

# **Отчёт по лабораторной работе №5**

**Основы информационной безопасности**

Надежда Александровна Рогожина

# Содержание

1	Цель работы	5
2	Теоретическое введение	6
3	Выполнение лабораторной работы	8
4	Выводы	17
	Список литературы	18

# Список иллюстраций

3.1	Вход в систему . . . . .	8
3.2	Вход в систему . . . . .	8
3.3	simpleid.c . . . . .	9
3.4	компиляция . . . . .	9
3.5	проверка работоспособности . . . . .	9
3.6	id . . . . .	9
3.7	simpleid2.c . . . . .	10
3.8	simpleid2 . . . . .	10
3.9	смена прав от root . . . . .	10
3.10	проверка . . . . .	10
3.11	id VS simpleid2 . . . . .	11
3.12	SetGID-бит . . . . .	11
3.13	readfile . . . . .	11
3.14	компиляция и проверка прав . . . . .	12
3.15	Смена владельца и прав . . . . .	12
3.16	попытка №1 . . . . .	12
3.17	SetUID . . . . .	13
3.18	смена владельца и прав . . . . .	13
3.19	попытка №2 . . . . .	13
3.20	/tmp . . . . .	14
3.21	file01.txt . . . . .	14
3.22	guest2 . . . . .	14
3.23	попытка №3 . . . . .	14
3.24	попытка №4 . . . . .	15
3.25	попытка №5 . . . . .	15
3.26	смена прав /tmp . . . . .	15
3.27	повтор . . . . .	16
3.28	возвращение прав . . . . .	16

## **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 Теоретическое введение

Setuid – это бит разрешения, который позволяет пользователю запускать исполняемый файл с правами владельца этого файла. Другими словами, использование этого бита позволяет нам поднять привилегии пользователя в случае, если это необходимо. Классический пример использования этого бита в операционной системе это команда `sudo`.

```
root@ruvds-hrc [~]# which sudo /usr/bin/sudo
root@ruvds-hrc [~]# ls -l /usr/bin/sudo
-rwsr-xr-x 1 root root 125308 Feb 20 14:15 /usr/bin/sudo
```

Как мы видим на месте, где обычно установлен классический бит `x` (на исполнение), у нас выставлен специальный бит `s`. Это позволяет обычному пользователю системы выполнять команды с повышенными привилегиями без необходимости входа в систему как `root`, разумеется зная пароль пользователя `root`.

Принцип работы `Setgid` очень похож на `setuid` с отличием, что файл будет запускаться пользователем от имени группы, которая владеет файлом. Иллюстрирует работу этого бита команда `crontab`:

```
root@ruvds-hrc [~]# which crontab
/usr/bin/crontab
root@ruvds-hrc [~]# ls -l /usr/bin/crontab
-rwxr-sr-x 1 root crontab 34021 Feb 12 2017 /usr/bin/crontab
```

Последний специальный бит разрешения – это `Sticky Bit`. В случае, если этот бит установлен для папки, то файлы в этой папке могут быть удалены только

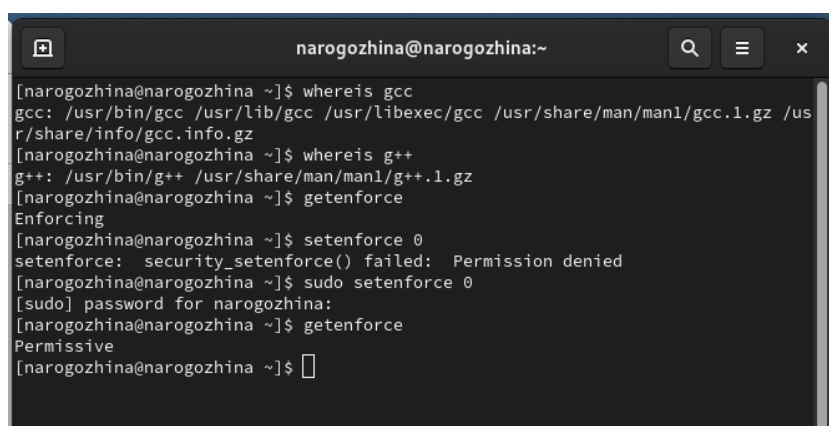
их владельцем. Пример использования этого бита в операционной системе это системная папка /tmp . Эта папка разрешена на запись любому пользователю, но удалять файлы в ней могут только пользователи, являющиеся владельцами этих файлов.

```
root@ruvds-hrc [~]# ls -ld /tmp
drwxrwxrwt 8 root root 4096 Mar 25 10:22 /tmp
```

Символ «t» указывает, что на папку установлен Sticky Bit.

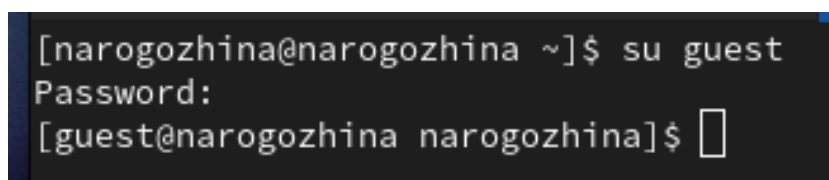
### 3 Выполнение лабораторной работы

1. Войдем в систему от имени пользователя guest (рис. [3.1], [3.2]).

A terminal window titled 'narogozhina@narogozhina:~' with search, menu, and close buttons. It shows the following commands and output:

```
[narogozhina@narogozhina ~]$ 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
[narogozhina@narogozhina ~]$ whereis g++
g++: /usr/bin/g++ /usr/share/man/man1/g++.1.gz
[narogozhina@narogozhina ~]$ getenforce
Enforcing
[narogozhina@narogozhina ~]$ setenforce 0
setenforce: security_setenforce() failed: Permission denied
[narogozhina@narogozhina ~]$ sudo setenforce 0
[sudo] password for narogozhina:
[narogozhina@narogozhina ~]$ getenforce
Permissive
[narogozhina@narogozhina ~]$
```

Рис. 3.1: Вход в систему

A terminal window showing the command to switch to the 'guest' user:

```
[narogozhina@narogozhina ~]$ su guest
Password:
[guest@narogozhina narogozhina]$
```

Рис. 3.2: Вход в систему

2. Создадим файл simpleid.c (рис. [3.3]).





Рис. 3.3: simpleid.c

3. Скомпилируем и убедимся, что файл программы создан (рис. [3.4], рис. [3.2]).



Рис. 3.4: компиляция

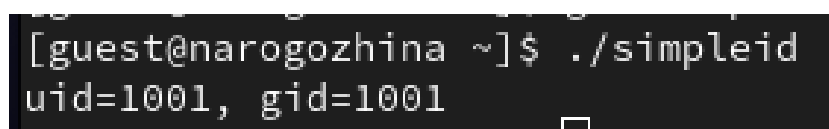


Рис. 3.5: проверка работоспособности

4. Выполните системную команду id и сравните с результатом вывода нашей программы (рис. [3.6]).

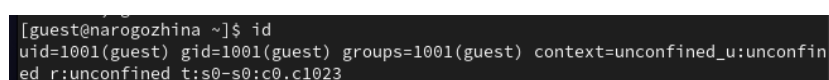
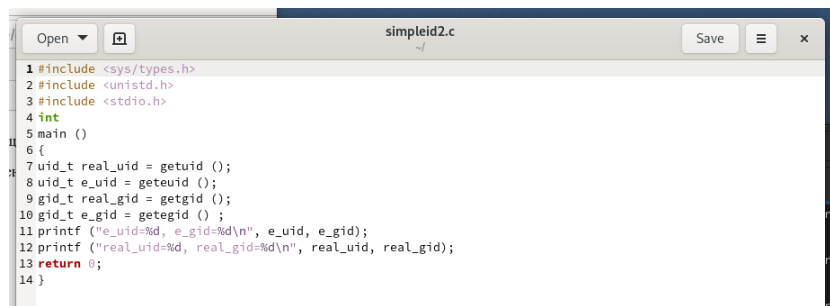


