

# Day2\_Materials\_RU

## 1. Общие сведения

**Date:** 20.08.2025

**Тема:** Работа с файлами и разрешениями.

**Цель дня:** Закрепить базовые команды Linux, научиться работать с файлами и папками, изучить права доступа, подготовить первый автоматизационный скрипт.

## 2. Пройденный материал

Команда	Описание
<code>ls</code>	Показывает содержимое каталога.
<code>cd</code>	Меняет рабочий каталог на уровень выше.
<code>pwd</code>	Показывает абсолютный путь к текущему каталогу.
<code>tree</code>	Показывает структуру каталогов.
<code>stat</code>	Отображает размер, права и даты изменения файла.

## 3. Повтор + Новые команды

Сегодня добавляем:

- `ls` ( `ls -la` ) — Показывает содержимое каталога.

```
leprecha@Ubuntu-DevOps:~$ ls -la
-rw----- 1 leprecha sysadmin 8217 Aug 20 17:07 .bash_history
```

Показывает содержимое каталога.

- 1.) `l` — детальный список.
- 2.) `a` — показывает скрытые файлы.

- `cd` — переход между каталогами.

```
leprecha@Ubuntu-DevOps:~$ cd /etc
leprecha@Ubuntu-DevOps:/etc$
```

- `pwd` — вывод текущего пути.

```
leprecha@Ubuntu-DevOps:~$ pwd
/home/leprecha
```

- `tree` — вывод структуры каталогов.

```
tree /etc | head -n 5
```

```
leprecha@Ubuntu-DevOps:~$ tree /etc | head -n 5
/etc
├── adduser.conf
├── alsa
│   └── conf.d
│       └── 50-pipewire.conf → /usr/share/alsa/alsa.conf.d/50-pipewire.conf
```

Показывает структуру каталогов (первые 5 строк).

- `stat` — подробная информация о файле.

```
stat /etc/passwd
```

```
leprecha@Ubuntu-DevOps:~$ stat /etc/passwd
File: /etc/passwd
Size: 2959    Blocks: 8      IO Block: 4096   regular file
```

```
Device: 259,2 Inode: 5507826 Links: 1
Access: (0644/-rw-r--r--) Uid: ( 0/ root) Gid: ( 0/ root)
Access: 2025-08-20 16:58:14.464296261 +0100
Modify: 2025-08-19 15:51:08.037927698 +0100
Change: 2025-08-19 15:51:08.038485928 +0100
Birth: 2025-08-19 15:51:08.037485933 +0100
```

## 4. Работа с nano

Редактор nano — простой консольный текстовый редактор в Linux. Подходит для создания и правки конфигурационных и текстовых файлов.

`nano filename.txt` — Открывает файл, если он существует, создаёт новый, если нет.

Клавиши	Действие
Ctrl + O	Сохранить файл
Ctrl + X	Выйти из nano
Ctrl + G	Показать справку
Ctrl + W	Поиск текста
Ctrl + K	Вырезать текущую строку
Ctrl + U	Вставить вырезанный текст
Ctrl + C	Показать позицию курсора
Ctrl + _	Перейти к строке/колодке
Alt + ,	Switch to previous file
Alt + .	Switch to next file

## Практика

1. Создай файл `filename.txt` в домашней папке.
2. Напиши 2–3 предложения о себе на английском.
3. Сохрани изменения и выйди из nano.
4. Скопируй файл в `/tmp`.

5. Проверь содержимое скопированного файла с помощью `cat`.

```
leprecha@Ubuntu-DevOps:~$ nano filename.txt
leprecha@Ubuntu-DevOps:~$ cp filename.txt /tmp/filename.txt
leprecha@Ubuntu-DevOps:~$ cat /tmp/filename.txt
How are you today? I'm fine.
```

## 5. Копирование, перемещение, удаление

Команды позволяют управлять файлами и папками — копировать, перемещать, переименовывать и удалять.

Команда	Описание
<code>cp file.txt backup.txt</code>	Копирует <code>file.txt</code> в <code>backup.txt</code>
<code>cp file.txt /home/user/</code>	Копирует файл в указанную папку
<code>cp -r myfolder /home/user/</code>	Рекурсивно копирует папку со всем содержимым
<code>mv file.txt /home/user/</code>	Перемещает файл в указанную папку
<code>mv oldname.txt newname.txt</code>	Переименовывает файл
<code>mv myfolder /home/user/</code>	Перемещает папку в указанное место
<code>rm file.txt</code>	Удаляет файл
<code>rm file1.txt file2.txt</code>	Удаляет несколько файлов
<code>rm -r myfolder</code>	Рекурсивно удаляет папку и её содержимое
<code>rm -rf myfolder</code>	Удаляет папку без подтверждения (опасно!)

