Отчет по лабораторной работе № 5

По дисциплине Информационная безопасность

Выполнил: Максимов Алексей Александрович

Группа: НПИ-бд-01-20

Российский Университет Дружбы Народов #### г. Москва

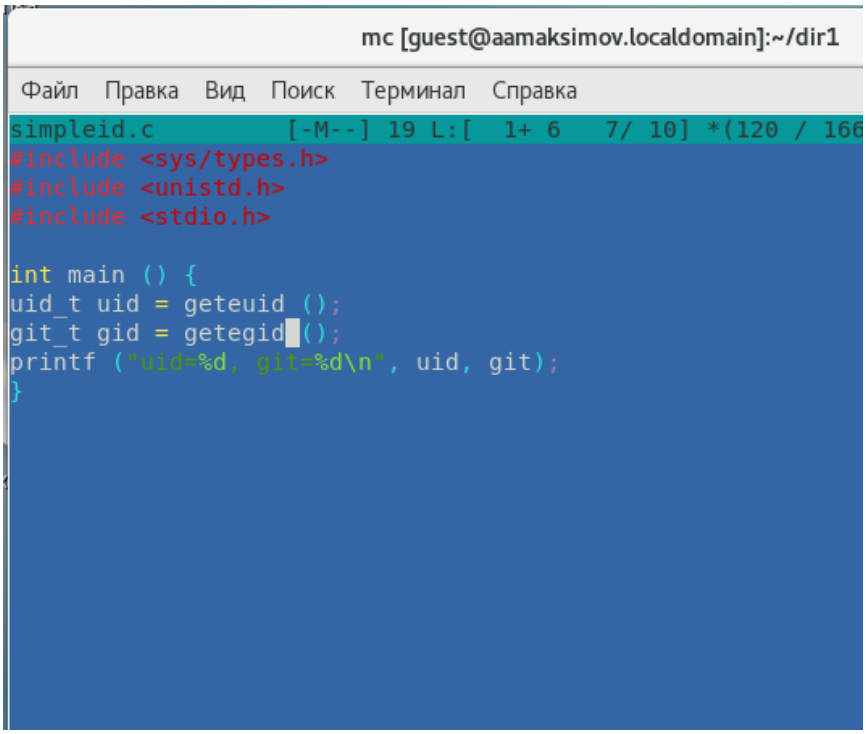
Цель работы

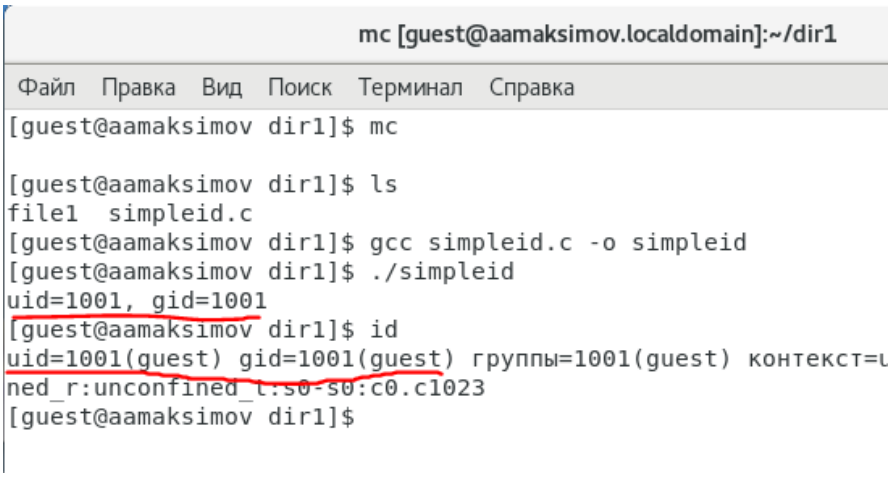
Изучение механизмов изменения идентификаторов, применения SetUID- и Sticky-битов. Получение практических навыков работы в консоли с дополнительными атрибутами. Рассмотрение работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.