Рис. 3.6: id

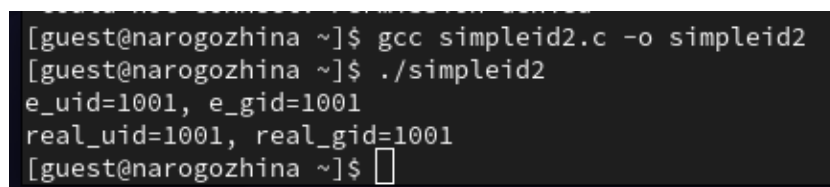
5. Программа simpleid2.c (рис. [3.7]).



```
1 #include <sys/types.h>
2 #include <unistd.h>
3 #include <stdio.h>
4 int
5 main ()
6 {
7     uid_t real_uid = getuid ();
8     uid_t e_uid = geteuid ();
9     gid_t real_gid = getgid ();
10    gid_t e_gid = getegid ();
11    printf ("e_uid=%d, e_gid=%d\n", e_uid, e_gid);
12    printf ("real_uid=%d, real_gid=%d\n", real_uid, real_gid);
13    return 0;
14 }
```

Рис. 3.7: simpleid2.c

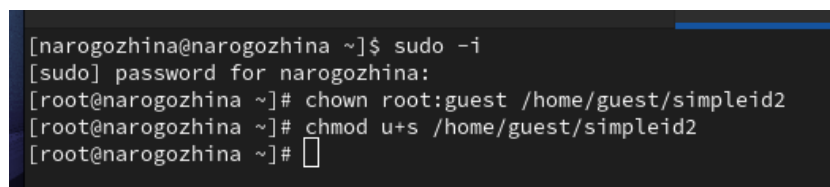
6. Компиляция и выполнение (рис. [3.8]).



```
[guest@narogozhina ~]$ gcc simpleid2.c -o simpleid2
[guest@narogozhina ~]$ ./simpleid2
e_uid=1001, e_gid=1001
real_uid=1001, real_gid=1001
[guest@narogozhina ~]$
```

Рис. 3.8: simpleid2

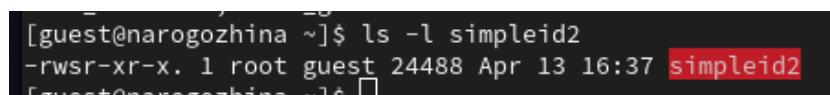
7. Смена владельца и прав (рис. [3.9]).



```
[narogozhina@narogozhina ~]$ sudo -i
[sudo] password for narogozhina:
[root@narogozhina ~]# chown root:guest /home/guest/simpleid2
[root@narogozhina ~]# chmod u+s /home/guest/simpleid2
[root@narogozhina ~]#
```

Рис. 3.9: смена прав от root

8. Проверка (рис. [3.10]).



```
[guest@narogozhina ~]$ ls -l simpleid2
-rwsr-xr-x. 1 root guest 24488 Apr 13 16:37 simpleid2
[guest@narogozhina ~]$
```

Рис. 3.10: проверка

9. Запуск двух команд (рис. [3.11]).

```
-rwsr-xr-x. 1 root guest 24488 Apr 13 16:37 simpleid2
[guest@narogozhina ~]$ ./simpleid2
e_uid=0, e_gid=1001
real_uid=1001, real_gid=1001
[guest@narogozhina ~]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@narogozhina ~]$
```

Рис. 3.11: id VS simpleid2

10. То же самое, только для SetGID-бита (рис. [3.12]).

```
[guest@narogozhina ~]$ ls -l simpleid2
-rwsr-xr-x. 1 root guest 24488 Apr 13 16:37 simpleid2
[guest@narogozhina ~]$ ./simpleid2
e_uid=0, e_gid=1001
real_uid=1001, real_gid=1001
[guest@narogozhina ~]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@narogozhina ~]$
```

