

# Лабораторная работа №5

## Основы информационной безопасности

---

Мурашов И. В., НКАбд-03-23

19 апреля 2025

Российский университет дружбы народов, Москва, Россия

- Мурашов Иван Вячеславович
- Студент, 2 курс, группа НКАбд-03-23
- Российский университет дружбы народов
- 1132236018@rudn.ru
- <https://neve7mind.github.io>

Изучение механизмов изменения идентификаторов, применения SetUID- и Sticky-битов. Получение практических навыков работы в кон- соли с дополнительными атрибутами. Рассмотрение работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.

# Выполнение лабораторной работы

Для лабораторной работы необходимо проверить, установлен ли компилятор gcc, команда `gcc -v` позволяет это сделать. Также осуществляется отключение системы запретов с помощью `setenforce 0` (рис. 1).

```
[ivmurashov@ivmurashov ~]$ gcc -v
Using built-in specs.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=/usr/libexec/gcc/x86_64-redhat-linux/11/lto-wrapper
OFFLOAD_TARGET_NAMES=nvptx-none
OFFLOAD_TARGET_DEFAULT=1
Target: x86_64-redhat-linux
Configured with: ../configure --enable-bootstrap --enable-host-pie --enable-host-bind-now --enable-languages=c,c++,fortran,lto --prefix=/usr --mandir=/usr/share/man --infodir=/usr/share/info --with-bugurl=https://bugs.rockylinux.org/ --enable-shared --enable-threads=posix --enable-checking=release --with-system-zlib --enable-__cxa_atexit --disable-libunwind-exceptions --enable-gnu-unique-object --enable-linker-build-id --with-gcc-major-version-only --enable-plugin --enable-initfini-array --without-isl --enable-multilib --with-linker-hash-style=gnu --enable-offload-targets=nvptx-none --without-cuda-driver --enable-gnu-indirect-function --enable-cet --with-tune=generic --with-arch_64=x86-64-v2 --with-arch_32=x86-64 --build=x86_64-redhat-linux --with-build-config=bootstrap-lto --enable-link-serialization=1
Thread model: posix
Supported LTO compression algorithms: zlib zstd
gcc version 11.5.0 20240719 (Red Hat 11.5.0-5) (GCC)
[ivmurashov@ivmurashov ~]$ sudo setenforce
[sudo] password for ivmurashov:
sudo: a password is required
[ivmurashov@ivmurashov ~]$ sudo setenforce 0
[sudo] password for ivmurashov:
[ivmurashov@ivmurashov ~]$ getenforce
Permissive
```

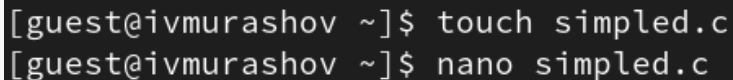
**Рис. 1:** Подготовка к лабораторной работе

Осуществляется вход от имени пользователя guest (рис. 2).

```
[ivmurashov@ivmurashov ~]$ sudo su guest  
[sudo] password for ivmurashov:  
[guest@ivmurashov ivmurashov]$
```

**Рис. 2:** Вход от имени пользователя guest

Создание файла `simplified.c` и запись в файл кода (рис. 3)

A terminal window with a dark background and light gray text. It shows two commands being executed in sequence. The first command creates a file named 'simplified.c' using the 'touch' command. The second command opens the file 'simplified.c' using the 'nano' text editor. A small white cursor is visible at the end of the second line.

```
[guest@ivmurashov ~]$ touch simplified.c  
[guest@ivmurashov ~]$ nano simplified.c
```

**Рис. 3:** Создание файла

```
C++ Листинг 1 #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; }
```

## Выполнение лабораторной работы

Содержимое файла выглядит следующим образом (рис. 4)

```
GNU nano 5.6.1
#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. 4: Содержимое файла



Компилирую файл, проверяю, что он скомпилировался (рис. 5)

