

Лабораторная работа №5

Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов

Парфенова Е. Е.

1 октября 2024

Российский университет дружбы народов, Москва, Россия

Информация

- Парфенова Елизавета Евгеньвна
- студент
- Российский университет дружбы народов
- 1032216437@pfur.ru
- <https://github.com/parfenovae>



Вводная часть

Важность знания дополнительных атрибутов и их назначения в операционной системе Linux для эффективной и беспроблемной работы с директориями и файлами

Цель: 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.

Задачи:

- Изменение идентификаторов и применение SetUID-, SetGID- и Sticky-битов.
- Проверка выполнения различных операций при разных дополнительных атрибутах

Теоретическое введение

Кроме прав чтения, выполнения и записи, есть еще три дополнительных атрибута.

1. SetUID – это бит разрешения, который позволяет пользователю запускать исполняемый файл с правами владельца этого файла. Другими словами, использование этого бита позволяет нам поднять привилегии пользователя в случае, если это необходимо. Например, права “-rwsr-xr-x”: на месте, где обычно установлен классический бит x (на исполнение), у нас выставлен специальный бит s. Команда, с помощью которой устанавливается этот доп.атрибут: `chmod u+s 'filename'`

2. SetGID - это бит разрешения, который позволяет пользователю запускать исполняемый файл от имени группы, которая владеет файлом. Например, права “-rwxr-sr-x”: на месте, где обычно установлен классический бит x (на исполнение группой), у нас выставлен специальный бит s. Команда, с помощью которой устанавливается этот доп.атрибут:
chmod g+s 'filename'

3. Sticky Bit - специальный бит разрешения, который позволяет только владельцу удалять файлы в папке, на которой этот бит установлен. Пример использования этого бита в операционной системе это системная папка `/tmp`. Эта папка разрешена на запись любому пользователю, но удалять файлы в ней могут только пользователи, являющиеся владельцами этих файлов.

Выполнение лабораторной работы

Подготовка лабораторного стенда

```
[guest@eeparfenova ~]$ su - eeparfenova
Password:
[eeparfenova@eeparfenova ~]$ yum install gcc
Error: This command has to be run with superuser privileges (under the root user on most systems).
[eeparfenova@eeparfenova ~]$ su -
Password:
[root@eeparfenova ~]# yum install gcc
Rocky Linux 9 - BaseOS                2.5 MB/s | 2.3 MB      00:00
Rocky Linux 9 - AppStream             4.7 MB/s | 8.0 MB      00:01
Rocky Linux 9 - Extras                 28 kB/s | 15 kB       00:00
Package gcc-11.4.1-3.el9.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@eeparfenova ~]# setenforce 0
[root@eeparfenova ~]# getenforce
Permissive
[root@eeparfenova ~]#
```

Рис. 1: Подготовка лабораторного стенда



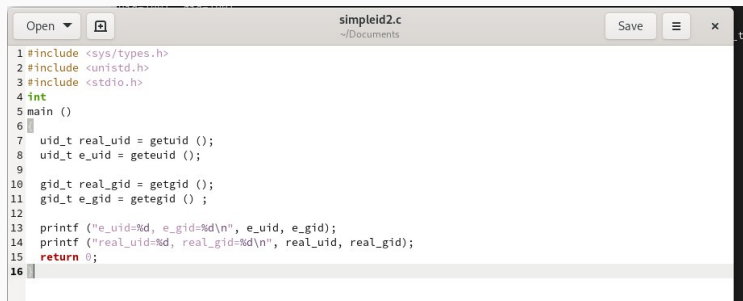
```
1 #include <sys/types.h>
2 #include <unistd.h>
3 #include <stdio.h>
4 int
5 main ()
6 {
7     uid_t uid = geteuid ();
8     gid_t gid = getegid ();
9     printf ("uid=%d, gid=%d\n", uid, gid);
10    return 0;
11 }
```

Рис. 2: Программа simpleid.c

Вывод: вывод на экран при выполнении обеих команд совпадает

```
[guest@eeparfenova Documents]$ gcc simpleid.c -o simpleid
[guest@eeparfenova Documents]$ ./simpleid
uid=1001, gid=1001
[guest@eeparfenova Documents]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@eeparfenova Documents]$
```

Рис. 3: Вывод программы simpleid и команды id

A screenshot of a code editor window titled 'simpleid2.c' with a path of '~/Documents'. The window contains C code for a program that prints effective and real user and group IDs. The code is as follows:

```
1 #include <sys/types.h>
2 #include <unistd.h>
3 #include <stdio.h>
4 int
5 main ()
6 {
7     uid_t real_uid = getuid ();
8     uid_t e_uid = geteuid ();
9
10    gid_t real_gid = getgid ();
11    gid_t e_gid = getegid ();
12
13    printf ("e_uid=%d, e_gid=%d\n", e_uid, e_gid);
14    printf ("real_uid=%d, real_gid=%d\n", real_uid, real_gid);
15    return 0;
16 }
```

Рис. 4: Программа simpleid2.c

Вывод: совпадает с данными в предыдущих случаях

```
[guest@eeparfenova Documents]$ gcc simpleid2.c -o simpleid2
[guest@eeparfenova Documents]$ ./simpleid2
e_uid=1001, e_gid=1001
real_uid=1001, real_gid=1001
[guest@eeparfenova Documents]$
```

Рис. 5: Вывод программы simpleid2

Вывод: результаты вывода программы и команды для суперпользователя одинаковы

```
[guest@deeparfenova Documents]$ su -  
Password:  
[root@deeparfenova ~]# chown root:guest /home/guest/Documents/simpleid2  
[root@deeparfenova ~]# chmod u+s /home/guest/Documents/simpleid2  
[root@deeparfenova ~]# ls -l simpleid2  
ls: cannot access 'simpleid2': No such file or directory  
[root@deeparfenova ~]# cd Doc  
-bash: cd: Doc: No such file or directory  
[root@deeparfenova ~]# ls -l /home/guest/Documents/simpleid2  
-rwsr-xr-x. 1 root guest 24488 Oct  1 13:16 /home/guest/Documents/simpleid2  
[root@deeparfenova ~]# ./simpleid2  
-bash: ./simpleid2: No such file or directory  
[root@deeparfenova ~]# ./home/guest/Documents/simpleid2  
-bash: ./home/guest/Documents/simpleid2: No such file or directory  
[root@deeparfenova ~]# /home/guest/Documents/simpleid2  
e_uid=0, e_gid=0  
real_uid=0, real_gid=0  
[root@deeparfenova ~]# id  
uid=0(root) gid=0(root) groups=0(root) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023  
[root@deeparfenova ~]#
```

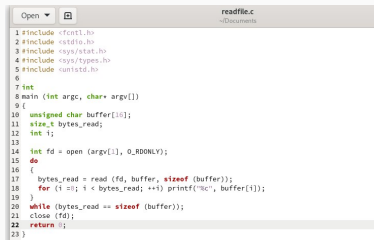
Рис. 6: Смена владельца файла simpleid2 и установка SetUID-бит. Сравнение выводов программы ./simpleid2 и id

Вывод: результаты снова одинаковы

```
[root@eeparfenova ~]# chmod u-s /home/guest/Documents/simpleid2
[root@eeparfenova ~]# chmod g+s /home/guest/Documents/simpleid2
[root@eeparfenova ~]# exit
logout
[guest@eeparfenova Documents]$ ls -l simpleid2
-rwxr-sr-x. 1 root guest 24488 Oct 1 13:16 simpleid2
[guest@eeparfenova Documents]$ ./simpleid2
e_uid=1001, e_gid=1001
real_uid=1001, real_gid=1001
[guest@eeparfenova Documents]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@eeparfenova Documents]$
```

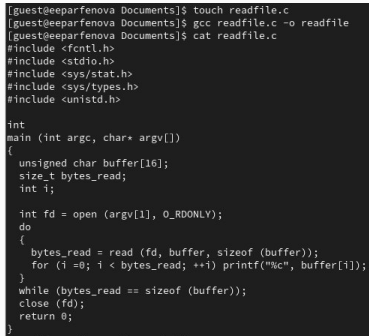
Рис. 7: Манипуляции с установленным SetGID-битом

Программа readfile.c



```
1 #include <fcntl.h>
2 #include <stdio.h>
3 #include <sys/stat.h>
4 #include <sys/types.h>
5 #include <unistd.h>
6
7 int
8 main (int argc, char* argv[])
9 {
10     unsigned char buffer[16];
11     size_t bytes_read;
12     int i;
13
14     int fd = open (argv[1], O_RDONLY);
15     do
16     {
17         bytes_read = read (fd, buffer, sizeof (buffer));
18         for (i = 0; i < bytes_read; ++i) printf("%c", buffer[i]);
19     }
20     while (bytes_read == sizeof (buffer));
21     close (fd);
22     return 0;
23 }
```

Рис. 8: Программа readfile.c



```
[guest@eeparfenova Documents]$ touch readfile.c
[guest@eeparfenova Documents]$ gcc readfile.c -o readfile
[guest@eeparfenova Documents]$ 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;
}
```

Рис. 9: Компиляция и чтение файла readfile.c

Изменение владельца и прав readfile.c

```
[guest@eeparfenova Documents]$ su -
Password:
[root@eeparfenova ~]# chmod u+s /home/guest/Documents/readfile.c
[root@eeparfenova ~]# chmod u-s /home/guest/Documents/readfile.c
[root@eeparfenova ~]# chown root:guest /home/guest/Documents/readfile.c
[root@eeparfenova ~]# chmod 700 /home/guest/Documents/readfile.c
[root@eeparfenova ~]# cat readfile.c
cat: readfile.c: No such file or directory
[root@eeparfenova ~]# cat /home/guest/Documents/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;
}
```

Рис. 10: Смена владельца и прав файла readfile.c

```
[root@eeparfenova ~]# exit
logout
[guest@eeparfenova Documents]$ cat readfile.c
cat: readfile.c: Permission denied
[guest@eeparfenova Documents]$
```

Рис. 11: Попытка чтения файла readfile.c от имени guest

```
[guest@eeparfenova Documents]$ su -  
Password:  
[root@eeparfenova ~]# chown root:guest /home/guest/Documents/readfile  
[root@eeparfenova ~]# chmod u+s /home/guest/Documents/readfile  
[root@eeparfenova ~]# ls -l /home/guest/Documents/readfile  
ls: cannot access '-': No such file or directory  
ls: cannot access 'l': No such file or directory  
/home/guest/Documents/readfile  
[root@eeparfenova ~]# ls -l /home/guest/Documents/readfile  
-rwsr-xr-x. 1 root guest 24432 Oct  1 13:27 /home/guest/Documents/readfile  
[root@eeparfenova ~]#
```

Рис. 12: Смена владельца у программы readfile и установка SetUID-бита

Вывод : readfile имеет все права пользователя root

```
badm: ./readfile: no such file or directory
[root@eeparfenova ~]# exit
logout
[guest@eeparfenova Documents]$ ./readfile readfile.c
#include <fcntl.h>
```

Рис. 13: Чтение файла readfile.c программой readfile

```
[guest@eeparfenova Documents]$ ./readfile /etc/shadow
root:60j6TQ0yPr8NThrrCa1t0Gye8uTSL0q8kPCF/xyGwMRMz8KNE4zRAQq11eoZW5hC9Idst0LkeJF6Zu/4yJ/GU150heg9kASTn0JVOjgD::
0:989898:7:::
btoc+13820:0:99999:7:::
clomoc+15820:0:99999:7:::
cdm+13820:0:99999:7:::
```

Рис. 14: Чтение файла /etc/shadow программой readfile

```
[guest@eeparfenova Documents]$ cd ..  
[guest@eeparfenova ~]$ ls -l / | grep tmp  
drwxrwxrwt. 16 root root 4096 Oct  1 13:36 tmp  
[guest@eeparfenova ~]$ echo "test" > /tmp/file01.txt  
[guest@eeparfenova ~]$ ls -l /tmp/file01.txt  
-rw-r--r--. 1 guest guest 5 Oct  1 13:39 /tmp/file01.txt  
[guest@eeparfenova ~]$ chmod o+rw /tmp/file01.txt  
[guest@eeparfenova ~]$ ls -l /tmp/file01.txt  
-rw-r--rw-. 1 guest guest 5 Oct  1 13:39 /tmp/file01.txt
```

Рис. 15: Права файла file01.txt

Вывод : ни одна из операций кроме чтения недоступна

```
[guest@eeparfenova ~]$ su - guest2
Password:
[guest2@eeparfenova ~]$ cat /tmp/file01.txt
test
[guest2@eeparfenova ~]$ echo "test2" > /tmp/file01.txt
-bash: /tmp/file01.txt: Permission denied
[guest2@eeparfenova ~]$ cat /tmp/file01.txt
test
[guest2@eeparfenova ~]$ echo "test3" > /tmp/file01.txt
-bash: /tmp/file01.txt: Permission denied
[guest2@eeparfenova ~]$ cat /tmp/file01.txt
test
[guest2@eeparfenova ~]$ rm /tmp/file01.txt
rm: remove write-protected regular file '/tmp/file01.txt'? y
rm: cannot remove '/tmp/file01.txt': Operation not permitted
```

Рис. 16: Манипуляции с файлом file01.txt

Вывод : стало доступно удаление файла

```
rm: cannot remove '/tmp/file01.txt': Operation not permitted
[guest2@eeparfenova ~]$ su -
Password:
[root@eeparfenova ~]# chmod -t /tmp
[root@eeparfenova ~]# exit
logout
[guest2@eeparfenova ~]$ ls -l / | grep tmp
drwxrwxrwx. 17 root root 4096 Oct  1 13:43 tmp
[guest2@eeparfenova ~]$ cat /tmp/file01.txt
test
[guest2@eeparfenova ~]$ echo "test2" > /tmp/file01.txt
-bash: /tmp/file01.txt: Permission denied
[guest2@eeparfenova ~]$ echo "test2" >> /tmp/file01.txt
-bash: /tmp/file01.txt: Permission denied
[guest2@eeparfenova ~]$ echo "test3" > /tmp/file01.txt
-bash: /tmp/file01.txt: Permission denied
[guest2@eeparfenova ~]$ cat /tmp/file01.txt
test
[guest2@eeparfenova ~]$ rm /tmp/file01.txt
rm: remove write-protected regular file '/tmp/file01.txt'? y
[guest2@eeparfenova ~]$
```

Рис. 17: Манипуляции с файлом file01.txt без Sticky-бит на директории

```
[guest2@eeparfenova ~]$ su -  
Password:  
[root@eeparfenova ~]# chmod +t /tmp  
[root@eeparfenova ~]# exit  
logout  
[guest2@eeparfenova ~]$
```

Рис. 18: Возвращение атрибута t на директорию tmp

Вывод

В результате выполнения лабораторной работы мы:

- изучили механизм изменения идентификаторов, применения SetUID-, SetGID- и Sticky-битов
- получили практические навыки работы в консоли с дополнительными атрибутами
- рассмотрели работу механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов