Лабораторная работа №5

Основы информационной безопасности

Анастасия Андреевна Кондрашова

Содержание

Список иллюстраций

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 Выполнение лабораторной работы

1. Вошли в систему от имени пользователя guest2.
2. Создали программу simpleid.c.
3. Скомплилируйте программу и убедитесь, что файл программы создан с помощью команды gcc simpleid.c -o simpleid.
4. Выполните программу simpleid.
5. Выполните системную программу id и сравнили полученный нами результат с данными предыдущего пункта. Данные иденичны.

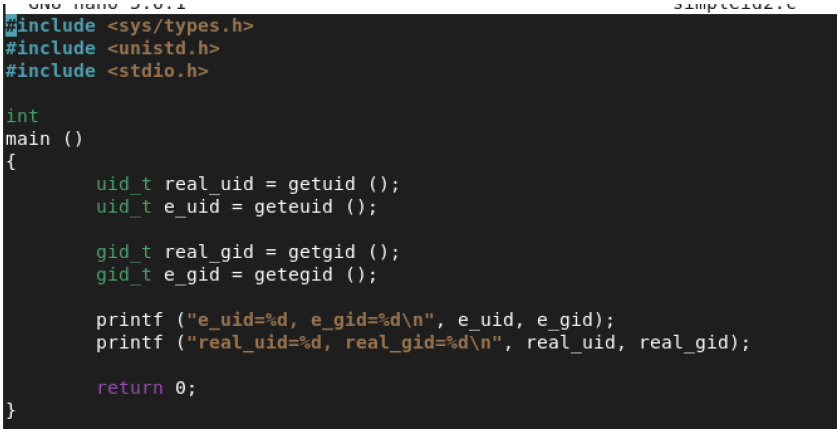
```
Last login: Sat Sep 17 20:56:56 MSK 2022 on pts/1
[guest2@aakondrashova ~]$ simpleid.c
bash: simpleid.c: command not found...
[guest2@aakondrashova ~]$ touch simpleid.c
[guest2@aakondrashova ~]$ nano simpleid.c
[guest2@aakondrashova ~]$ gcc simpleid.c -o simpleid
[guest2@aakondrashova ~]$ ./simpleid
uid=1002, gid=1002/n[guest2@aakondrashova ~]$ id
uid=1002(guest2) gid=1002(guest2) groups=1002(guest2),1001(guest
023
```

Рис. 2.1: Выполнение программы simpleid

```
  GNU nano 5.6.1
#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;
}
```

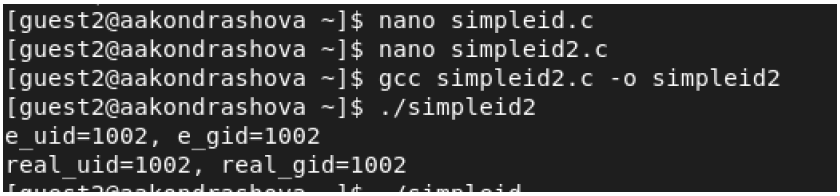
Рис. 2.2: Программа simpleid

6. Усложнили программу, добавив вывод действительных идентификаторов.Получившуюся программу назвали simpleid2.c.
7. Скомпилируйте и запустили simpleid2.c с помощью команды gcc simpleid2.c -o 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. 2.3: Программа simpleid2



```
[guest2@aakondrashova ~]$ nano simpleid.c
[guest2@aakondrashova ~]$ nano simpleid2.c
[guest2@aakondrashova ~]$ gcc simpleid2.c -o simpleid2
[guest2@aakondrashova ~]$ ./simpleid2
e_uid=1002, e_gid=1002
real_uid=1002, real_gid=1002
```

Рис. 2.4: Выполнение программы simpleid2

8. От имени суперпользователя выполнили команды (для этого использовали повышение своих прав с помощью sudo): chown root:guest /home/guest/simpleid2 chmod u+s /home/guest/simpleid2
9. Выполнили проверку правильности установки новых атрибутов и смены владельца файла simpleid2.
10. Запустите simpleid2 и id.

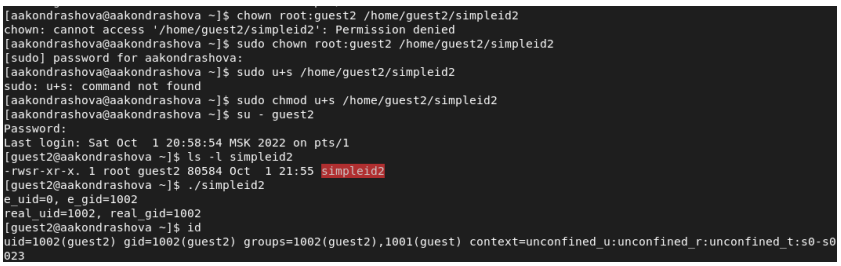


Рис. 2.5: Выполнение программы simpleid2

12. Создали программу readfile.c:
13. Откомпилировали её.

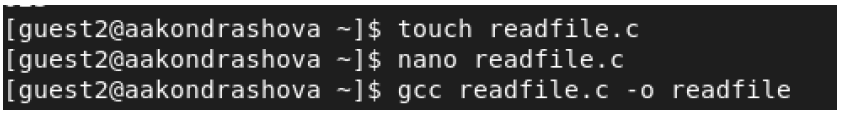


Рис. 2.6: Создание программы readfile.c

Рис. 2.7: Программа readfile.c

14. Сменили владельца у файла readfile.c и изменили права так, чтобы только суперпользователь (root) мог прочитать его, а guest2 не мог.

```
[aakondrashova@aakondrashova ~]$ sudo chmod 700 /home/guest2/readfile.c
[sudo] password for aakondrashova:
Sorry, try again.
[sudo] password for aakondrashova:
[aakondrashova@aakondrashova ~]$ sudo chown root:guest2 /home/guest2/readfile.c
```

Рис. 2.8: Смена прав

15. Проверили, что пользователь guest2 не может прочитать файл readfile.c.

```
[aakondrashova@aakondrashova ~]$ su - guest2
Password:
Last login: Sat Oct  1 21:59:07 MSK 2022 on pts/1
[guest2@aakondrashova ~]$ cat readfile.c
cat: readfile.c: Permission denied
```

Рис. 2.9: Проверка невозможности чтения

16. Сменили у программы readfile владельца и установите SetU'D-бит.

```
[aakondrashova@aakondrashova ~]$ sudo chown root:guest2 /home/guest2/readfile.c
[aakondrashova@aakondrashova ~]$ sudo chown u+s /home/guest2/readfile
chown: invalid user: ‘u+s’
[aakondrashova@aakondrashova ~]$ sudo chown root:guest2 /home/guest2/readfile
[aakondrashova@aakondrashova ~]$ sudo chown u+s /home/guest2/readfile
chown: invalid user: ‘u+s’
[aakondrashova@aakondrashova ~]$ sudo chmod u+s /home/guest2/readfile
```

Рис. 2.10: Смена владельца и установка SetU'D-бит

17. Проверили, что программа readfile может прочитать файл readfile.c

```
[guest2@aakondrashova ~]$ ./readfile readfile.c
#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]);
        }
```

Рис. 2.11: Проверка возможности чтения файла readfile.c

18. Проверьте, что программа readfile может прочитать файл /etc/shadow

Рис. 2.12: Проверка возможности чтения файла /etc/shadow

- Исследование Sticky-бита

1. Выяснили, установлен ли атрибут Sticky на директории /tmp, для чего выполнили команду ls -l / | grep tmp
2. От имени пользователя guest создали файл file01.txt в директории /tmp со словом test.
3. Просмотрели атрибуты у только что созданниого файла и разрешили чтение и запись для категории пользователей «все остальные».

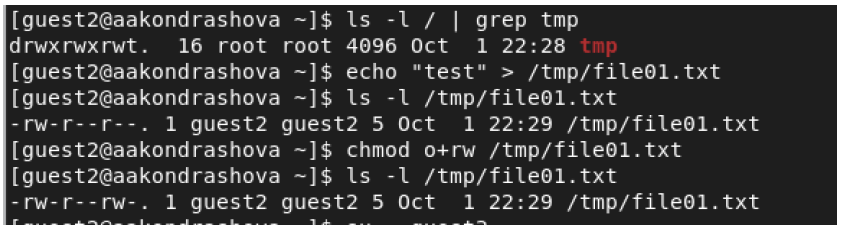


Рис. 2.13: Атрибуты на директории /tmp и файла file01.txt

4. От пользователя guest3 (не являющегося владельцем) попробовали прочитать файл /tmp/file01.txt. Нам это удалось.
5. От пользователя guest3 попробовали дозаписать в файл /tmp/file01.txt слово test2. Слово перезаписалось, а не дозаписалось.

6. От пользователя guest3 попробовали перезаписать в файл /tmp/file01.txt слово test3. Слово перезаписалось.
7. От пользователя guest3 попробовали удалить файл /tmp/file01.txt командой
8. Повысили свои права до суперпользователя и выполнили после этого команду, снимающую атрибут t (Sticky-бит) с директории /tmp.

```
[guest3@aakondrashova ~]$ cat /tmp/file01.txt
test
[guest3@aakondrashova ~]$ echo "test2" > /tmp/file01.txt
[guest3@aakondrashova ~]$ cat /tmp/file01.txt
test2
[guest3@aakondrashova ~]$ echo "test3" > /tmp/file01.txt
[guest3@aakondrashova ~]$ cat /tmp/file01.txt
test3
[guest3@aakondrashova ~]$ rm /tmp/file01.txt
rm: cannot remove '/tmp/file01.txt': Operation not permitted
[guest3@aakondrashova ~]$ su -
Password:
[root@aakondrashova ~]# chmod -t /tmp
[root@aakondrashova ~]# exit
logout
```

Рис. 2.14: Пункты 4-8

9. От пользователя guest3 проверили, что атрибута t у директории /tmp нет.
10. Повторили предыдущие шаги, удалось удалить файл.
11. Повысили свои права до суперпользователя и вернули атрибут t на директорию /tmp:

```
[guest3@aakondrashova ~]$ ls -l / | grep tmp
drwxrwxrwx.  16 root root 4096 Oct  1 22:35 tmp
[guest3@aakondrashova ~]$ cat /tmp/file01.txt
test3
[guest3@aakondrashova ~]$ echo "test2" > /tmp/file01.txt
[guest3@aakondrashova ~]$ cat /tmp/file01.txt
test2
[guest3@aakondrashova ~]$ echo "test3" > /tmp/file01.txt
[guest3@aakondrashova ~]$ cat /tmp/file01.txt
test3
[guest3@aakondrashova ~]$ rm /tmp/file01.txt
[guest3@aakondrashova ~]$ su -
Password:
Last login: Sat Oct  1 22:34:34 MSK 2022 on pts/1
[root@aakondrashova ~]# chmod +t /tmp
[root@aakondrashova ~]# exit
logout
[guest3@aakondrashova ~]$
```

Рис. 2.15: Пункты 9-11

3 Выводы

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