

# **Отчет по лабораторной работе №5**

**Лабораторная работа №5: Дискреционное разграничение прав в Linux.  
Исследование влияния дополнительных атрибутов**

Голова Варвара Алексеевна, НФИбд-03-18

2021, 29 November

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Выполнение работы</b>	<b>5</b>
2.1	Создание программы . . . . .	5
2.2	Компиляция . . . . .	5
2.3	Программа . . . . .	6
2.4	Компиляция . . . . .	6
2.5	Программа readfile.c . . . . .	7
2.6	Атрибуты . . . . .	8
<b>3</b>	<b>Выводы</b>	<b>11</b>

# List of Figures

2.1	Simpleid.c . . . . .	5
2.2	Компиляция . . . . .	5
2.3	Simpleid . . . . .	6
2.4	Id . . . . .	6
2.5	Simpleid2.c . . . . .	6
2.6	Компиляция . . . . .	6
2.7	Команды . . . . .	7
2.8	Проверка . . . . .	7
2.9	Запуск программ . . . . .	7
2.10	Readfile.c . . . . .	8
2.11	Права . . . . .	8
2.12	Атрибуты . . . . .	9
2.13	Атрибуты File01 . . . . .	9
2.14	Атрибут t . . . . .	10

# 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 Выполнение работы

### 2.1 Создание программы

Создала программу 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;
}
```

Figure 2.1: Simpleid.c

### 2.2 Компиляция

Скомпилировала программу и убедилась, что файл программы создан.

```
[quest_1@vaqolova ~]$ gcc simpleid.c -o simpleid
```

Figure 2.2: Компиляция

Выполнила программу simpleid.

```
[guest_1@vagolova ~]$ ./simpleid  
uid=1003, gid=1003
```

Figure 2.3: Simpleid

Выполнила системную программу id. Результаты совпадают.

```
uid=1003(guest_1) gid=1003(guest_1) группы=1003(guest_1) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

Figure 2.4: Id

## 2.3 Программа

Усложнила программу, добавив вывод действительных идентификаторов, получившуюся программу назвала simpleid2.c.

```
#include <sys/types.h>  
#include <unistd.h>  
#include <stdio.h>  
  
int  
main ()  
{  
    uid_t real_uid = getuid ();  
    gid_t e_gid = getegid ();  
  
    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;  
}
```

Figure 2.5: Simpleid2.c

## 2.4 Компиляция

Скомпилировала simpleid2.c

```
[guest_1@vagolova ~]$ gcc simpleid2.c -o simpleid2
```

Figure 2.6: Компиляция

От имени суперпользователя выполнила команды. Команда su позволяет выполнять команды от имени суперпользователя.

```
[guest_1@vago-lova ~]$ su
Пароль:
[root@vago-lova guest_1]# chown root:guest_1 /home/guest_1/simpleid2
[root@vago-lova guest_1]# chmod u+s /home/guest_1/simpleid2
```

Figure 2.7: Команды

Выполнила проверку правильности установки новых атрибутов и смены владельца файла simpleid2:

```
[guest_1@vago-lova ~]$ ls -l simpleid2
-rwsrwxr-x. 1 root guest_1 17648 ноя 13 15:59 simpleid2
```

Figure 2.8: Проверка

Запустила simpleid2 и id. Результаты совпадают.

```
[guest_1@vago-lova ~]$ ./simpleid2
e uid=0, e gid=1003
real uid=1003, real gid=1003
[guest_1@vago-lova ~]$ id
uid=1003(guest_1) gid=1003(guest_1) группы=1003(guest_1) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

Figure 2.9: Запуск программ

## 2.5 Программа readfile.c

Создала программу 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;
}

