

отчёта по лабораторной работе 5

**Дискреционное разграничение прав в Linux. Исследование влияния
дополнительных атрибутов**

Неустроева Ирина Николаевна

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	13

Список иллюстраций

2.1	Создание программы simpleid.c	6
2.2	Компиляция программы и проверка на создание файла	6
2.3	Выполнение программ simpleid и id	7
2.4	Создание программы simpleid2.c	7
2.5	Запуск программы simpleid2.c	7
2.6	Выполнение команд	8
2.7	Проверка правильности установки атрибутов и смены владельца файла. Запуск simpleid2 и id	8
2.8	Создание программы readfile.c	8
2.9	Откомпилирование программы	9
2.10	Смена владельца у файла readfile.c	9
2.11	Отказ в доступе на чтение	9
2.12	Попытка прочесть файл	9
2.13	Проверка на установку атрибута	9
2.14	Создание файла с текстом внутри	10
2.15	Разрешение в доступе на чтение и запись для категории все остальные	10
2.16	Попытки от пользователя guest2 прочитать, дозаписать и удалить файл	10
2.17	Отказ в доступе	11
2.18	Проверка атрибута	11
2.19	Повторение предыдущих шагов	11
2.20	Повысили свои права до суперпользователя	12

Список таблиц

1 Цель 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

2 Выполнение лабораторной работы

Вошли в систему от имени пользователя guest и создали программу simpleid.c

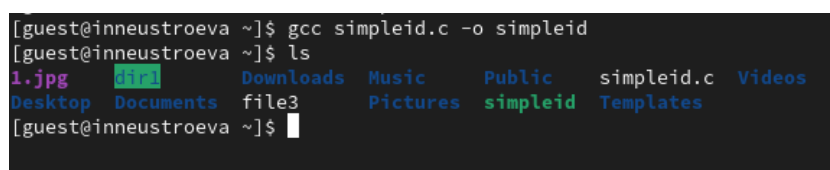


```
guest@inneustroeva:~ — nano simpleid.c
GNU nano 5.6.1 simpleid.c
#include <sys/types.h>
#include <unistd.h>
#include <stdio.h>

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

Рис. 2.1: Создание программы simpleid.c

Скомпилировали программу и убедились, что файл программы создан



```
[guest@inneustroeva ~]$ gcc simpleid.c -o simpleid
[guest@inneustroeva ~]$ ls
1.jpg  dir  Downloads  Music  Public  simpleid.c  Videos
Desktop Documents file3  Pictures  simpleid  Templates
[guest@inneustroeva ~]$
```

Рис. 2.2: Компиляция программы и проверка на создание файла

Выполнили программу simpleid.c и программу id. Вывод программ одинаковый

```

[guest@inneustroeva ~]$ ./simpleid
uid=1001, gid=1001
[guest@inneustroeva ~]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconf
ined_r:unconfined_t:s0-s0:c0.c1023
[guest@inneustroeva ~]$

```

Рис. 2.3: Выполнение программ simpleid и id

Создали новую программу simpleid2.c, добавили вывод действительных идентификаторов

```

GNU nano 5.6.1 simpleid2.c
#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. 2.4: Создание программы simpleid2.c

Скомпилировали и запустили simpleid2.c

```

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

```

Рис. 2.5: Запуск программы simpleid2.c

От имени суперпользователя выполнили команды

```
permis...
[root@inneustroeva ~]# chown root:guest /home/guest/simpleid2
[root@inneustroeva ~]# chmod u+s /home/guest/simpleid2
[root@inneustroeva ~]#
```

Рис. 2.6: Выполнение команд

Выполнили проверку правильности установки новых атрибутов и смены владельца файла simpleid2. Запустили simpleid2 и id

```
[guest@inneustroeva ~]$ ls -l simpleid2
-rwsr-xr-x. 1 root guest 17656 Apr 19 16:11 simpleid2
[guest@inneustroeva ~]$ ./simpleid2
e_uid=0, e_gid=1001
real_uid=1001, real_gid=1001
[guest@inneustroeva ~]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@inneustroeva ~]$
```

Рис. 2.7: Проверка правильности установки атрибутов и смены владельца файла. Запуск simpleid2 и id

Создали программу readfile.c

```
GNU nano 5.6.1                                     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[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. 2.8: Создание программы readfile.c

Откомпилировали программу


```
[guest@inneustroeva ~]$ gcc readfile.c -o readfile
[guest@inneustroeva ~]$
```

Рис. 2.9: Откомпилирование программы

Сменили владельца у файла `readfile.c` и изменили права так, чтобы только суперпользователь (`root`) мог прочитать его, а `guest` не мог

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

Рис. 2.10: Смена владельца у файла readfile.c

Проверили, что пользователь `guest` не может прочитать файл `readfile.c`

[File 'readfile.c' is unwritable]

Рис. 2.11: Отказ в доступе на чтение

Проверили, что программа readfile прочитать файл /etc/shadow не может

[illegible]

Рис. 2.12: Попытка прочесть файл

Выяснили, что атрибут Sticky на директории /tmp установлен

```
[root@inneustroeva ~]# ls -l / | grep tmp
drwxrwxrwt. 21 root root 4096 Apr 19 17:03 tmp
[root@inneustroeva ~]#
```

Рис. 2.13: Проверка на установку атрибута

От имени пользователя guest создали файл file01.txt в директории /tmp со словом test

```
[guest@inneustroeva ~]$ echo "test" > /tmp/file01.txt
[guest@inneustroeva ~]$
```

Рис. 2.14: Создание файла с текстом внутри

Просмотрели атрибуты у только что созданного файла и разрешили доступ на чтение и запись для категории пользователей все остальные

```
[guest@inneustroeva ~]$ ls -l /tmp/file01.txt
-rw-r--r--. 1 guest guest 5 Apr 19 17:07 /tmp/file01.txt
[guest@inneustroeva ~]$ chmod o+rw /tmp/file01.txt
[guest@inneustroeva ~]$ ls -l /tmp/file01.txt
-rw-r--rw-. 1 guest guest 5 Apr 19 17:07 /tmp/file01.txt
[guest@inneustroeva ~]$
```

Рис. 2.15: Разрешение в доступе на чтение и запись для категории все остальные

От пользователя guest2 (не являющегося владельцем) получилось прочитать файл /tmp/file01.txt. Дозаписать в файл /tmp/file01.txt слово test2 не удалось. Не получилось записать в файл /tmp/file01.txt слово test3. Не удалось удалить файл /tmp/file01.txt.

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

Рис. 2.16: Попытки от пользователя guest2 прочитать, дозаписать и удалить файл

Повысили свои права до суперпользователя и сняли атрибут t (Sticky-бит) с директории /tmp. Покинули режим суперпользователя командой

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

Рис. 2.17: Отказ в доступе

От пользователя guest2 проверили, что атрибута t нет у директории /tmp

```
[guest2@inneustroeva ~]$ ls -l / | grep tmp
drwxrwxrwx. 21 root root 4096 Apr 19 17:26 tmp
[guest2@inneustroeva ~]$
```

Рис. 2.18: Проверка атрибута

Повторили предыдущие шаги и выяснили, что можем только прочитать файл и удалить его.

```
[guest2@inneustroeva ~]$ cat /tmp/file01.txt
test
[guest2@inneustroeva ~]$ echo "test2" > /tmp/file01.txt
bash: /tmp/file01.txt: Permission denied
[guest2@inneustroeva ~]$ cat /tmp/file01.txt
test
[guest2@inneustroeva ~]$ echo "test3" > /tmp/file01.txt
bash: /tmp/file01.txt: Permission denied
[guest2@inneustroeva ~]$ cat /tmp/file01.txt
test
[guest2@inneustroeva ~]$ rm /tmp/file01.txt
rm: remove write-protected regular file '/tmp/file01.txt'? y
[guest2@inneustroeva ~]$
```

Рис. 2.19: Повторение предыдущих шагов

Повысили свои права до суперпользователя и вернули атрибут t на директорию /tmp:

```
rm: remove write-protected regular file
[guest2@inneustroeva ~]$ su -
Password:
[root@inneustroeva ~]# chmod +t /tmp
[root@inneustroeva ~]# exit
logout
[guest2@inneustroeva ~]$
```

Рис. 2.20: Повысили свои права до суперпользователя

3 Выводы

Изучили механизм изменения идентификаторов, применения SetUID- и Sticky-битов. Получили практические навыки работы в консоли с дополнительными атрибутами. Рассмотрели работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.