Рис. 3.12: SetGID-бит

11. Программа readfile (рис. [3.13]).

```
1 #include <fcntl.h>
2 #include <stdio.h>
3 #include <sys/stat.h>
4 #include <sys/types.h>
5 #include <unistd.h>
6 int
7 main (int argc, char* argv[])
8 {
9     unsigned char buffer[16];
10    size_t bytes_read;
11    int i;
12    int fd = open (argv[1], O_RDONLY);
13    do
14    {
15        bytes_read = read (fd, buffer, sizeof (buffer));
16        for (i = 0; i < bytes_read; ++i) printf("%c", buffer[i]);
17    }
18    while (bytes_read == sizeof (buffer));
19    close (fd);
20    return 0;
21 }
```

Рис. 3.13: readfile

12. Компиляция и проверка прав (рис. [3.14]).

```
[guest@narogozhina ~]$ gcc readfile.c -o readfile
[guest@narogozhina ~]$ ls -l readfile
-rwxr-xr-x. 1 guest guest 24432 Apr 13 16:41 readfile
[guest@narogozhina ~]$
```

Рис. 3.14: компиляция и проверка прав

13. Смена владельца и прав у readfile (рис. [3.15]).

```
[root@narogozhina ~]# chown -R root:guest /home/guest/readfile
[root@narogozhina ~]# ls -l /home/guest/readfile
-rwxr-xr-x. 1 root guest 24432 Apr 13 16:41 /home/guest/readfile
[root@narogozhina ~]# chmod g-r /home/guest/readfile
[root@narogozhina ~]# chmod g-x /home/guest/readfile
[root@narogozhina ~]# chmod 711 /home/guest/readfile
[root@narogozhina ~]# ls -l /home/guest/readfile
-rwx--x--x. 1 root guest 24432 Apr 13 16:41 /home/guest/readfile
[root@narogozhina ~]#
```

Рис. 3.15: Смена владельца и прав

14. Попытка прочитать (рис. [3.16]).

```
[guest@narogozhina ~]$ cat readfile
cat: readfile: Permission denied
[guest@narogozhina ~]$
```

Рис. 3.16: попытка №1

15. Пример работы SetUID-бита (рис. [3.17]).

```

cat: readfile: permission denied
[guest@narogozhina ~]$ touch myfile
[guest@narogozhina ~]$ ls -l myfile
-rw-r--r--. 1 guest guest 0 Apr 13 17:08 myfile
[guest@narogozhina ~]$ chmod g+w myfile
[guest@narogozhina ~]$ ls -l myfile
-rw-rw-r--. 1 guest guest 0 Apr 13 17:08 myfile
[guest@narogozhina ~]$ chmod u+s myfile
[guest@narogozhina ~]$ ls -l myfile
-rwSr--r--. 1 guest guest 0 Apr 13 17:08 myfile
[guest@narogozhina ~]$

```

Рис. 3.17: SetUID

16. Смена прав и владельца файла readfile (рис. [3.18]).

```

[root@narogozhina ~]# ls -l /home/guest/readfile
-rwx--x--x. 1 root guest 24432 Apr 13 16:41 /home/guest/readfile
[root@narogozhina ~]# chown root:guest /home/guest/readfile
[root@narogozhina ~]# ls -l /home/guest/readfile
-rwx--x--x. 1 root guest 24432 Apr 13 16:41 /home/guest/readfile
[root@narogozhina ~]# chmod u+s /home/guest/readfile
[root@narogozhina ~]# ls -l /home/guest/readfile
-rws--x--x. 1 root guest 24432 Apr 13 16:41 /home/guest/readfile
[root@narogozhina ~]#

