

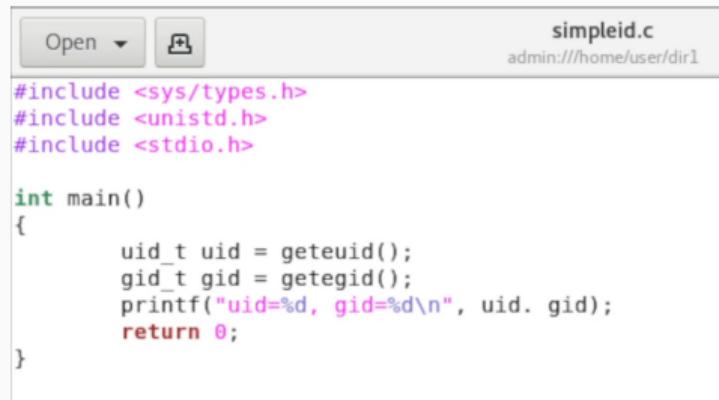
Лабораторная работа № 5

Новосельцев Данила Сергеевич
2023, 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



The screenshot shows a window of an Emacs text editor. The title bar at the top right says "simpleid.c" and "admin:///home/user/dir1". On the left side of the title bar are "Open" and a file icon. The main area contains the following C code:

```
#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. 1: Код программы в редакторе Emacs

Ход работы

```
[root@user dir1]# gcc simpleid.c -o simpleid
[root@user dir1]# ./simpleid
uid=0, gid=0
```

Рис. 2: Успешная компиляция и запуск simpleid. Запуск системной программы id

```
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. 3: Код программы в редакторе Emacs

```
[root@user dir1]# gcc simpleid.c -o simpleid
[root@user dir1]# ./simpleid
e_uid=0, e_gid=0
real_uid=0, real_gid=0
```

Рис. 4: Успешная компиляция

Ход работы

```
[root@user dir1]# chown root:guest simpleid
[root@user dir1]# chmod u+s simpleid
bash: chmod: command not found...
Similar command is: 'kmod'
[root@user dir1]# chmod u+s simpleid
[root@user dir1]# ls -l simpleid
-rwsr-xr-x. 1 root guest 8616 Nov 12 01:34 simpleid
[root@user dir1]# ./simpleid
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[root@user dir1]# id
uid=0(root) gid=0(root) groups=0(root) context=unconfined_u:u
0-s0:c0.c1023
[root@user dir1]# chmod u-s simpleid
[root@user dir1]# chmod g+s simpleid
[root@user dir1]# ls -l simpleid
-rwrxr-xr-x. 1 root guest 8616 Nov 12 01:34 simpleid
[root@user dir1]# ./simpleid
e_uid=0, e_gid=1001
real_uid=0, real_gid=0
[root@user dir1]# id
uid=0(root) gid=0(root) groups=0(root) context=unconfined_u:u
0-s0:c0.c1023
```

Рис. 5: Успешная компиляция с новым владельцем файла

Ход работы

```
#include <fcntl.h>
#include <stdio.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <unistd.h>

int main(int args, 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;
}
```

Рис. 6: Код программы в редакторе Emacs

```
[root@user dir1]# gcc readfile.c -o readfile
[root@user dir1]# chown root readfile.c
[root@user dir1]# chmod 700 readfile.c
[root@user dir1]# si guest
bash: si: command not found...
Similar command is: 'ci'
[root@user dir1]# su guest
[guest@user dir1]$ cat readfile.c
cat: readfile.c: Permission denied
```

Рис. 7: Смена прав и владельца и проверка чтения файла

```
[root@user dir1]# chown root:guest readfile  
[root@user dir1]# chmod u+s readfile  
[root@user dir1]# su guest
```

Рис. 8: Смена прав и владельца

Ход работы

```
[root@user dir1]# ./readfile readfile.c
#include <fcntl.h>
#include <stdio.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <unistd.h>

int main(int args, 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;
}
```

Рис. 9: Проверка функционала файла

Ход работы

Рис. 10: Чтение файла /etc/shadow

```
[guest@user dir1]$ ls -l / | grep tmp  
drwxrwxrwt. 16 root root 4096 Nov 12 08:47 tmp  
[guest@user dir1]$ echo "test" >> /tmp/file01.txt  
[guest@user dir1]$ chmod o+rw /tmp/file01.txt  
chmod: cannot access '/tmp/file01.txt': No such file or direc  
[quest@user dir1]$ chmod o+rw /tmp/file01.txt
```

Рис. 11: Смена прав и владельца

```
[guest@user dir1]$ ls -l /tmp/file01.txt
-rw-r--rw-. 1 root root 10 Nov 12 08:52 /tmp/file01.txt
[guest@user dir1]$ █
```

Рис. 12: проверка атрибутов программы

```
[guest@user dir1]$ su guest2
Password:
[guest2@user dir1]$ cat tmp/file01.txt
cat: tmp/file01.txt: No such file or directory
[guest2@user dir1]$ cat /tmp/file01.txt
test
test
[guest2@user dir1]$ echo "test" /tmp/file01.txt
test /tmp/file01.txt
[guest2@user dir1]$ echo "test" >> /tmp/file01.txt
[guest2@user dir1]$ cat /tmp/file01.txt
test
test
test
[guest2@user dir1]$ echo "test3" > /tmp/file01.txt
[guest2@user dir1]$ cat /tmp/file01.txt
test3
[guest2@user dir1]$ rm /tmp/file01.txt
rm: cannot remove '/tmp/file01.txt': Operation not permitted
```

Рис. 13: Редактирование файла

Ход работы

```
[guest2@user dir1]$ su
Password:
[root@user dir1]# chmod -t /tmp
[root@user dir1]# exit
exit
[guest2@user dir1]$ ls -l / | grep tmp
drwxrwxrwx. 16 root root 4096 Nov 12 09:02 tmp
[guest2@user dir1]$ rm /tmp/file01.txt
[guest2@user dir1]$ su
Password:
[root@user dir1]# chmod +t /tmp
[root@user dir1]# exit
exit
[guest2@user dir1]$ ls -l / | grep tmp
drwxrwxrwt. 17 root root 4096 Nov 12 09:03 tmp
```

Рис. 14: Редактирование файла в режиме суперпользователя

Изучил механизмы изменения идентификаторов, применения SetUID- и Sticky-битов, получил практических навыков работы в консоли с дополнительными атрибутами, а также рассмотрел работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.