

Лабораторная работа №5

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

Валиева Найля Разимовна

Содержание

| | | |
|---|--------------------------------|----|
| 1 | Цель работы | 6 |
| 2 | Задание | 7 |
| 3 | Выполнение лабораторной работы | 8 |
| 4 | Выводы | 20 |
| 5 | Список литературы | 21 |

Список таблиц

Список иллюстраций

| | | |
|------|--|----|
| 3.1 | Установка компилятора gcc | 8 |
| 3.2 | Отключение системы запретов | 8 |
| 3.3 | Проверка названий компиляторов | 9 |
| 3.4 | Создание программы simpleid.c (часть 1) | 9 |
| 3.5 | Создание программы simpleid.c (часть 2) | 9 |
| 3.6 | Компиляция программы | 10 |
| 3.7 | Выполнение созданной программы | 10 |
| 3.8 | Выполнение системной программы id | 10 |
| 3.9 | Усложнение программы | 11 |
| 3.10 | Переименование программы | 11 |
| 3.11 | Компиляция и запуск файла | 11 |
| 3.12 | Смена владельца и атрибутов от имени суперпользователя | 12 |
| 3.13 | Использование оператора su | 12 |
| 3.14 | Проверка правильности установления атрибутов | 12 |
| 3.15 | Проверка id пользователя и группы | 12 |
| 3.16 | Повторение операций относительно SetGID-бита | 13 |
| 3.17 | Создание программы readfile.c (часть 1) | 13 |
| 3.18 | Создание программы readfile.c (часть 2) | 13 |
| 3.19 | Компиляция программы | 13 |
| 3.20 | Смена владельца и изменение прав файла | 14 |
| 3.21 | Попытка прочесть файл | 14 |
| 3.22 | Смена владельца и установка SetUID-бита | 14 |
| 3.23 | Проверка чтения файла (часть 1) | 14 |
| 3.24 | Проверка чтения файла (часть 2) | 15 |
| 3.25 | Проверка чтения файла /etc/shadow | 15 |
| 3.26 | Проверка нахождения атрибута Sticky на директории /tmp | 15 |
| 3.27 | Создание файла и внесение записи в него | 16 |
| 3.28 | Просмотр атрибутов файла и установление прав на чтение и запись для категории “все остальные” | 16 |
| 3.29 | Чтение файла от имени пользователя guest2 | 16 |
| 3.30 | Дозапись слова в файл от имени пользователя guest2 | 17 |
| 3.31 | Проверка содежимого в файле от имени пользователя guest2 | 17 |
| 3.32 | Перезапись информации в файл от имени пользователя guest2 | 17 |
| 3.33 | Проверка содежимого в файле от имени пользователя guest2 | 17 |
| 3.34 | Попытка удаления файла от имени пользователя guest2 | 17 |
| 3.35 | Повышение прав до суперпользователя. Снятие атрибута t | 18 |
| 3.36 | Выход из режима суперпользователя | 18 |

| | |
|---|----|
| 3.37 Проверка отсутствия атрибута t | 18 |
| 3.38 Повтор предыдущих шагов | 19 |
| 3.39 Переход в режим суперпользователя и возврат атрибута t | 19 |

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. [1]

2 Задание

1. Подготовить лабораторный стенд
2. Рассмотреть компиляцию программ
3. Создать программы
4. Исследовать Sticky-бит

