

# Лабораторная работа №2

Дискреционное разграничение прав в Linux. Основные атрибуты

---

Буллет Т. А.

14 февраля 2025

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Буллер Татьяна Александровна
- студент направления Бизнес-информатика
- Российский университет дружбы народов

## Вводная часть

---

- Операционная система linux, дисрибутив Rocky
- Среда виртуализации VirtualBox

- Получение практических навыков работы в консоли с атрибутами файлов,
- закрепление теоретических основ дискреционного разграничения доступа в современных системах с открытым кодом на базе ОС Linux


- Процессор **pandoc** для входного формата Markdown
- Среда виртуализации VirtualBox

## Выполнение лабораторной работы

---



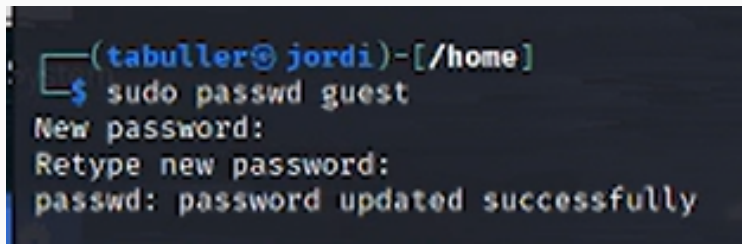
Для создания пользователей используем команду `useradd` с опцией `-m`, чтобы вместе с пользователем создать его домашнюю директорию и в дальнейшем иметь возможность зайти на рабочий стол под этой учетной записью.

A terminal window with a dark background. The prompt is `(tabuller@jordi)-[/home]` in blue and white. Below the prompt, the command `$ sudo useradd -m guest` is entered in white text. A green cursor is visible at the end of the command.

```
(tabuller@jordi)-[/home]  
$ sudo useradd -m guest
```

Рис. 1: Создание пользователя

Далее задаем пароль для пользователя, который необходимо подтвердить, введя повторно.

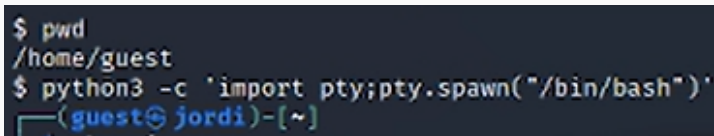
A terminal window with a dark background. The prompt is `(tabuller@jordi)-[/home]`. The user enters `$ sudo passwd guest`. The terminal then displays `New password:`, `Retype new password:`, and finally `passwd: password updated successfully`.

```
(tabuller@jordi)-[/home]
$ sudo passwd guest
New password:
Retype new password:
passwd: password updated successfully
```

Рис. 2: Пароль для пользователя

## Вход в систему от имени созданного пользователя

С помощью команды `su guest` получаем неинтерактивную оболочку пользователя и переводим ее в интерактивную: в таком виде с ней удобнее работать. Команда `pwd` помогает определить, что в данный момент мы находимся в домашней директории созданного ранее пользователя.



```
$ pwd
/home/guest
$ python3 -c 'import pty;pty.spawn("/bin/bash")'
(guest@jordi)-[~]
```

Рис. 3: Вход в систему и рабочая директория

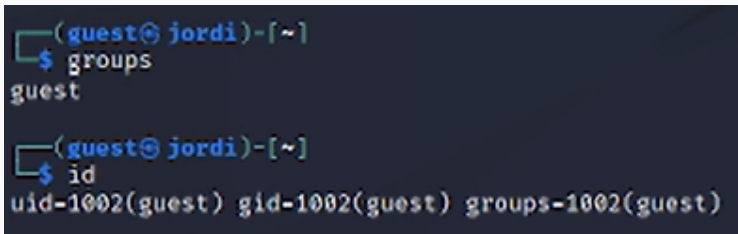
Команда `whoami` выводит в терминал имя пользователя, от имени которого она была запущена. В нашем случае - `guest`.

