

# **Отчёт по лабораторной работе № 7**

**Простейший вариант**

Арфонос Дмитрий

# Содержание

1	Цель работы	4
2	Выполнение лабораторной работы	5
3	Вывод	16

## Список иллюстраций

2.1	создаю файл . . . . .	5
2.2	less . . . . .	5
2.3	вывод содержимого . . . . .	5
2.4	вывод последних 10 строк . . . . .	6
2.5	содержимое . . . . .	6
2.6	Создание нового каталога . . . . .	6
2.7	переименование. . . . .	6
2.8	что-то... . . . .	7
2.9	создание . . . . .	7
2.10	телепортация . . . . .	7
2.11	Отобразим опции ls . . . . .	7
2.12	опции . . . . .	8
2.13	Опции . . . . .	8
2.14	опции . . . . .	8
2.15	опции . . . . .	9
2.16	опции . . . . .	9
2.17	.. . . .	9
2.18	– . . . .	9
2.19	– . . . .	10
2.20	– . . . .	10
2.21	.. . . .	10
2.22	– . . . .	10
2.23	– . . . .	11
2.24	– . . . .	11
2.25	.. . . .	11
2.26	.. . . .	11
2.27	– . . . .	12
2.28	– . . . .	13
2.29	– . . . .	14
2.30	.. . . .	15

# 1 Цель работы

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.

---

## 2 Выполнение лабораторной работы

1 Создаю текстовый файл с командой touch и смотрю содержимое. (рис. 2.1)

```
darfonos@fedora: ~/lab7$ touch lab07.txt
darfonos@fedora: ~/lab7$ cat lab07.txt
darfonos@fedora: ~/lab7$ mc
[1]+  Остановлен /usr/bin/mc -P "$MC_PWD_FILE" "$@"
darfonos@fedora: ~/lab7$ nano lab07.txt
darfonos@fedora: ~/lab7$ cat lab07.txt
hello world!!
darfonos@fedora: ~/lab7$
```

Рис. 2.1: создаю файл

2

Для просмотра файлов постранично удобнее использовать команду less(рис. 2.2)

```
darfonos@fedora: ~/lab7$ less lab07.txt
[2]+  Остановлен less lab07.txt
darfonos@fedora: ~/lab7$
```

Рис. 2.2: less

3 Команда head выводит по умолчанию первые 10 строк файла (рис. 2.3)

```
darfonos@fedora: ~/lab7$ head [-n] lab07.txt
head: невозможно открыть '[-n]' для чтения: Нет такого файла или каталога
==> lab07.txt <==
hello world!!
darfonos@fedora: ~/lab7$
```

Рис. 2.3: вывод содержимого

4 Команда tail выводит умолчанию 10 последних строк файла.(рис. 2.4)

```

darfonos@fedora:~/lab7$ tail [-n] lab07.txt
tail: невозможно открыть '[-n]' для чтения: Нет такого файла или каталога
==> lab07.txt <==
hello world!!

```

Рис. 2.4: вывод последних 10 строк

Да, есть!

5 Скопирую файл `/usr/include/sys/io.h` в домашний каталог и назову его `equipment`. (рис. 2.5)

```

darfonos@fedora:~/lab7$ cp /usr/include/sys/io.h /home/darfonos/lab7/equipment.h
darfonos@fedora:~/lab7$ ls
equipment.h  lab07.txt
darfonos@fedora:~/lab7$

```

Рис. 2.5: содержимое

6 В домашнем каталоге создаю директорию `~/ski.places` и Перемещу файл `equipment` в этот каталог(рис. 2.6)

```

darfonos@fedora:~$ cp /home/darfonos/lab7/equipment.h ~/ski.places
darfonos@fedora:~$ ls
equipment.h  lab7  work  Документы  Изображения  Общедоступные  Шаблоны
io.h         ski.places  Видео  Загрузки  Музыка  'Рабочий стол'
darfonos@fedora:~$ cd ~/ski.places.
bash: cd: /home/darfonos/ski.places.: Нет такого файла или каталога
darfonos@fedora:~$ cd ~/ski.places
darfonos@fedora:~/ski.places$ ls
equipment.h
darfonos@fedora:~/ski.places$

```

Рис. 2.6: Создание нового каталога

7 Переименовываю файл `~/ski.places/equipment` в `~/ski.places/equiplist`.(рис. 2.7)

```

darfonos@fedora:~/ski.places$ mv ~/ski.places/equipment.h ~/ski.places/equiplist
darfonos@fedora:~/ski.places$ ls
equiplist
darfonos@fedora:~/ski.places$

```

Рис. 2.7: переименование.

