

Презентация по лабораторной работе № 5

Информационная безопасность

Адебайо Р. А.

06.10.2023

Российский университет дружбы народов, Москва, Россия

Информация

- Адебайо Ридвануллахи Айофе
- студент группы НКНбд-01-20
- Факультет физико-математических и естественных наук
- Российский университет дружбы народов
- Страничка на GitHub
- Страничка на LinkedIn

Цель лабораторной 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.

Создание программы

Сначала созданы и выполнены две программы, имитирующие команду 'id', для отображения различных идентификаторов пользователя и группы. Затем, с правами суперпользователя, установлены биты SetUID и SetGID для этих программ. После этого, при выполнении программы, они получают соответствующие привилегии суперпользователя и группы. Это демонстрирует, как изменение битов SetUID и SetGID может повлиять на выполнение программ и их привилегии.

```
[guest@raadebayjo ~]$ gcc simpleid.c -o simpleid
[guest@raadebayjo ~]$ ./simpleid
uid=1001, gid=1001
[guest@raadebayjo ~]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s8-s8;c8.c1823
```

Рис. 1: 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;  
}
```

"simpleid.c" 11L, 174B

Создание программы(2)

Сначала создали программу для чтения файла (readfile.c) и скомпилировали её. Затем изменили права доступа к программе так, чтобы только пользователь root мог её читать, а гость - нет. Убедились, что гость не имеет доступа к файлу readfile.c через выполнение программы. Далее сменили владельца программы readfile и установили бит SetUID. После этого с помощью программы удалось прочитать файлы readfile.c и /etc/shadow. Этот процесс иллюстрирует изменение прав доступа и привилегий программы в системе.

```
[guest@raadebayjo ~]$ touch readfile.c
[guest@raadebayjo ~]$ vim readfile.c
[guest@raadebayjo ~]$ vim readfile.c
[guest@raadebayjo ~]$ gcc readfile.c -o readfile
readfile.c: In function 'main':
readfile.c:17:66: error: expected ';' before ')' token
17 |         for(i=0; i<bytes_read; ++i) printf("%c", buffer[i]);
   |                                                             ^
[guest@raadebayjo ~]$ vim readfile.c
[guest@raadebayjo ~]$ gcc readfile.c -o readfile
[guest@raadebayjo ~]$
```

Рис. 3: readfile.c

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

int main(int argc, char* argv[])
{
    unsigned char buffer[10];
    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;
}
```

"readfile.c" 24L, 406B 23,1

Рис. 4: Код

Исследование Sticky-бита

Сначала мы создали файл в каталоге /tmp, разрешив чтение и запись для всех пользователей. Затем, от имени пользователя guest2, мы попытались прочитать, дозаписать и переписать файл. Однако нам не удалось удалить файл.

Затем, суперпользователь снял Sticky-бит с каталога tmp и мы повторили действия с файлом. В этот раз удаление файла стало возможным.

Наконец, суперпользователь вернул Sticky-бит на каталог tmp, обеспечивая тем самым ограниченный доступ к файлам в этом каталоге, даже для суперпользователя. Эти действия демонстрируют влияние Sticky-бита на возможности удаления файлов в каталоге.

```
[guest2@raadebayjo ~]$ ls -l / | grep tmp
drwxrwxrwx. 17 root root 4096 Oct 3 21:21 tmp
[guest2@raadebayjo ~]$ cat /tmp/file01.txt
test3
[guest2@raadebayjo ~]$ echo "test2" > /tmp/file01.txt
[guest2@raadebayjo ~]$ rm /tmp/file01.txt
[guest2@raadebayjo ~]$
```

root@raadebayjo:~

```
drwxrwxrwt. 17 root root 4096 Oct 3 21:04 tmp
[raadebayjo@raadebayjo ~]$ su -
Password:
[root@raadebayjo ~]# chmod -t /tmp
[root@raadebayjo ~]#
```

```
[root@raadebayjo ~]# chmod +t /tmp
[root@raadebayjo ~]# exit
logout
[raadebayjo@raadebayjo ~]$
```

Рис. 6: sticky-bit(2)

Вывод

В ходе выполнения данной лабораторной работы я изучил механизмы изменения идентификаторов, применение SetUID- и Sticky-битов. Получил практические навыки работы в консоли с дополнительными атрибутами. Рассмотрел работу механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.