

# **Лабораторная работа №5**

**Основы информационной безопасности**

Колчева Юлия Вячеславовна

# Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	15
5	Список литературы	16

## Список иллюстраций

3.1	Работа с консолью . . . . .	7
3.2	Работа с консолью . . . . .	8
3.3	Работа с консолью . . . . .	8
3.4	Код первой программы . . . . .	8
3.5	Работа с консолью . . . . .	9
3.6	Вторая программа . . . . .	9
3.7	Работа с консолью . . . . .	9
3.8	Работа с консолью . . . . .	10
3.9	Работа с консолью . . . . .	10
3.10	Работа с консолью . . . . .	10
3.11	Работа с консолью . . . . .	10
3.12	Работа с консолью . . . . .	11
3.13	Работа с консолью . . . . .	11
3.14	Работа с консолью . . . . .	11
3.15	Работа с консолью . . . . .	12
3.16	Работа с консолью . . . . .	12
3.17	Работа с консолью . . . . .	12
3.18	Работа с консолью . . . . .	13
3.19	Работа с консолью . . . . .	14
3.20	Работа с консолью . . . . .	14

## **Список таблиц**

# 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 Задание

Часть 1

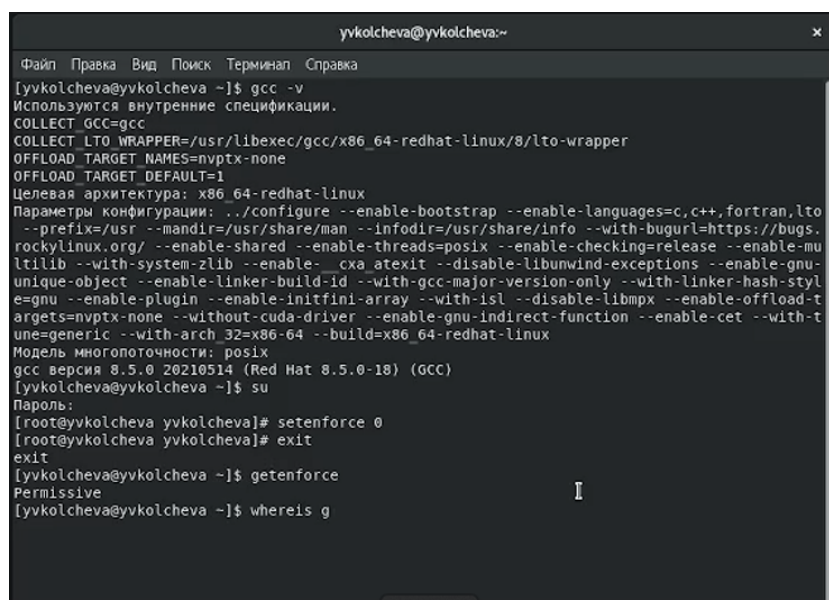
Создание программы

Часть 2

Исследование Sticky-бита

### 3 Выполнение лабораторной работы

Для начала я убедилась, что компилятор gcc установлен, используя команду “gcc -v”. Затем отключила систему запретов до очередной перезагрузки системы командой “sudo setenforce 0”, после чего команда “getenforce” вывела “Permissive”(рис. [3.1])



```
yvkolcheva@yvkolcheva:~  
Файл Правка Вид Поиск Терминал Справка  
[yvkolcheva@yvkolcheva ~]$ gcc -v  
Используются внутренние спецификации.  
COLLECT_GCC=gcc  
COLLECT_LTO_WRAPPER=/usr/libexec/gcc/x86_64-redhat-linux/8/lto-wrapper  
OFFLOAD_TARGET_NAMES=nvptx-none  
OFFLOAD_TARGET_DEFAULT=1  
Целевая архитектура: x86_64-redhat-linux  
Параметры конфигурации: ../configure --enable-bootstrap --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 --enable-mu  
ltilib --with-system-zlib --enable-cxx-exceptions --enable-gnu-unique-object --enable-linker-build-id --with-gcc-major-version-only --with-linker-hash-styl  
e=gnu --enable-plugin --enable-initfini-array --with-isl --disable-lto --enable-offload-t  
argets=nvptx-none --without-cuda-driver --enable-gnu-indirect-function --enable-cet --with-t  
une=generic --with-arch_32=x86-64 --build=x86_64-redhat-linux  
Модель многопоточности: posix  
gcc версия 8.5.0 20210514 (Red Hat 8.5.0-18) (GCC)  
[yvkolcheva@yvkolcheva ~]$ su  
Пароль:  
[root@yvkolcheva yvkolcheva]# setenforce 0  
[root@yvkolcheva yvkolcheva]# exit  
exit  
[yvkolcheva@yvkolcheva ~]$ getenforce  
Permissive  
[yvkolcheva@yvkolcheva ~]$ whereis g
```