8 Создаю в домашнем каталоге файл `abc1` и скопирую его в каталог `~/ski.places`, назову его `equiplist2`.(рис. 2.8)

```
darfonos@fedora:~$ touch abc1
darfonos@fedora:~$ ls
abc1      fb.h      ski.places  Видео      Загрузки    Музыка      'Рабочий стол'
equipment.h lab7      work        Документы  Изображения Общедоступные Шаблоны
darfonos@fedora:~$ mc

[7]+ Остановлен /usr/bin/mc -P "$MC_PWD_FILE" "$@"
darfonos@fedora:~$ cp ~/abc1 ~/ski.places/equiplist2
darfonos@fedora:~$ ls ~/ski.places
equiplist equiplist2
darfonos@fedora:~$
```

Рис. 2.8: что-то...

9

Создаю каталог с именем equipment в каталоге ~/ski.places.(рис. 2.9)

```
darfonos@fedora:~$ mkdir ~/ski.places/equipment
darfonos@fedora:~$ ls ~/ski.places
equiplist equiplist2 equipment
darfonos@fedora:~$
```

Рис. 2.9: создание

10

Перемещу файлы ~/ski.places/equiplist и equiplist2 в каталог ~/ski.places/equipment.(рис. 2.10)

```
darfonos@fedora:~/ski.places$ mv ~/ski.places/equiplist ~/ski.places/equiplist2 ~/ski.places/equipment
darfonos@fedora:~/ski.places$ ls
equipment
darfonos@fedora:~/ski.places$ ls equipment/
equiplist equiplist2
darfonos@fedora:~/ski.places$
```

Рис. 2.10: телепортация

11

Создал и переместите каталог ~/newdir в каталог ~/ski.places назовиву его plans.(рис. 2.11)

```
darfonos@fedora:~/ski.places$ mkdir ~/newdir
darfonos@fedora:~/ski.places$ mv -R ~/newdir ~/ski.places/plans
mv: неверный ключ - «R»
по команде «mv --help» можно получить дополнительную информацию.
darfonos@fedora:~/ski.places$ mv ~/newdir ~/ski.places/plans
darfonos@fedora:~/ski.places$ ls
equipment plans
darfonos@fedora:~/ski.places$
```

Рис. 2.11: Отобразим опции ls

12

Создаю необходимые каталоги и файлы для определения опций chmod(рис. 2.12)

```
darfonos@fedora:~/ski.places/plans$ touch play
darfonos@fedora:~/ski.places/plans$ touch my_os
darfonos@fedora:~/ski.places/plans$ touch feathers
darfonos@fedora:~/ski.places/plans$ ls
australia feathers my_os play
darfonos@fedora:~/ski.places/plans$
```

Рис. 2.12: опции

### 13

Используя команду chmod изменяю права пользователей по образцу drwxr-r- к папке australia.(рис. 2.13)

```
darfonos@fedora:~/ski.places/plans$ chmod g-x australia
darfonos@fedora:~/ski.places/plans$ chmod o-x australia
darfonos@fedora:~/ski.places/plans$ ls -l
итого 0
drwxr--r--. 1 darfonos darfonos 0 map 23 20:44 australia
-rw-r--r--. 1 darfonos darfonos 0 map 23 20:42 feathers
-rw-r--r--. 1 darfonos darfonos 0 map 23 20:42 my_os
drwxr-xr-x. 1 darfonos darfonos 0 map 23 20:44 play
```

Рис. 2.13: Опции

### 14

Используя команду chmod изменяю права пользователей по образцу drwx-x-x к папке play(рис. 2.14)

```
darfonos@fedora:~/ski.places/plans$ chmod g-r play
darfonos@fedora:~/ski.places/plans$ chmod o-r play
darfonos@fedora:~/ski.places/plans$ ls -l
итого 0
drwxr--r--. 1 darfonos darfonos 0 map 23 20:44 australia
-rw-r--r--. 1 darfonos darfonos 0 map 23 20:42 feathers
-rw-r--r--. 1 darfonos darfonos 0 map 23 20:42 my_os
drwx-x-x. 1 darfonos darfonos 0 map 23 20:44 play
```

Рис. 2.14: опции

### 15

Используя команду chmod изменяю права пользователей по образцу -r-xr-r- к файлу my\_os(рис. 2.15)



```

darfonos@fedora:~/ski.places/plans$ chmod g-x my_os
darfonos@fedora:~/ski.places/plans$ chmod u+x my_os
darfonos@fedora:~/ski.places/plans$ ls -l
итого 0
drwxr--r--. 1 darfonos darfonos 0 map 23 20:44 australia
-rw-r--r--. 1 darfonos darfonos 0 map 23 20:42 feathers
-r-xr--r--. 1 darfonos darfonos 0 map 23 20:42 my_os
drwx--x--x. 1 darfonos darfonos 0 map 23 20:44 play

```

Рис. 2.15: опции

## 16

Используя команду `chmod` изменяю права пользователей по образцу `-r-xr-r-` к файлу `feathers`(рис. 2.16)

```

darfonos@fedora:~/ski.places/plans$ chmod g+w feathers
darfonos@fedora:~/ski.places/plans$ ls -l
итого 0
drwxr--r--. 1 darfonos darfonos 0 map 23 20:44 australia
-rw-rw-r--. 1 darfonos darfonos 0 map 23 20:42 feathers
-r-xr--r--. 1 darfonos darfonos 0 map 23 20:42 my_os
drwx--x--x. 1 darfonos darfonos 0 map 23 20:44 play
darfonos@fedora:~/ski.places/plans$

```

Рис. 2.16: опции

## 18

Посмотрю содержимое файла `/etc/passwd`. (рис. 2.17)

```

darfonos@fedora:/etc$ sudo cat /etc/passwd
cat: /etc/passwd: Нет такого файла или каталога
darfonos@fedora:/etc$

```

Рис. 2.17: ..

- нет такого файла

## 19

Скопирую файл `~/feathers` в файл `~/file.old`.(рис. 2.18)

```

darfonos@fedora:~$ cp ~/feathers ~/file.old

```

Рис. 2.18: –

20

Перемещу файл ~/file.old в каталог ~/play.(рис. 2.19)

```
darfonos@fedora:~$ cp -r ~/file.old ~/play
darfonos@fedora:~$
```

Рис. 2.19: –

21

Перемещу каталог ~/fun в каталог ~/play и назову его games.(рис. 2.20)

```
darfonos@fedora:~$ mv ~/fun ~/play/games
darfonos@fedora:~$
```

Рис. 2.20: –

22

Лишаю владельца файла ~/feathers права на чтение. (рис. 2.21)

```
darfonos@fedora:~$ chmod u-r feathers
darfonos@fedora:~$ ls -l feathers
--w-r--r--. 1 darfonos darfonos 0 map 23 20:59 feathers
darfonos@fedora:~$
```

Рис. 2.21: ..

23

Если вы попытаюсь просмотреть файл ~/feathers командой cat то откажут в доступе. (рис. 2.22)

```
darfonos@fedora:~$ cat feathers
cat: feathers: Отказано в доступе
darfonos@fedora:~$
```

Рис. 2.22: –

- если попытаюсь скопировать файл ~/feathers, то также будет отказано в доступе.

24

Даю владельцу файла ~/feathers право на чтение.(рис. 2.23)

```
darfonos@fedora:~$ chmod u+r feathers
darfonos@fedora:~$ ls -l feathers
-rw-r--r--. 1 darfonos darfonos 0 map 23 20:59 feathers
darfonos@fedora:~$
```

Рис. 2.23: –

25

Лишаю владельца каталога ~/play права на выполнение.(рис. 2.24)

```
darfonos@fedora:~$ chmod u-x play
darfonos@fedora:~$ ls -l play
ls: невозможно получить доступ к 'play/file.old': Отказано в доступе
ls: невозможно получить доступ к 'play/games': Отказано в доступе
итого 0
-???????? ? ? ? ? ? file.old
d???????? ? ? ? ? ? games
```

Рис. 2.24: –

26

Пытаюсь Перейти в каталог ~/play, и мне отказывают в доступе.(рис. 2.25)

```
darfonos@fedora:~$ cd play
bash: cd: play: Отказано в доступе
darfonos@fedora:~$
```

Рис. 2.25: ..

27

Даю владельцу каталога ~/play право на выполнение. (рис. 2.26)

```
darfonos@fedora:~$ chmod u+x play
darfonos@fedora:~$
```

Рис. 2.26: ..

28

Прочитаю man по команде mount (рис. 2.27)

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

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|rslave|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
services.

The standard form of the mount command is:

    mount -t type device dir

This tells the kernel to attach the filesystem found on device (which
is of type type) at the directory dir. The option -t type is optional.
The mount command is usually able to detect a filesystem. The root
permissions are necessary to mount a filesystem by default. See section
"Non-superuser mounts" below for more details. The previous contents
(if any) and owner and mode of dir become invisible, and as long as
this filesystem remains mounted, the pathname dir refers to the root of
the filesystem on device.

If only the directory or the device is given, for example:
```

Рис. 2.27: –

- Для просмотра используемых в операционной системе файловых систем можно воспользоваться командой mount без параметров. В результате её применения можно получить примерно следующее:

29

Прочитаю ман по команде fsck(рис. 2.28)

```
darfonos@fedora:~ -- man fsck
FSCK(8) System Administration FSCK(8)

NAME
    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 a filesystem label or UUID specifier
    (e.g., UUID=8868ab76-88c5-4a83-98b8-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

    2      System should be rebooted

    4      Filesystem errors left uncorrected

    8      Operational error

    16     Usage or syntax error

    32     Checking canceled by user request

    128    Shared-library error

    The exit status returned when multiple filesystems are checked is the bit-wise OR of the exit statuses for each
```

Рис. 2.28: –

- С помощью команды `fsck` можно проверить (а в ряде случаев восстановить) целостность файловой системы:

30

Прочитаю ман по команде `mkfs`(рис. 2.29)

```
mkfs(8)                                System Administration                                mkfs(8)

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.

    fs-options
        Filesystem-specific options to be passed to the real filesystem builder.

    -V, --verbose
        Produce verbose output, including all filesystem-specific commands that are executed. Specifying this option more than once inhibits execution of any filesystem-specific commands. This is really only useful for testing.

    -h, --help
        Display help text and exit.

    -V, --version
        Print version and exit. (Option -V will display version information only when it is the only parameter, otherwise it will work as --verbose.)

BUGS
    All generic options must precede and not be combined with filesystem-specific options. Some filesystem-specific programs do not automatically detect the device size and require the size parameter to be specified.
```

Рис. 2.29: –

- mkfs используется для создания файловой системы Linux на некотором устройстве, обычно в разделе жёсткого диска. В качестве аргумента filesystem для файловой системы может выступать или название устройства (например, /dev/hda1, /dev/sdb2) или точка монтирования (например, /, /usr, /home).

31

Прочитаю man по команде kill(рис. 2.30)

```
darfonos@fedora:~ -- man kill
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.

  If signal is 0, then no actual signal is sent, but error checking is still performed.
ARGUMENTS
  The list of processes to be signaled can be a mixture of names and PIDs.

  pid
  Each pid can be expressed in one of the following ways:

  n
  where n is larger than 0. The process with PID n is signaled.

  0
  All processes in the current process group are signaled.

  -1
  All processes with a PID larger than 1 are signaled.

  -n
  where n is larger than 1. All processes in process group n are signaled. When an argument of the form '-n' is given, and it is meant to denote a process group, either a signal must be specified first, or the argument must be preceded by a '--' option, otherwise it will be taken as the signal to send.

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

Рис. 2.30: ..

- Команда Kill посылает указанный сигнал указанному процессу. Если не указано ни одного сигнала, посылается сигнал SIGTERM. Сигнал SIGTERM завершает лишь те процессы, которые не обрабатывают его приход. Для других процессов может быть необходимым послать сигнал SIGKILL, поскольку этот сигнал перехватить невозможно.

## **3 Вывод**

В ходе лабораторной работы я ознакомился с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобрел практические навыки по применению команд для работы с файлами и каталогами.