

Отчет по лабораторной работе №5

Дерябина Мария

2021

RUDN University, Moscow, Russian Federation

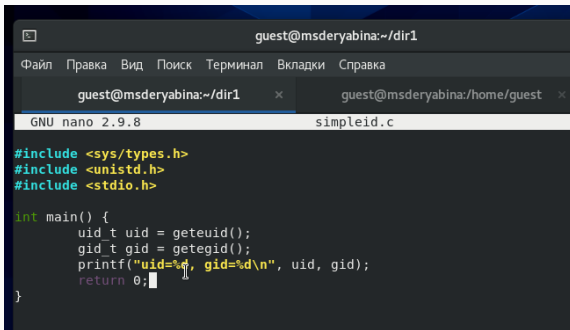
Изучить механизмы изменения идентификаторов, применения SetUID- и Sticky-биты.

Получить практические навыки работы в консоли с дополнительными атрибутами.

Рассмотреть работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.

Выполнение. Исследование SetUID- и SetGID-битов

Вошла в систему от пользователя guest и создала программу simpleid.c



```
guest@msderyabina:~/dir1
Файл  Правка  Вид  Поиск  Терминал  Вкладки  Справка
guest@msderyabina:~/dir1  ×  guest@msderyabina:/home/guest  ×
GNU nano 2.9.8  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;
}
```

Figure 1: Код программы simpleid.c

Выполнение. Исследование SetUID- и SetGID-битов

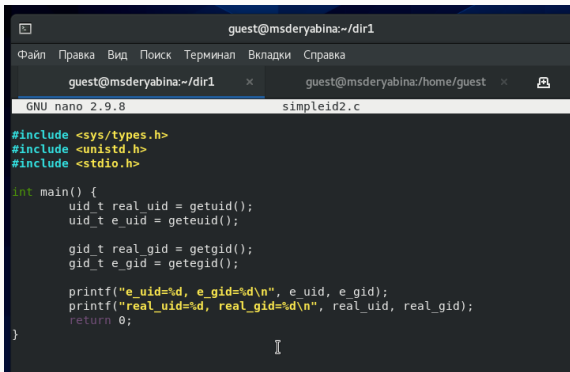
Скомпилировала и выполнила программу. Полученный результат совпал с выводом команды `id`

```
[guest@msderyabina dir1]$ gcc simpleid.c -o simpleid
[guest@msderyabina dir1]$ ./simpleid
uid=1001, gid=1001
[guest@msderyabina dir1]$ id
uid=1001(guest) gid=1001(guest) rгруппы=1001(guest) контекст=unconfined_u:unconfi
ned_r:unconfined_t:s0-s0:c0.c1023
[guest@msderyabina dir1]$
```

Figure 2: Компиляция и выполнение программы `simpleid.c`

Выполнение. Исследование SetUID- и SetGID-битов

Добавила в программу вывод действительных идентификаторов, назвала ее simpleid2.c

A screenshot of a terminal window with a dark background. The window title is 'guest@msderyabina:~/dir1'. The menu bar includes 'Файл', 'Правка', 'Вид', 'Поиск', 'Терминал', 'Вкладки', and 'Справка'. There are two tabs: 'guest@msderyabina:~/dir1' (active) and 'guest@msderyabina:/home/guest'. The editor is GNU nano 2.9.8, editing the file 'simpleid2.c'. The code is as follows:

```
#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;
}
```

Figure 3: Код программы simpleid2.c

Скомпилировала и запустила программу simpleid2.c.

Действительные идентификаторы совпали с эффективными

```
[guest@msderyabina dir1]$ gcc simpleid2.c -o simpleid2
[guest@msderyabina dir1]$ ./simpleid2
e_uid=1001, e_gid=1001
real_uid=1001, real_gid=1001
[guest@msderyabina dir1]$
```

Figure 4: Компиляция и выполнение программы simpleid2.c

Выполнение. Исследование SetUID- и SetGID-битов

От имени суперпользователя изменила владельца программы simpleid2 на root и добавила атрибут SetUID

Проверила правильность установки новых атрибутов и смены владельца файла simpleid2 и запустила simpleid2. Теперь вывод программы отличается от вывода команды id. Действительные идентификаторы остались прежними, а эффективный идентификатор пользователя теперь равен 0 - это идентификатор суперпользователя. Это значит, что пользователь guest использует права суперпользователя во время выполнения программы

```
[guest@msderyabina dir1]$ ls -l simpleid2
-rwsrwxr-x. 1 root guest 17648 ноя 13 13:35 simpleid2
[guest@msderyabina dir1]$ ./simpleid2
e uid=0, e gid=1001
real uid=1001, real gid=1001
[guest@msderyabina dir1]$ id
uid=1001(guest) gid=1001(guest) группы=1001(guest) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023

[guest@msderyabina dir1]$
[guest@msderyabina dir1]$
```

Figure 5: Вывод программы simpleid2 с атрибутом SetUID

Выполнение. Исследование SetUID- и SetGID-битов

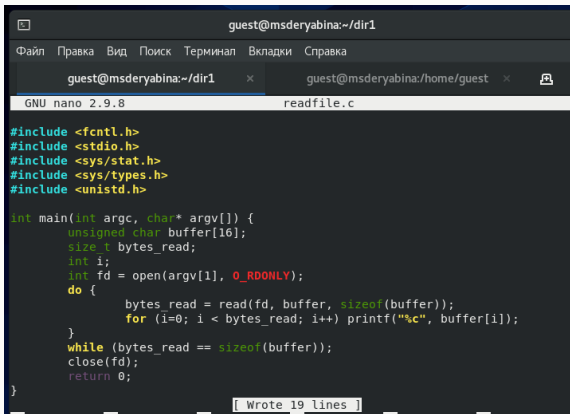
Прodelала то же самое относительно SetGID-бита. Результат оказался аналогичным, теперь при выполнении simpleid2 от пользователя guest эффективный идентификатор группы равен идентификатору группы суперпользователя

```
[guest@msderyabina dir1]$ ls -l simpleid2
-rwsrwsr-x. 1 root root 17648 ноя 13 13:35 simpleid2
[guest@msderyabina dir1]$ ./simpleid2
e_uid=0, e_gid=0
real_uid=1001, real_gid=1001
[guest@msderyabina dir1]$
```

Figure 6: Вывод программы simpleid2 с атрибутом SetUID и SetGID

Выполнение. Исследование SetUID- и SetGID-битов

Создала программу readfile.c



```
guest@msderyabina:~/dir1
Файл  Правка  Вид  Поиск  Терминал  Вкладки  Справка
guest@msderyabina:~/dir1  x  guest@msderyabina:/home/guest  x  [icon]
GNU nano 2.9.8  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;
}
```

[Wrote 19 lines]

Figure 7: Код программы readfile.c

Выполнение. Исследование SetUID- и SetGID-битов

Сменила владельца у файла readfile.c и изменила права так, чтобы только суперпользователь мог прочитать его, а guest не мог

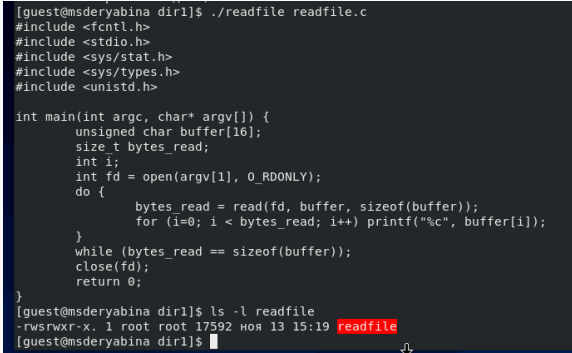
Сменила у программы readfile владельца на root и установила SetUID-бит

```
[root@msderyabina guest]# chown root:root /home/guest/dir1/readfile
[root@msderyabina guest]# chmod u+s /home/guest/dir1/readfile
[root@msderyabina guest]#
```

Figure 8: Добавление SetUID-бита к программе readfile

Выполнение. Исследование SetUID- и SetGID-битов

Теперь с помощью программы readfile можно от имени пользователя guest прочитать файл readfile.c



```
[guest@msderyabina dir1]$ ./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 {
        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;
}
[guest@msderyabina dir1]$ ls -l readfile
-rwsrwxr-x. 1 root root 17592 ноя 13 15:19 readfile
[guest@msderyabina dir1]$
```

Figure 9: Чтение файла readfile.c с помощью readfile

Исследование Sticky-бита

Посмотрела, что на директории /tmp установлен атрибут Sticky. От имени пользователя guest создала файл file01.txt в директории /tmp со словом “test”. Посмотрела атрибуты у file01.txt и разрешила чтение и запись для категории пользователей “other”. От пользователя guest2 попробовала выполнить различные действия - прочитать файл, дозаписать текст в файл, переписать текст в файле, удалить файл. Получилось сделать все, кроме удаления файла

```

[guest2@msderyabina guest]$ cat /tmp/file01.txt
test

[guest2@msderyabina guest]$
[guest2@msderyabina guest]$ echo "test2" > /tmp/file01.txt
[guest2@msderyabina guest]$ cat /tmp/file01.txt
test2
[guest2@msderyabina guest]$ echo "test3" >> /tmp/file01.txt
[guest2@msderyabina guest]$ cat /tmp/file01.txt
test2
test3
[guest2@msderyabina guest]$ rm /tmp/file01.txt
rm: невозможно удалить '/tmp/file01.txt': Операция не позволена
[guest2@msderyabina guest]$
[guest2@msderyabina guest]$
```

Figure 10: Выполнение операций над file01.txt от имени guest2

Исследование Sticky-бита

От имени суперпользователя сняла Sticky-бит с директории /tmp.

Повторила предыдущие шаги. В этот раз удалось удалить file01.txt.

Таким образом, со снятым атрибутом Sticky можно удалить из директории файл от имени пользователя, не являющегося его владельцем

```
[guest2@msderyabina guest]$ ls -l / | grep tmp
drwxrwxrwx. 13 root root 4096 ноя 13 15:48 tmp
[guest2@msderyabina guest]$ cat /tmp/file01.txt
test2
test3
[guest2@msderyabina guest]$ echo "test2" > /tmp/file01.txt
[guest2@msderyabina guest]$ cat /tmp/file01.txt
test2
[guest2@msderyabina guest]$ rm /tmp/file01.txt
[guest2@msderyabina guest]$ ls -l / | grep tmp
drwxrwxrwt. 13 root root 4096 ноя 13 15:56 tmp
[guest2@msderyabina guest]$
```

Figure 11: Выполнение операций над file01.txt со снятым атрибутом Sticky

Я изучила механизмы изменения идентификаторов, применения SetUID- и Sticky-биты.

Получила практические навыки работы в консоли с дополнительными атрибутами.

Рассмотрела работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.