

# Лабораторная работа №5

Анализ файловой системы Linux.

---

Захаренко А. В.

11.03.2023

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Захаренко Анастасия Викторовна
- студентка первого курса бакалавриата
- направление бизнес-информатика, факультет физико-математических и естественных наук
- Российский университет дружбы народов

## Элементы презентации

---

Ознакомление с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобретение практических навыков по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы

- Выполнить все примеры, приведённые в первой части описания лабораторной работы.
- Выполнить действия, представленные в лаб.5

Команды для работы с файлами и каталогами

Копирование файлов и каталогов Команда `cp` используется для копирования файлов и каталогов. Формат команды: `cp [-опции] исходный_файл целевой_файл`

Перемещение и переименование файлов и каталогов Команды `mv` и `mkdir` предназначены для перемещения и переименования файлов и каталогов. Формат команды `mv mv [-опции] старый_файл новый_файл`

Права доступа Каждый файл или каталог имеет права доступа. В сведениях о файле или каталоге указываются: – тип файла (символ (-) обозначает файл, а символ (d) — каталог); – права для владельца файла (r — разрешено чтение, w — разрешена запись, x — разрешено выполнение, - — право доступа отсутствует); – права для членов группы (r — разрешено чтение, w — разрешена запись, x — разрешено выполнение, - — право доступа отсутствует); – права для всех остальных (r — разрешено чтение, w — разрешена запись, x — разрешено выполнение, - — право доступа отсутствует).



1. Выполните все примеры, приведённые в первой части описания лабораторной работы

Копирование файлов и каталогов

```
[avzakharenko@fedora ~]$ cd
[avzakharenko@fedora ~]$ touch abc1
[avzakharenko@fedora ~]$ cp abc1 april
[avzakharenko@fedora ~]$ cp abc1 may
[avzakharenko@fedora ~]$ mkdir monthly
[avzakharenko@fedora ~]$ cp april may monthly
[avzakharenko@fedora ~]$ cp monthly/may monthly/june
[avzakharenko@fedora ~]$ ls monthly
april  june  may
[avzakharenko@fedora ~]$
```



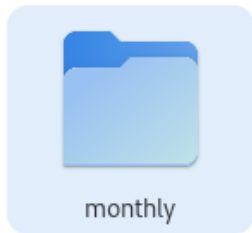
abc1



april



may



monthly

Рис. 1: вид1



april



june



may

Рис. 2: вид2

```
[avzakharenko@fedora ~]$ mkdir monthly.00  
[avzakharenko@fedora ~]$ cp -r monthly monthly.00  
[avzakharenko@fedora ~]$ cp -r monthly.00 /tmp
```

Рис. 3: коп.monthly.00

## Перемещение и переименование файлов и каталогов

```
[avzakharenko@fedora ~]$ mv april july
[avzakharenko@fedora ~]$ mv july monthly.00
[avzakharenko@fedora ~]$ ls monthly.00
july  monthly
[avzakharenko@fedora ~]$ mv monthly.00 monthly.01
[avzakharenko@fedora ~]$ mkdir reports
[avzakharenko@fedora ~]$ mv monthly.01 reports
[avzakharenko@fedora ~]$ mv reports/monthly.01 reports/monthly
```

Рис. 4: выполнение

```
[avzakharenko@fedora ~]$ cd
[avzakharenko@fedora ~]$ touch may
[avzakharenko@fedora ~]$ s -l may
bash: s: команда не найдена...
[avzakharenko@fedora ~]$ ls -l may
-rw-r--r--. 1 avzakharenko avzakharenko 0 map 7 01:31 may
[avzakharenko@fedora ~]$ chmod u+x may
[avzakharenko@fedora ~]$ ls -l may
-rwxr--r--. 1 avzakharenko avzakharenko 0 map 7 01:31 may
[avzakharenko@fedora ~]$ mkdir monthly
mkdir: невозможно создать каталог «monthly»: Файл существует
[avzakharenko@fedora ~]$ cd
[avzakharenko@fedora ~]$ mkdir monthly
mkdir: невозможно создать каталог «monthly»: Файл существует
[avzakharenko@fedora ~]$ chmod u-x may
[avzakharenko@fedora ~]$ ls -l may
-rw-r--r--. 1 avzakharenko avzakharenko 0 map 7 01:31 may
[avzakharenko@fedora ~]$ cd
[avzakharenko@fedora ~]$ ls -l monthly
итого 0
-rw-r--r--. 1 avzakharenko avzakharenko 0 map 7 01:20 april
-rw-r--r--. 1 avzakharenko avzakharenko 0 map 7 01:22 june
-rw-r--r--. 1 avzakharenko avzakharenko 0 map 7 01:20 may
[avzakharenko@fedora ~]$ chmod g-r, o-r monthly
chmod: неверный режим: «g-r,»
По команде «chmod --help» можно получить дополнительную информацию.
```

```
cd[avzakharenko@fedora ~]$ cd
[avzakharenko@fedora ~]$ touch abc1
[avzakharenko@fedora ~]$ chmod g+w abc1
[avzakharenko@fedora ~]$ ls -l abc1
-rw-rw-r--. 1 avzakharenko avzakharenko 0 map  7 01:42 abc1
[avzakharenko@fedora ~]$
```

Рис. 6: изменение прав2

2. Выполняем следующие действия: 2.1. Скопируем файл `/usr/include/sys/io.h` в домашний каталог и называем его `equipment`. 2.2. В домашнем каталоге создаем директорию `~/ski.plases`. 2.3. Перемещаем файл `equipment` в каталог `~/ski.plases`. 2.4. Переименуем файл `~/ski.plases/equipment` в `~/ski.plases/equiplist`. 2.5. Создаем в домашнем каталоге файл `abc1` и копируем его в каталог `~/ski.plases`, называем его `equiplist2`. 2.6. Создаем каталог с именем `equipment` в каталоге `~/ski.plases`. 2.7. Перемещаем файлы `~/ski.plases/equiplist` и `equiplist2` в каталог `~/ski.plases/equipment`.



```
[avzakharenko@fedora ~]$ cp /usr/include/sys/io.h /home/avzakharenko
[avzakharenko@fedora ~]$ mv io.h equipment
[avzakharenko@fedora ~]$ mkdir ~/ski.plases
[avzakharenko@fedora ~]$ mv equipment ~/ski.plases
[avzakharenko@fedora ~]$ mv ~/ski.plases ~/ski.plases/equiplist
mv: невозможно перенести '/home/avzakharenko/ski.plases' в свой собственный подк
аталог, '/home/avzakharenko/ski.plases/equiplist'
[avzakharenko@fedora ~]$ cd
[avzakharenko@fedora ~]$ mv ~/ski.plases/equipment ~/ski.plases/equiplist
[avzakharenko@fedora ~]$ touch abc1
[avzakharenko@fedora ~]$ cp abc1 ~/ski.plases
[avzakharenko@fedora ~]$ mv ~/ski.plases/abc1 ~/ski.plases/equiplist2
```

Рис. 7: выполнение2



abc1



may



monthly



reports



ski.places



work



Видео



Документы

Рис. 8: вид3

2.8. Создаем и перемещаем каталог ~/newdir в каталог ~/ski.plases и называем его plans.

```
[avzakharenko@fedora ski.plases]$ cd  
[avzakharenko@fedora ~]$ mkdir newdir  
[avzakharenko@fedora ~]$ mv newdir ski.plases  
[avzakharenko@fedora ~]$ mv newdir plans  
mv: не удалось выполнить stat для 'newdir': Нет такого файла или каталога  
[avzakharenko@fedora ~]$ mv ~/ski.plases/newdir ~/ski.plases/plans  
[avzakharenko@fedora ~]$
```

