

Отчёт по лабораторной работе №5

**Дискреционное разграничение прав в Linux. Исследование влияния
дополнительных атрибутов**

Виктория Михайловна Шутенко

Содержание

1	Цель работы	5
2	Подготовка к выполнению лабораторной работы №5	6
3	Ход работы	7
3.1	Создание программы	7
3.2	Исследование Sticky-бита	12
4	Итоги выполнения лабораторной работы	15

List of Figures

2.1	Подготовка лабораторного стенда.	6
3.1	Код simpleid.c.	7
3.2	Компилирование и запуск файла simpleid.c. Выполнение команды id	8
3.3	Код simpleid2.c	8
3.4	Компилирование и запуск simpleid2.c.	8
3.5	Выполнение команд.	9
3.6	Проверка правильности.	9
3.7	Запуск simpleid2 и id.	9
3.8	Код readfile.c.	10
3.9	Компиляция и запуск readfile.c.	10
3.10	Запуск readfile.c.	11
3.11	Запуск readfile.c.	11
3.12	Просмотр атрибутов file01.txt	12
3.13	Запись файла и проверка.	13
3.14	Удаление файла, вход и выход из режима суперпользователя. . .	14

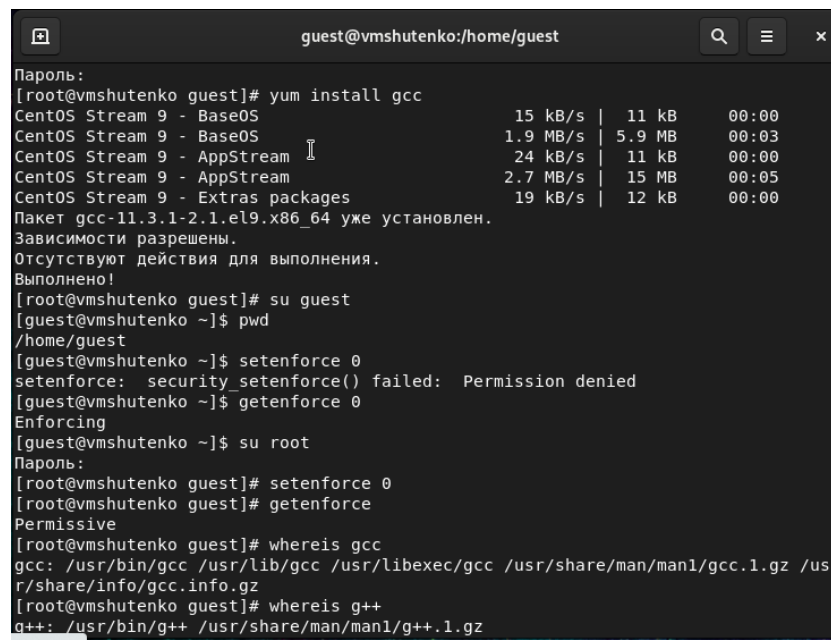
List of Tables

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.¹

2 Подготовка к выполнению лабораторной работы №5

Я выполнила подготовительные действия перед началом выполнения лабораторной работы - проверила наличие gcc и g++



```
guest@vmshutenko:/home/guest
Пароль:
[root@vmshutenko guest]# yum install gcc
CentOS Stream 9 - BaseOS                               15 kB/s | 11 kB      00:00
CentOS Stream 9 - BaseOS                               1.9 MB/s | 5.9 MB    00:03
CentOS Stream 9 - AppStream                             24 kB/s | 11 kB      00:00
CentOS Stream 9 - AppStream                             2.7 MB/s | 15 MB     00:05
CentOS Stream 9 - Extras packages                       19 kB/s | 12 kB      00:00
Пакет gcc-11.3.1-2.1.el9.x86_64 уже установлен.
Зависимости разрешены.
Отсутствуют действия для выполнения.
Выполнено!
[root@vmshutenko guest]# su guest
[guest@vmshutenko ~]$ pwd
/home/guest
[guest@vmshutenko ~]$ setenforce 0
setenforce: security setenforce() failed: Permission denied
[guest@vmshutenko ~]$ getenforce 0
Enforcing
[guest@vmshutenko ~]$ su root
Пароль:
[root@vmshutenko guest]# setenforce 0
[root@vmshutenko guest]# getenforce
Permissive
[root@vmshutenko guest]# 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
[root@vmshutenko guest]# whereis g++
g++: /usr/bin/g++ /usr/share/man/man1/g++.1.gz
```

Figure 2.1: Подготовка лабораторного стенда.

3 Ход работы

3.1 Создание программы

1. Вошла в систему от имени пользователя guest.
2. Создала программу simpleid.c:

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

Figure 3.1: Код simpleid.c.

3. Скомпилировала программу и убедилась, что файл программы создан:

```
gcc simpleid.c -o simpleid
```

4. Выполнила программу simpleid:

```
./simpleid
```

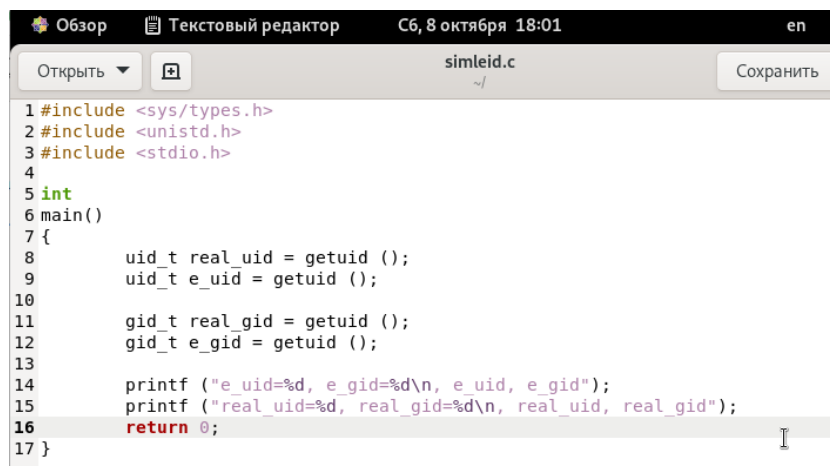
5. Выполнила системную программу id:

id

```
[root@vmshutenko guest]# gcc simpleid.c -o simpleid
[root@vmshutenko guest]# ls
dir1      simpleid  Документы  Изображения  Общедоступные  Шаблоны
simpleid.c Видео     Загрузки   Музыка        'Рабочий стол'
[root@vmshutenko guest]# ./simpleid
uid=1928818808, gid=1928818824
, uid, gid[root@vmshutenko guest]# id
uid=0(root) gid=0(root) группы=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:
s0-s0:c0.c1023
[root@vmshutenko guest]# id
uid=0(root) gid=0(root) группы=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:
s0-s0:c0.c1023
[root@vmshutenko guest]# S
```

Figure 3.2: Компилирование и запуск файла simpleid.c. Выполнение команды id

6. Усложнила программу, добавив вывод действительных идентификаторов:



```
Обзор  Текстовый редактор  С6, 8 октября 18:01  en
Открыть  Сохранить
simpleid.c
~/
1 #include <sys/types.h>
2 #include <unistd.h>
3 #include <stdio.h>
4
5 int
6 main()
7 {
8     uid_t real_uid = getuid ();
9     uid_t e_uid = getuid ();
10
11     gid_t real_gid = getuid ();
12     gid_t e_gid = getuid ();
13
14     printf ("e_uid=%d, e_gid=%d\n, e_uid, e_gid");
15     printf ("real_uid=%d, real_gid=%d\n, real_uid, real_gid");
16     return 0;
17 }
```

Figure 3.3: Код simpleid2.c

7. Скомпилировала и запустила simpleid2.c:

```
gcc simpleid2.c -o simpleid2
```

```
./simpleid2
```

```
[root@vmshutenko guest]# gcc simpleid2.c -o simpleid2
[root@vmshutenko guest]# ./simpleid2
e_uid=817137464, e_gid=817137480
, e_uid, e_gidreal_uid=100, real_gid=0
, real_uid, real_gid[root@vmshutenko guest]# id
uid=0(root) gid=0(root) группы=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:
s0-s0:c0.c1023
```

Figure 3.4: Компилирование и запуск simpleid2.c.

8. От имени суперпользователя выполнила команды:

```
chown root:guest /home/guest/simpleid2
```

```
chmod u+s /home/guest/simpleid2
```

```
[root@vmshutenko guest]# chown root:guest /home/guest/simpleid2
[root@vmshutenko guest]# chmod u+s /home/guest/simpleid2
```

Figure 3.5: Выполнение команд.

9. Использовала `sudo` и временно повысила свои права.

10. Выполнила проверку правильности установки новых атрибутов и смены владельца файла `simpleid2`:

```
ls -l simpleid2
```

```
[root@vmshutenko guest]# ls -l simpleid2
-rwsr-xr-x. 1 root guest 25856 окт  8 18:25 simpleid2
```

Figure 3.6: Проверка правильности.

11. Запустила `simpleid2` и `id`:

```
./simpleid2
```

```
id
```

```
[root@vmshutenko guest]# ./simpleid2
e_uid=-1953519000, e_gid=-1953518984
, e_uid, e_gidreal_uid=100, real_gid=0
, real_uid, real_gid[root@vmshutenko guest]# ./simpleid2
e_uid=-1760983624, e_gid=-1760983608
, e_uid, e_gidreal_uid=100, real_gid=0
, real_uid, real_gid[root@vmshutenko guest]#
[root@vmshutenko guest]# id
uid=0(root) gid=0(root) группы=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[root@vmshutenko guest]#
```

Figure 3.7: Запуск `simpleid2` и `id`.

12. Проделала тоже самое относительно SetGID-бита.

13. Создала программу `readfile.c`:

simpleid2.c	simpleid.c	readfile.c
<pre> 1 #include <fcntl.h> 2 #include <sys/types.h> 3 #include <unistd.h> 4 #include <stdio.h> 5 #include <sys/stat.h> 6 7 int main(int argc, char* argv[]) 8 { 9 unsigned char buffer[16]; 10 size_t bytes_read; 11 int i; 12 13 int fd = open (argv[1], O_RDONLY); 14 do{ 15 bytes_read = read (fd, buffer, sizeof (buffer)); 16 for (i=0; i < bytes_read; ++i) printf("%c", buffer[i]); 17 } 18 19 while (bytes_read == sizeof (buffer)); 20 close (fd); 21 return 0; 22 }</pre>		

Figure 3.8: Код readfile.c.

14. Откомпилировала её.

`gcc readfile.c -o readfile`

```

guest@vmshutenko:/home/guest x guest@vmshutenko:~ x
[root@vmshutenko guest]# touch readfile.c
[root@vmshutenko guest]# chmod 777 readfile.c
[root@vmshutenko guest]# gcc readfile.c -o readfile
[root@vmshutenko guest]# ./readfile
V>@000lK0Bc000KS{00000lv@000
000l00d@p@0000
000-/0c00#00%0x86 64./readfileSHELL=/bin/bashSESSION_MANAGER=local/unix:@/tmp/.ICE-unix
/1473,unix:/tmp/.ICE-unix/1473COLORTERM=truecolorHISTCONTROL=ignoredupsXDG_MENU_PR
EFIX=gnome-HOSTNAME=vmshutenkoHISTSIZE=1000SSH_AUTH_SOCK=/run/user/1001/keyring/sshXMOD
IFIERS=@im=ibusDESKTOP_SESSION=gnomePWD=/home/guestXDG_SESSION_DESKTOP=gnomeLOGNAME=gue
stXDG_SESSION_TYPE=waylandSYSTEMD_EXEC_PID=1493GDM_LANG=ru_RU.UTF-8HOME=/rootUSERNAME=g
uestLANG=ru_RU.UTF-8LS_COLORS=rs=0:di=01;34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35:b
d=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=3
4;42:st=37;44:ex=01;32:*.tar=01;31:*.tgz=01;31:*.arc=01;31:*.arj=01;31:*.taz=01;31:*.l
h
a=01;31:*.lzh=01;31:*.lzh=01;31:*.lzh=01;31:*.tlz=01;31:*.txz=01;31:*.tzo=01;31:*.t7z=
01;31:*.zip=01;31:*.z=01;31:*.dz=01;31:*.gz=01;31:*.lrz=01;31:*.lz=01;31:*.lzo=01;31:*.
xz=01;31:*.zst=01;31:*.tzt=01;31:*.bz2=01;31:*.bz=01;31:*.tbz=01;31:*.tbz2=01;31:*.tz
=01;31:*.deb=01;31:*.rpm=01;31:*.jar=01;31:*.war=01;31:*.ear=01;31:*.sar=01;31:*.rar=01;
31:*.alz=01;31:*.ace=01;31:*.zoo=01;31:*.cpio=01;31:*.7z=01;31:*.rz=01;31:*.cab=01;31:*.
wim=01;31:*.swm=01;31:*.dwm=01;31:*.esd=01;31:*.jpg=01;35:*.jpeg=01;35:*.mjpg=01;35:*.
mjpeg=01;35:*.gif=01;35:*.bmp=01;35:*.pbm=01;35:*.pgm=01;35:*.ppm=01;35:*.tga=01;35:*.x
bm=01;35:*.xpm=01;35:*.tif=01;35:*.tiff=01;35:*.png=01;35:*.svg=01;35:*.svgz=01;35:*.m
g=01;35:*.pcx=01;35:*.mov=01;35:*.mpg=01;35:*.mpeg=01;35:*.m2v=01;35:*.mkv=01;35:*.webm
=01;35:*.webp=01;35:*.ogm=01;35:*.mp4=01;35:*.m4v=01;35:*.mp4v=01;35:*.vob=01;35:*.qt=0
1;35:*.nuv=01;35:*.wmv=01;35:*.asf=01;35:*.rm=01;35:*.rmvb=01;35:*.flc=01;35:*.avi=01;3

```

Figure 3.9: Компиляция и запуск readfile.c.

15. Сменила владельца у файла readfile.c и изменила права так, чтобы только суперпользователь (root) мог прочитать его, а guest не мог.

3.2 Исследование Sticky-бита

1. Выяснила, установлен ли атрибут Sticky на директории /tmp, для чего выполнила команду

```
ls -l / | grep tmp
```

2. От имени пользователя guest создала файл file01.txt в директории /tmp со словом test:

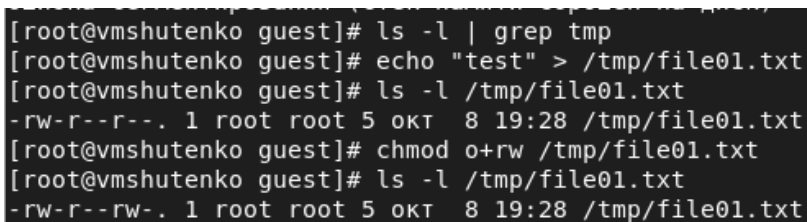
```
echo "test" > /tmp/file01.txt
```

3. Просмотрела атрибуты у только что созданного файла и разрешила чтение и запись для категории пользователей «все остальные»:

```
ls -l /tmp/file01.txt
```

```
chmod o+rw /tmp/file01.txt
```

```
ls -l /tmp/file01.txt
```



```
[root@vmshutenko guest]# ls -l | grep tmp
[root@vmshutenko guest]# echo "test" > /tmp/file01.txt
[root@vmshutenko guest]# ls -l /tmp/file01.txt
-rw-r--r--. 1 root root 5 окт  8 19:28 /tmp/file01.txt
[root@vmshutenko guest]# chmod o+rw /tmp/file01.txt
[root@vmshutenko guest]# ls -l /tmp/file01.txt
-rw-r--rw-. 1 root root 5 окт  8 19:28 /tmp/file01.txt
```

Figure 3.12: Просмотр атрибутов file01.txt

4. От пользователя guest2 (не являющегося владельцем) попробовала прочитать файл /tmp/file01.txt:

```
cat /tmp/file01.txt
```

5. От пользователя guest2 попробовала дозаписать в файл /tmp/file01.txt слово test2 командой

```
echo "test2" > /tmp/file01.txt
```

6. Проверила содержимое файла командой

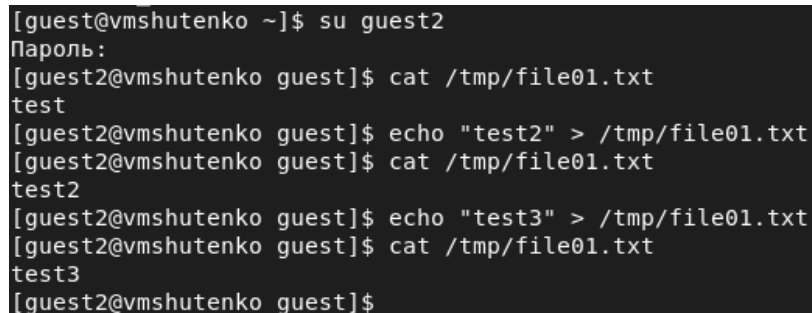
```
cat /tmp/file01.txt
```

7. От пользователя guest2 попробовала записать в файл /tmp/file01.txt слово test3, стерев при этом всю имеющуюся в файле информацию командой

```
echo "test3" > /tmp/file01.txt
```

8. Проверила содержимое файла командой

```
cat /tmp/file01.txt
```



```
[guest@vmshutenko ~]$ su guest2
Пароль:
[guest2@vmshutenko guest]$ cat /tmp/file01.txt
test
[guest2@vmshutenko guest]$ echo "test2" > /tmp/file01.txt
[guest2@vmshutenko guest]$ cat /tmp/file01.txt
test2
[guest2@vmshutenko guest]$ echo "test3" > /tmp/file01.txt
[guest2@vmshutenko guest]$ cat /tmp/file01.txt
test3
[guest2@vmshutenko guest]$
```

Figure 3.13: Запись файла и проверка.

9. От пользователя guest2 попробовала удалить файл /tmp/file01.txt командой

```
rm /tmp/file01.txt
```

10. Повысила свои права до суперпользователя следующей командой

```
su -
```

и выполнила после этого команду, снимающую атрибут t (Sticky-бит) с директории /tmp:

```
chmod -t /tmp
```

11. Покинула режим суперпользователя командой

exit

12. От пользователя guest2 проверила, что атрибута t у директории /tmp нет:

ls -l / | grep tmp

```
[guest2@vmshutenko guest]$ rm /tmp/file01.txt
rm: невозможно удалить '/tmp/file01.txt': Операция не позволена
[guest2@vmshutenko guest]$ su -
Пароль:
[root@vmshutenko ~]# chmod -t /tmp
[root@vmshutenko ~]# exit
выход
[guest2@vmshutenko guest]$ ls -l | grep tmp
[guest2@vmshutenko guest]$ cat /tmp/file01.txt
test3
[guest2@vmshutenko guest]$ rm /tmp/file01.txt
```

Figure 3.14: Удаление файла, вход и выход из режима суперпользователя.

4 Итоги выполнения лабораторной работы

- Получили практические навыки работы в консоли с расширенными атрибутами файлов;
- Заполнили таблицу.