и верните атрибут `t` на директорию `/tmp`:

```
su -  
chmod +t /tmp  
exit
```

## 3 Вывод

В рамках лабораторной работы изучили механизмы изменения идентификаторов, применения SetUID- и Sticky-битов, Получили практические навыки работы в консоли с дополнительными атрибутами. Рассмотрели работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.

## Список литературы

1. Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. СПб.: Питер, 2015. 1120 с.