

## Лабораторная работа №1

Разважный Георгий НПИбд-02-19

## 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.

## Выполнение лабораторной работы

1. Вошел в систему от имени пользователя guest и создал программу simpleid.c (рис. 1).

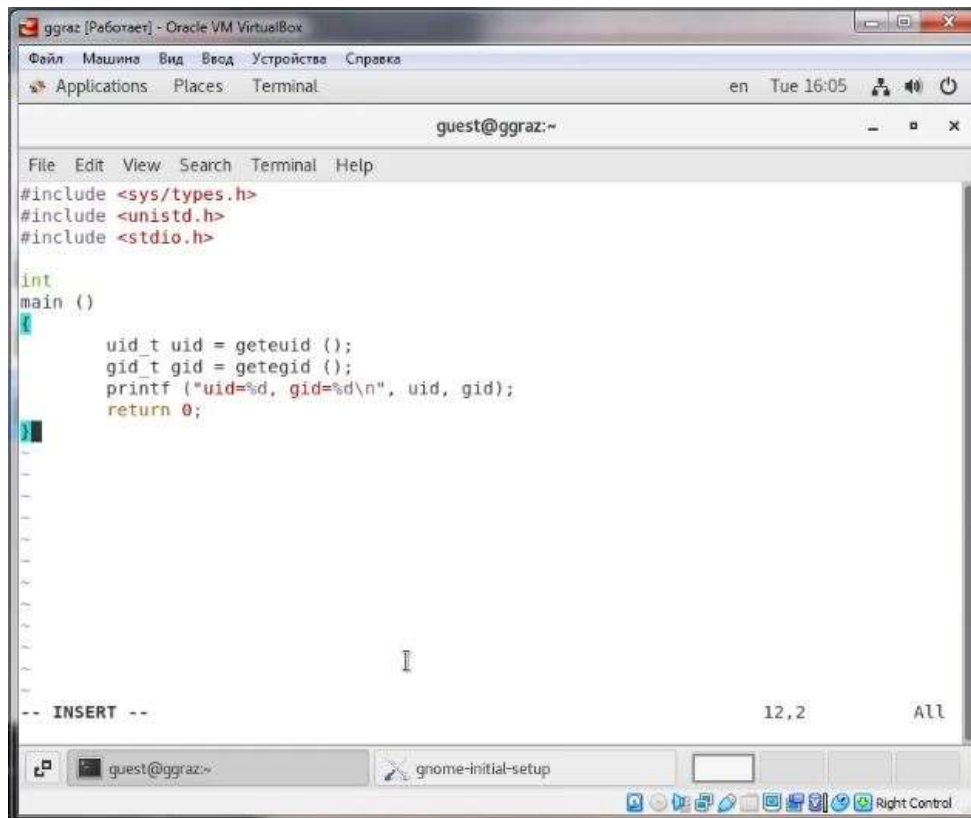
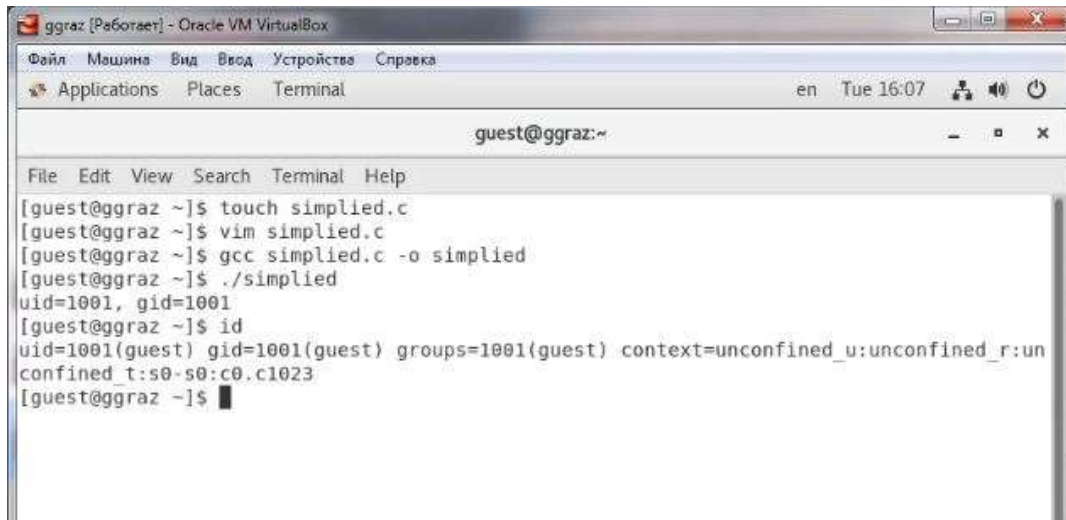


Рис.1

2. Скомпилировал программу и убедился, что файл программы создан. Выполнил программу `simpleid`. Выполнил системную программу `id`. В отличие от команды `id`, моя программа не выводит контекст и все группы, в которые пользователь (рис. 2).

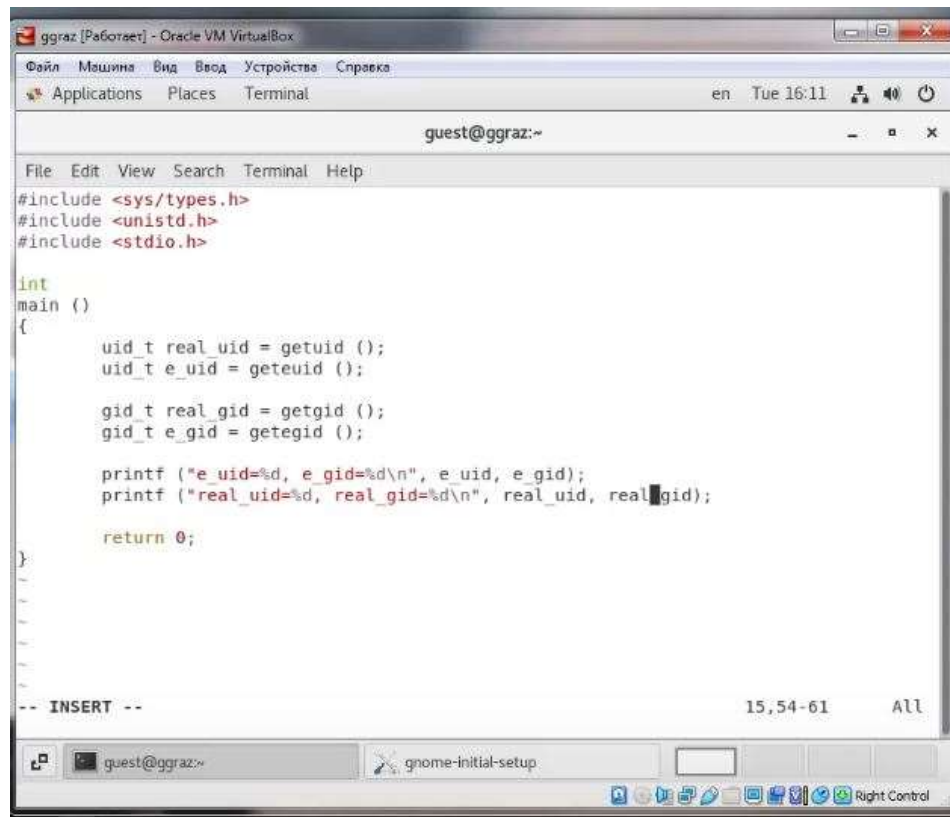


```
ggraz [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка
Applications  Places  Terminal
en  Tue 16:07
guest@ggraz:~

File Edit View Search Terminal Help
[guest@ggraz ~]$ touch simplified.c
[guest@ggraz ~]$ vim simplified.c
[guest@ggraz ~]$ gcc simplified.c -o simplified
[guest@ggraz ~]$ ./simplified
uid=1001, gid=1001
[guest@ggraz ~]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@ggraz ~]$
```

Рис.2

3. Усложнил программу, добавив вывод действительных идентификаторов (рис. 3).



The screenshot shows a VirtualBox window titled "ggraz [Работает] - Oracle VM VirtualBox". Inside, there is a terminal window titled "guest@ggraz:~" with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal displays a C program that includes `<sys/types.h>`, `<unistd.h>`, and `<stdio.h>`. The `main` function calls `getuid`, `geteuid`, `getgid`, and `getegid` to retrieve user and group IDs, then prints them using `printf`. The status bar at the bottom of the terminal shows "-- INSERT --", "15,54-61", and "All".