```

Рис. 3.18: смена владельца и прав

17. Попытка прочитать файлы (рис. [3.19]).

```

bash: readfile: command not found...
[guest@narogozhina ~]$ readfile readfile.c
bash: readfile: command not found...
[guest@narogozhina ~]$ readfile /etc/shadow
bash: readfile: command not found...
[guest@narogozhina ~]$ ls
Desktop  Downloads  Pictures  readfile.c  simpleid2.c  Videos
dir1     Music      Public    simpleid     simpleid.c
Documents  myfile     readfile  simpleid2    Templates
[guest@narogozhina ~]$

```

Рис. 3.19: попытка №2

18. Права доступа директории /tmp (рис. [3.20]).

```
[guest@narogozhina ~]$ ls -l / | grep tmp
drwxrwxrwt. 18 root root 4096 Apr 13 17:07 tmp
[guest@narogozhina ~]$
```

Рис. 3.20: /tmp

19. Попытка записи в новый файл и просмотр его атрибутов (рис. [3.21]).

```
[guest@narogozhina ~]$ echo "test" > /tmp/file01.txt
[guest@narogozhina ~]$ ls =l /tmp/file01.txt
ls: cannot access '=l': No such file or directory
/tmp/file01.txt
[guest@narogozhina ~]$ ls -l /tmp/file01.txt
-rw-r--r--. 1 guest guest 5 Apr 13 17:12 /tmp/file01.txt
[guest@narogozhina ~]$ chmod o+rw /tmp/file01.txt
[guest@narogozhina ~]$ ls -l /tmp/file01.txt
-rw-r--rw-. 1 guest guest 5 Apr 13 17:12 /tmp/file01.txt
[guest@narogozhina ~]$
```

Рис. 3.21: file01.txt

20. Вход от имени guest2 (рис. [3.22]).

```
[narogozhina@narogozhina ~]$ su guest2
Password:
[guest2@narogozhina narogozhina]$ cat /tmp/file01.txt
test
[guest2@narogozhina narogozhina]$
```

Рис. 3.22: guest2

21. Попытка дозаписи в файл (рис. [3.23]).

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

Рис. 3.23: попытка №3

22. Попытка перезаписи файла (рис. [3.24]).

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

Рис. 3.24: попытка №4

23. Попытка удаления файла (рис. [3.25]).

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

Рис. 3.25: попытка №5

24. Смена прав директории (рис. [3.26]).

```
[root@narogozhina ~]# chmod -t /tmp
```

Рис. 3.26: смена прав /tmp

25. Проверка (рис. [??]).

```
[guest2@narogozhina ~]$ ls -l / |grep tmp
drwxrwxrwx. 18 root root 4096 Apr 13 17:16 tmp
[guest2@narogozhina ~]$
```

26. Повторение всех предыдущих действий (рис. [3.27]).

```
[guest2@narogozhina ~]$ echo "test2" > /tmp/file01.txt
bash: /tmp/file01.txt: Permission denied
[guest2@narogozhina ~]$ echo "test3" >> /tmp/file01.txt
bash: /tmp/file01.txt: Permission denied
[guest2@narogozhina ~]$ rm /tmp/file01.txt
rm: remove write-protected regular file '/tmp/file01.txt'? y
[guest2@narogozhina ~]$ ls -l /tmp |grep file01.txt
[guest2@narogozhina ~]$
```

Рис. 3.27: повтор

27. Возвращение прав обратно (рис. [3.28]).

```
[root@narogozhina ~]# chmod +t /tmp
[root@narogozhina ~]# ls -l / |grep tmp
drwxrwxrwt. 18 root root 4096 Apr 13 17:17 tmp
[root@narogozhina ~]#
```

Рис. 3.28: возвращение прав



## 4 Выводы

В ходе работы мы изучили механизмы изменения идентификаторов, применения SetUID- и Sticky-битов, получили практические навыки работы в консоли с дополнительными атрибутами, а также рассмотрели работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.

## **Список литературы**