```
[guest@ivmurashov ~]$ gcc simplified.c -o simplified  
[guest@ivmurashov ~]$ ls  
Desktop  dir1  Documents  Downloads  Music  Pictures  Public  simplified  simplified.c  Templates  test  Videos
```

Рис. 5: Компиляция файла

Запускаю исполняемый файл. В выводе файла выписаны номера пользователя и групп, от вывода при вводе `if`, они отличаются только тем, что информации меньше (рис. 6)

```
[guest@ivmurashov ~]$ ./simplified
uid=1003, gid=1001
[guest@ivmurashov ~]$ id
uid=1003(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

**Рис. 6:** Сравнение команд

Создание, запись в файл и компиляция файла `simplified2.c`. Запуск программы (рис. 7)

```
[guest@ivmurashov ~]$ touch simplified2.c
[guest@ivmurashov ~]$ nano simplified2.c
[guest@ivmurashov ~]$ gcc simplified2.c -o simplified2
[guest@ivmurashov ~]$ ./simplified2
e_uid=1003, e_gid=1001
real_uid=1003, real_gid=1001
```

**Рис. 7:** Создание и компиляция файла

```
C++ Листинг 2 #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; }
```

(рис. 8)

## Выполнение лабораторной работы

```
GNU nano 5.6.1
#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;
}
```

Рис. 8: Содержимое файла

С помощью `chown` изменяю владельца файла на суперпользователя, с помощью `chmod` изменяю права доступа (рис. 9)

```
[ivmurashov@ivmurashov ~]$ sudo chown root:guest /home/guest/simplified2
[sudo] password for ivmurashov:
[ivmurashov@ivmurashov ~]$ sudo chmod u+s /home/guest/simplified2
chmod: cannot access '/home/guest/simplified2': No such file or directory
[ivmurashov@ivmurashov ~]$ sudo chmod u+s /home/guest/simplified2
[ivmurashov@ivmurashov ~]$ sudo ls -l /home/guest/simplified2
-rwsr-xr-x. 1 root guest 17656 Apr 19 02:43 /home/guest/simplified2
[ivmurashov@ivmurashov ~]$
```

**Рис. 9:** Смена владельца файла и прав доступа к файлу

Сравнение вывода программы и команды `id`, наша команда снова вывела только ограниченное количество информации(рис. 10)

```
[ivmurashov@ivmurashov ~]$ sudo /home/guest/simplied2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[ivmurashov@ivmurashov ~]$ id
uid=1000(ivmurashov) gid=1000(ivmurashov) groups=1000(ivmurashov),10(wheel) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[ivmurashov@ivmurashov ~]$ sudo id
uid=0(root) gid=0(root) groups=0(root) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

**Рис. 10:** Запуск файла

## Создание и компиляция файла readfile.c (рис. 11)

```
guest@ivmurashov ~]$ touch readfile.c
guest@ivmurashov ~]$ nano readfile.c
guest@ivmurashov ~]$ gcc readfile.c -o readfile
guest@ivmurashov ~]$ ls
Desktop dir1 Documents Downloads Music Pictures Public readfile readfile.c simpled simpled2 simpled2.c simpled.c Templates test Videos
guest@ivmurashov ~]$
```

**Рис. 11:** Создание и компиляция файла



```
С++ Листинг 3 #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; }
```

# Выполнение лабораторной работы

(рис. 12)

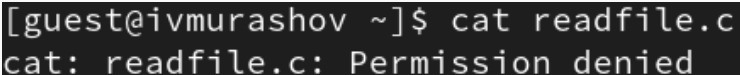
```
GNU nano 5.6.1
#include <fcntl.h>
#include <stdio.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <unistd.h>
int
main (int argc, char* argv[])
{
    unsigned char buffer[16];
    size_t bytes_read;
    int i;
    int fd = open (argv[1], O_RDONLY);
    do
    {
        bytes_read = read (fd, buffer, sizeof (buffer));
        for (i = 0; i < bytes_read; ++i) printf("%c", buffer[i]);
    }
    while (bytes_read == sizeof (buffer));
    close (fd);
```

Снова от имени суперпользователя меняю владельца файла readfile. Далее меняю права доступа так, чтобы пользователь guest не смог прочесть содержимое файла (рис. 13)

```
[ivmurashov@ivmurashov ~]$ sudo chown root:guest /home/guest/readfile.c
[ivmurashov@ivmurashov ~]$ sudo chmod u+s /home/guest/readfile.c
[ivmurashov@ivmurashov ~]$ sudo chmod 700 /home/guest/readfile.c
[ivmurashov@ivmurashov ~]$ sudo chmod -r /home/guest/readfile.c
[ivmurashov@ivmurashov ~]$ sudo chmod u+s /home/guest/readfile.c
```

**Рис. 13:** Смена владельца файла и прав доступа к файлу

Проверка прочесть файл от имени пользователя guest. Прочесть файл не удастся (рис. 14)

A terminal window with a dark background. The prompt is [guest@ivmurashov ~]\$. The command entered is cat readfile.c. The output is cat: readfile.c: Permission denied.

```
[guest@ivmurashov ~]$ cat readfile.c  
cat: readfile.c: Permission denied
```

**Рис. 14:** Попытка прочесть содержимое файла

## Выполнение лабораторной работы

Попытка прочесть тот же файл с помощью программы readfile, в ответ получаем “отказано в доступе” (рис. 15)

[illegible]

**Рис. 15:** Попытка прочесть содержимое файла программой

## Выполнение лабораторной работы

Попытка прочесть файл `\etc\shadow` с помощью программы, все еще получаем отказ в доступе (рис. 16)

[illegible]

**Рис. 16:** Попытка прочесть содержимое файла программой

Пробуем прочесть эти же файлы от имени суперпользователя и чтение файлов проходит успешно (рис. 17)

```
[ivmurashov@ivmurashov ~]$ sudo /home/guest/readfile /etc/shadow
root:$6$60qMQX5xSjGTsc10$h1mujm5M/JnFwD7g0JtVw8u6vRwfId4wglieq0b4zFDwm/8VtpsW0/SoNlala.NTmqTHY0Rx0TwZLC290gE2U.:0:99999:7:::
bin:*:19820:0:99999:7:::
daemon:*:19820:0:99999:7:::
adm:*:19820:0:99999:7:::
```

**Рис. 17:** Чтение файла от имени суперпользователя

Проверяем папку tmp на наличие атрибута Sticky, т.к. в выводе есть буква t, то атрибут установлен (рис. 18)

```
[ivmurashov@ivmurashov ~]$ ls -l / | grep tmp  
drwxrwxrwt. 34 root root 4096 Apr 19 02:55 tmp
```

**Рис. 18:** Проверка атрибутов директории tmp



От имени пользователя guest создаю файл с текстом, добавляю права на чтение и запись для других пользователей (рис. 19)

```
[guest@ivmurashov ~]$ echo "test" > /tmp/file01.txt
[guest@ivmurashov ~]$ ls -l /tmp/file01.txt
-rw-r--r--. 1 guest guest 5 Apr 19 02:56 /tmp/file01.txt
[guest@ivmurashov ~]$ chmod o+rw /tmp/file01.txt
[guest@ivmurashov ~]$ ls -l /tmp/file01.txt
-rw-r--rw-. 1 guest guest 5 Apr 19 02:56 /tmp/file01.txt
```

**Рис. 19:** Создание файла, изменение прав доступа

## Выполнение лабораторной работы

Вхожу в систему от имени пользователя guest2, от его имени могу прочитать файл file01.txt, но перезаписать информацию в нем не могу (рис. 20)

```
[ivmurashov@ivmurashov ~]$ su guest2
Password:
[guest2@ivmurashov ivmurashov]$ cd
[guest2@ivmurashov ~]$ cat /tmp/file01.txt
test
[guest2@ivmurashov ~]$ echo "test2" >> /tmp/file01.txt
bash: /tmp/file01.txt: Permission denied
[guest2@ivmurashov ~]$ cat /tmp/file01.txt
test
[guest2@ivmurashov ~]$
```

**Рис. 20:** Попытка чтения файла

Также невозможно добавить в файл file01.txt новую информацию от имени пользователя guest2 (рис. 21)

```
[guest2@ivmurashov ~]$ echo "test3" > /tmp/file01.txt
bash: /tmp/file01.txt: Permission denied
[guest2@ivmurashov ~]$ cat /tmp/file01.txt
test
[guest2@ivmurashov ~]$
```

**Рис. 21:** Попытка записи в файл

Далее пробуем удалить файл, снова получаем отказ (рис. 22)

```
[guest2@ivmurashov ~]$ rm /tmp/file01.txt  
rm: remove write-protected regular file '/tmp/file01.txt'? y  
rm: cannot remove '/tmp/file01.txt': Operation not permitted
```

**Рис. 22:** Попытка удалить файл

От имени суперпользователя снимаем с директории атрибут Sticky (рис. 23)

```
[guest2@ivmurashov ~]$ su -  
Password:  
[root@ivmurashov]~# chmod -t /tmp  
[root@ivmurashov]~# exit
```

**Рис. 23:** Смена атрибутов файла

Проверяем, что атрибут действительно снят (рис. 24)

```
[guest2@ivmurashov ~]$ ls -l / | grep tmp  
drwxrwxrwx. 36 root root 4096 Apr 19 03:01 tmp
```

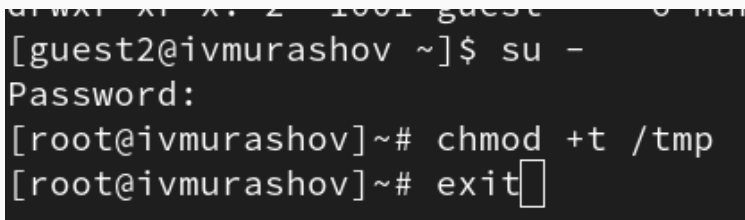
**Рис. 24:** Проверка атрибутов директории

## Выполнение лабораторной работы

Далее был выполнен повтор предыдущих действий. По результатам без Sticky-бита запись в файл и дозапись в файл осталась невозможной, зато удаление файла прошло успешно (рис. 25)

```
[guest2@ivmurashov ~]$ cat /tmp/file01.txt
test
[guest2@ivmurashov ~]$ echo "test2" >> /tmp/file01.txt
bash: /tmp/file01.txt: Permission denied
[guest2@ivmurashov ~]$ cat /tmp/file01.txt
test
[guest2@ivmurashov ~]$ echo "test3" > /tmp/file01.txt
bash: /tmp/file01.txt: Permission denied
[guest2@ivmurashov ~]$ cat /tmp/file01.txt
test
[guest2@ivmurashov ~]$ rm /tmp/file01.txt
rm: remove write-protected regular file '/tmp/file01.txt'? y
[guest2@ivmurashov ~]$ ls -l / | grep tmp
drwxrwxrwx. 36 root root 4096 Apr 19 03:02 tmp
[guest2@ivmurashov ~]$ ls -l
total 0
[guest2@ivmurashov ~]$ ls -l /home/guest
total 76
drwxr-xr-x. 2 1001 guest      6 Mar  7 12:53 Desktop
drwxr-xr-x. 2 guest guest    19 Apr  5 14:52 dir1
drwxr-xr-x. 2 1001 guest      6 Mar  7 12:53 Documents
drwxr-xr-x. 2 1001 guest      6 Mar  7 12:53 Downloads
drwxr-xr-x. 2 1001 guest      6 Mar  7 12:53 Music
drwxr-xr-x. 2 1001 guest      6 Mar  7 15:19 Pictures
drwxr-xr-x. 2 1001 guest      6 Mar  7 12:53 Public
```

Возвращение директории tmp атрибута t от имени суперпользователя (рис. 26)



```
guest2@ivmurashov ~$ su -  
Password:  
[root@ivmurashov]~# chmod +t /tmp  
[root@ivmurashov]~# exit
```

**Рис. 26:** Изменение атрибутов



## **Выводы**

---

Я изучил механизм изменения идентификаторов, применила SetUID- и Sticky-биты. Получил практические навыки работы в кон- соли с дополнительными атрибутами. Рассмотрел работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.