A terminal window with a dark background. The prompt is `(guest@jordi)=[~]` in blue. Below it, the command `$ whoami` is entered in blue. The output `guest` is displayed in white below the command.

```
(guest@jordi)=[~]  
$ whoami  
guest
```

Рис. 4: `whoami`

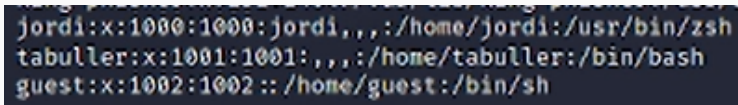
Узнать, в каких группах состоит пользователь, можно двумя способами: командой `id` (выводит `id` пользователя и его имя, `id` группы пользователя и ее названия, а также `id` и названия прочих групп, куда входит данный пользователь) либо командой `groups`. Последняя, в отличие от первой, выводит на экран только названия групп.



```
(guest@jordi)-[~]  
$ groups  
guest  
  
(guest@jordi)-[~]  
$ id  
uid-1002(guest) gid-1002(guest) groups-1002(guest)
```

Рис. 5: Группы пользователя

Запросив содержимое файла `/etc/passwd` командой `cat`, мы можем видеть всех пользователей машины, их `uid` и `gid`. Для созданного пользователя `uid` и `gid` совпадают: 1002.



```
jordi:x:1000:1000:jordi,,,:/home/jordi:/usr/bin/zsh
tabuller:x:1001:1001:,,,:/home/tabuller:/bin/bash
guest:x:1002:1002::/home/guest:/bin/sh
```

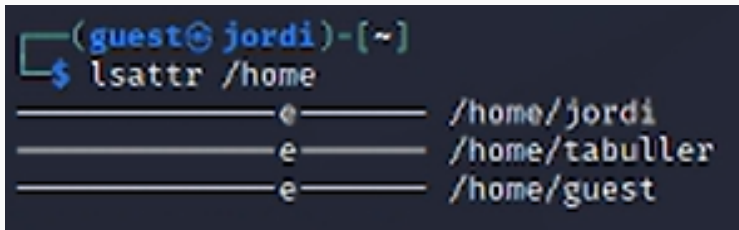
Рис. 6: Содержимое файла `/etc/passwd`

Командой `ls -l /home/` определим существующие в системе домашние директории пользователей. Видим, что их три: `guest`, созданный по ходу выполнения лабораторной работы, `tabuller` - основной пользователь для выполнения лабораторных работ и `jordi`. Кто он и зачем существует - дело десятое. `jordi`. В каждой из директорий владелец имеет право на чтение, запись и исполнение. В директориях `tabuller` и `jordi` право на чтение и исполнение файлов также имеют участники группы и остальные пользователи машины.

```
(guest@jordi)-[~]  
$ ls -l /home/  
total 12  
drwx----- 15 guest      guest      4096 Feb 14 05:26 guest  
drwxr-xr-x 20 jordi      jordi      4096 Feb 14 03:32 jordi  
drwxr-xr-x 19 tabuller   tabuller   4096 Feb 14 05:26 tabuller
```

Рис. 7: Домашние директории пользователей

С помощью lsattr можем увидеть, что у каждой из домашних директорий установлен атрибут е.



```
(guest@jordi)-[~]  
$ lsattr /home  
____e_____ /home/jordi  
____e_____ /home/tabuller  
____e_____ /home/guest
```

Рис. 8: Атрибуты директорий

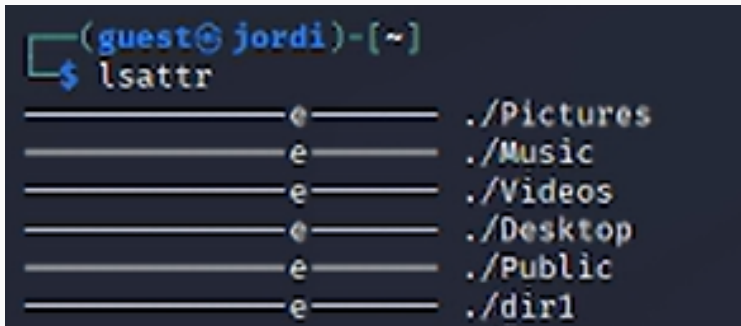


## Права директорий и файлов

В домашней директории пользователя guest создадим директорию dir1. Можно увидеть, что сразу после создания директория выдает все права владельцу и права на чтение и исполнение членам группы и прочим пользователям.

```
(guest@jordi)-[~]  
$ mkdir dir1  
  
(guest@jordi)-[~]  
$ ls -l dir1/  
total 0  
  
(guest@jordi)-[~]  
$ ls -l dir1  
total 0  
  
(guest@jordi)-[~]  
$ ls -l  
total 36  
drwxr-xr-x 2 guest guest 4096 Feb 14 05:26 Desktop  
drwxr-xr-x 2 guest guest 4096 Feb 14 05:30 dir1  
drwxr-xr-x 2 guest guest 4096 Feb 14 05:26 Documents
```

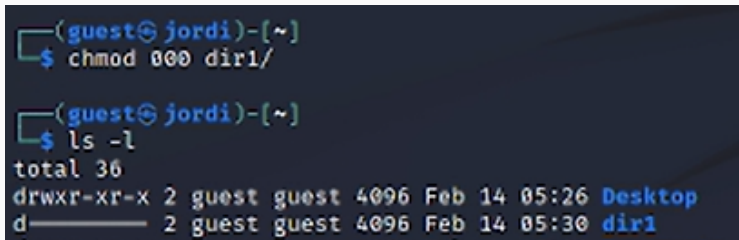
По умолчанию также у всех директорий внутри домашней установлен атрибут `e`: то же самое, что мы видели и в директории `/home`

A terminal window with a dark background. The prompt is `(guest@jordi)-[~]`. The command `$ lsattr` has been entered. The output shows six lines, each representing a directory in the user's home folder. Each line consists of a permissions string (all containing `e`), a link count, and the directory name.

```
(guest@jordi)-[~]  
$ lsattr  
-----e----- ./Pictures  
-----e----- ./Music  
-----e----- ./Videos  
-----e----- ./Desktop  
-----e----- ./Public  
-----e----- ./dir1
```

Рис. 10: Атрибуты директорий внутри домашнего каталога пользователя

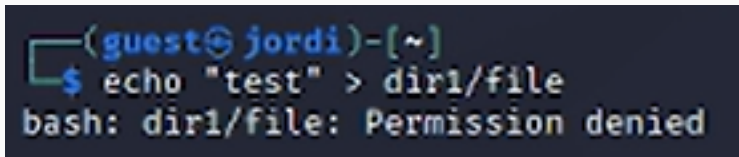
Снимем все права с созданной нами директории. Можем видеть, что вывод команды `ls -l` теперь показывает только то, что перед нами директория. Никто не имеет права ни на чтение, ни на запись, ни на исполнение файлов. Владелец в том числе.

A terminal window with a dark background. The prompt is `(guest@jordi)-[~]`. The user enters `$ chmod 000 dir1/`. The prompt is `(guest@jordi)-[~]`. The user enters `$ ls -l`. The output shows the total size of 36, followed by two lines: `drwxr-xr-x 2 guest guest 4096 Feb 14 05:26 Desktop` and `d----- 2 guest guest 4096 Feb 14 05:30 dir1`.

```
(guest@jordi)-[~]  
$ chmod 000 dir1/  
  
(guest@jordi)-[~]  
$ ls -l  
total 36  
drwxr-xr-x 2 guest guest 4096 Feb 14 05:26 Desktop  
d----- 2 guest guest 4096 Feb 14 05:30 dir1
```

Рис. 11: У гостей нет прав

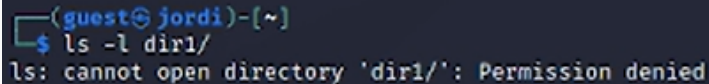
При попытке создать файл внутри директории и перенаправить в него вывод команды echo сталкиваемся с ошибкой: на эту директорию у пользователя больше нет прав.

A terminal window with a dark background. The prompt is `(guest@jordi)-[~]`. The user enters the command `$ echo "test" > dir1/file`. The output is `bash: dir1/file: Permission denied`.

```
(guest@jordi)-[~]  
$ echo "test" > dir1/file  
bash: dir1/file: Permission denied
```

Рис. 12: У гостей нет прав создать файл

При попытке прочитать содержимое директории тоже ничего не увидим: права на чтение сняты вместе со всеми остальными.

A terminal window with a dark background. The prompt is `(guest@jordi)-[~]`. The user enters `$ ls -l dir1/`. The output is `ls: cannot open directory 'dir1/': Permission denied`.

```
(guest@jordi)-[~]  
$ ls -l dir1/  
ls: cannot open directory 'dir1/': Permission denied
```

Рис. 13: У гостей нет прав читать директорию

На основе полученной информации была заполнена таблица о правах доступа владельца.

## Выводы

---

Получены практические навыки работы в консоли с атрибутами файлов, закреплены теоретические основы дискреционного разграничения доступа в современных системах с открытым кодом на базе ОС Linux