

# Презентация по лабораторной работе №5

---

КОНЕ СИРИКИ

2023 Sep 29th

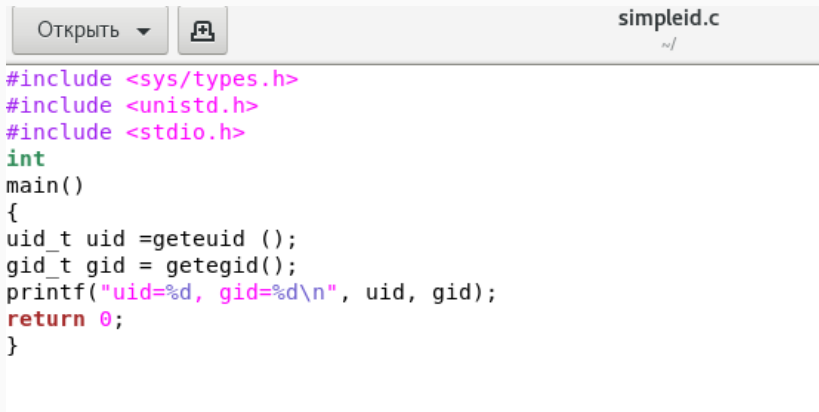
## Цель работы

---

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.

```
[kone@user ~]$ su guest
Пароль:
[guest@user kone]$ setenceforce 0
bash: setenceforce: команда не найдена...
[guest@user kone]$ su
Пароль:
su: Сбой при проверке подлинности
[guest@user kone]$ █
```

Рис. 1: 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;
}
```

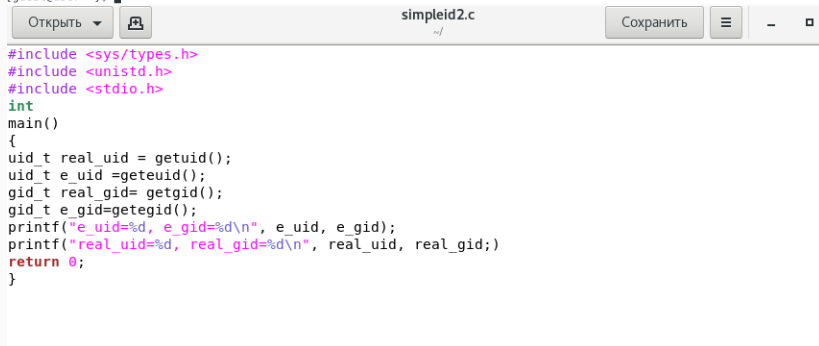
Рис. 2: 2

```
[guest@user ~]$ gcc simpleid.c -o simpleid
[guest@user ~]$ ./simpleid
uid=1001, gid=1001
[guest@user ~]$ id
uid=1001(guest) gid=1001(guest) rpyннм=1001(guest) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@user ~]$ █
```

Рис. 3: 3

## 4 and 5

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



```
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;
}
```

```
[guest@user ~]$ gcc simpleid.c -o simpleid2
[guest@user ~]$ ./simpleid2
uid=1001, gid=1001
[guest@user ~]$ id
uid=1001(guest) gid=1001(guest) rpyrnnw=1001(guest) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@user ~]$
```

Рис. 4: 6





The image shows a code editor window with the title 'readfile.c'. The window has a menu bar with 'Открыть' (Open) and 'Сохранить' (Save) buttons. The code is written in C and includes several system headers. It defines a 'main' function that takes command-line arguments, opens a file, reads its contents into a buffer, and prints the buffer's contents character by character.

```
#include <fcntl.h>
#include <stdio.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <sys/unistd.h>
#include <sys/stdio.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]);
    }
    }
```

```
(gedit:5815): dconf-CRITICAL **: 20:19:13.927: unable to create directory '/run/user/1000/dconf': Отказано в доступе. d
[guest@user ~]$ gcc readfile.c -o readfile
[guest@user ~]$ ./readfile.c
bash: ./readfile.c: Отказано в доступе
[guest@user ~]$ id
uid=1001(guest) gid=1001(guest) группы=1001(guest) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@user ~]$
```

Рис. 6: 8

# Результат выполнения лабораторной работы

```
gedit:5815): dconf-CRITICAL **: 20:19:13.927: unable to create directory '/run/user/1000/dconf': Отказано в доступе. dconf will not work properly.
guest@user ~]$ gcc readfile.c -o readfile
guest@user ~]$ ./readfile.c
ash: ./readfile.c: Отказано в доступе
guest@user ~]$ id
id=1001(guest) gid=1001(guest) rpnw=1001(guest) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
guest@user ~]$ ls -l / | grep tmp
-rwxrwxrwt. 29 root root 4096 окт  5 20:19 tmp
guest@user ~]$ echo "test" >> /tmp/file01.txt
guest@user ~]$ chmod o+rx /tmp/file01.txt
guest@user ~]$ su guest2
apony:
guest2@user guest]$ cat /tmp/file01.txt
test
guest2@user guest]$ echo "test3"> /tmp/file01.txt
ash: /tmp/file01.txt: Отказано в доступе
guest2@user guest]$ echo "test3">> /tmp/file01.txt
ash: /tmp/file01.txt: Отказано в доступе
guest2@user guest]$ echo "test3"> /tmp/file01.txt
ash: /tmp/file01.txt: Отказано в доступе
guest2@user guest]$
```

Рис. 7: 9

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