Рис. 10: выполнение3

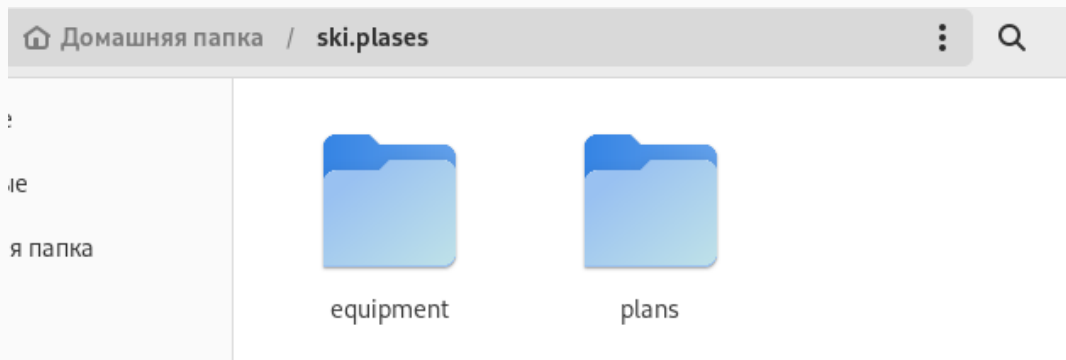


Рис. 11: вид5

3. Определите опции команды `chmod`, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет:
- 3.1. `drwxr-r-` ... `australia` 3.2. `drwx-x-x` ... `play` 3.3. `-r-xr-r-` ... `my_os` 3.4. `-rw-rw-r-` ... `feathers`

```

[avzakharenko@fedora ~]$ cd
[avzakharenko@fedora ~]$ ls -l
итого 0
-rw-rw-r--. 1 avzakharenko avzakharenko  0 мар  7 01:55 abc1
-rw-r--r--. 1 avzakharenko avzakharenko  0 мар  7 01:31 may
drwxr-xr-x. 1 avzakharenko avzakharenko 24 мар  7 01:22 monthly
drwxr-xr-x. 1 avzakharenko avzakharenko 14 мар  7 01:29 reports
drwxr-xr-x. 1 avzakharenko avzakharenko 28 мар  7 02:04 ski.plases
drwxr-xr-x. 1 avzakharenko avzakharenko 10 фев 21 18:14 work
drwxr-xr-x. 1 avzakharenko avzakharenko  0 фев 18 19:55 Видео
drwxr-xr-x. 1 avzakharenko avzakharenko  0 фев 18 19:55 Документы
drwxr-xr-x. 1 avzakharenko avzakharenko 132 мар  4 18:13 Загрузки
drwxr-xr-x. 1 avzakharenko avzakharenko  50 фев 18 22:10 Изображения
drwxr-xr-x. 1 avzakharenko avzakharenko  0 фев 18 19:55 Музыка
drwxr-xr-x. 1 avzakharenko avzakharenko  0 фев 18 19:55 Общедоступные
drwxr-xr-x. 1 avzakharenko avzakharenko  0 фев 18 19:55 'Рабочий стол'
drwxr-xr-x. 1 avzakharenko avzakharenko  0 фев 18 19:55 Шаблоны

```

Рис. 12: список

```
[avzakharenko@fedora ~]$ mkdir australia
```

Рис. 13: australia1

```
drwxr-xr-x. 1 avzakharenko avzakharenko 0 map 7 02:07 australia
```

Рис. 14: australia2

```
[avzakharenko@fedora ~]$ chmod g-x australia
```

Рис. 15: australia3

```
[avzakharenko@fedora ~]$ chmod o-x australia
```

Рис. 16: australia4

```
drwxr--r--. 1 avzakharenko avzakharenko 0 map 7 02:07 australia
```

Рис. 17: australia5

```
[avzakharenko@fedora ~]$ mkdir play  
[avzakharenko@fedora ~]$ chmod g-r play
```

Рис. 18: play1

```
[avzakharenko@fedora ~]$ chmod o-r play
```

Рис. 19: play2

```
drwx--x--x. 1 avzakharenko avzakharenko 0 map 7 02:17 play
```

Рис. 20: play3



```
[avzakharenko@fedora ~]$ touch my_os  
[avzakharenko@fedora ~]$ touch feathers
```

Рис. 21: my\_os/feathers

```
[avzakharenko@fedora ~]$ chmod g+w feathers
```

Рис. 22: feathers2

```
-rw-rw-r--. 1 avzakharenko avzakharenko 0 map 7 02:23 feathers
```