### 1). Копирование

В Linux за копирование отвечает команда `cp`.

```
leprecha@Ubuntu-DevOps:~$ cp filename.txt backup.txt
leprecha@Ubuntu-DevOps:~$ cat backup.txt
```

How are you today? I'm fine.

Копировать файл в папку:

```
leprecha@Ubuntu-DevOps:~$ tree /home/leprecha/Documents/  
/home/leprecha/Documents/  
└── filename.txt
```

Копировать папку со всем содержимым:

( `-r` or `--recursive` — рекурсивно, обязательно для директорий).

```
leprecha@Ubuntu-DevOps:~$ cp -r Folder /home/leprecha/Documents/  
leprecha@Ubuntu-DevOps:~$ tree /home/leprecha/Documents/  
/home/leprecha/Documents/  
├── filename.txt  
└── Folder
```

## 2). Перемещение

За перемещение (и переименование) отвечает `mv`.

```
leprecha@Ubuntu-DevOps:~$ mv filename.txt /home/leprecha/Documents/Fo  
lder/  
leprecha@Ubuntu-DevOps:~$ tree /home/leprecha/Documents/  
/home/leprecha/Documents/  
├── filename.txt  
└── Folder  
    └── filename.txt
```

2 directories, 2 files

```
leprecha@Ubuntu-DevOps:~$
```

Переименование:

```
leprecha@Ubuntu-DevOps:~$ mv filename.txt newfile.txt
leprecha@Ubuntu-DevOps:~$ ls
backup.txt  DevOps  Downloads  Music    Pictures  snap    Videos
Desktop    Documents  Folder    newfile.txt  Public    Templates
```

Перемещение папки:

```
leprecha@Ubuntu-DevOps:~$ mv Folder /home/leprecha/Music/
leprecha@Ubuntu-DevOps:~$ ls /home/leprecha/Music/
Folder
```

### 3). Удаление

За удаление отвечает `rm`.

Удалить файл:

```
leprecha@Ubuntu-DevOps:~$ rm newfile.txt
```

Удалить несколько файлов:

```
leprecha@Ubuntu-DevOps:~$ rm backup.txt file.txt
```

Удалить папку и всё внутри:

```
leprecha@Ubuntu-DevOps:~$ rm -r /home/leprecha/Music/Folder/
```

## Практика

1. Создай папку `lab2_files` в домашней папке.
2. Создай в ней три файла: `file1.txt`, `file2.txt`, `file3.txt`.
3. Скопируй `file1.txt` в `/tmp`.
4. Перемести `file2.txt` в `/tmp` и переименуй в `file2_moved.txt`.
5. Удали `file3.txt`.
6. Удали папку `lab2_files`.

```
leprecha@Ubuntu-DevOps:~$ mkdir ~/lab2_files
leprecha@Ubuntu-DevOps:~$ touch ~/lab2_files/file1.txt ~/lab2_files/file2.txt
~/lab2_files/file3.txt
leprecha@Ubuntu-DevOps:~$ cp ~/lab2_files/file1.txt /tmp
leprecha@Ubuntu-DevOps:~$ mv ~/lab2_files/file2.txt /tmp/file2_moved.txt
leprecha@Ubuntu-DevOps:~$ rm ~/lab2_files/file3.txt
leprecha@Ubuntu-DevOps:~$ rm -r ~/lab2_files
leprecha@Ubuntu-DevOps:~$ ls
Desktop Documents Music  Public Templates
DevOps  Downloads Pictures snap  Videos
leprecha@Ubuntu-DevOps:~$ ls /tmp
file1.txt
file2_moved.txt
```

## 6. Права доступа

В Linux права доступа определяют, кто и что может делать с файлом или папкой.

### 1). Что такое права доступа

В Linux каждый файл и каталог имеют три группы прав доступа:

1. **Владелец** (user, `u`) — пользователь, которому принадлежит файл.

2. **Группа** (group, **g**) — пользователи, входящие в ту же группу, что и владелец.
3. **Остальные** (others, **o**) — все остальные пользователи.

Каждая группа может иметь три типа прав:

- **r** (read) — право на чтение файла.
- **w** (write) — право на изменение файла.
- **x** (execute) — право на выполнение файла как программы.

## Формат прав в **ls -l**

```
leprecha@Ubuntu-DevOps:~$ ls -l
drwxr-xr-x 2 leprecha sysadmin 4096 Aug 20 17:23 Desktop
drwx----- 6 leprecha sysadmin 4096 Aug 19 21:00 snap
```

- Первый символ: тип ( **-** — файл, **d** — каталог).
- Затем три группы по 3 символа: **владелец, группа, остальные**.
- **r** — чтение, **w** — запись, **x** — выполнение, — нет доступа.

```
-rw-rw-r-- 1 leprecha sysadmin 492 Aug 20 19:31 learnlinux.spec
```

## Расшифровка

- Первый символ указывает тип объекта ( **-** = файл, **d** = каталог).
- Последующие символы обозначают права доступа:
  - **rw-** — владелец (**чтение, запись**, без выполнения).
  - **rw-** — группа (**чтение, запись**, без выполнения).
  - **r--** — остальные (**чтение**, без записи, без выполнения).

## 2). Как менять права



В Linux права для каждого файла и папки можно менять с помощью команды `chmod`.

Примеры:

- `chmod u+x file` — добавить выполнение владельцу.
- `chmod g-w file` — убрать запись у группы.
- `chmod o+r file` — добавить чтение для остальных.
- `chmod 755 file` — установить права в числовой (восьмеричной) форме.

```
leprecha@Ubuntu-DevOps:~$ chmod u+x learnlinux.spec
leprecha@Ubuntu-DevOps:~$ ls -l
-rwxr--r-- 1 leprecha sysadmin  0 Aug 20 19:09 learnlinux.spec
```

`rwx` — владелец (**ч**тение, **з**апись, **в**ыполнение).

## В числовой форме

Права кодируются числами:

- $r = 4$
- $w = 2$
- $x = 1$

Складываем:

- `rwx` =  $4+2+1 = 7$
- `rw-` =  $4+2+0 = 6$
- `r-x` =  $4+0+1 = 5$

Пример: `chmod 555`

```
leprecha@Ubuntu-DevOps:~$ chmod 555 learnlinux.spec
leprecha@Ubuntu-DevOps:~$ ls -l
-r-xr-xr-x 1 leprecha sysadmin  0 Aug 20 19:09 learnlinux.spec
```

## Изменение прав

Команда	Описание
<code>chmod 755 file</code>	rwх для владельца, rx для группы и остальных
<code>chmod u+x file</code>	Добавить выполнение владельцу
<code>chmod g-w file</code>	Убрать запись у группы
<code>chmod o-r file</code>	Убрать чтение у остальных

## Как менять владельца

Сменить владельца файла на пользователя `helpme` с помощью `sudo chown` :

```
leprecha@Ubuntu-DevOps:~$ sudo chown helpme learnlinux.spec
leprecha@Ubuntu-DevOps:~$ ls -l learnlinux.spec
-r-xr-xr-x 1 helpme sysadmin 0 Aug 20 19:09 learnlinux.spec
```

## Смена владельца и группы

Команда	Описание
<code>chown user file</code>	Сменить владельца файла
<code>chgrp group file</code>	Сменить группу файла
<code>chown user:group file</code>	Сменить владельца и группу

## Практика

1. Создай файл `test_permissions.txt` .
2. Проверь права.
3. Выдай владельцу полный доступ, группе — только чтение, остальным — нет доступа.
4. Проверь права.
5. Сменить владельца файла.

```
leprecha@Ubuntu-DevOps:~$ touch test_permissions.txt
leprecha@Ubuntu-DevOps:~$ ls -l test_permissions.txt
-rw-r--r-- 1 leprecha sysadmin 0 Aug 20 19:25 test_permissions.txt
leprecha@Ubuntu-DevOps:~$ chmod 740 test_permissions.txt
leprecha@Ubuntu-DevOps:~$ sudo chown helpme test_permissions.txt
leprecha@Ubuntu-DevOps:~$ ls -l test_permissions.txt
-rwxr----- 1 helpme sysadmin 0 Aug 20 19:25 test_permissions.txt
```