

# Лабораторная работа № 5

Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов

---

Морозов М. Е.

12 сентября 2024

Российский университет дружбы народов, Москва, Россия

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.

В настоящее время sticky bit используется в основном для каталогов, чтобы защитить в них файлы. Из такого каталога пользователь может удалить только те файлы, владельцем которых он является. Примером может служить каталог `/tmp`, в который запись открыта для всех пользователей, но нежелательно удаление чужих файлов. Установка атрибута производится утилитой `chmod`.

## Выполнение лабораторной работы

---

Создали файл `simple.id` и записали в него код из лабораторной. После запуска получили `uid` и `gid` нашего пользователя. Усложним скрипт, добавив вывод `real uid` и `gid`. Теперь выводятся и `real uid` и `gid`, все совпадает с результатами предыдущих шагов. Пропишем `chown` и `chmod`. `chown` изменяет владельца файла, а `chmod u+s` позволяет запускать файл с правами владельца. Теперь при запуске файла от имени `guest` получаем `e_uid root`

Проделаем то же самое с SetGID-битом. Вывод такой же. Создадим файл `readfile.c` как в лабораторной и скомпилируем его. Меняем владельца на `root` и забираем все права у всех кроме владельца. Проверяем, `guest` не может прочесть `readfile.c`. Попробуем прочитать `readfile.c` с помощью `readfile`.

# Выполнение лабораторной работы

## Результаты работы 1 (рис. 1)

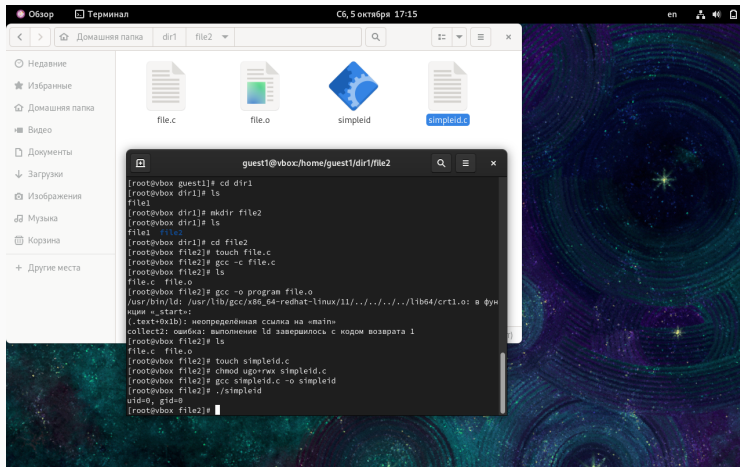
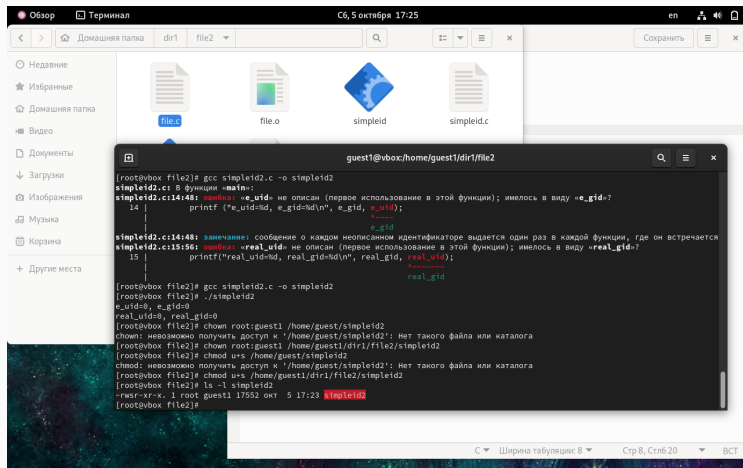


Figure 1: Результат 1

# Выполнение лабораторной работы

## Результаты работы 2 (рис. 2)



```
Обзор Терминал Сб, 5 октября 17:25 en
< > Домашняя папка dir1 file2
Недавние
Избранные
Домашняя папка
Видео
Документы
Загрузки
Изображения
Музыка
Корзина
Другие места

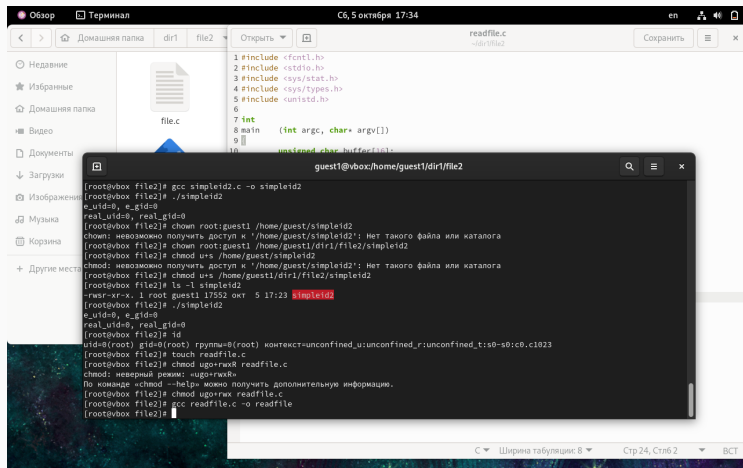
file.c file.o simpleid simpleid.c

guest1@vbox:/home/guest1/dir1/file2
[root@vbox file2]# gcc simpleid2.c -o simpleid2
simpleid2.c: в функции «main»:
simpleid2.c:14:48: ошибка: «e_uid» не описан (первое использование в этой функции); имелось в виду «e_gid»?
14 |         printf ("e_uid=%d, e_gid=%d\n", e_gid, e_uid);
    |                                                ^~~~~
simpleid2.c:14:48: замечание: сообщение о каждом неопisanном идентификаторе выдается один раз в каждой функции, где он встречается
simpleid2.c:15:56: ошибка: «real_uid» не описан (первое использование в этой функции); имелось в виду «real_gid»?
15 |         printf("real_uid=%d, real_gid=%d\n", real_gid, real_uid);
    |                                                        ^~~~~~
[root@vbox file2]# gcc simpleid2.c -o simpleid2
[root@vbox file2]# ./simpleid2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[root@vbox file2]# chown root:guest1 /home/guest1/simpleid2
chown: невозможно получить доступ к '/home/guest1/simpleid2': Нет такого файла или каталога
[root@vbox file2]# chown root:guest1 /home/guest1/dir1/file2/simpleid2
[root@vbox file2]# chmod u+s /home/guest1/simpleid2
chmod: невозможно получить доступ к '/home/guest1/simpleid2': Нет такого файла или каталога
[root@vbox file2]# chmod u+s /home/guest1/dir1/file2/simpleid2
[root@vbox file2]# ls -l simpleid2
-rwsr-xr-x. 1 root guest1 17552 окт  5 17:23 simpleid2
[root@vbox file2]#
```

Figure 2: Результат 2



## Результаты работы 3 (рис. 3)



The screenshot shows a file manager window with a sidebar on the left containing icons for 'Недавние', 'Избранные', 'Домашняя папка', 'Видео', 'Документы', 'Загрузки', 'Изображения', 'Музыка', 'Корзина', and 'Другие места'. The main pane shows a directory structure with 'file1' and 'file2'. A file named 'file.c' is highlighted. To the right, the code of 'readfile.c' is displayed, showing a C program that includes `<fcntl.h>`, `<stdio.h>`, `<sys/stat.h>`, `<sys/types.h>`, and `<unistd.h>`. It defines a `main` function that takes `argc` and `argv` and contains a `printf` statement: `printf("unconfined_char buffer[361]");`.

Overlaid on the file manager is a terminal window titled 'guest1@vbox:/home/guest1/dir1/file2'. The terminal shows the following commands and output:

```
[root@vbox file2]# gcc simpleid2.c -o simpleid2
[root@vbox file2]# ./simpleid2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[root@vbox file2]# chown root:guest1 /home/guest/simpleid2
chown: невозможно получить доступ к '/home/guest/simpleid2': Нет такого файла или каталога
[root@vbox file2]# chown root:guest1 /home/guest1/dir1/file2/simpleid2
[root@vbox file2]# chmod u+s /home/guest/simpleid2
chmod: невозможно получить доступ к '/home/guest/simpleid2': Нет такого файла или каталога
[root@vbox file2]# chmod u+s /home/guest1/dir1/file2/simpleid2
[root@vbox file2]# ls -l simpleid2
-rwxr-xr-x. 1 root guest1 17552 окт 5 17:23 simpleid2
[root@vbox file2]# ./simpleid2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[root@vbox file2]# id
uid=0(root) gid=0(root) tty=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[root@vbox file2]# touch readfile.c
[root@vbox file2]# chmod ugo+rwx readfile.c
chmod: неверный режим: «ugo+rwxR»
По команде «chmod --help» можно получить дополнительную информацию.
[root@vbox file2]# chmod ugo+rwx readfile.c
[root@vbox file2]# gcc readfile.c -o readfile
[root@vbox file2]#
```

Figure 3: Результат 3

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. Stickybit [Электронный ресурс]. Wikimedia Foundation, Inc., 2024. URL: [https://ru.wikipedia.org/wiki/Sticky\\_bit](https://ru.wikipedia.org/wiki/Sticky_bit).