Рис. 23: feathers3

```
[avzakharenko@fedora ~]$ chmod u-w my_os
```

Рис. 24: my\_os2

```
[avzakharenko@fedora ~]$ chmod u+x my_os
```

Рис. 25: my\_os3

```
-r-xr--r--. 1 avzakharenko avzakharenko 0 map 7 02:23 my_os
```

Рис. 26: my\_os4

4. Проделаем приведённые ниже упражнения: 4.1. Посмотрим содержимое файла `/etc/password`. 4.2. Скопируем файл `~/feathers` в файл `~/file.old`. 4.3. Переместим файл `~/file.old` в каталог `~/play`. 4.4. Скопируем каталог `~/play` в каталог `~/fun`. 4.5. Переместим каталог `~/fun` в каталог `~/play` и назовем его `games`. 4.6. Лишаем владельца файла `~/feathers` права на чтение. 4.7. Что произойдёт, если вы попытаетесь просмотреть файл `~/feathers` командой `cat?`-отказ в доступе 4.8. Что произойдёт, если вы попытаетесь скопировать файл `~/feathers?` 4.9. Даем владельцу файла `~/feathers` право на чтение.

```
[avzakharenko@fedora ~]$ ls /etc/password
ls: невозможно получить доступ к '/etc/password': Нет такого файла или каталога
[avzakharenko@fedora ~]$ ls ~/etc/password
ls: невозможно получить доступ к '/home/avzakharenko/etc/password': Нет такого файла или каталога
[avzakharenko@fedora ~]$ cp ~/feathers ~/file.old
[avzakharenko@fedora ~]$ mv ~/file.old ~/play
[avzakharenko@fedora ~]$ cp ~/play ~/fun
cp: не указан -r; пропускается каталог '/home/avzakharenko/play'
[avzakharenko@fedora ~]$ cd
[avzakharenko@fedora ~]$ mkdir fun
[avzakharenko@fedora ~]$ cp ~/play ~/fun
cp: не указан -r; пропускается каталог '/home/avzakharenko/play'
[avzakharenko@fedora ~]$ rm ~/fun
rm: невозможно удалить '/home/avzakharenko/fun': Это каталог
[avzakharenko@fedora ~]$ cp -r ~/play ~/fun
[avzakharenko@fedora ~]$ mv ~/fun ~/play
[avzakharenko@fedora ~]$ mv ~/play/fun ~/play/games
[avzakharenko@fedora ~]$ chmod u-r feathers
[avzakharenko@fedora ~]$ ls feathers
feathers
[avzakharenko@fedora ~]$ cat ~/feathers
cat: /home/avzakharenko/feathers: Отказано в доступе
[avzakharenko@fedora ~]$ cp ~/feathers
cp: после '/home/avzakharenko/feathers' пропущен операнд, задающий целевой файл
По команде «cp --help» можно получить дополнительную информацию.
[avzakharenko@fedora ~]$ chmod u+r feathers
[avzakharenko@fedora ~]$ chmod u+x play
[avzakharenko@fedora ~]$ cd play
[avzakharenko@fedora play]$ cd
[avzakharenko@fedora ~]$ chmod u-x play
```

Рис. 27: выполнение4

4.10. Лишаем владельца каталога ~/play права на выполнение. 4.11. Перейдите в каталог ~/play. Что произошло?-отказ в доступе 4.12. Дайте владельцу каталога ~/play право на выполнение

```
[avzakharenko@fedora ~]$ chmod u-x play
[avzakharenko@fedora ~]$ cd play
bash: cd: play: Отказано в доступе
[avzakharenko@fedora ~]$ chmod u+x play
[avzakharenko@fedora ~]$
```