3 Выполнение лабораторной работы

1. Предварительно установила компилятор gcc с помощью команды `yum install gcc` (рис - @fig:001).

```
Зависимости установлены:
cloog-ppl.i686 0:0.15.7-1.2.el6      cpp.i686 0:4.4.7-23.el6
mpfr.i686 0:2.4.1-6.el6              ppl.i686 0:0.10.2-11.el6

Готово!
[root@nrvalieva Рабочий стол]# gcc -v
Используются внутренние спецификации.
Целевая архитектура: i686-redhat-linux
Параметры конфигурации: ../configure --prefix=/usr --mandir=/usr/share/man --inf
odir=/usr/share/info --with-bugurl=http://bugzilla.redhat.com/bugzilla --enable-
bootstrap --enable-shared --enable-threads=posix --enable-checking=release --wit
h-system-zlib --enable-_cxa_atexit --disable-libunwind-exceptions --enable-gnu-
unique-object --enable-languages=c,c++,objc,obj-c++,java,fortran,ada --enable-ja
va-awt=gtk --disable-dssi --with-java-home=/usr/lib/jvm/java-1.5.0-gcj-1.5.0.0/j
re --enable-libgcj-multifile --enable-java-maintainer-mode --with-ecj-jar=/usr/s
hare/java/eclipse-ecj.jar --disable-libjava-multilib --with-ppl --with-cloog --w
ith-tune=generic --with-arch=i686 --build=i686-redhat-linux
Модель многопоточности: posix
gcc версия 4.4.7 20120313 (Red Hat 4.4.7-23) (GCC)
[root@nrvalieva Рабочий стол]# setenforce 0
[root@nrvalieva Рабочий стол]# getenforce
Permissive
[root@nrvalieva Рабочий стол]# █
```

Рис. 3.1: Установка компилятора gcc

Отключила систему защиты SELinux с помощью команды `setenforce 0`. После этого команда `getenforce` вывела `Permissive` (рис @fig:002).

```
gcc версия 4.4.7 20120313 (Red Hat 4.4.7-23) (GCC)
[root@nrvalieva Рабочий стол]# setenforce 0
[root@nrvalieva Рабочий стол]# getenforce
Permissive
[root@nrvalieva Рабочий стол]# █
```

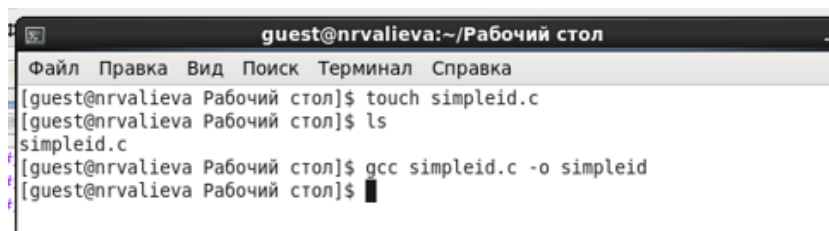
Рис. 3.2: Отключение системы запретов

2. Изучила компиляцию программ. Компилятор языка C называется gcc. Компилятор языка C++ называется g++ и запускается с параметрами почти так же, как gcc. Проверила это с помощью команд `whereis gcc` и `whereis g++` (рис @fig:003).

```
[root@nrvalieva Рабочий стол]# whereis gcc
gcc: /usr/bin/gcc /usr/lib/gcc /usr/libexec/gcc /usr/share/man/man1/gcc.1.gz
[root@nrvalieva Рабочий стол]# whereis g++
g++:
[root@nrvalieva Рабочий стол]#
```

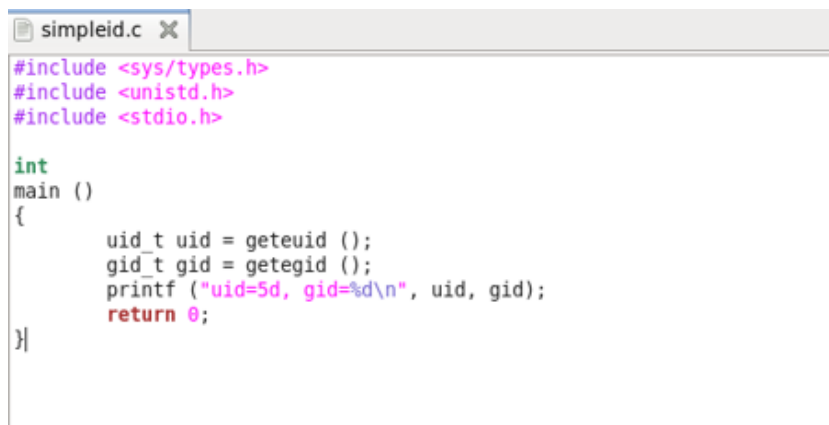
Рис. 3.3: Проверка названий компиляторов

