

Лабораторная работа №5

Титаренко Анастасия Дмитриевна

Содержание

Цель работы	1
Выполнение лабораторной работы	1
Создание программы.....	1
Исследование Sticky-бита.....	5
Вывод.....	7
Список литературы	7

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

Создание программы

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

```
guest@adtitarenko:~  
Файл Правка Вид Поиск Терминал Справка  
#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;  
}  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
"simpleid.c" 13L, 181C 13,0-1 Весь
```

Листинг программы simpleid.c

3. Скомпилировала программу и убедилась, что файл программы создан: gcc simpleid.c -o simpleid

```
[guest@adtitarenko ~]$ gcc simpleid.c -o simpleid  
[guest@adtitarenko ~]$ ls  
dir1 simpleid simpleid.c  
[guest@adtitarenko ~]$
```

Компилирование программы simpleid.c

4. Выполнила программу simpleid: ./simpleid
5. Выполнила системную программу id: id Результаты двух команд совпадают.

```
[guest@adtitarenko ~]$ ./simpleid  
uid=1001, gid=1001  
[guest@adtitarenko ~]$ id  
uid=1001(guest) gid=1001(guest) группы=1001(guest) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

Сравнение результата команд ./simpleid и id

6. Усложнила программу, добавив вывод действительных идентификаторов. Получившуюся программу назвала simpleid2.c.

```
guest@adtitarenko:~  
Файл Правка Вид Поиск Терминал Справка  
#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;  
}  
~  
~  
~  
"simpleid2.c" [Новый] 19L, 313C записано 19,0-1 Весь
```

Листинг программы simpleid2.c

7. Скомпилировала и запустила simpleid2.c:

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

Компилирование и запуск программы simpleid2.c

8. От имени суперпользователя выполнила команды: `chown root:guest /home/guest/simpleid2` `chmod u+s /home/guest/simpleid2`

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

Изменение прав доступа файла simpleid2

9. Использовала `sudo` или повысила временно свои права с помощью `su`.
10. Выполнила проверку правильности установки новых атрибутов и смены владельца файла simpleid2: `ls -l simpleid2`

```
[root@adtitarenko guest]#  
[root@adtitarenko guest]# ls -l simpleid2  
-rwsrwxr-x. 1 root guest 8616 окт  8 18:09 simpleid2  
[root@adtitarenko guest]#
```

Проверка правильности установки новых атрибутов и смены владельца файла simpleid2

11. Запустила simpleid2 и `id`: `./simpleid2 id` Результаты двух команд совпадают.

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

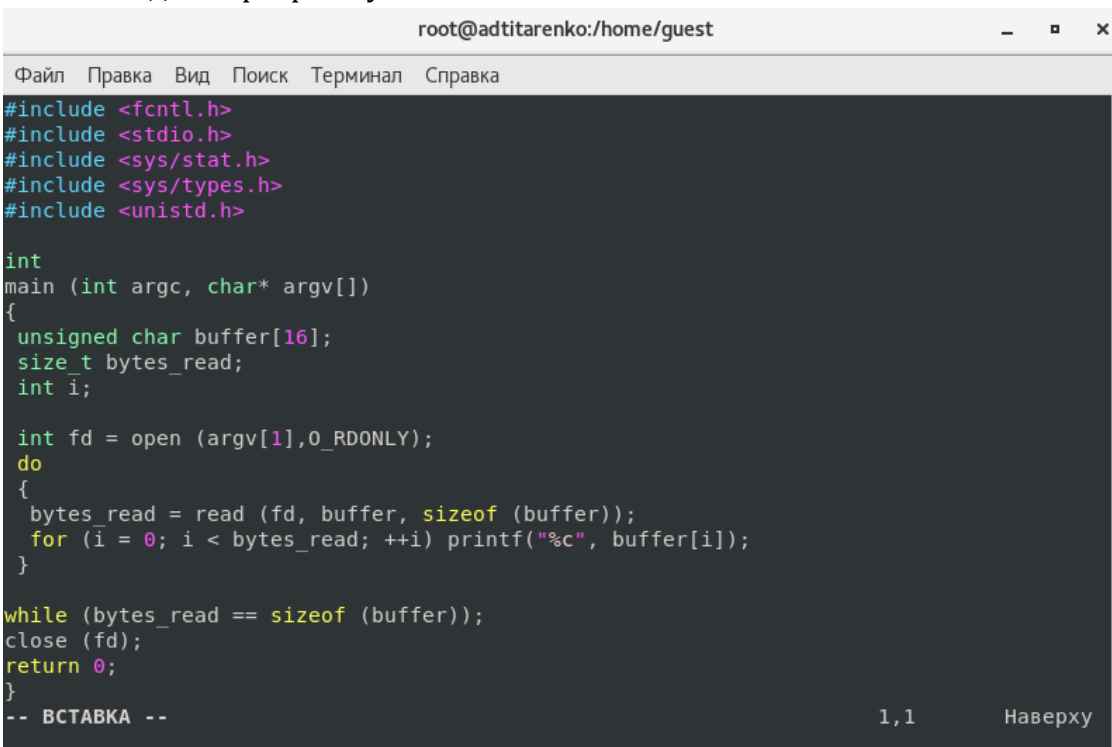
Сравнение результата команд `./simpleid2` и `id`

12. Проделала тоже самое относительно SetGID-бита.

```
[root@adtitarenko guest]# chmod g+s /home/guest/simpleid2
[root@adtitarenko guest]# ls -l simpleid2
-rwsrwsr-x. 1 root guest 8616 окт  8 18:09 simpleid2
[root@adtitarenko guest]# ./simpleid2
e_uid=0, e_gid=1001
real_uid=0, real_gid=0
```

Изменения относительно SetGID-бита файла `simpleid2`

13. Создала программу `readfile.c`:



```
root@adtitarenko:/home/guest
Файл  Правка  Вид  Поиск  Терминал  Справка
#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;
}
-- ВСТАВКА -- 1,1 Наверху
```

Листинг программы `readfile.c`

14. Откомпилировала её. `gcc readfile.c -o readfile`

```
[root@adtitarenko guest]#
[root@adtitarenko guest]#
[root@adtitarenko guest]# gcc readfile.c -o readfile
```

Компилирование программы `readfile.c`

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

16. Проверила, что пользователь `guest` не может прочитать файл `readfile.c`.

```
[root@adtitarenko guest]#
[root@adtitarenko guest]#
[root@adtitarenko guest]# chown root readfile.c
[root@adtitarenko guest]# chmod og-rwx readfile.c
[root@adtitarenko guest]#
[root@adtitarenko guest]# su - guest
Последний вход в систему:Сб окт  8 17:52:24 MSK 2022на pts/0
[guest@adtitarenko ~]$
[guest@adtitarenko ~]$ cat readfile.c
cat: readfile.c: Отказано в доступе
```

Изменения прав файла readfile.c

17. Сменила у программы readfile владельца и установите SetU'D-бит.

18. Проверила, может ли программа readfile прочитать файл readfile.c

```
[root@adtitarenko guest]# chmod u+s /home/guest/readfile
[root@adtitarenko guest]# ./readfile
[guest@adtitarenko ~]$ cat readfile.c
cat: readfile.c: Отказано в доступе
```

Установка SetU'D-бита и изменения владельца файла readfile.c

Исследование Sticky-бита

1. Выяснила, установлен ли атрибут Sticky на директории /tmp, для чего выполнила команду `ls -l / | grep tmp`

```
[root@adtitarenko guest]# ls -l / | grep tmp
drwxrwxrwt. 17 root root 4096 окт  8 18:34 tmp
[root@adtitarenko guest]#
[root@adtitarenko guest]# su - guest
Последний вход в систему:Сб окт  8 18:33:35 MSK 2022на pts/0
[guest@adtitarenko ~]$
```

Выполнение команды: `ls -l / | grep tmp`

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

```
[guest@adtitarenko ~]$
[guest@adtitarenko ~]$ echo "test" > /tmp/file01.txt
[guest@adtitarenko ~]$
[guest@adtitarenko ~]$ ls /tmp
anaconda.log
file01.txt
```

Создание файла 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`

```
[guest@adtitarenko ~]$
[guest@adtitarenko ~]$ ls -l /tmp/file01.txt
-rw-rw-r--. 1 guest guest 5 окт  8 18:38 /tmp/file01.txt
[guest@adtitarenko ~]$ chmod o+rw /tmp/file01.txt
[guest@adtitarenko ~]$ ls -l /tmp/file01.txt
-rw-rw-rw-. 1 guest guest 5 окт  8 18:38 /tmp/file01.txt
[guest@adtitarenko ~]$
```

Просмотр атрибутов и изменение прав файла file01.txt

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

```
[guest2@adtitarenko ~]$  
[guest2@adtitarenko ~]$  
[guest2@adtitarenko ~]$ cat /tmp/file01.txt  
test
```

Попытка прочитать файл file01.txt от пользователя, не являющегося владельцем

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

```
[guest2@adtitarenko ~]$  
[guest2@adtitarenko ~]$ cat /tmp/file01.txt  
test2  
[guest2@adtitarenko ~]$ █
```

Изменение файла file01.txt и просмотр содержимого

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 Операцию удалось выполнить.
8. Проверила содержимое файла командой cat /tmp/file01.txt

```
[guest2@adtitarenko ~]$  
[guest2@adtitarenko ~]$ echo "test3" > /tmp/file01.txt  
[guest2@adtitarenko ~]$ cat /tmp/file01.txt  
test3
```

Изменение файла file01.txt и просмотр содержимого

9. От пользователя guest2 попробовала удалить файл /tmp/file01.txt командой rm /tmp/file01.txt Файл удалить не получилось.

```
[guest2@adtitarenko ~]$  
[guest2@adtitarenko ~]$  
[guest2@adtitarenko ~]$ rm /tmp/file01.txt  
rm: невозможно удалить «/tmp/file01.txt»: Операция не позволена
```

Попытка удалить файл 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

```
[guest2@adtitarenko ~]$ su  
Пароль:  
[root@adtitarenko guest2]# chmod -t /tmp  
[root@adtitarenko guest2]# exit  
exit
```

Сняла атрибут -t с директории /tmp и покинула режим суперпользователя

12. От пользователя guest2 проверила, что атрибута t у директории /tmp нет: ls -l / | grep tmp

```
[guest2@adtitarenko ~]$ ls -l / | grep tmp  
drwxrwxrwx. 17 root root 4096 окт  8 18:44 tmp  
[guest2@adtitarenko ~]$  
[guest2@adtitarenko ~]$
```

Проверка: атрибута -t нет у директории /tmp

13. Повторила предыдущие шаги.

14. Удалось удалить файл от имени пользователя, не являющегося его владельцем.

```
[guest2@adtitarenko ~]$ cat /tmp/file01.txt
test3
[guest2@adtitarenko ~]$ echo "test2" > /tmp/file01.txt
[guest2@adtitarenko ~]$ cat /tmp/file01.txt
test2
[guest2@adtitarenko ~]$ echo "test3" > /tmp/file01.txt
[guest2@adtitarenko ~]$ cat /tmp/file01.txt
test3
[guest2@adtitarenko ~]$ rm /tmp/file01.txt
[guest2@adtitarenko ~]$
```

Повтор предыдущих шагов

15. Повысила свои права до суперпользователя и верните атрибут `t` на директорию `/tmp`: `su - chmod +t /tmp exit`

```
[root@adtitarenko guest2]# chmod +t /tmp
[root@adtitarenko guest2]# exit
exit
[guest2@adtitarenko ~]$
```

Вернула атрибут `-t` с директории `/tmp`

Вывод

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

Список литературы

1. [Лабораторная работа № 5. Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов](#)