

ОТЧЕТ по лабораторной работе №5

дисциплина: Информационная безопасность

Студент: Васильева Юлия

Группа: НФИбд-03-18

МОСКВА 2021г.

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.

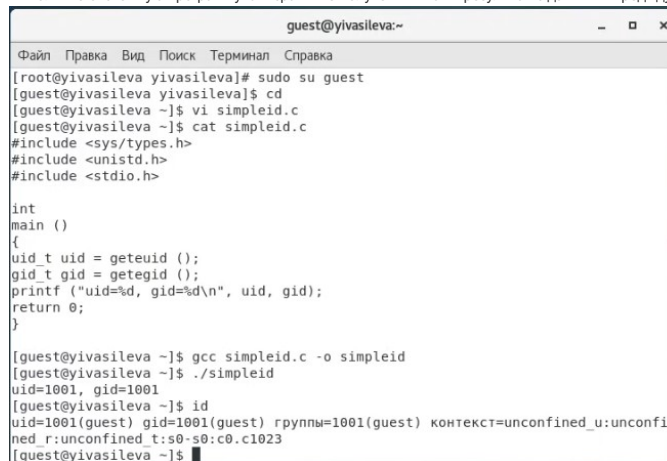
Выполнение лабораторной работы

1. Войдите в систему от имени пользователя guest.
2. Создайте программу 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;
}
```

3. Скомпилируйте программу и убедитесь, что файл программы создан: gcc simpleid.c -o simpleid
4. Выполните программу simpleid: ./simpleid
5. Выполните системную программу id и сравните полученный вами результат с данными предыдущего пункта задания.



```
guest@yivasileva:~
Файл  Правка  Вид  Поиск  Терминал  Справка
[root@yivasileva yivasileva]# sudo su guest
[guest@yivasileva yivasileva]$ cd
[guest@yivasileva ~]$ vi simpleid.c
[guest@yivasileva ~]$ cat 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;
}

[guest@yivasileva ~]$ gcc simpleid.c -o simpleid
[guest@yivasileva ~]$ ./simpleid
uid=1001, gid=1001
[guest@yivasileva ~]$ id
uid=1001(guest) gid=1001(guest) rpyнны=1001(guest) контекст=unconfined_u:unconfi
ned_r:unconfined_t:s0-s0:c0.c1023
[guest@yivasileva ~]$
```

Данные одинаковы.

6. Усложните программу, добавив вывод действительных идентификаторов:

```
include <sys/types.h>
include <unistd.h>
include <stdio.h>

int
main ()
{
    uid_t realuid = getuid ();
    uid_t euid = geteuid ();
    gid_t realgid = getgid ();
    gid_t egid = getegid ();
    printf ("euid=%d, egid=%d\n", euid, egid);
    printf ("realuid=%d, realgid=%d\n", realuid,
    ,→ realgid);
    return 0;
}
```

Получившуюся программу назовите simpleid2.c.

7. Скомпилируйте и запустите simpleid2.c:
gcc simpleid2.c -o simpleid2
./simpleid2
8. От имени суперпользователя выполните команды:

- ```
chown root:guest /home/guest/simpleid2
chmod u+s /home/guest/simpleid2
```
- Используйте `sudo` или повысьте временно свои права с помощью `su`. Поясните, что делают эти команды.
  - Выполните проверку правильности установки новых атрибутов и смены владельца файла `simpleid2`:  
`ls -l simpleid2`
  - Запустите `simpleid2` и `id`:  
`./simpleid2`  
`id`  
Сравните результаты.

```

guest@yivasileva:/home/guest
Файл Правка Вид Поиск Терминал Справка
[guest@yivasileva ~]$ gcc simpleid.c -o simpleid
[guest@yivasileva ~]$ gcc simpleid2.c -o simpleid2
[guest@yivasileva ~]$./simpleid2
e_uid=1001, e_gid=1001
real_uid=1001, real_gid=1001
[guest@yivasileva ~]$ su
Пароль:
su: Сбой при проверке подлинности
[guest@yivasileva ~]$ su
Пароль:
[root@yivasileva guest]# chown root:guest /home/guest/simpleid2
[root@yivasileva guest]# chmod u+s /home/guest/simpleid2
[root@yivasileva guest]# ls -l simpleid2
-rwsrwxr-x. 1 root guest 8512 ноя 13 20:51 simpleid2
[root@yivasileva guest]# ./simpleid2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[root@yivasileva guest]# id
uid=0(root) gid=0(root) группы=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[root@yivasileva guest]# chmod g+s /home/guest/simpleid2
[root@yivasileva guest]# ls -l simpleid2
-rwsrwsr-x. 1 root guest 8512 ноя 13 20:51 simpleid2

```

- Проделайте тоже самое относительно SetGID-бита.  

```

[root@yivasileva guest]# ./simpleid2
e_uid=0, e_gid=1001
real_uid=0, real_gid=1001
[root@yivasileva guest]# id
uid=0(root) gid=0(root) группы=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023

```
- Создайте программу `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 bytesread;
 int i;
 int fd = open (argv[1], O_RDONLY);
 do
 {
 bytesread = read (fd, buffer, sizeof (buffer));
 for (i = 0; i < bytesread; ++i) printf("%c", buffer[i]);
 }
 while (bytesread == sizeof (buffer));
 close (fd);
 return 0;
}

```
- Откомпилируйте её.  
`gcc readfile.c -o readfile`
- Смените владельца у файла `readfile.c` (или любого другого текстового файла в системе) и измените права так, чтобы только суперпользователь (`root`) мог прочитать его, а `guest` не мог.
- Проверьте, что пользователь `guest` не может прочитать файл `readfile.c`.
- Смените у программы `readfile` владельца и установите SetUD-бит.
- Проверьте, может ли программа `readfile` прочитать файл `readfile.c`? Может
- Проверьте, может ли программа `readfile` прочитать файл `/etc/shadow`? Может  

```

[root@yivasileva guest]# sudo chown root:guest readfile
[root@yivasileva guest]# sudo chmod u+s readfile
[root@yivasileva guest]# ./readfile.c
bash: ./readfile.c: Отказано в доступе
[root@yivasileva guest]# su
[root@yivasileva guest]# ./readfile /etc/shadow
root:6E2J2gd100uBd8qEC$giM/Hag6y518LkaI6T0IwGCVQGfP.dol3omBm2JBEKy/Wx5/uSsitP
IicqthqWE4n3/ONSZo50cur9ghKYEr1::0:99999:7:::
bin:*.18353:0:99999:7:::
daemon:*.18353:0:99999:7:::
adm:*.18353:0:99999:7:::
lp:*.18353:0:99999:7:::
sync:*.18353:0:99999:7:::
shutdown:*.18353:0:99999:7:::
halt:*.18353:0:99999:7:::
mail:*.18353:0:99999:7:::
operator:*.18353:0:99999:7:::
games:*.18353:0:99999:7:::
ftp:*.18353:0:99999:7:::
nobody:*.18353:0:99999:7:::
systemd-network:!!:18899:::

```

## 5.3.2. Исследование Sticky-бита

- Выясните, установлен ли атрибут Sticky на директории `/tmp`, для чего выполните команду  
`ls -l / | grep tmp`
- От имени пользователя `guest` создайте файл `file01.txt` в директории `/tmp` со словом `test`:

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

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

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

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

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

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

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

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

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

Удалось ли вам выполнить операцию?

6. Проверьте содержимое файла командой

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

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

Удалось ли вам выполнить операцию? Да

8. Проверьте содержимое файла командой

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

```
guest2@yivasileva:/home/yivasileva -
Файл Правка Вид Поиск Терминал Справка
[yivasileva@yivasileva ~]$ ls -l / | grep tmp
drwxrwxrwt. 15 root root 4096 ноя 13 21:19 tmp
[yivasileva@yivasileva ~]$ su guest
Пароль:
su: Сбой при проверке подлинности
[yivasileva@yivasileva ~]$ su guest
Пароль:
[guest@yivasileva yivasileva]$ echo "test" > /tmp/file01.txt
[guest@yivasileva yivasileva]$ ls -l /tmp/file01.txt
-rw-rw-r--. 1 guest guest 5 ноя 13 21:26 /tmp/file01.txt
[guest@yivasileva yivasileva]$ chmod o+rw /tmp/file01.txt
[guest@yivasileva yivasileva]$ ls -l /tmp/file01.txt
-rw-rw-rw-. 1 guest guest 5 ноя 13 21:26 /tmp/file01.txt
[guest@yivasileva yivasileva]$ su guest2
Пароль:
su: Сбой при проверке подлинности
[guest@yivasileva yivasileva]$ su guest2
Пароль:
[guest2@yivasileva yivasileva]$ cat /tmp/file01.txt
test
[guest2@yivasileva yivasileva]$ echo "test2" > /tmp/file01.txt
[guest2@yivasileva yivasileva]$ cat /tmp/file01.txt
test2
[guest2@yivasileva yivasileva]$ █
```

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

Удалось ли вам удалить файл? Нет

10. Повысьте свои права до суперпользователя следующей командой su -

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

```
chmod -t /tmp
```

11. Покиньте режим суперпользователя командой exit

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

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

13. Повторите предыдущие шаги. Какие наблюдаются изменения?

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

15. Повысьте свои права до суперпользователя и верните атрибут t на директорию /tmp:

```
su -
chmod +t /tmp
exit
[guest2@yivasileva yivasileva]$ rm /tmp/file01.txt
rm: невозможно удалить «/tmp/file01.txt»: Операция не позволена
[guest2@yivasileva yivasileva]$ su -
Пароль:
Последний вход в систему:Сб ноя 13 21:22:13 MSK 2021на pts/1
[root@yivasileva ~]# chmod -t /tmp
[root@yivasileva ~]# exit
logout
[guest2@yivasileva yivasileva]$ ls -l / | grep tmp
drwxrwxrwt. 15 root root 4096 ноя 13 21:31 tmp
[guest2@yivasileva yivasileva]$ echo "test" > /tmp/file01.txt
[guest2@yivasileva yivasileva]$ cat /tmp/file01.txt
test
[guest2@yivasileva yivasileva]$ echo "test2" > /tmp/file01.txt
[guest2@yivasileva yivasileva]$ cat /tmp/file01.txt
test2
[guest2@yivasileva yivasileva]$ rm /tmp/file01.txt
[guest2@yivasileva yivasileva]$ su -
```

## Вывод

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