3. Вошла в систему от имени пользователя `guest` и создала программу `simpleid.c` (рис @fig:004 и рис @fig:005).



```
guest@nrvalieva:~/Рабочий стол
Файл Правка Вид Поиск Терминал Справка
[guest@nrvalieva Рабочий стол]$ touch simpleid.c
[guest@nrvalieva Рабочий стол]$ ls
simpleid.c
[guest@nrvalieva Рабочий стол]$ gcc simpleid.c -o simpleid
[guest@nrvalieva Рабочий стол]$
```

Рис. 3.4: Создание программы `simpleid.c` (часть 1)



```
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;
}
```

Рис. 3.5: Создание программы `simpleid.c` (часть 2)

Скомпилировала программу и убедилась, что файл программы создан с помощью команды `gcc simpleid.c -o simpleid` (рис @fig:006)

```
[guest@nrvalieva Рабочий стол]$ gcc simpleid.c -o simpleid
[guest@nrvalieva Рабочий стол]$
```

Рис. 3.6: Компиляция программы

Выполнила программу `simpleid` (рис @fig:007)

```
[guest@nrvalieva Рабочий стол]$ gcc simpleid.c -o simpleid
[guest@nrvalieva Рабочий стол]$ ./simpleid
uid=5d, gid=501
```

Рис. 3.7: Выполнение созданной программы

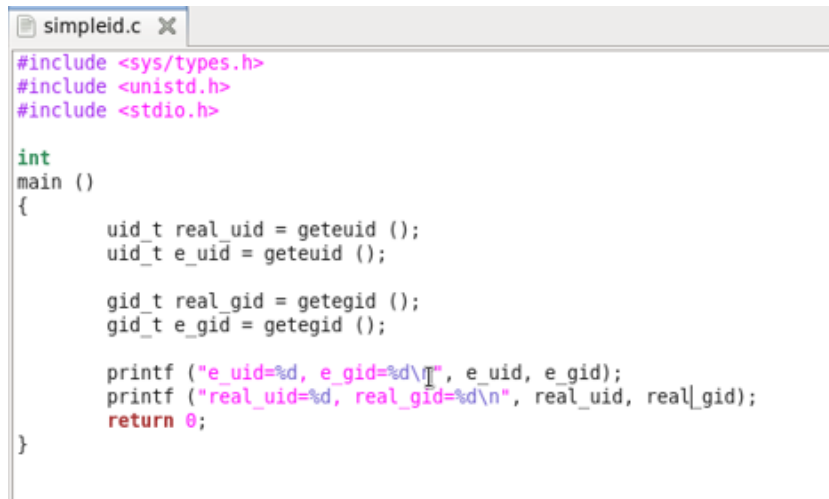
Выполнила системную программу `id` (рис @fig:008)

```
[guest@nrvalieva Рабочий стол]$ id
uid=501(guest) gid=501(guest) группы=501(guest) контекст=unconfined_u:unconfined
r:unconfined t:s0-s0:c0.c1023
```

Рис. 3.8: Выполнение системной программы `id`

Вывод обеих команд совпадает.

Усложнила программу, добавив вывод действительных идентификаторов (рис @fig:009)



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

int
main ()
{
    uid_t real_uid = geteuid ();
    uid_t e_uid = geteuid ();

    gid_t real_gid = getegid ();
    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;
}
```

Рис. 3.9: Усложнение программы

Получившуюся программу назвала simpleid2.c (рис @fig:010)

```
[guest@nrvalieva Рабочий стол]$ mv simpleid.c simpleid2.c
[guest@nrvalieva Рабочий стол]$ ls
simpleid  simpleid2.c  simpleid.c~
[guest@nrvalieva Рабочий стол]$ gcc simpleid2.c -o simpleid2
[guest@nrvalieva Рабочий стол]$ ls
```

Рис. 3.10: Переименование программы

Скомпилировала и запустила simpleid2.c (рис @fig:011)

```
[guest@nrvalieva Рабочий стол]$ gcc simpleid2.c -o simpleid2
[guest@nrvalieva Рабочий стол]$ ls
simpleid  simpleid2  simpleid2.c  simpleid.c~
[guest@nrvalieva Рабочий стол]$ ./simpleid2
e_uid=501, e_gid=501
real_uid=501, real_gid=501
[guest@nrvalieva Рабочий стол]$ █
```

Рис. 3.11: Компиляция и запуск файла

От имени суперпользователя выполнила следующие команды (рис @fig:012)

```
[root@nrvalieva Рабочий стол]# chown root:guest simpleid2
[root@nrvalieva Рабочий стол]# ls
simpleid simpleid2 simpleid2.c simpleid.c~
[root@nrvalieva Рабочий стол]# chmod u+s simpleid2
[root@nrvalieva Рабочий стол]# ls -l
итого 24
-rwxrwxr-x. 1 guest guest 4890 Ноя 13 21:17 simpleid
-rwsrwxr-x. 1 root guest 4971 Ноя 13 21:21 simpleid2
-rw-rw-r--. 1 guest guest 315 Ноя 13 21:20 simpleid2.c
-rw-rw-r--. 1 guest guest 180 Ноя 13 21:17 simpleid.c~
[root@nrvalieva Рабочий стол]#
```

Рис. 3.12: Смена владельца и атрибутов от имени суперпользователя

Использовала su для временного повышения своих прав (рис @fig:013)

```
[guest@nrvalieva Рабочий стол]$ su
Пароль:
```

Рис. 3.13: Использование оператора su

Команда su используется для получения прав суперпользователя.

Выполнила проверку правильности установки новых атрибутов и смены владельца файла simpleid2 (рис @fig:014)

```
-rwsrwxr-x. 1 root guest 4971 Ноя 13 21:21 simpleid2
```

Рис. 3.14: Проверка правильности установления атрибутов

Запустила simpleid2 и id (рис @fig:015)

```
[root@nrvalieva Рабочий стол]# ./simpleid2
e_uid=0, e_gid=0
real uid=0, real gid=0
[root@nrvalieva Рабочий стол]# id
uid=0(root) gid=0(root) группы=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

Рис. 3.15: Проверка id пользователя и группы

Проделала то же самое относительно SetGID-бита (рис @fig:016)

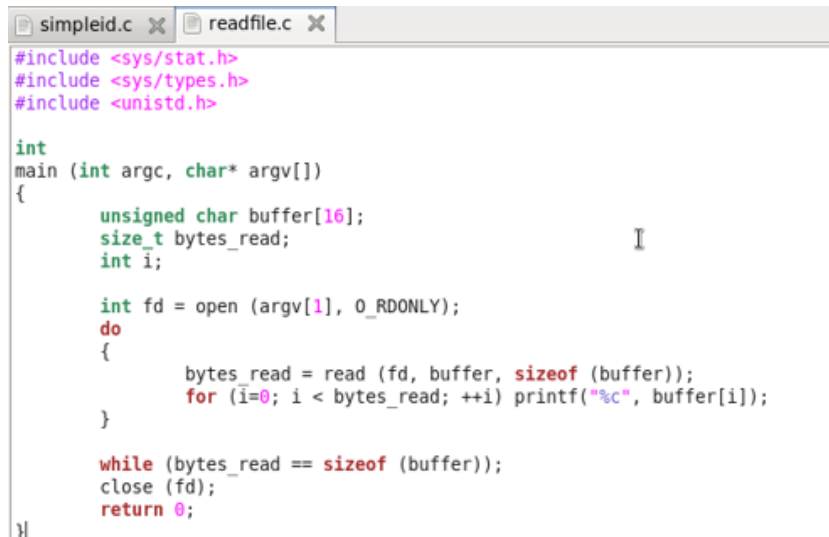
```
[root@nrvalieva Рабочий стол]# chmod g+s simpleid2
[root@nrvalieva Рабочий стол]# ls -l
итого 24
-rwxrwxr-x. 1 guest guest 4890 Ноя 13 21:17 simpleid
-rwsrwsr-x. 1 root  guest 4971 Ноя 13 21:21 simpleid2
-rw-rw-r--. 1 guest guest 315  Ноя 13 21:20 simpleid2.c
-rw-rw-r--. 1 guest guest 180  Ноя 13 21:17 simpleid.c~
```

Рис. 3.16: Повторение операций относительно SetGID-бита

Создала программу `readfile.c` (рис @fig:017 и рис @fig:018)

```
[guest@nrvalieva Рабочий стол]$ touch readfile.c
[guest@nrvalieva Рабочий стол]$ █
```

Рис. 3.17: Создание программы `readfile.c` (часть 1)



```
simpleid.c x readfile.c x
#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;
}
```

Рис. 3.18: Создание программы `readfile.c` (часть 2)

Откомпилировала созданную программу (рис @fig:019)

```
[guest@nrvalieva Рабочий стол]$ touch readfile.c
[guest@nrvalieva Рабочий стол]$ gcc readfile.c -o readfile
[guest@nrvalieva Рабочий стол]$ ls
readfile  readfile.c~  simpleid2  simpleid.c~
readfile.c  simpleid  simpleid2.c
```

Рис. 3.19: Компиляция программы

Сменила владельца у файла `readfile.c` и изменила права так, чтобы только суперпользователь мог прочитать его, а `guest` не мог (рис @fig:020)

```
[root@nrvalieva Рабочий стол]# chown root:root readfile.c
[root@nrvalieva Рабочий стол]# chmod 700 readfile.c
```

Рис. 3.20: Смена владельца и изменение прав файла

Проверила, что пользователь `guest` не может прочитать файл `readfile.c` (рис @fig:021)

```
[guest@nrvalieva Рабочий стол]$ cat readfile.c
cat: readfile.c: Отказано в доступе
```

Рис. 3.21: Попытка прочесть файл

Сменила у программы `readfile` владельца и установила SetUID-бит (рис @fig:022)

```
[root@nrvalieva Рабочий стол]# chown root:root readfile
[root@nrvalieva Рабочий стол]# chmod u+s readfile
```

Рис. 3.22: Смена владельца и установка SetUID-бита

Проверила, может ли программа `readfile` прочитать файл `readfile.c`. Да, может. (рис @fig:023 и рис @fig:024)

```
[root@nrvalieva Рабочий стол]# ./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;
```

Рис. 3.23: Проверка чтения файла (часть 1)

От имени пользователя `guest` создала файл `file01.txt` в директории `/tmp` со словом `test` (рис @fig:027):

```
[guest@nrvalieva ~]$ echo "test" > /tmp/file01.txt
```

Рис. 3.27: Создание файла и внесение записи в него

Просмотрела атрибуты у только что созданного файла и разрешила чтение и запись для категории пользователей “все остальные” (рис @fig:028):

```
[guest@nrvalieva ~]$ ls -l tmp/file01.txt
-rw-rw-r--. 1 guest guest 5 Ноя 13 21:37 tmp/file01.txt
[guest@nrvalieva ~]$ chmod o+rw tmp/file01.txt
[guest@nrvalieva ~]$ ls [l tmp/file01.txt
ls: невозможно получить доступ к [l: Нет такого файла или каталога
tmp/file01.txt
[guest@nrvalieva ~]$ ls -l tmp/file01.txt
-rw-rw-rw-. 1 guest guest 5 Ноя 13 21:37 tmp/file01.txt
[guest@nrvalieva ~]$ █
```

Рис. 3.28: Просмотр атрибутов файла и установление прав на чтение и запись для категории “все остальные”

От имени пользователя `guest2` (не являющегося владельцем) прочитала файл `/tmp/file01.txt` (рис @fig:029):

```
[guest@nrvalieva ~]$ su - guest2
Пароль:
[guest2@nrvalieva ~]$ cat /tmp/file01.txt
test
[guest2@nrvalieva ~]$ █
```

Рис. 3.29: Чтение файла от имени пользователя `guest2`

От имени пользователя `guest2` дозаписала в файл `/tmp/file01.txt` слово `test2` (рис @fig:030):


```
[guest2@nrvalieva ~]$ echo "test" >> /tmp/file01.txt
[guest2@nrvalieva ~]$ cat /tmp/file01.txt
test
test
```

Рис. 3.30: Дозапись слова в файл от имени пользователя guest2

Проверила содержимое файла (рис @fig:031):

```
[guest2@nrvalieva ~]$ echo "test" >> /tmp/file01.txt
[guest2@nrvalieva ~]$ cat /tmp/file01.txt
test
test
```

Рис. 3.31: Проверка содежимого в файле от имени пользователя guest2

От имени пользователя guest2 записала в файл /tmp/file01.txt слово test3, стерев при этом всю имеющуюся в файле информацию (рис @fig:032):

```
[guest2@nrvalieva ~]$ echo "test3" > /tmp/file01.txt
[guest2@nrvalieva ~]$ cat /tmp/file01.txt
test3
```

Рис. 3.32: Перезапись информации в файл от имени пользователя guest2

Проверила содержимое файла (рис @fig:033):

```
[guest2@nrvalieva ~]$ echo "test3" > /tmp/file01.txt
[guest2@nrvalieva ~]$ cat /tmp/file01.txt
test3
```

Рис. 3.33: Проверка содежимого в файле от имени пользователя guest2

От имени пользователя guest2 попробовала удалить файл /tmp/file01.txt (рис @fig:034):

```
[guest2@nrvalieva ~]$ rm /tmp/file01.txt
rm: невозможно удалить «/tmp/file01.txt»: Операция не допускается
[guest2@nrvalieva ~]$
```

Рис. 3.34: Попытка удаления файла от имени пользователя guest2

Мне не удалось удалить файл.

Повысила свои права до суперпользователя и выполнила после этого команду, снимающую атрибут `t` (Sticky-бит) с директории `/tmp` (рис @fig:035):

```
[guest2@nrvalieva ~]$ su -  
Пароль:  
[root@nrvalieva ~]# chmod -t /tmp  
[root@nrvalieva ~]# exit  
logout
```

Рис. 3.35: Повышение прав до суперпользователя. Снятие атрибута `t`

Покинула режим суперпользователя командой `exit` (рис @fig:036):

```
[root@nrvalieva ~]# exit  
logout
```

Рис. 3.36: Выход из режима суперпользователя

От имени пользователя `guest2` проверила, что атрибута `t` у директории `/tmp` нет (рис @fig:037):

```
[guest2@nrvalieva ~]$ ls -l / | grep tmp  
drwxrwxrwx. 24 root root 4096 Ноя 13 21:37 tmp
```

Рис. 3.37: Проверка отсутствия атрибута `t`

Повторила предыдущие шаги (рис @fig:038):

```
[guest2@nrvalieva ~]$ echo "test" > /tmp/file01.txt
[guest2@nrvalieva ~]$ cat /tmp/file01.txt
test
[guest2@nrvalieva ~]$ echo "test2" >> /tmp/file01.txt
[guest2@nrvalieva ~]$ cat /tmp/file01.txt
test
test2
[guest2@nrvalieva ~]$ echo "test3" > /tmp/file01.txt
[guest2@nrvalieva ~]$ cat /tmp/file01.txt
test3
[guest2@nrvalieva ~]$ rm /tmp/file01.txt
[guest2@nrvalieva ~]$ █
```

Рис. 3.38: Повтор предыдущих шагов

Как видно из рисунка, удалось выполнить все команды, которые были рассмотрены выше, включая удаление.

Повысила свои права до суперпользователя и вернула атрибут `t` на директорию `/tmp` (рис @fig:039):

```
[guest2@nrvalieva ~]$ su -
Пароль:
[root@nrvalieva ~]# chmod +t /tmp
```

Рис. 3.39: Переход в режим суперпользователя и возврат атрибута `t`

4 Выводы

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

5 Список литературы

1. Кулябов Д. С., Королькова А. В., Геворкян М. Н. Информационная безопасность компьютерных сетей. Лабораторная работа № 5. Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов