

Лабораторная работа №5.

Дискреционноеразграничение прав в Linux. Исследование влияния дополнительных атрибутов

Евдокимов Иван Андреевич. НФИбд-01-20

11 сентября, 2023, Москва, Россия

Российский Университет Дружбы Народов

Цель лабораторной работы

Цель лабораторной работы

Целью данной лабораторной работы является изучить механизмы 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.

Подготовка лабораторного стенда

Установили компилятор gcc. Отключили систему запретов до очередной перезагрузки системы. После этого команда `getenforce` вывел *Permissive*. (@fig:001)

```
[guest2@iven guest]$ su root
Пароль:
[root@iven guest]# yum install gcc
Загружены модули: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
* base: centos-mirror.rbc.ru
* epel: mirror.yandex.ru
* extras: mirror.surf
* updates: mirror.surf
Пакет gcc-4.8.5-44.el7.x86_64 уже установлен, и это последняя версия.
Выполнять нечего
[root@iven guest]#
```

Рис. 1: Установка компилятора gcc

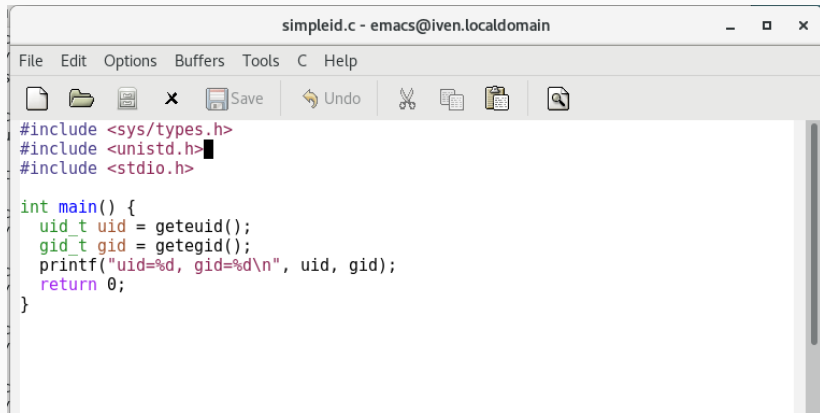
Вошли в систему от имени пользователя guest и создали программу simpleid.c. (@fig:002)

```
[eangreevich@iven ~]$ su guest
Пароль:
[guest@iven eangreevich]$ touch simpleid.c
```

Рис. 2: Программа simpleid.c

Создание программы

Скомпилировали программу и убедились, что файл программы создан, выполнили программу `simpleid`, а затем выполнили системную программу `id`. Обе программы выводят одинаковые значения для `uid` и `gid`. (@fig:003)

The image shows a screenshot of an Emacs editor window. The title bar at the top reads "simpleid.c - emacs@iven.localdomain". Below the title bar is a menu bar with "File", "Edit", "Options", "Buffers", "Tools", "C", and "Help". Underneath the menu bar is a toolbar with icons for file operations (new, open, save, close), editing (undo, redo, cut, copy, paste), and search. The main area of the window contains C code for a program named "simpleid.c". The code includes headers for `<sys/types.h>`, `<unistd.h>`, and `<stdio.h>`. The `main` function calls `geteuid()` and `getegid()` to retrieve the effective user and group IDs, prints them using `printf`, and then returns 0.

```
simpleid.c - emacs@iven.localdomain
File Edit Options Buffers Tools C Help
[Icons: New, Open, Save, Close, Undo, Redo, Cut, Copy, Paste, Find]

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

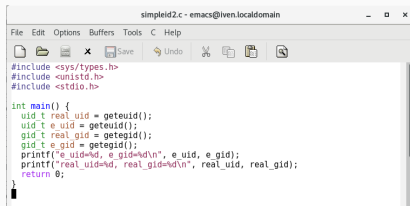
int main() {
    uid_t uid = geteuid();
    gid_t gid = getegid();
    printf("uid=%d, gid=%d\n", uid, gid);
    return 0;
}
```

Усложнили программу, добавив вывод действительных идентификаторов и назвали ее simpleid2.c. (@fig:004)