Ход работы

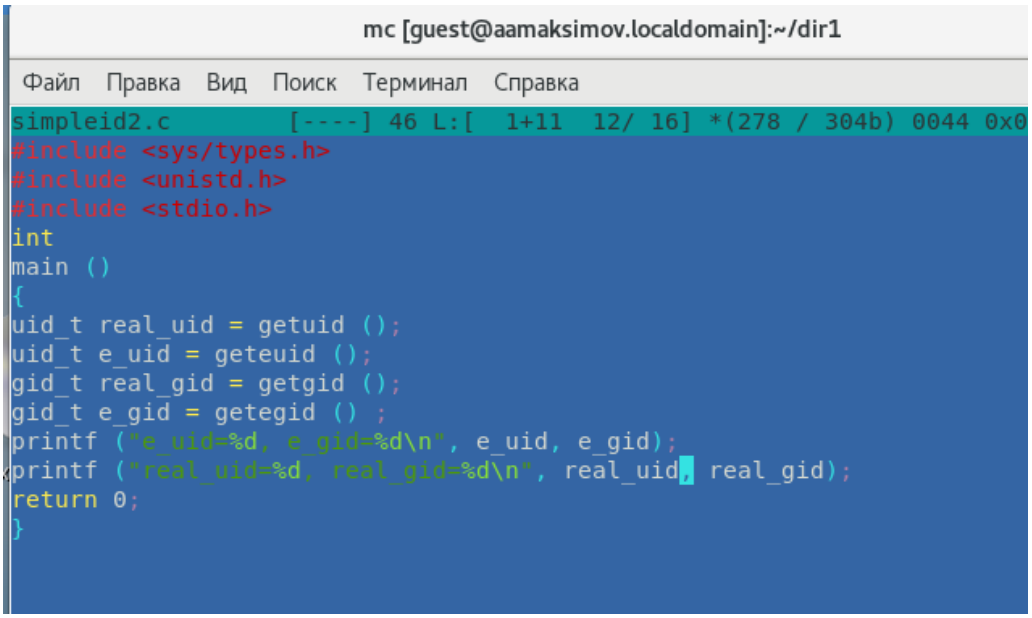
Выполнение лабораторной работы

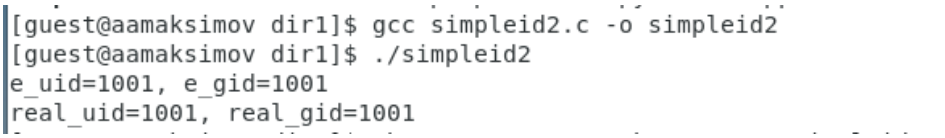
Создаем программу simpleid, компилируем и проверяем верность выведенных данных

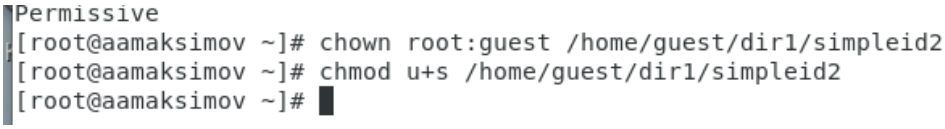




Создаем программу simpleid2, компилируем и проверяем верность выведенных данных, меняем владельца файла



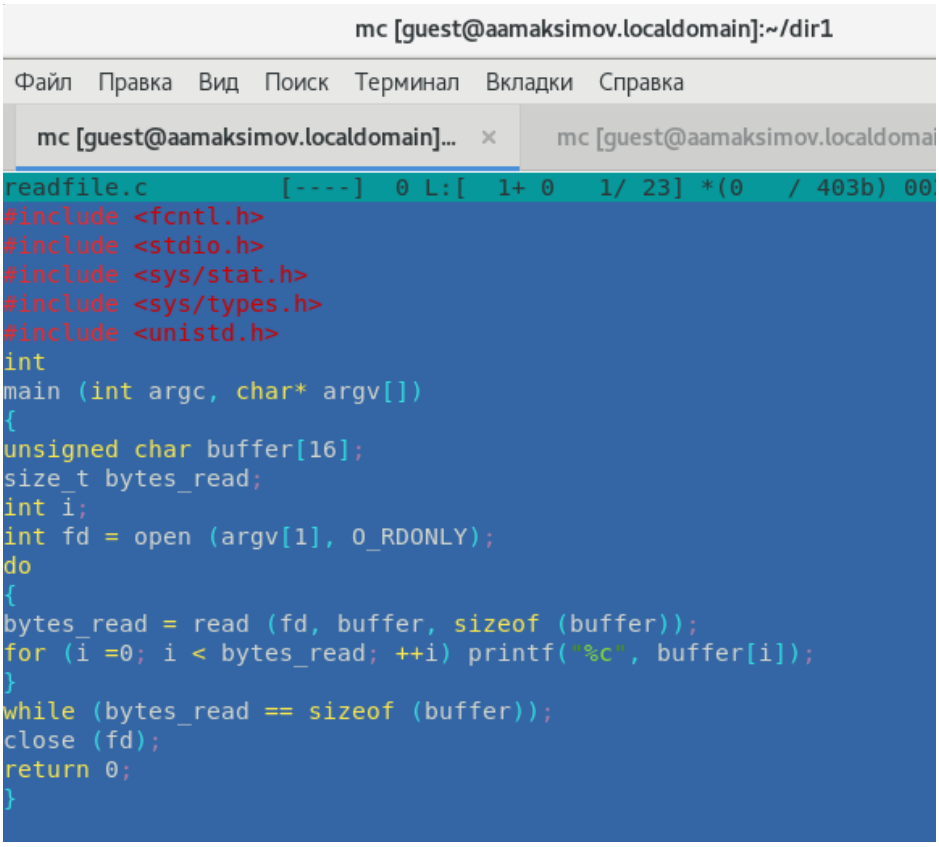




```
[guest@aamaksimov dir1]$ ls -l simpleid2
-rwsrwxr-x. 1 root guest 8616 сен 25 13:54 simpleid2
[guest@aamaksimov dir1]$ ./simpleid2
e_uid=0, e_gid=1001
real_uid=1001, real_gid=1001
[guest@aamaksimov dir1]$ id
uid=1001(guest) gid=1001(guest) группы=1001(guest) контекст=unconfined_u:un
ned_r:unconfined_t:s0-s0:c0.c1023
```

Создаем программу simpleid2, компилируем, меняем владельца файла

возниклас проблема с пониманием синтаксиса активации программы



```
[guest@aamaksimov dir1]$ touch readfile.c
[guest@aamaksimov dir1]$ gcc readfile.c -o readfile
[guest@aamaksimov dir1]$ cat 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]);
}
while (bytes_read == sizeof (buffer));
close (fd);
return 0;
}
```

```
[root@aamaksimov dir1]# chown root:guest /home/guest/dir1/readfile.c
[root@aamaksimov dir1]# chmod u+s /home/guest/dir1/readfile.c
[root@aamaksimov dir1]#
```

Исследование Sticky-бита

проверяем наличие t атрибута на папке tmp, создаем файл в папке от пользователя 1 и пытаемся работать с ним от пользователя 2

```
[guest@aamaksimov dir1]$ ls -l / | grep tmp
drwxrwxrwt.  21 root root 4096 сен 25 14:48 tmp
[guest@aamaksimov dir1]$ echo "test" > /tmp/file01.txt
```

```
[guest2@aamaksimov guest]$ ls -l / | grep tmp
drwxrwxrwt.  21 root root 4096 сен 25 15:06 tmp
[guest2@aamaksimov guest]$ cat /tmp/file01.txt
test
[guest2@aamaksimov guest]$ echo "test3" > /tmp/file01.txt
[guest2@aamaksimov guest]$ cat /tmp/file01.txt
test3
[guest2@aamaksimov guest]$ rm /tmp/file01.txt
rm: невозможно удалить «/tmp/file01.txt»: Операция не позволена
[guest2@aamaksimov guest]$
```

Затем убираем атрибут t и повторяем действия

```
Последний вход в систему:Пн сен 25 15:03:03 MSK 2023на pts/5
[root@aamaksimov ~]# chmod -t /tmp
[root@aamaksimov ~]# exit
logout
```

```
[guest2@aamaksimov guest]$ ls -l / | grep tmp
drwxrwxrwx.  21 root root 4096 сен 25 15:03 tmp
[guest2@aamaksimov guest]$ cat /tmp/file01.txt
test2
[guest2@aamaksimov guest]$ echo "test3" > /tmp/file01.txt
[guest2@aamaksimov guest]$ cat /tmp/file01.txt
test3
[guest2@aamaksimov guest]$ rm /tmp/file01.txt
[guest2@aamaksimov guest]$ cat /tmp/file01.txt
cat: /tmp/file01.txt: Нет такого файла или каталога
[guest2@aamaksimov guest]$ ls -l / | grep tmp
drwxrwxrwx.  21 root root 4096 сен 25 15:04 tmp
```

Выводы

Изучили механизмы изменения идентификаторов. 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.