

Лабораторная работа 5

Попов Дмитрий Павлович, НФИбд-01-19

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ПРЕЗЕНТАЦИЯ ПО ЛАБОРАТОРНОЙ РАБОТЕ №5

дисциплина: Информационная безопасность

Преподаватель: Кулябов Дмитрий Сергеевич

Студент: Попов Дмитрий Павлович

Группа: НФИбд-01-19

МОСКВА

2022 г.

Прагматика выполнения лабораторной работы

Прагматика выполнения лабораторной работы

- ▶ работа с дополнительными атрибутами

Цель работы

Цель работы

Изучение механизмов изменения идентификаторов, применения SetUID и Sticky-битов.

Выполнение лабораторной работы

1. Создание программы simpleid.c и выполнение

1. Создание программы simpleid.c и выполнение

```
[guest@dpopov ~]$ touch simpleid.c
[guest@dpopov ~]$ nano simpleid.c
[guest@dpopov ~]$ 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@dpopov ~]$ gcc simpleid.c -o simpleid
[guest@dpopov ~]$ ./simpleid
uid=1001, gid=1001
[guest@dpopov ~]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined r:unconfined t:s0-s0:c0.c1023
```

2. Создание программы simpleid2.c и выполнение

2. Создание программы simpleid2.c и выполнение

```
[guest@dpopov ~]$ cat simpleid2.c
#include <sys/types.h>
#include <unistd.h>
#include <stdio.h>

int
main ()
{
    uid_t real_uid = getuid ();
    gid_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@dpopov ~]$ gcc simpleid2.c -o simpleid2
[guest@dpopov ~]$ ./simpleid2
e_uid=1001, e_gid=1001
real_uid=1001, real_gid=1001
```

3. Установка новых атрибутов и запуск simplifiedid2

3. Установка новых атрибутов и запуск simpleid2

```
[guest@dpopov ~]$ su -  
Password:  
[root@dpopov ~]# chown root:guest /home/guest/simpleid2  
[root@dpopov ~]# chmod u+s /home/guest/simpleid2  
[guest@dpopov ~]$ ls -l simpleid2  
-rwsrwxr-x. 1 root guest 26008 Oct  8 18:13 simpleid2  
[guest@dpopov ~]$ ./simpleid2  
e_uid=0, e_gid=1001  
real_uid=1001, real_gid=1001  
[guest@dpopov ~]$ id  
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

4. Создание программы readfile.c

4. Создание программы readfile.c

```
[guest@dpopov ~]$ 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;
}
```

Figure 1: readfile.c

5. Смена владельца readfile

5. Смена владельца readfile

```
[root@dpopov ~]# chown root:guest /home/guest/readfile.c  
[root@dpopov ~]# chmod 700 /home/guest/readfile.c  
[guest@dpopov ~]$ cat readfile.c  
cat: readfile.c: Permission denied
```

6. Смена владельца readfile и установил SetU'D-бит

6. Смена владельца readfile и установил SetU'D-бит

```
[root@dpopov ~]# chown root:guest /home/guest/readfile  
[root@dpopov ~]# chmod u+s /home/guest/readfile
```

```
[guest@dpopov ~]$ ./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
```

```
[guest@dpopov ~]$ ./readfile /etc/shadow
```

```
root:$6$N/bw.rcfZ3wL2m/o$ZIFKfxPpxjiGz5z8dCmko6XasmPGd  
X9HWXXnxBX4nTfp0PcWAECiQLJnx3W1::0:99999:7:::
```

```
bin:*:191123:0:99999:7:::
```

```
daemon:*:10123:0:99999:7:::
```

7. Проверка sticky бит на категории tmp.

7. Проверка sticky бит на категории tmp.

```
[guest@dpopov ~]$ ls -l / | grep tmp  
[guest@dpopov ~]$ ls -l / | grep tmp  
drwxrwxrwt. 17 root root 4096 Oct  8 18:17 tmp  
[guest@dpopov ~]$ echo "test" > /tmp/file01.txt  
[guest@dpopov ~]$ ls -l /tmp/file01.txt  
-rw-rw-r--. 1 guest guest 5 Oct  8 18:26 /tmp/file01.txt
```

Figure 2: sticky

8. Выполнение различных операций от guest2

8. Выполнение различных операций от guest2

```
[guest2@dpopov ~]$ echo "test" > /tmp/file01.txt
[guest2@dpopov ~]$ echo "test2" >> /tmp/file01.txt
[guest2@dpopov ~]$ cat /tmp/file01.txt
test
test2
[guest2@dpopov ~]$ echo "test3" > /tmp/file01.txt
[guest2@dpopov ~]$ cat /tmp/file01.txt
test3
```

Figure 3: guest2 file01

9. Снятие атрибут t (Sticky-бит)
сдиректории /tmp и выполнение
предыдущих шагов

9. Снятие атрибут t (Sticky-бит) с директории /tmp и выполнение предыдущих шагов

```
[guest2@dpopov ~]$ su -
```

```
Password:
```

```
[root@dpopov ~]# chmod -t /tmp
```

```
[root@dpopov ~]# exit
```

```
logout
```

```
[guest2@dpopov ~]$ echo "test2" >> /tmp/file01.txt
```

```
[guest2@dpopov ~]$ cat /tmp/file01.txt
```

```
test3
```

```
test2
```

```
[guest2@dpopov ~]$ rm /tmp/file01.txt
```

```
[guest2@dpopov ~]$ su -
```

Вывод

Вывод

Выполнив данную лабораторную работу, я получил практические навыки работы в консоли с дополнительными атрибутами. Рассмотрел работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.