Рис. 28: выполнение5

## 5. Прочитаем ман по командам mount, fsck, mkfs, kill.

```
MOUNT(8)                                System Administration                                MOUNT(8)

NAME
    mount - mount a filesystem

SYNOPSIS
    mount [-h|-V]

    mount [-l] [-t fstype]

    mount -a [-fFnrsvw] [-t fstype] [-O optlist]

    mount [-fnrsvw] [-o options] device|mountpoint

    mount [-fnrsvw] [-t fstype] [-o options] device mountpoint

    mount --bind|--rbind|--move olddir newdir

    mount
    --make-[shared|slave|private|unbindable|rshared|rsave|rprivate|runbindable]
    mountpoint

DESCRIPTION
    All files accessible in a Unix system are arranged in one big tree, the
    file hierarchy, rooted at /. These files can be spread out over several
    devices. The mount command serves to attach the filesystem found on
    some device to the big file tree. Conversely, the umount(8) command
    will detach it again. The filesystem is used to control how data is
    stored on the device or provided in a virtual way by network or other

Manual page mount(8) line 1 (press h for help or q to quit)
```

`fsck - check and repair a Linux filesystem`

#### SYNOPSIS

```
fsck [-lsAVRTMNP] [-r [fd]] [-C [fd]] [-t fstype] [filesystem...] [--]  
[fs-specific-options]
```

#### DESCRIPTION

**fsck** is used to check and optionally repair one or more Linux filesystems. filesystem can be a device name (e.g., /dev/hdc1, /dev/sdb2), a mount point (e.g., /, /usr, /home), or an filesystem label or UUID specifier (e.g., UUID=8868abf6-88c5-4a83-98b8-bfc24057f7bd or LABEL=root). Normally, the **fsck** program will try to handle filesystems on different physical disk drives in parallel to reduce the total amount of time needed to check all of them.

If no filesystems are specified on the command line, and the **-A** option is not specified, **fsck** will default to checking filesystems in /etc/fstab serially. This is equivalent to the **-As** options.

The exit status returned by **fsck** is the sum of the following conditions:

- 0  
No errors
- 1  
Filesystem errors corrected

Manual page fsck(8) line 4 (press h for help or q to quit)

**NAME**

**mkfs** - build a Linux filesystem

**SYNOPSIS**

**mkfs** [*options*] [-t *type*] [*fs-options*] *device* [*size*]

**DESCRIPTION**

This **mkfs** frontend is deprecated in favour of filesystem specific **mkfs.<type> utils**.

**mkfs** is used to build a Linux filesystem on a device, usually a hard disk partition. The *device* argument is either the device name (e.g., */dev/hda1*, */dev/sdb2*), or a regular file that shall contain the filesystem. The *size* argument is the number of blocks to be used for the filesystem.

The exit status returned by **mkfs** is 0 on success and 1 on failure.

In actuality, **mkfs** is simply a front-end for the various filesystem builders (**mkfs.fstype**) available under Linux. The filesystem-specific builder is searched for via your **PATH** environment setting only. Please see the filesystem-specific builder manual pages for further details.

**OPTIONS**

**-t, --type *type***

Specify the *type* of filesystem to be built. If not specified, the default filesystem type (currently ext2) is used.

Manual page mkfs(8) line 1 (press h for help or q to quit)



```
KILL(1)                                User Commands                                KILL(1)
```

**NAME**

kill - terminate a process

**SYNOPSIS**

```
kill [-signal|-s signal|-p] [-q value] [-a] [--timeout milliseconds
signal] [--] pid|name...
```

```
kill -l [number] | -L
```

**DESCRIPTION**

The command **kill** sends the specified signal to the specified processes or process groups.

If no signal is specified, the **TERM** signal is sent. The default action for this signal is to terminate the process. This signal should be used in preference to the **KILL** signal (number 9), since a process may install a handler for the TERM signal in order to perform clean-up steps before terminating in an orderly fashion. If a process does not terminate after a **TERM** signal has been sent, then the **KILL** signal may be used; be aware that the latter signal cannot be caught, and so does not give the target process the opportunity to perform any clean-up before terminating.

Most modern shells have a builtin **kill** command, with a usage rather similar to that of the command described here. The **--all**, **--pid**, and **--queue** options, and the possibility to specify processes by command name, are local extensions.

Manual page kill(1) line 1 (press h for help or q to quit)

Я ознакомилась с файловой системой Linux, её структурой, именами и содержанием каталогов. Я приобрела практические навыки по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы