

# **Лабораторная работа №5**

**Информационная безопасность**

НВЕ МАНГЕ ХОСЕ ХЕРСОН МИКО; НКАбд-03-22

# Содержание

|          |                                       |           |
|----------|---------------------------------------|-----------|
| <b>1</b> | <b>Цель работы</b>                    | <b>5</b>  |
| <b>2</b> | <b>Теоретическое введение</b>         | <b>6</b>  |
| <b>3</b> | <b>Подготовка к выполнению работы</b> | <b>7</b>  |
| <b>4</b> | <b>Выполнение лабораторной работы</b> | <b>8</b>  |
| 4.1      | Создание программы . . . . .          | 8         |
| 4.2      | Исследование Sticky-бита . . . . .    | 12        |
| <b>5</b> | <b>Выводы</b>                         | <b>15</b> |

# Список иллюстраций

|      |  |    |
|------|--|----|
| 3.1  | Проверка установки ПО . . . . .                  | 7  |
| 3.2  | setenforce 0 . . . . .                           | 7  |
| 4.1  | Вход в систему от другого пользователя . . . . . | 8  |
| 4.2  | Создание программы . . . . .                     | 8  |
| 4.3  | Заполнение элементарной программы . . . . .      | 9  |
| 4.4  | Компиляция и запуск программы . . . . .          | 9  |
| 4.5  | Команда id . . . . .                             | 9  |
| 4.6  | Заполнение программы . . . . .                   | 10 |
| 4.7  | Компиляция и запуск программы . . . . .          | 10 |
| 4.8  | Поменяла владельца программы . . . . .           | 10 |
| 4.9  | ls -l . . . . .                                  | 11 |
| 4.10 | Сравнение результатов . . . . .                  | 11 |
| 4.11 | Создание программы . . . . .                     | 11 |
| 4.12 | Компиляция программы . . . . .                   | 12 |
| 4.13 | Смена владельца . . . . .                        | 12 |
| 4.14 | Результат работы программы . . . . .             | 12 |
| 4.15 | Выполнение задач . . . . .                       | 13 |
| 4.16 | Работа с файлами . . . . .                       | 13 |
| 4.17 | Работа с атрибутом t . . . . .                   | 14 |
| 4.18 | Возвращение Sticky . . . . .                     | 14 |

## Список таблиц

# 1 Цель работы

Целью 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.

## 2 Теоретическое введение

Setuid, Setgid и Sticky Bit - это специальные типы разрешений позволяют задавать расширенные права доступа на файлы или каталоги.

### 3 Подготовка к выполнению работы

Я проверила, установлен ли у меня gcc командой **yum install gcc**. Он установлен и обновлен до последней версии (рис. 3.1).

Проверка установки ПО

Рис. 3.1: Проверка установки ПО

Помимо этого, я отключила систему запретов до очередной перезагрузки системы командой **setenforce 0**. После этого команда **getenforce** выводит Permissive (рис. 3.2).

```
jose@jose-IdeaPad-3-15ITL6:~$ setenforce 0
setenforce: SELinux is disabled
jose@jose-IdeaPad-3-15ITL6:~$ sudo su
[sudo] contraseña para jose:
root@jose-IdeaPad-3-15ITL6:/home/jose# setenforce 0
Orden «setenforce» no encontrada. Quizá quiso decir:
  la orden «setenforce» del paquete deb «selinux-utils (3.5-1)»
Pruebe con: apt install <nombre del paquete deb>
root@jose-IdeaPad-3-15ITL6:/home/jose# getenforce
Disabled
root@jose-IdeaPad-3-15ITL6:/home/jose# █
```

Рис. 3.2: setenforce 0

## 4 Выполнение лабораторной работы

### 4.1 Создание программы

Я вошла в систему от имени пользователя guest (рис. 4.1).

```
jose@jose-IdeaPad-3-15ITL6:~$ su - guest
Contraseña:
```

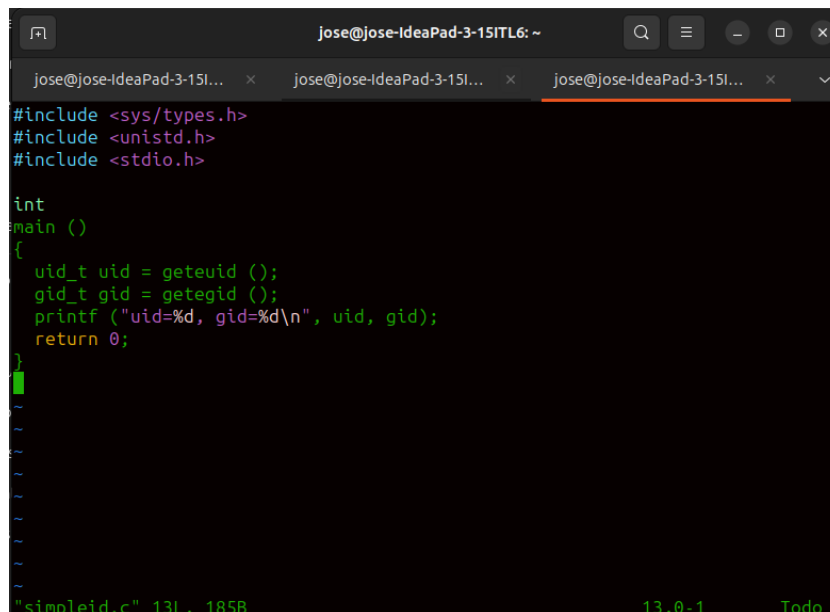
Рис. 4.1: Вход в систему от другого пользователя

Далее создала программу simpleid.c и заполнила ее (рис. 4.2), (рис. 4.3).

```
jose@jose-IdeaPad-3-15ITL6:~$ pwd
/home/jose
jose@jose-IdeaPad-3-15ITL6:~$ touch simpleid.c
jose@jose-IdeaPad-3-15ITL6:~$ ls
'ARCHIVOS ANTIGUOS'      simpleid3.c
config.yaml              simpleid.c
Descargas                 snap
directorio_nuevo         tl-errmess.vbs
Documentos               tlmgrgui.pl
Escritorio                tlmgr.pl
gcc                       'trabajo 1 de Cisco.pkt'
Imágenes                 uninstall-windows.pl
install-tl                uninstq.vbs
install-tl-20240309      Untitled10.ipynb
```

Рис. 4.2: Создание программы



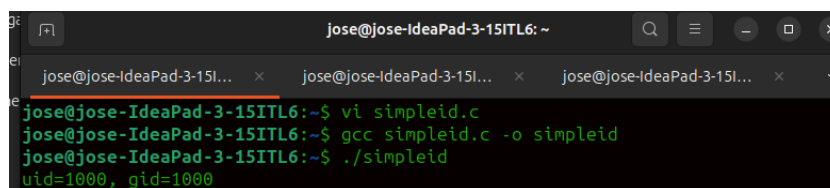
A terminal window titled 'jose@jose-IdeaPad-3-15ITL6: ~' with three tabs. The active tab shows the code for 'simpleid.c'. The code includes headers for types, unistd, and stdio. The main function calls getuid and getgid, then prints the values and returns 0. The status bar at the bottom shows '"simpleid.c" 13L, 185B', '13,0-1', and 'Todo'.

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

int
main ()
{
    uid_t uid = getuid ();
    gid_t gid = getgid ();
    printf ("uid=%d, gid=%d\n", uid, gid);
    return 0;
}
```

Рис. 4.3: Заполнение элементарной программы

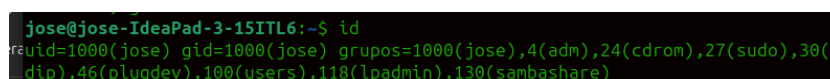
Скомпилировала файл через **gcc simpleid.c -o simpleid** и выполнила программу simpleid (рис. 4.4).

A terminal window titled 'jose@jose-IdeaPad-3-15ITL6: ~' with three tabs. The active tab shows the commands to compile and run the program. The output shows the user ID and group ID as 1000.

```
jose@jose-IdeaPad-3-15ITL6:~$ vi simpleid.c
jose@jose-IdeaPad-3-15ITL6:~$ gcc simpleid.c -o simpleid
jose@jose-IdeaPad-3-15ITL6:~$ ./simpleid
uid=1000, gid=1000
```

Рис. 4.4: Компиляция и запуск программы

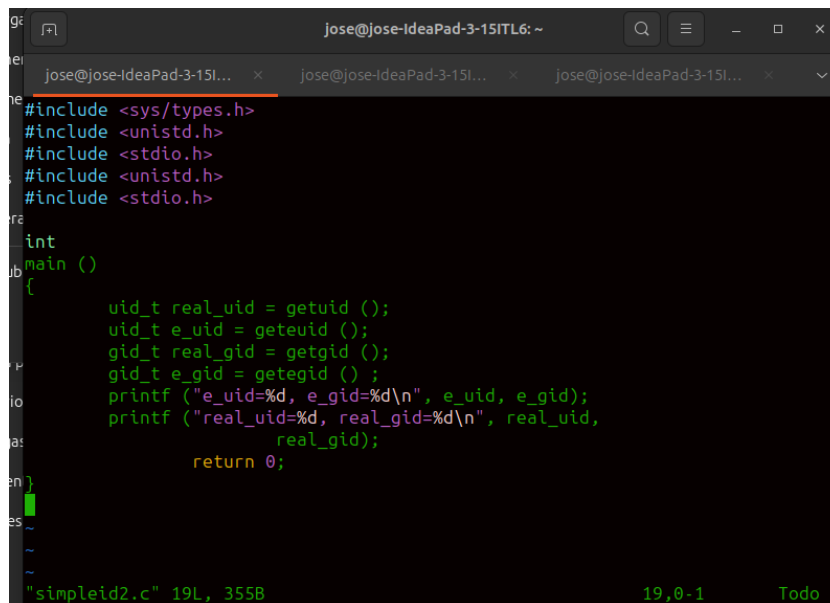
Выполнила системную программу id. Результаты похожи. Gid и uid одинаковые, однако команда id дает больше информации (рис. 4.5).

A terminal window titled 'jose@jose-IdeaPad-3-15ITL6: ~' with one tab. The active tab shows the command 'id' and its output, which lists the user, group, and supplementary groups with their IDs.

```
jose@jose-IdeaPad-3-15ITL6:~$ id
uid=1000(jose) gid=1000(jose) grupos=1000(jose),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),100(users),118(lpadmin),130(sambashare)
```

Рис. 4.5: Команда id

Усложнила программу, добавив вывод действительных идентификаторов (рис. 4.6).

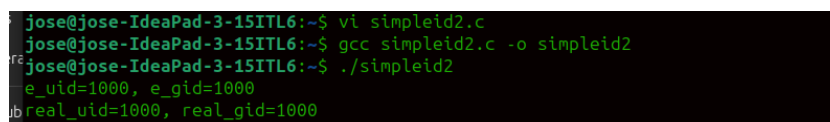


```
#include <sys/types.h>
#include <unistd.h>
#include <stdio.h>
#include <unistd.h>
#include <stdio.h>

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. 4.6: Заполнение программы

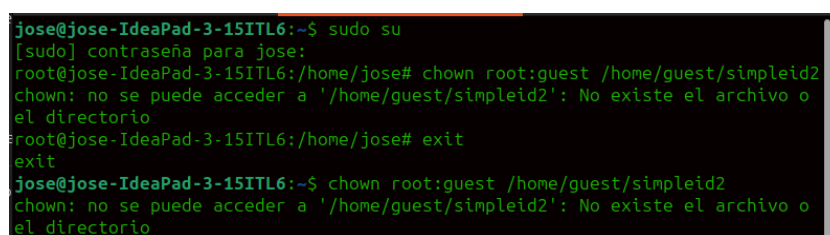
Скомпилировала и запустила simpleid2.c (рис. 4.7).



```
jose@jose-IdeaPad-3-15ITL6:~$ vi simpleid2.c
jose@jose-IdeaPad-3-15ITL6:~$ gcc simpleid2.c -o simpleid2
jose@jose-IdeaPad-3-15ITL6:~$ ./simpleid2
e_uid=1000, e_gid=1000
real_uid=1000, real_gid=1000
```

Рис. 4.7: Компиляция и запуск программы

От имени суперпользователя выполнила команды **chown root:guest /home/guest/simpleid2**, **chmod u+s /home/guest/simpleid2** (рис. 4.8).



```
jose@jose-IdeaPad-3-15ITL6:~$ sudo su
[sudo] contraseña para jose:
root@jose-IdeaPad-3-15ITL6:/home/jose# chown root:guest /home/guest/simpleid2
chown: no se puede acceder a '/home/guest/simpleid2': No existe el archivo o
el directorio
root@jose-IdeaPad-3-15ITL6:/home/jose# exit
exit
jose@jose-IdeaPad-3-15ITL6:~$ chown root:guest /home/guest/simpleid2
chown: no se puede acceder a '/home/guest/simpleid2': No existe el archivo o
el directorio
```

Рис. 4.8: Поменяла владельца программы

Выполнила проверку правильности установки новых атрибутов и смены владельца файла simpleid2 (рис. 4.9).

```
jose@jose-IdeaPad-3-15ITL6:~$ sudo su
root@jose-IdeaPad-3-15ITL6:/home/jose# ls -li simpleid2
11282144 simpleid2
root@jose-IdeaPad-3-15ITL6:/home/jose# ./simpleid2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
```

Рис. 4.9: ls -li

Запустила simpleid2 и id. Результаты похожи. Gid и uid одинаковые, однако команда id дает больше информации (рис. 4.10).

```
root@jose-IdeaPad-3-15ITL6:/home/jose# id
uid=0(root) gid=0(root) grupos=0(root)
```

Рис. 4.10: Сравнение результатов

Создала программу readfile.c (рис. 4.11).

```
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <unistd.h>
#include <stdio.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. 4.11: Создание программы

Откомпилировала её (рис. 4.12).

```

#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <unistd.h>
#include <stdio.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. 4.12: Компиляция программы

Сменила владельца у файла readfile.c и изменила права так, чтобы только суперпользователь (root) мог прочитать его, а aleksandrovaux не мог. Проверила, может ли пользователь прочитать файл readfile.c. Не может (рис. 4.13).

```

jose@jose-IdeaPad-3-15ITL6:~$ cat /home/guest/readfile.c
cat: /home/guest/readfile.c: No existe el archivo o el directorio
jose@jose-IdeaPad-3-15ITL6:~$

```

Рис. 4.13: Смена владельца

Программа readfile в целом может прочитать файл /etc/shadow (рис. 4.14).

```

t@p@p@h0 86_64./readfile/etc/shadowSHELL=/bin/bashHISTCONTR
OL=ignoredupsHISTSIZE=1000HOSTNAME=localhostPWD=/home/guestLOGNAME=guestXAUTHORI
TY=/home/guest/.xauthkf0TjdHOME=/home/guestLANG=en_US.UTF-8LS_COLORS=rs=0:di=01
34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35:bd=40;33;01:cd=40;33;01:or=40;31;01
:mi=01;37;41:su=37;41:sg=30;43:ca=30;41:tw=30;42:ow=34;42:st=37;44:ex=01;32:*.ta
r=01;31:*.tgz=01;31:*.arc=01;31:*.arj=01;31:*.taz=01;31:*.lha=01;31:*.lz4=01;31:
*.lzh=01;31:*.lzma=01;31:*.tlz=01;31:*.txz=01;31:*.tzo=01;31:*.t7z=01;31:*.zip=0
1;31:*.zst=01;31:*.dz=01;31:*.gz=01;31:*.lrz=01;31:*.lz=01;31:*.lzo=01;31:*.xz=01;
31:*.zst=01;31:*.bz2=01;31:*.bz=01;31:*.tbz=01;31:*.tbz2=01;31:*.tz

```

Рис. 4.14: Результат работы программы

## 4.2 Исследование Sticky-бита

Выяснила, установлен ли атрибут Sticky на директории /tmp. Установлен. От имени пользователя guest создала файл file01.txt в директории /tmp со сло-

вом test через **echo "test" > /tmp/file01.txt**. Затем просмотрите атрибуты у только что созданного файла и разрешила чтение и запись для категории пользователей «все остальные» при помощи утилиты **chmod o+rw /tmp/file01.txt** (рис. 4.15).

```
root@jose-IdeaPad-3-15ITL6:/home/jose# ls -l / | grep tmp
drwxrwxrwt  22 root root 20480 abr 13 19:50 tmp
root@jose-IdeaPad-3-15ITL6:/home/jose# echo "test" > /tmp/file01.txt
root@jose-IdeaPad-3-15ITL6:/home/jose# ls -l /tmp/file01.txt
-rw-r--r--  1 root root 5 abr 13 19:55 /tmp/file01.txt
root@jose-IdeaPad-3-15ITL6:/home/jose# chmod o+rw /tmp/file01.txt
root@jose-IdeaPad-3-15ITL6:/home/jose# ls -l /tmp/file01.txt
-rw-r--rw-  1 root root 5 abr 13 19:55 /tmp/file01.txt
```

Рис. 4.15: Выполнение задач

От пользователя guest2 (не являющегося владельцем) попробовала прочитать файл /tmp/file01.txt. Попробовала дозаписать в файл /tmp/file01.txt слово test2 командой **echo "test2" > /tmp/file01.txt**. Мне отказано в доступе. То же самое попробовала сделать с test3, но мне снова отказано в доступе. Файл не записался (рис. 4.16).

```
root@jose-IdeaPad-3-15ITL6:/home/jose# echo "test" > /tmp/file01.txt
root@jose-IdeaPad-3-15ITL6:/home/jose# ls -l /tmp/file01.txt
-rw-r--r--  1 root root 5 abr 13 19:55 /tmp/file01.txt
root@jose-IdeaPad-3-15ITL6:/home/jose# chmod o+rw /tmp/file01.txt
root@jose-IdeaPad-3-15ITL6:/home/jose# ls -l /tmp/file01.txt
-rw-r--rw-  1 root root 5 abr 13 19:55 /tmp/file01.txt
root@jose-IdeaPad-3-15ITL6:/home/jose# cat /tmp/file01.txt
test
root@jose-IdeaPad-3-15ITL6:/home/jose# echo "test2" > /tmp/file01.txt
root@jose-IdeaPad-3-15ITL6:/home/jose# cat /tmp/file01.txt
test2
root@jose-IdeaPad-3-15ITL6:/home/jose# echo "test3" > /tmp/file01.txt
root@jose-IdeaPad-3-15ITL6:/home/jose# cat /tmp/file01.txt
test3
root@jose-IdeaPad-3-15ITL6:/home/jose# rm /tmp/file01.txt
rm: no se puede borrar '/tmp/file01.txt': No existe el archivo o el directori
```

Рис. 4.16: Работа с файлами

От пользователя guest2 попробовала удалить файл /tmp/file01.txt. Мне отказали в доступе. Потом я повысила свои права до суперпользователя командой **su -** и сняла атрибут t (Sticky-бит) с директории /tmp. От пользователя guest2 проверила, что атрибута t у директории /tmp нет. Повторила предыдущие шаги. Файл удалился (рис. 4.17).

```
jose@jose-IdeaPad-3-15ITL6:~$ ls /tmp
dbus-7mU2RtIM
dbus-AXjCbRLA
file01.txt
gdm3-config-err-iHWzMu
lu46305p3l.tmp
lu6719kjrj.tmp
OSL_PIPE_1000_SingleOfficeIPC_8fe2e4c27819f62b4f6b37cdf51ee64
snap-private-tmp
systemd-private-6460e344d3f54866a1ba99bd8a88dc34-blutetooth.service-VJ9oyy
systemd-private-6460e344d3f54866a1ba99bd8a88dc34-colord.service-9kvaWo
systemd-private-6460e344d3f54866a1ba99bd8a88dc34-fwupd.service-t54q2K
systemd-private-6460e344d3f54866a1ba99bd8a88dc34-geoclue.service-dGQs6d
systemd-private-6460e344d3f54866a1ba99bd8a88dc34-ModemManager.service-id7Lu4
systemd-private-6460e344d3f54866a1ba99bd8a88dc34-polkit.service-Vc5KfX
systemd-private-6460e344d3f54866a1ba99bd8a88dc34-power-profiles-daemon.service-t0NEr0
systemd-private-6460e344d3f54866a1ba99bd8a88dc34-switcheroo-control.service-MHM7xF
systemd-private-6460e344d3f54866a1ba99bd8a88dc34-systemd-logind.service-5CzErl
systemd-private-6460e344d3f54866a1ba99bd8a88dc34-systemd-sd.service-YRi380
```

Рис. 4.17: Работа с атрибутом t

Вернула атрибут t (рис. 4.18).

```
jose@jose-IdeaPad-3-15ITL6:~$ ls /tmp
root@jose-IdeaPad-3-15ITL6:/home/jose# chmod +t /tmp
root@jose-IdeaPad-3-15ITL6:/home/jose# exit
exit
jose@jose-IdeaPad-3-15ITL6:~$
```

Рис. 4.18: Возвращение Sticky

## 5 Выводы

Я изучила механизмы изменения идентификаторов, применения SetUID- и Sticky-битов. Получила практических навыков работы в консоли с дополнительными атрибутами.