Рис. 3.1: Работа с консолью

Проверила успешное выполнение команд “whereis gcc” и “whereis g++” (их рас-  
положение) (рис. [3.2])

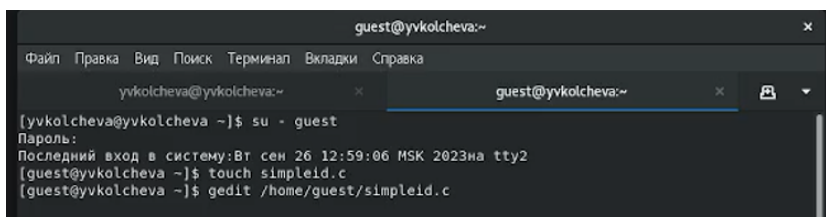
```

[yvkolcheva@yvkolcheva ~]$ whereis gcc
gcc: /usr/bin/gcc /usr/lib/gcc /usr/libexec/gcc /usr/share/man/man1/gcc.1.gz /usr/share/info/gcc.info.gz
[yvkolcheva@yvkolcheva ~]$ whereis g++
g++: /usr/bin/g++ /usr/share/man/man1/g++.1.gz
[yvkolcheva@yvkolcheva ~]$

```

Рис. 3.2: Работа с консолью

Вошла в систему от имени пользователя guest командой “su - guest”. Создала программу simpleid.c командой “touch simpleid.c” и открыла её в редакторе командой “gedit /home/guest/simpleid.c” (рис. [3.3])




```

guest@yvkolcheva:~
Файл Правка Вид Поиск Терминал Вкладки Справка
yvkolcheva@yvkolcheva:~ guest@yvkolcheva:~
[yvkolcheva@yvkolcheva ~]$ su - guest
Пароль:
Последний вход в систему:Вт сен 26 12:59:06 MSK 2023на tty2
[guest@yvkolcheva ~]$ touch simpleid.c
[guest@yvkolcheva ~]$ gedit /home/guest/simpleid.c

```

Рис. 3.3: Работа с консолью

Код программы выглядит следующим образом (рис. [3.4])



```

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. 3.4: Код первой программы

Скомпилировала программу и убедилась, что файл программы был создан командой “gcc simpleid.c -o simpleid”. Выполнила программу simpleid командой “./simpleid”, а затем выполнила системную программу id командой “id”. Результаты, полученные в результате выполнения обеих команд, совпадают (uid=1001 и gid=1001) (рис. [3.5])



```

** (gedit:4873): WARNING **: 14:34:03.626: Set document metadata failed: Setting attribute m
etadata::gedit-position not supported
[guest@yvkolcheva ~]$ gcc simpleid.c -o simpleid
[guest@yvkolcheva ~]$ ./simpleid
uid=1001, gid=1001
[guest@yvkolcheva ~]$

```

Рис. 3.5: Работа с консолью

Усложнила программу, добавив вывод действительных идентификаторов (рис. [3.7])



```

Обзор  Текстовый редактор  Пн, 2 октября 14:44
simpleid2.c
#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. 3.6: Вторая программа

Получившуюся программу назвала simpleid2.c (рис. [3.8])

```

etadata::gedit-position not supported
[guest@yvkolcheva ~]$ rename simpleid.c simpleid2.c /home/guest/simpleid.c
[guest@yvkolcheva ~]$ ls
Desktop  Documents  Music  Public  simpleid2.c  Videos
dir1    Downloads  Pictures  simpleid  Templates
[guest@yvkolcheva ~]$ gcc

```

Рис. 3.7: Работа с консолью

Скомпилировала и запустила simpleid2.c командами “gcc simpleid2.c -o sipleid2” и “./simpleid2” (рис. [3.8])

```
[guest@yvkolcheva ~]$ gcc simpleid2.c -o simpleid2
[guest@yvkolcheva ~]$ ./simpleid2
e uid=1001, e gid=1001
real uid=1001, real gid=1001
[guest@yvkolcheva ~]$
```

Рис. 3.8: Работа с консолью

От имени суперпользователя выполнила команды “sudo chown root:guest /home/guest/simpleid2” и “sudo chmod u+s /home/guest/simpleid2”, затем выполнила проверку правильности установки новых атрибутов и смены владельца файла simpleid2 командой “sudo ls -l /home/guest/simpleid2” (рис. 3.9). Этими командами была произведена смена пользователя файла на root и установлен SetUID-бит. (рис. [3.9])

```
yvkolcheva@yvkolcheva:~ x guest@yvkolcheva:~ yvkolcheva@yvkolcheva/h... x
[yvkolcheva@yvkolcheva ~]$ su
Пароль:
[root@yvkolcheva yvkolcheva]# chown root:guest /home/guest/simpleid2
[root@yvkolcheva yvkolcheva]# chmod u+s /home/guest/simpleid2
[root@yvkolcheva yvkolcheva]# ls -l /home/guest/simpleid2
-rwsrwxr-x. 1 root guest 18312 окт  2 14:45 /home/guest/simpleid2
[root@yvkolcheva yvkolcheva]#
```

Рис. 3.9: Работа с консолью

Запустила программы simpleid2 и id. Теперь появились различия в uid (рис. [3.10])

```
[guest@yvkolcheva ~]$ ./simpleid2
e uid=0, e gid=1001
real uid=1001, real gid=1001
[guest@yvkolcheva ~]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@yvkolcheva ~]$
```

Рис. 3.10: Работа с консолью

Проделала тоже самое относительно SetGID-бита. Также можем заметить различия с предыдущим пунктом (рис. [3.11]) (рис. [3.12])

```
-rwsrwxr-x. 1 root guest 18312 окт  2 14:45 /home/guest/simpleid2
[root@yvkolcheva yvkolcheva]# chown root:guest /home/guest/simpleid2
[root@yvkolcheva yvkolcheva]# chmod g+s /home/guest/simpleid2
[root@yvkolcheva yvkolcheva]#
```

Рис. 3.11: Работа с консолью

```
[guest@yvkolcheva ~]$ ./simpleid2
e uid=1001, e_gid=1001
real uid=1001, real_gid=1001
[guest@yvkolcheva ~]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfi
ned t:s0-s0:c0.c1023
[guest@yvkolcheva ~]$
```

Рис. 3.12: Работа с консолью

Создаем программу readfile.c (рис. [??])



```
#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;
}
```

Рис. 3.13: Работа с консолью

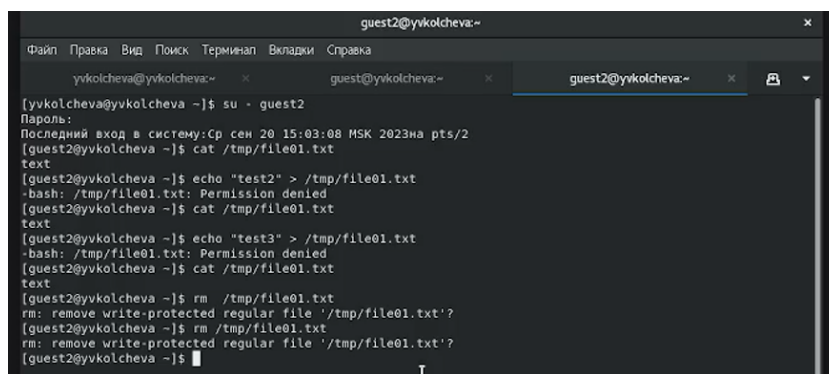
Скомпилировала созданную программу командой “gcc readfile.c -o readfile”. Сменила владельца у файла readfile.c командой “sudo chown root:guest /home/guest/readfile.c” и поменяла права так, чтобы только суперпользователь мог прочитать его, а guest не мог, с помощью команды “sudo chmod 700 /home/guest/readfile.c”. Теперь убедилась, что пользователь guest не может прочитать файл readfile.c командой “cat readfile.c”, получив отказ в доступе (рис. [3.14]) (рис. [3.15])

```
[guest@yvkolcheva ~]$ touch readfile.c
[guest@yvkolcheva ~]$ gcc simpleid2.c -o simpleid2
```

Рис. 3.14: Работа с консолью



От имени пользователя guest2 попробовала прочитать файл командой “cat /tmp/file01.txt” - это удалось. Далее попыталась дозаписать в файл слово test2, проверить содержимое файла и записать в файл слово test3, стеревав при этом всю имеющуюся в файле информацию - эти операции удалось выполнить только в случае, если еще дополнительно разрешить чтение и запись для группы пользователей командой “chmod g+rw /tmp/file01.txt”. От имени пользователя guest2 попробовала удалить файл - это не удастся ни в каком из случаев, возникает ошибка (рис. [3.18])



```
guest2@yvkolcheva:~  
Файл  Правка  Вид  Поиск  Терминал  Вкладки  Справка  
yvkolcheva@yvkolcheva:~  guest@yvkolcheva:~  guest2@yvkolcheva:~  
[yvkolcheva@yvkolcheva ~]$ su - guest2  
Пароль:  
Последний вход в систему: Ср сен 20 15:03:08 MSK 2023 на pts/2  
[guest2@yvkolcheva ~]$ cat /tmp/file01.txt  
text  
[guest2@yvkolcheva ~]$ echo "test2" > /tmp/file01.txt  
-bash: /tmp/file01.txt: Permission denied  
[guest2@yvkolcheva ~]$ cat /tmp/file01.txt  
text  
[guest2@yvkolcheva ~]$ echo "test3" > /tmp/file01.txt  
-bash: /tmp/file01.txt: Permission denied  
[guest2@yvkolcheva ~]$ cat /tmp/file01.txt  
text  
[guest2@yvkolcheva ~]$ rm /tmp/file01.txt  
rm: remove write-protected regular file '/tmp/file01.txt'?  
[guest2@yvkolcheva ~]$ rm /tmp/file01.txt  
rm: remove write-protected regular file '/tmp/file01.txt'?  
[guest2@yvkolcheva ~]$
```

Рис. 3.18: Работа с консолью

Повысила права до суперпользователя командой “su -” и выполнила команду, снимающую атрибут t с директории /tmp “chmod -t /tmp”. После чего покинула режим суперпользователя командой “exit”. Повторила предыдущие шаги. Теперь мне удалось удалить файл file01.txt от имени пользователя, не являющегося его владельцем (рис. [3.19])

```

[yvkolcheva@yvkolcheva ~]$ su - guest2
Пароль:
Последний вход в систему: Ср сен 20 15:03:08 MSK 2023 на pts/2
[guest2@yvkolcheva ~]$ cat /tmp/file01.txt
text
[guest2@yvkolcheva ~]$ echo "test2" > /tmp/file01.txt
-bash: /tmp/file01.txt: Permission denied
[guest2@yvkolcheva ~]$ cat /tmp/file01.txt
text
[guest2@yvkolcheva ~]$ echo "test3" > /tmp/file01.txt
-bash: /tmp/file01.txt: Permission denied
[guest2@yvkolcheva ~]$ cat /tmp/file01.txt
text
[guest2@yvkolcheva ~]$ rm /tmp/file01.txt
rm: remove write-protected regular file '/tmp/file01.txt'?
[guest2@yvkolcheva ~]$ rm /tmp/file01.txt
rm: remove write-protected regular file '/tmp/file01.txt'?
[guest2@yvkolcheva ~]$ rm /tmp/file01.txt
rm: remove write-protected regular file '/tmp/file01.txt'?
[guest2@yvkolcheva ~]$ rm /tmp/file01.txt
rm: remove write-protected regular file '/tmp/file01.txt'?
[guest2@yvkolcheva ~]$ ls
[guest2@yvkolcheva ~]$ ls /tmp
systemd-private-0b0aab320cfb412f9a31accfd0b4663b-chrond.service-z7H3JR
systemd-private-0b0aab320cfb412f9a31accfd0b4663b-colord.service-IMyrr7

```

Рис. 3.19: Работа с консолью

Повысила свои права до суперпользователя и вернула атрибут `t` на директорию `/tmp` (рис. [3.20])

```

[yvkolcheva@yvkolcheva ~]$ echo "test" > /tmp/file01.txt
[yvkolcheva@yvkolcheva ~]$ ls -l /tmp/file01.txt
-rw-rw-r--. 1 yvkolcheva yvkolcheva 5 окт 2 15:14 /tmp/file01.txt
[yvkolcheva@yvkolcheva ~]$ rm /tmp/file01.txt
[yvkolcheva@yvkolcheva ~]$ ls -l / | grep tmp
drwxr-xr-t. 16 root root 4096 окт 2 15:27 tmp
[yvkolcheva@yvkolcheva ~]$ ls -l / | grep tmp
drwxr-xr-t. 16 root root 4096 окт 2 15:41 tmp
[yvkolcheva@yvkolcheva ~]$

```

Рис. 3.20: Работа с консолью

## 4 Выводы

В ходе выполнения данной лабораторной работы я изучила механизмы изменения идентификаторов, применение 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

Лабораторная работа №5

Стандартные права SetUID, SetGID, Sticky в Linux [Электронный ресурс]. URL:  
<https://linux-notes.org/standartny-e-prava-unix-suid-sgid-sticky-bity/>