```

Figure 2.10: Readfile.c

Сменила владельца у файла readfile.c и изменила права так, чтобы только суперпользователь мог прочитать его, а guest\_1 не мог. Проверила, что пользователь guest\_1 не может прочитать файл readfile.c. Сменила у программы readfile владельца и установите SetU'D-бит.

```

[guest_1@vagolova ~]$ chmod 000 /home/guest_1/readfile.c
[guest_1@vagolova ~]$ cat readfile.c
cat: readfile.c: Отказано в доступе
[guest_1@vagolova ~]$ chown root:root /home/guest_1/readfile.c
chown: изменение владельца '/home/guest_1/readfile.c': Операция не позволена
[guest_1@vagolova ~]$ sudo chown root:root /home/guest_1/readfile.c
[sudo] пароль для guest_1:
попробуйте ещё раз.
[sudo] пароль для guest_1:
guest_1 is not in the sudoers file. This incident will be reported.
[guest_1@vagolova ~]$ su
Пароль:
[root@vagolova guest_1]# chown root:root /home/guest_1/simpleid2
[root@vagolova guest_1]# chown root:guest_1 /home/guest_1/readfile
[root@vagolova guest_1]# chmod u+s /home/guest_1/readfile

```

Figure 2.11: Права

## 2.6 Атрибуты

Выяснила, установлен ли атрибут Sticky на директории /tmp. Атрибут установлен. От имени пользователя guest\_1 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»

```
[guest_1@vagolova ~]$ ls -l / | grep tmp
drwxrwxrwt. 13 root root 4096 ноя 13 16:35 tmp
[guest_1@vagolova ~]$ echo "test" > /tmp/file01.txt
[guest_1@vagolova ~]$ ls -l /tmp/file01.txt
-rw-rw-r--. 1 guest_1 guest_1 5 ноя 13 16:36 /tmp/file01.txt
[guest_1@vagolova ~]$ chmod o+rw /tmp/file01.txt
[guest_1@vagolova ~]$ ls -l /tmp/file01.txt
-rw-rw-rw-. 1 guest_1 guest_1 5 ноя 13 16:36 /tmp/file01.txt
```

Figure 2.12: Атрибуты

От пользователя guest\_2 попробовала прочитать файл /tmp/file01.txt. Это удалось. От пользователя guest\_2 попробовала дозаписать в файл /tmp/file01.txt слово test2. Это также удалось. Проверила содержимое файла командой. От пользователя guest\_2 попробовала записать в файл /tmp/file01.txt слово test3, стерев при этом всю имеющуюся в файле информацию. Проверила содержимое файла командой. От пользователя guest\_2 попробовала удалить файл /tmp/file01.txt. Это не удалось. Повысила свои права до суперпользователя и выполнила после этого команду, снимающую атрибут t (Sticky-бит) с директории /tmp. Покинула режим суперпользователя. От пользователя guest\_2 проверила, что атрибута t у директории /tmp нет.

```
[guest_1@vagolova ~]$ ls -l / | grep tmp
drwxrwxrwt. 13 root root 4096 ноя 13 16:35 tmp
[guest_1@vagolova ~]$ echo "test" > /tmp/file01.txt
[guest_1@vagolova ~]$ ls -l /tmp/file01.txt
-rw-rw-r--. 1 guest_1 guest_1 5 ноя 13 16:36 /tmp/file01.txt
[guest_1@vagolova ~]$ chmod o+rw /tmp/file01.txt
[guest_1@vagolova ~]$ ls -l /tmp/file01.txt
-rw-rw-rw-. 1 guest_1 guest_1 5 ноя 13 16:36 /tmp/file01.txt
```

Figure 2.13: Атрибуты File01

Повторила предыдущие шаги. Мне удалось удалить файл от имени пользователя, не являющегося его владельцем. Повысила свои права до суперпользователя и вернула атрибут t на директорию /tmp.

```
[guest_2@vagolova ~]$ cat /tmp/file01.txt
test3
[guest_2@vagolova ~]$ echo "test2" > /tmp/file01.txt
[guest_2@vagolova ~]$ cat /tmp/file01.txt
test2
[guest_2@vagolova ~]$ rm /tmp/file01.txt
[guest_2@vagolova ~]$ su
Пароль:
[root@vagolova guest_2]# chmod +t /tmp
[root@vagolova guest_2]# exit
exit
[guest_2@vagolova ~]$ ls -l / | grep tmp
drwxrwxrwt. 16 root root 4096 ноя 13 16:45 tmp
[guest_2@vagolova ~]$
```

Figure 2.14: Атрибут t

## 3 Выводы

Я изучила механизмы изменения идентификаторов, применения SetUID- и Sticky-битов. Получила практические навыки работы в консоли с дополнительными атрибутами. Рассмотрела работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.