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

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

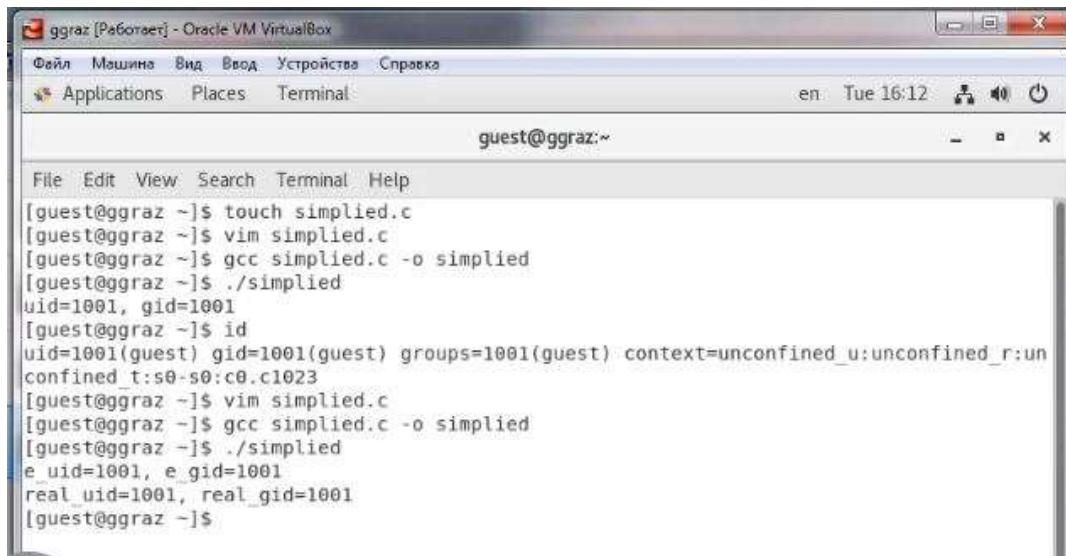
    gid_t real_gid = getgid ();
    gid_t e_gid = getegid ();

    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.3

4. Получившуюся программу назвал simpleid2.c. Скомпилировал и запустил simpleid2.c (рис. 4).

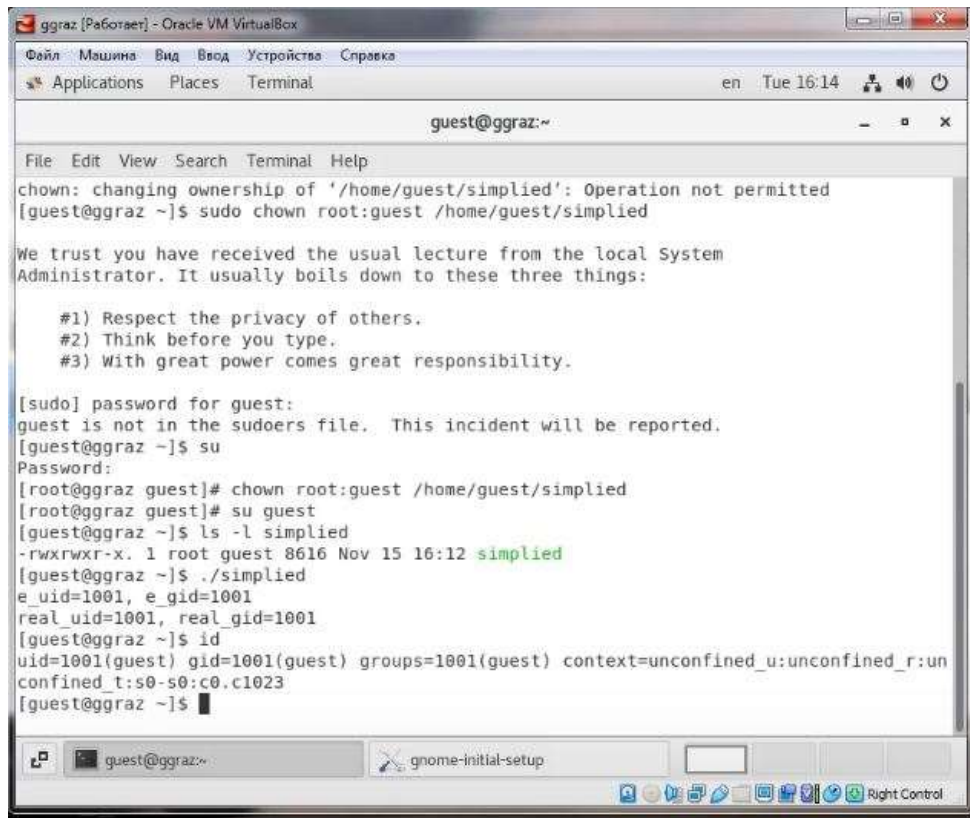


```
ggraz [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка
Applications  Places  Terminal
en  Tue 16:12
guest@ggraz:~

File Edit View Search Terminal Help
[guest@ggraz ~]$ touch simplified.c
[guest@ggraz ~]$ vim simplified.c
[guest@ggraz ~]$ gcc simplified.c -o simplified
[guest@ggraz ~]$ ./simplified
uid=1001, gid=1001
[guest@ggraz ~]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@ggraz ~]$ vim simplified.c
[guest@ggraz ~]$ gcc simplified.c -o simplified
[guest@ggraz ~]$ ./simplified
e_uid=1001, e_gid=1001
real_uid=1001, real_gid=1001
[guest@ggraz ~]$
```

Рис.4

5. От имени суперпользователя выполнил команды `chown root:guest /home/guest/simpleid2` и `chmod u+s /home/guest/simpleid2`. Первая команда меняет владельца файла `simpleid2` на группу `guest`. Вторая команда меняет права доступа к файлу `simpleid2` для пользователя и установленные атрибуты SUID или SGID позволяют запускать файл на выполнение с правами владельца файла или группы соответственно. Выполнил проверку правильности установки новых атрибутов и смены владельца файла `simpleid2`. Запустил `simpleid2` и `id`. Сравнил результаты (рис. 5).



```
ggraz [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройство  Справка
Applications  Places  Terminal
en  Tue 16:14

guest@ggraz:~

File  Edit  View  Search  Terminal  Help

chown: changing ownership of '/home/guest/simplified': Operation not permitted
[guest@ggraz ~]$ sudo chown root:guest /home/guest/simplified

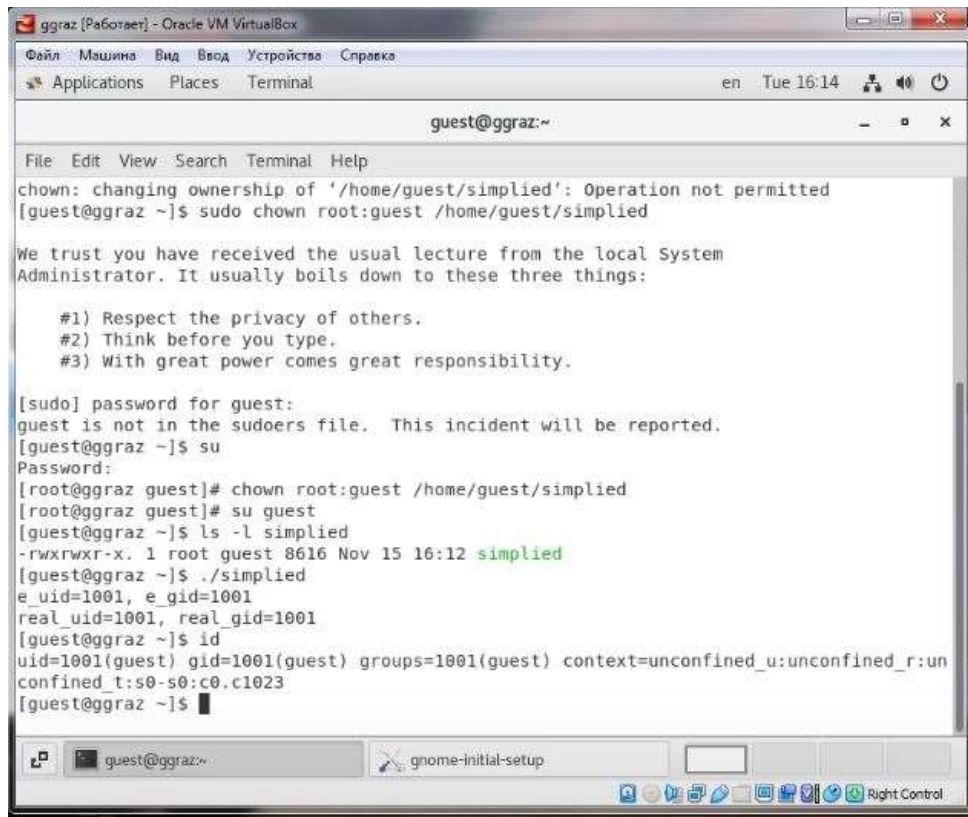
We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

#1) Respect the privacy of others.
#2) Think before you type.
#3) With great power comes great responsibility.

[sudo] password for guest:
guest is not in the sudoers file. This incident will be reported.
[guest@ggraz ~]$ su
Password:
[root@ggraz guest]# chown root:guest /home/guest/simplified
[root@ggraz guest]# su guest
[guest@ggraz ~]$ ls -l simplified
-rwxrwxr-x. 1 root guest 8616 Nov 15 16:12 simplified
[guest@ggraz ~]$ ./simplified
e_uid=1001, e_gid=1001
real_uid=1001, real_gid=1001
[guest@ggraz ~]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@ggraz ~]$
```

Рис. 5.

6. Проделал тоже самое относительно SetGID-бита (рис. 6).



```
ggraz [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка
Applications  Places  Terminal
en  Tue 16:14

guest@ggraz:~

File Edit View Search Terminal Help
chown: changing ownership of '/home/guest/simplified': Operation not permitted
[guest@ggraz ~]$ sudo chown root:guest /home/guest/simplified

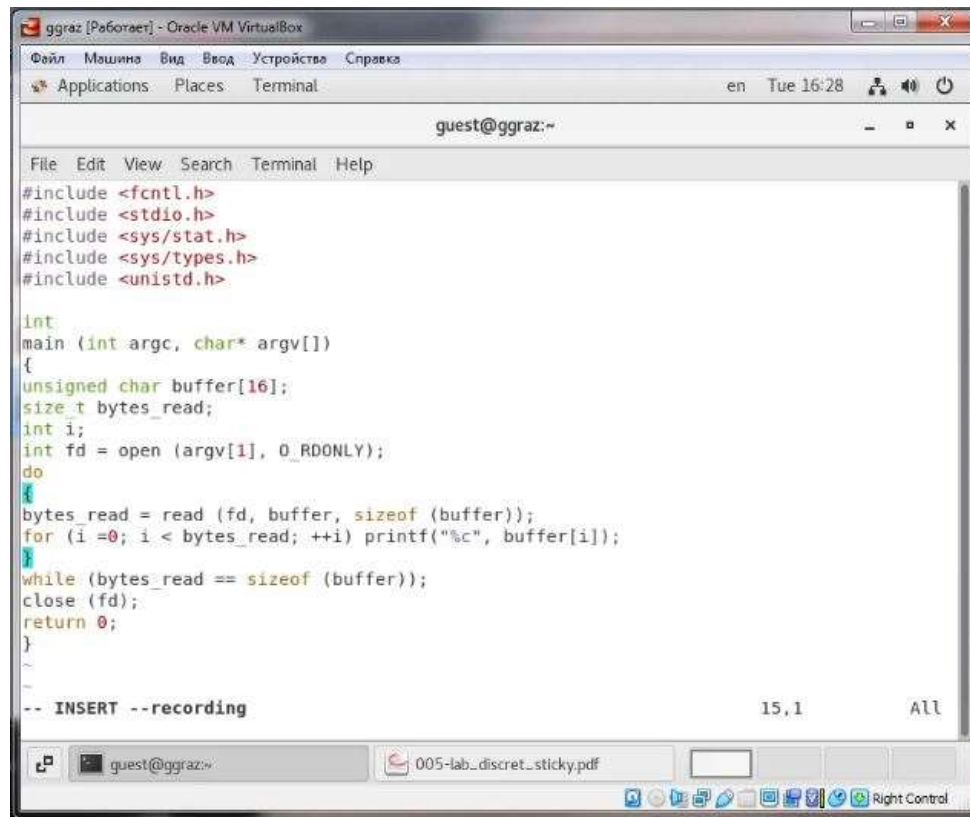
We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for guest:
guest is not in the sudoers file. This incident will be reported.
[guest@ggraz ~]$ su
Password:
[root@ggraz guest]# chown root:guest /home/guest/simplified
[root@ggraz guest]# su guest
[guest@ggraz ~]$ ls -l simplified
-rwxrwxr-x. 1 root guest 8616 Nov 15 16:12 simplified
[guest@ggraz ~]$ ./simplified
e_uid=1001, e_gid=1001
real_uid=1001, real_gid=1001
[guest@ggraz ~]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@ggraz ~]$
```

Рис. 6.

7. Создал программу readfile.c (рис. 7).



The screenshot shows a terminal window titled "ggraz [Работаer] - Oracle VM VirtualBox". The window has a menu bar with "Файл", "Машина", "Вид", "Ввод", "Устройство", and "Справка". Below the menu bar is a toolbar with "Applications", "Places", and "Terminal". The terminal prompt is "guest@ggraz:~". The code displayed is as follows:

```
#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;
}

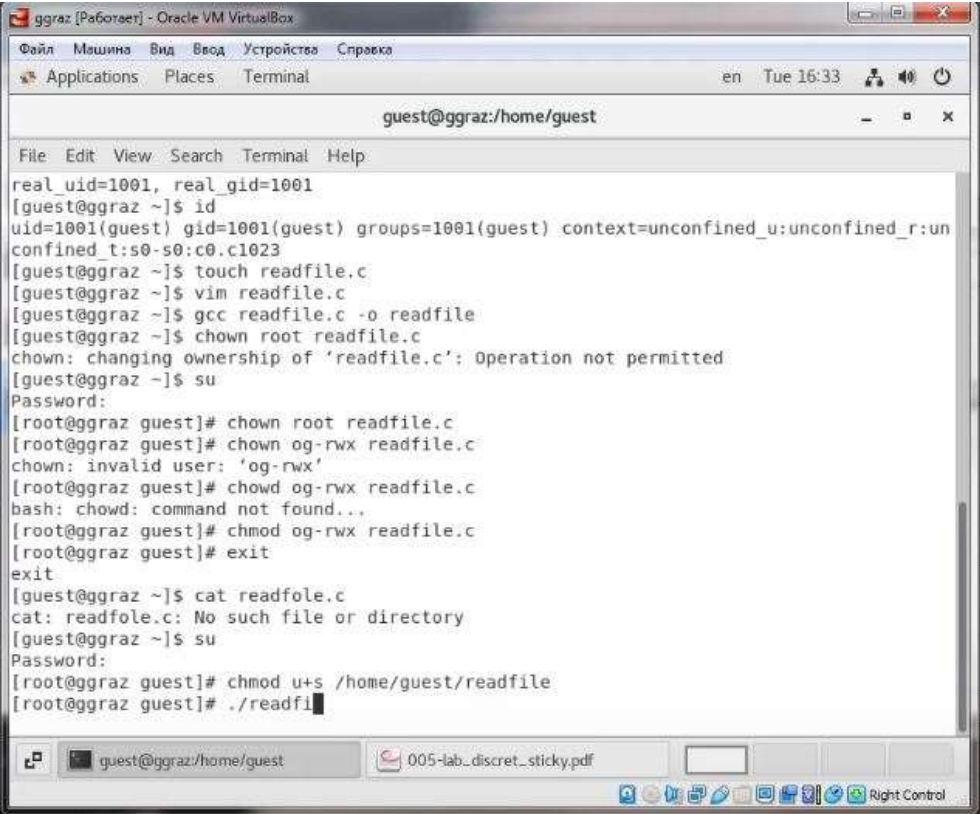
~
-- INSERT --recording 15,1 All
```

The terminal window also shows a taskbar at the bottom with icons for the terminal, a PDF file named "005-lab\_discret\_sticky.pdf", and a "Right Control" button.

Рис. 7.



8. Откомпилировал программу. Сменил владельца у файла readfile.c и изменил права так, чтобы только суперпользователь (root) мог прочитать его, а guest не мог. Проверил, что пользователь guest не может прочитать файл readfile.c. Сменил у программы readfile владельца и установила SetU'D-бит. Проверил, может ли программа readfile прочитать файл readfile.c (рис. 8).



```
ggraz [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка
Applications  Places  Terminal
en  Tue 16:33

guest@ggraz:/home/guest

File Edit View Search Terminal Help
real_uid=1001, real_gid=1001
[guest@ggraz ~]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@ggraz ~]$ touch readfile.c
[guest@ggraz ~]$ vim readfile.c
[guest@ggraz ~]$ gcc readfile.c -o readfile
[guest@ggraz ~]$ chown root readfile.c
chown: changing ownership of 'readfile.c': Operation not permitted
[guest@ggraz ~]$ su
Password:
[root@ggraz guest]# chown root readfile.c
[root@ggraz guest]# chown og-rwx readfile.c
chown: invalid user: 'og-rwx'
[root@ggraz guest]# chowd og-rwx readfile.c
bash: chowd: command not found...
[root@ggraz guest]# chmod og-rwx readfile.c
[root@ggraz guest]# exit
exit
[guest@ggraz ~]$ cat readfole.c
cat: readfole.c: No such file or directory
[guest@ggraz ~]$ su
Password:
[root@ggraz guest]# chmod u+s /home/guest/readfile
[root@ggraz guest]# ./readfi
```