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

Рис. 4: Программа simpleid2.c

Скомпилировали и запустили simpleid2.c. (@fig:005)



```
simpleid2.c - emacs@iven.localdomain
File Edit Options Buffers Tools C Help
[Icons] Save Undo [Icons]
#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;
}
```

Рис. 5: Выполнение программы simpleid2.c

Повысили временно свои права от имени суперпользователя. Выполнили проверку правильности установки новых атрибутов и смены владельца файла `simpleid2`. Запустили `simpleid2` и `id`. Значения вывода обеих программ совпадают. (@fig:006)

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

Рис. 6: Выполнение программ `simpleid2` и `id`

Создание программы

Проделили тоже самое относительно SetGID-бита. Значения вывода обеих программ совпадают, только в отличие от предыдущего пункта значение `e_gid = 1002`. (@fig:007)

```
[root@iven guest]# chown root:guest /home/guest/simpleid2
[root@iven guest]# chmod u+s /home/guest/simpleid2
[root@iven guest]# ls -l simpleid2
-rwsrwxr-x. 1 root guest 8512 окт 7 17:41 simpleid2
[root@iven guest]# ./simpleid2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[root@iven guest]# id
uid=0(root) gid=0(root) группы=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[root@iven guest]#
```

Рис. 7: Выполнение программ `simpleid2` и `id` относительно SetGID-бита

Создали программу readfile.c. (@fig:008)

```
[root@iven guest]# chmod g+s /home/guest/simpleid2
[root@iven guest]# ls -l simpleid2
-rwsrwsr-x. 1 root guest 8512 окт 7 17:41 simpleid2
[root@iven guest]# ./simpleid2
e_uid=0, e_gid=1001
real_uid=0, real_gid=1001
[root@iven guest]# id
uid=0(root) gid=0(root) группы=0(root) контекст=unconfined_u:unconfined_r:u
nconfined_t:s0-s0:c0.c1023
[root@iven guest]# █
```

Рис. 8: Программа readfile.c

Создание программы

Откомпилировали программу readfile.c. Сменили владельца у файла и изменили права так, чтобы только суперпользователь мог прочитать его. (@fig:009)



```
readfile.c - emacs@iven.localdomain
File Edit Options Buffers Tools C Help
[Icons: New, Open, Save, Undo, Cut, Copy, Paste, Find]
#include <sys/types.h>
#include <sys/stat.h>
#include <stdio.h>
#include <fcntl.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]);
    }
}
```

Проверили, что пользователь guest не может прочитать файл readfile.c.(@fig:010)

```
[guest@iven ~]$ su root
Пароль:
[root@iven guest]# chowd root /home/guest/readfile.c
bash: chowd: команда не найдена...
[root@iven guest]# chown root /home/guest/readfile.c
[root@iven guest]# chmod 700 /home/guest/readfile.c
[root@iven guest]#
```

Рис. 10: Проверка возможности чтения файла readfile.c пользователем guest

Сменили у программы readfile владельца и установили SetU'D-бит. (@fig:011)

```
[root@iven guest]# su guest
[guest@iven ~]$ cat readfile.c
cat: readfile.c: Отказано в доступе
[guest@iven ~]$
```

Рис. 11: Смена у программы readfile владельца и установка SetU'D-бита

Проверим, может ли программа readfile прочитать файл readfile.c. (@fig:012)

```
[guest@iven ~]$ su root
Пароль:
[root@iven guest]# chown root:guest /home/guest/readfile.c
[root@iven guest]# chmod u+s /home/guest/readfile.c
[root@iven guest]#
```

Рис. 12: Чтение программой readfile файла readfile.c

Создание программы

Проверим, может ли программа readfile прочитать файл /etc/shadow. (@fig:013)

```
[root@iven ~]# ./readfile readfile.c
#include <sys/types.h>
#include <sys/stat.h>
#include <stdio.h>
#include <fcntl.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. 13: Чтение программой readfile файла readfile.c

Выяснили, что установлен атрибут Sticky на директории /tmp. От имени пользователя guest создали файл file01.txt. (@fig:014)

