

# **Отчет по лабораторной работе №5**

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

Асеинова Елизавета

2022 Oct 4th

# Содержание

1. Цель работы	5
2. Выполнение лабораторной работы	6
3. Выводы	11
4. Список литературы	12

## **Список таблиц**

## Список иллюстраций

2.1. Simpleid.c . . . . .	6
2.2. Запуск simpleid.c . . . . .	6
2.3. Дополнение программы . . . . .	7
2.4. Запуск simpleid2.c . . . . .	7
2.5. Команды суперпользователя . . . . .	7
2.6. SetGID бит . . . . .	8
2.7. Запуск readfile.c . . . . .	8
2.8. Запуск readfile.c . . . . .	8
2.9. Запуск readfile.c 2 . . . . .	8
2.10. Проверка readfile.c . . . . .	9
2.11. Проверка readfile.c 2 . . . . .	9
2.12. Тестовый файл . . . . .	9
2.13. Изменение файла другим пользователем . . . . .	10
2.14. Снятие атрибута . . . . .	10

# 1. Цель работы

Целью данной работы является 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. [1]

## 2. Выполнение лабораторной работы

1. Выполнила вход в систему от имени пользователя guest1. Создала программу simpleid.c

```
1 #include <sys/types.h>
2 #include <unistd.h>
3 #include <stdio.h>
4 int
5 main ()
6 {
7     uid_t uid = geteuid ();
8     gid_t gid = getegid ();
9     printf ("uid=%d, gid=%d\n", uid, gid);
10    return 0;
11 }
```

Рис. 2.1.: Simpleid.c

2. Скомпилировала программу и убедилась, что файл программы создан. Выполнила программу. Для сравнения выполнила системную программу id.

```
[guest1@evaseinova ~]$ gcc simpleid.c -o simpleid
[guest1@evaseinova ~]$ ./simpleid
uid=1001, gid=1001
[guest1@evaseinova ~]$ id
uid=1001(guest1) gid=1001(guest1) groups=1001(guest1) context=unconfined_t:s0-s0:c0.c1023
```

Рис. 2.2.: Запуск simpleid.c

3. Усложнила программу, добавив вывод действительных идентификаторов.

```

1 #include <sys/types.h>
2 #include <unistd.h>
3 #include <stdio.h>
4 int
5 main ()
6 {
7     uid_t real_uid = getuid ();
8     uid_t e_uid = geteuid ();
9     gid_t real_gid = getgid ();
10    gid_t e_gid = getegid ();
11    printf ("e_uid=%d, e_gid=%d\n", e_uid, e_gid);
12    printf ("real_uid=%d, real_gid=%d\n", real_uid, real_gid);
13    return 0;
14 }

```

Рис. 2.3.: Дополнение программы

#### 4. Скомпилировала и запустила файл simpleid2.c

```

[guest1@evaseinova ~]$ gcc simpleid2.c -o simpleid2
[guest1@evaseinova ~]$ ./simpleid2
e_uid=1001, e_gid=1001
real_uid=1001, real_gid=1001

```

Рис. 2.4.: Запуск simpleid2.c

- От имени суперпользователя выполнила команды `chown root:guest1 /home/guest1/simpleid2` и `chmod u+s /home/guest1/simpleid2`. Выполнила проверку правильности установки новых атрибутов и смены владельца файла `simpleid2`. Запустила `simpleid2` и `id`. Проделала то же самое для SetGID бита.

```

[root@evaseinova guest1]# chown root:guest1 /home/guest1/simpleid2
[root@evaseinova guest1]# chmod u+s /home/guest1/simpleid2
[root@evaseinova guest1]# ls -l simpleid2
-rwsrwxr-x. 1 root guest1 26008 Oct  4 15:20 simpleid2
[root@evaseinova guest1]# ./simpleid2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[root@evaseinova guest1]# id
uid=0(root) gid=0(root) groups=0(root) context=unconfined_u:unconfined_r:unconfined_t:0:c1023

```

Рис. 2.5.: Команды суперпользователя

```
[root@evaseinova guest1]# chmod g+s /home/guest1/simpleid2
[root@evaseinova guest1]# ls -l simpleid2
-rwsrwsr-x. 1 root guest1 26008 Oct  4 15:20 simpleid2
[root@evaseinova guest1]# ./simpleid2
e_uid=0, e_gid=1001
real uid=0, real_gid=0
[root@evaseinova guest1]# id
uid=0(root) gid=0(root) groups=0(root) context=unconfined_u:unc
0.c1023
```

Рис. 2.6.: SetGID бит

## 6. Создала программу readfile.c

```
1 #include <sys/types.h>
2 #include <unistd.h>
3 #include <stdio.h>
4 #include <fcntl.h>
5 #include <sys/stat.h>
6
7 int main(int argc, char* argv[]) {
8     unsigned char buffer[16];
9     size_t bytes_read;
10    int i;
11
12    int fd = open(argv[1], O_RDONLY);
13    do {
14        bytes_read = read(fd, buffer, sizeof(buffer));
15        for (i=0; i<bytes_read; i++)
16            printf("%c", buffer[i]);
17    } while(bytes_read == sizeof(buffer));
18    close(fd);
19    return 0;
20 }
```

Рис. 2.7.: Запуск readfile.c

## 7. Скомпилировала программу. Сменила владельца у файла readfile.c и изменила права так, чтобы только суперпользователь мог прочитать его, а пользователь guest1 не мог.

```
[guest1@evaseinova ~]$ gcc readfile.c -o readfile
[guest1@evaseinova ~]$ su root
Password:
[root@evaseinova guest1]# chown root:guest1 /home/guest1/readfile.c
[root@evaseinova guest1]# chmod 700 /home/guest1/readfile.c
[root@evaseinova guest1]# cat 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[])
{
```

Рис. 2.8.: Запуск readfile.c

```
[root@evaseinova guest1]# su guest1
[guest1@evaseinova ~]$ cat readfile.c
cat: readfile.c: Permission denied
```

Рис. 2.9.: Запуск readfile.c 2



8. Сменила у программы readfile владельца и установила SetU'D-бит. Проверила, может ли программа readfile прочитать файл readfile.c, Проверила, может ли программа readfile прочитать файл /etc/shadow.

```
[root@evaseinova guest1]# chown root:guest1 /home/guest1/readfile
[root@evaseinova guest1]# chmod u+s /home/guest1/readfile
[root@evaseinova guest1]# su guest1
[guest1@evaseinova ~]$ ./readfile readfile.c
#include <fcntl.h>
#include <stdio.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <unistd.h>
int
```

Рис. 2.10.: Проверка readfile.c

```
[guest1@evaseinova ~]$ ./readfile /etc/shadow
root:$6$YxOzkm.l9jCETdX4$KkjUfQkyHj1AUBsP17GMbi0SVkFMTNwbsiH0
4VJfAoUxHM2yl4rvD/::0:99999:7:::
bin:*.19123:0:99999:7:::
daemon:*.19123:0:99999:7:::
adm:*.19123:0:99999:7:::
lp:*.19123:0:99999:7:::
sync:*.19123:0:99999:7:::
shutdown:*.19123:0:99999:7:::
halt:*.19123:0:99999:7:::
mail:*.19123:0:99999:7:::
operator:*.19123:0:99999:7:::
games:*.19123:0:99999:7:::
```

Рис. 2.11.: Проверка readfile.c 2

9. Выяснила, установлен ли атрибут Sticky на директории /tmp. От имени пользователя guest1 создала файл file01.txt в директории /tmp со словом test. Просмотрела атрибуты у только что созданного файла и разрешила чтение и запись для категории пользователей «все остальные».

```
[guest1@evaseinova ~]$ ls -l / | grep tmp
drwxrwxrwt. 17 root root 4096 Oct  4 15:44 tmp
[guest1@evaseinova ~]$ echo "test" > /tmp/file01.txt
[guest1@evaseinova ~]$ ls -l /tmp/file01.txt
-rw-rw-r--. 1 guest1 guest1 5 Oct  4 15:50 /tmp/file01.txt
[guest1@evaseinova ~]$ chmod o+rw /tmp/file01.txt
[guest1@evaseinova ~]$ ls -l /tmp/file01.txt
-rw-rw-rw-. 1 guest1 guest1 5 Oct  4 15:50 /tmp/file01.txt
```

Рис. 2.12.: Тестовый файл

10. От пользователя guest2 попробовала прочитать файл, дозаписать в файл слово test2. Проверила содержимое файла. Попробовала записать в файл слово test3, стерев при этом всю имеющуюся в файле информацию. Проверила содержимое файла. Попробовала удалить файл.

```
[guest1@evaseinova ~]$ su guest2
Password:
[guest2@evaseinova guest1]$ cat /tmp/file01.txt
test
[guest2@evaseinova guest1]$ echo "test2" > /tmp/file01.txt
[guest2@evaseinova guest1]$ cat /tmp/file01.txt
test2
[guest2@evaseinova guest1]$ echo "test3" > /tmp/file01.txt
[guest2@evaseinova guest1]$ cat /tmp/file01.txt
test3
[guest2@evaseinova guest1]$ rm /tmp/file01.txt
rm: cannot remove '/tmp/file01.txt': Operation not permitted
```

Рис. 2.13.: Изменение файла другим пользователем

11. Повысила свои права до суперпользователя и выполнила команду, снимающую атрибут t с директории /tmp. Покинула режим суперпользователя. От пользователя guest2 проверила, что атрибута t у директории /tmp нет. Попробовала прочитать файл, дозаписать в файл, удалить его. Повысила свои права до суперпользователя и вернула атрибут t на директорию /tmp.

```
[guest2@evaseinova guest1]$ su -
Password:
[root@evaseinova ~]# chmod -t /tmp
[root@evaseinova ~]# exit
logout
[guest2@evaseinova guest1]$ ls -l / | grep tmp
drwxrwxrwx. 18 root root 4096 Oct  4 15:54 tmp
[guest2@evaseinova guest1]$ cat /tmp/file01.txt
test3
[guest2@evaseinova guest1]$ echo "test2" > /tmp/file01.txt
[guest2@evaseinova guest1]$ cat /tmp/file01.txt
test2
[guest2@evaseinova guest1]$ echo "test3" > /tmp/file01.txt
[guest2@evaseinova guest1]$ cat /tmp/file01.txt
test3
[guest2@evaseinova guest1]$ rm /tmp/file01.txt
[guest2@evaseinova guest1]$ su -
Password:
[root@evaseinova ~]# chmod +t /tmp
[root@evaseinova ~]# exit
logout
```

Рис. 2.14.: Снятие атрибута

## 3. Выводы

В ходе работы мы 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.

## **4. Список литературы**

1. Методические материалы курса