Рис. 8.



10. От пользователя guest2 попробовал прочитать файл /tmp/file01.txt. От пользователя guest2 попробовал дозаписать в файл /tmp/file01.txt слово test2. Удалось выполнить операцию. Проверил содержимое файла. От пользователя guest2 попробовал записать в файл /tmp/file01.txt слово test3, стерев при этом всю имеющуюся в файле информацию. Удалось выполнить операцию. Проверил содержимое файла. От пользователя guest2 попробовал удалить файл /tmp/file01.tx. Не удалось выполнить операцию. Повысил свои права до суперпользователя и выполнила после этого команду, снимающую атрибут t (Sticky-бит) с директории /tmp. Покинул режим суперпользователя. От пользователя guest2 проверила, что атрибута t у директории /tmp нет. Повторил предыдущие шаги. Удалось успешно выполнить каждый шаг. Повысил свои права до суперпользователя и вернул атрибут t на директорию /tmp (рис. 10).

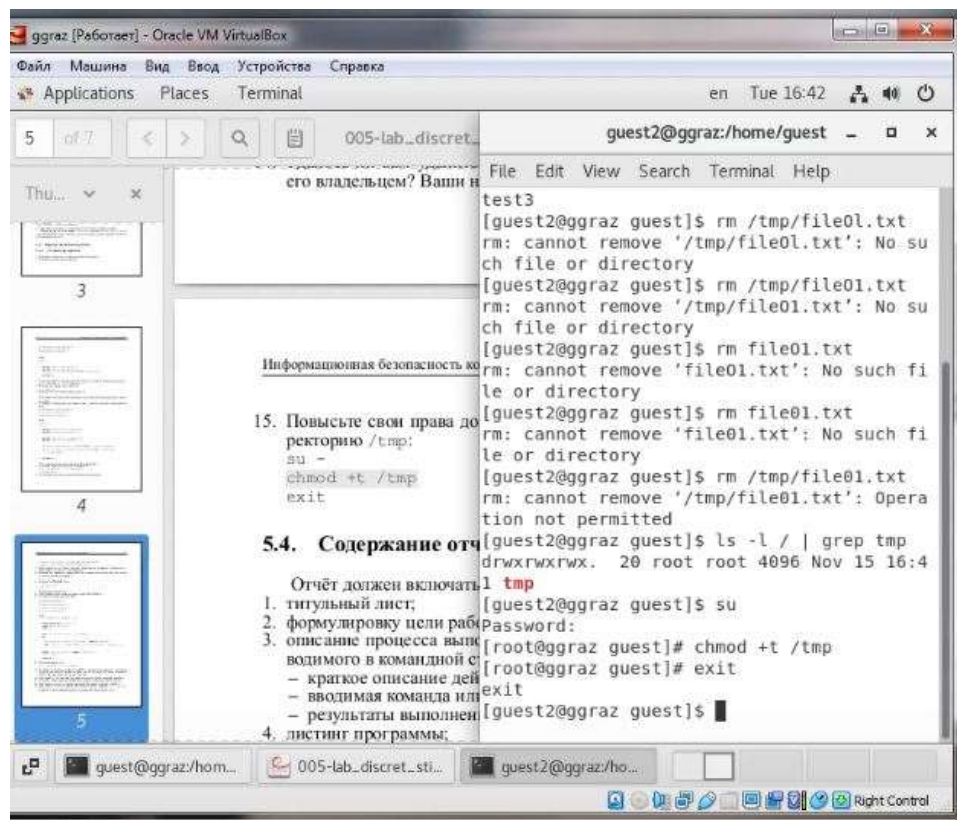


Рис. 10.

# Выводы

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