```
[root@iven guest]# ls -l |grep tmp
[root@iven guest]# su guest
[guest@iven ~]$ echo "test" > /tmp/file01.txt
[guest@iven ~]$ █
```

Рис. 14: Выполнение команды `ls -l / | grep tmp` и создание файла `file01.txt`

Просмотрели атрибуты у только что созданного файла и разрешили чтение и запись для категории пользователей «все остальные». (@fig:015)

```
[guest@iven ~]$ ls -l /tmp/file01.txt
-rw-rw-r--. 1 guest guest 5 окт  7 18:20 /tmp/file01.txt
[guest@iven ~]$ chmod o+rw /tmp/file01.txt
[guest@iven ~]$ ls -l /tmp/file01.txt
-rw-rw-rw-. 1 guest guest 5 окт  7 18:20 /tmp/file01.txt
[guest@iven ~]$ █
```

Рис. 15: Атрибуты файла file01.txt

От пользователя guest2 попробовали прочитать, дозаписать, записать, удалить файл /tmp/file01.txt. Выполнено все, кроме удаления файла. (@fig:016)

```
[guest@iven ~]$ su guest2
Пароль:
[guest2@iven guest]$ cat /tmp/file01.txt
test
[guest2@iven guest]$ echo "test2" >> /tmp/file01.txt
[guest2@iven guest]$ cat /tmp/file01.txt
test
test2
[guest2@iven guest]$ echo "test3" > /tmp/file01.txt
[guest2@iven guest]$ cat /tmp/file01.txt
test3
[guest2@iven guest]$ rm /tmp/file01.txt
rm: невозможно удалить «/tmp/file01.txt»: Операция не позволена
[guest2@iven guest]$ █
```

Рис. 16: Чтение, дозапись, запись, удаление файл /tmp/file01.txt

Повысили свои права до суперпользователя и сняли атрибут t (Sticky-бит) с директории /tmp. (@fig:017)

```
[guest2@iven guest]$ su -  
Пароль:  
Последний вход в систему:Сб окт  7 18:15:31 MSK 2023на pts/2  
[root@iven ~]# chmod -t /tmp  
[root@iven ~]# exit  
logout
```

Рис. 17: Сняли Sticky-бит с директории /tmp

Повторили предыдущие шаги. В данном случае получилось выполнить удаление файла. (@fig:018)

```
[guest2@iven guest]$ ls -l / | grep tmp  
drwxrwxrwx. 17 root root 4096 окт  7 18:27 tmp  
[guest2@iven guest]$
```

Рис. 18: Чтение, дозапись, запись, удаление файл /tmp/file01.txt без атрибута t

Повысили свои права до суперпользователя и вернули атрибут `t` на директорию `/tmp`. (@fig:019)

```
[guest2@iven guest]$ cat /tmp/file01.txt
test3
[guest2@iven guest]$ echo "test2" >> /tmp/file01.txt
[guest2@iven guest]$ cat /tmp/file01.txt
test3
test2
[guest2@iven guest]$ echo "test3" > /tmp/file01.txt
[guest2@iven guest]$ cat /tmp/file01.txt
test3
[guest2@iven guest]$ rm /tmp/file01.txt
[guest2@iven guest]$
```

Рис. 19: Вернули атрибут `t` на директорию `/tmp`

Выводы:

В ходе выполнения лабораторной работы мы приобрели изучили механизмы изменения идентификаторов, применения SetUID- 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.

1. Кулябов Д. С., Королькова А. В., Геворкян М. Н.
Дискреционное разграничение прав в Linux.
Исследование влияния дополнительных атрибутов
[Текст] / Кулябов Д. С., Королькова А. В., Геворкян М. Н. -
Москва: - 7 с. [^1]: Дискреционное разграничение прав в
Linux. Исследование влияния дополнительных
атрибутов.
2. Справочник 70 основных команд Linux: полное
описание с примерами
(<https://eternalhost.net/blog/sozдание-saytov/osnovnye-komandy-linux>)