

Лабораторная работа №5

Анализ файловой системы Linux. Команды для работы с файлами и каталогами

Латыпова Диана. НФИбд-02-21

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	8
4	Контрольные вопросы	22
5	Выводы	26

Список иллюстраций

3.1	Пример1. Копирование файла в текущем каталоге.	8
3.2	Пример1. Копирование нескольких файлов в каталог. Копирование файлов в произвольном каталоге.	9
3.3	Пример2. Копирование каталогов в текущем каталоге. Копирование каталогов в произвольном каталоге.	9
3.4	Пример3.	11
3.5	Пример4 (1,2).	12
3.6	Пример4(3) начало.	12
3.7	Пример4(3,4) продолжение.	12
3.8	2.1	13
3.9	2.2	13
3.10	2.3	13
3.11	2.4	13
3.12	2.5	14
3.13	2.6	14
3.14	2.7	14
3.15	2.8	15
3.16	Настройка прав доступа	15
3.17	Проверка настройки прав доступа	16
3.18	4.1	16
3.19	4.2	16
3.20	4.3	17
3.21	4.4	17
3.22	4.5	17
3.23	4.6-4.8	17
3.24	4.9	18
3.25	4.10-4.11	18
3.26	4.12	18
3.27	Характеристика	18
3.28	mount	19
3.29	fsck	20
3.30	mkfs	21
3.31	kill	21

Список таблиц

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 Задание

1. Выполните все примеры, приведённые в первой части описания лабораторной работы.
2. Выполните следующие действия, зафиксировав в отчёте по лабораторной работе используемые при этом команды и результаты их выполнения:
 - 2.1. Скопируйте файл `/usr/include/sys/io.h` в домашний каталог и назовите его `equipment`. Если файла `io.h` нет, то используйте любой другой файл в каталоге `/usr/include/sys/` вместо него.
 - 2.2. В домашнем каталоге создайте директорию `~/ski.places`.
 - 2.3. Переместите файл `equipment` в каталог `~/ski.places`.
 - 2.4. Переименуйте файл `~/ski.places/equipment` в `~/ski.places/equiplist`.
 - 2.5. Создайте в домашнем каталоге файл `abc1` и скопируйте его в каталог `~/ski.places`, назовите его `equiplist2`.
 - 2.6. Создайте каталог с именем `equipment` в каталоге `~/ski.places`.
 - 2.7. Переместите файлы `~/ski.places/equiplist` и `equiplist2` в каталог `~/ski.places/equipment`.
 - 2.8. Создайте и переместите каталог `~/newdir` в каталог `~/ski.places` и назовите его `plans`.
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

При необходимости создайте нужные файлы.

- 4. Прodelайте приведённые ниже упражнения, записывая в отчёт по лабораторной работе используемые при этом команды:
 - 4.1. Просмотрите содержимое файла /etc/password.
 - 4.2. Скопируйте файл ~/feathers в файл ~/file.old.
 - 4.3. Переместите файл ~/file.old в каталог ~/play.
 - 4.4. Скопируйте каталог ~/play в каталог ~/fun.
 - 4.5. Переместите каталог ~/fun в каталог ~/play и назовите его games.
 - 4.6. Лишите владельца файла ~/feathers права на чтение.
 - 4.7. Что произойдёт, если вы попытаетесь просмотреть файл ~/feathers командой cat?
 - 4.8. Что произойдёт, если вы попытаетесь скопировать файл ~/feathers?
 - 4.9. Дайте владельцу файла ~/feathers право на чтение.
 - 4.10. Лишите владельца каталога ~/play права на выполнение.
 - 4.11. Перейдите в каталог ~/play. Что произошло?
 - 4.12. Дайте владельцу каталога ~/play право на выполнение.
- 5. Прочитайте man по командам mount, fsck, mkfs, kill и кратко их охарактеризуйте, приведя примеры.

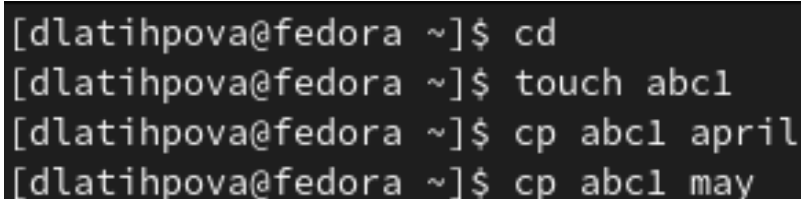
3 Выполнение лабораторной работы

1.Выполнила все примеры, которые были в первой части описания лабораторной работы

Пример 1:

1.Копирование файла в текущем каталоге. Скопировать файл ~/abc1 в файл april и в файл may(рис. 3.1):

```
1 cd
2 touch abc1
3 cp abc1 april
4 cp abc1 may
```

A screenshot of a terminal window showing four commands being executed. The prompt is [dlatihpova@fedora ~]\$. The commands are: cd, touch abc1, cp abc1 april, and cp abc1 may. The output of each command is not shown.

```
[dlatihpova@fedora ~]$ cd
[dlatihpova@fedora ~]$ touch abc1
[dlatihpova@fedora ~]$ cp abc1 april
[dlatihpova@fedora ~]$ cp abc1 may
```

Рис. 3.1: Пример1. Копирование файла в текущем каталоге.

2.Копирование нескольких файлов в каталог. Скопировать файлы april и may в каталог monthly(рис. 3.2):

```
1 mkdir monthly
2 cp april may monthly
```

3.Копирование файлов в произвольном каталоге. Скопировать файл monthly/may в файл с именем june(рис. 3.2):


```
1 cp monthly/may monthly/june
2 ls monthly
```

```
[dlatihpova@fedora ~]$ mkdir monthly
[dlatihpova@fedora ~]$ cp april may mounthly
cp: указанная цель 'mounthly' не является каталогом
[dlatihpova@fedora ~]$ cp april may monthly
[dlatihpova@fedora ~]$ cp monthly/may monthly/june
[dlatihpova@fedora ~]$ ls mounthly
ls: невозможно получить доступ к 'mounthly': Нет такого файла или каталога
[dlatihpova@fedora ~]$ ls monthly
april  june  may
```

Рис. 3.2: Пример1. Копирование нескольких файлов в каталог. Копирование файлов в произвольном каталоге.

Пример 2(рис. 3.3):

1.Копирование каталогов в текущем каталоге. Скопировать каталог monthly в каталог monthly.00:

```
1 mkdir monthly.00
2 cp -r monthly monthly.00
```

2.Копирование каталогов в произвольном каталоге. Скопировать каталог monthly.00 в каталог /tmp

```
1 cp -r monthly.00 /tmp
```

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

Рис. 3.3: Пример2. Копирование каталогов в текущем каталоге. Копирование каталогов в произвольном каталоге.

Пример 3(рис. 3.4):

1.Переименование файлов в текущем каталоге. Изменить название файла april на july в домашнем каталоге:

```
1 cd
```

```
2 mv april july
```

2.Перемещение файлов в другой каталог. Переместить файл july в каталог monthly.00:

```
1 mv july monthly.00
```

```
2 ls monthly.00
```

Результат:

```
1 april july june may
```

Если необходим запрос подтверждения о перезаписи файла, то нужно использовать опцию i.

3.Переименование каталогов в текущем каталоге. Переименовать каталог monthly.00 в monthly.01

```
1 mv monthly.00 monthly.01
```

4.Перемещение каталога в другой каталог. Переместить каталог monthly.01 в каталог reports:

```
1 mkdir reports
```

```
2 mv monthly.01 reports
```

5.Переименование каталога, не являющегося текущим. Переименовать каталог reports/monthly.01 в reports/monthly:

```
1 mv reports/monthly.01 reports/monthly
```

```

[dlatihpova@fedora ~]$ cd
[dlatihpova@fedora ~]$ mv april july
[dlatihpova@fedora ~]$ mv july monthly.00
[dlatihpova@fedora ~]$ ls monthly.00
july  monthly
[dlatihpova@fedora ~]$ mv monthly.00 monthly.01
[dlatihpova@fedora ~]$ mkdir reports
[dlatihpova@fedora ~]$ mv monthly.01 reports
[dlatihpova@fedora ~]$ mv reo=ports/monthly.01 reports/monthly
mv: не удалось выполнить stat для 'reo=ports/monthly.01': Нет та
лога
[dlatihpova@fedora ~]$ mv reoports/monthly.01 reports/monthly
mv: не удалось выполнить stat для 'reoports/monthly.01': Нет та
ога
[dlatihpova@fedora ~]$ mv reports/monthly.01 reports/monthly
[dlatihpova@fedora ~]$ 

```

Рис. 3.4: Пример3.

Пример 4:

1.Требуется создать файл ~/may с правом выполнения для владельца(рис. 3.5):

- 1 cd
- 2 touch may
- 3 ls -l may
- 4 chmod u+x may
- 5 ls -l may

2.Требуется лишить владельца файла ~/may права на выполнение(рис. 3.5):

- 1 chmod u-x may
- 2 ls -l may

```

[dlatihpova@fedora ~]$ cd
[dlatihpova@fedora ~]$ touch may
[dlatihpova@fedora ~]$ ls -l may
-rw-rw-r--. 1 dlatihpova dlatihpova 0 мая  4 07:28 may
[dlatihpova@fedora ~]$ chmod u+x may
[dlatihpova@fedora ~]$ ls-l may
bash: ls-l: command not found...
[dlatihpova@fedora ~]$ ls -l may
-rwxrw-r--. 1 dlatihpova dlatihpova 0 мая  4 07:28 may
[dlatihpova@fedora ~]$ chmod u-x may
[dlatihpova@fedora ~]$ ls -l may
-rw-rw-r--. 1 dlatihpova dlatihpova 0 мая  4 07:28 may

```

Рис. 3.5: Пример4 (1,2).

3.Требуется создать каталог monthly с запретом на чтение для членов группы и всех остальных пользователей(рис. 3.6)(рис. 3.7):

- 1 cd
- 2 mkdir monthly
- 3 chmod g-r, o-r monthly

4.Требуется создать файл ~/abc1 с правом записи для членов группы(рис. 3.7):

- 1 cd
- 2 touch abc1
- 3 chmod g+w abc1

```

[dlatihpova@fedora ~]$ cd
[dlatihpova@fedora ~]$ mkdir monthly
mkdir: невозможно создать каталог «monthly»: Файл существует

```

Рис. 3.6: Пример4(3) начало.

```

[dlatihpova@fedora ~]$ chmod g-r,o-r monthly
[dlatihpova@fedora ~]$ cd
[dlatihpova@fedora ~]$ touch abc1
[dlatihpova@fedora ~]$ chmod g+w abc1
[dlatihpova@fedora ~]$ 

```

Рис. 3.7: Пример4(3,4) продолжение.

2. После выполнения примеров, я, собственно, приступила к выполнению практикума по заданию.

2.1 Скопировала с помощью команды `cp /usr/include/sys/io.h` в домашний каталог и назвала его `equipment` (рис. 3.8).

`cp /usr/include/sys/io.h ~/equipment`

```
[dlatihpova@fedora ~]$ cp /usr/include/sys/io.h ~/equipment
```

Рис. 3.8: 2.1

2.2 В домашнем каталоге создала директорию `ski.places`, используя команду `mkdir` (рис. 3.9).

`mkdir ski.places`

```
[dlatihpova@fedora ~]$ mkdir ski.places
[dlatihpova@fedora ~]$ ls
'1 pr1.png'      '5 pr5 part3.png'  may      Видео      'Рабочий стол'
'2 pr2.png'      '6 pr 6.png'       monthly  Документы  Шаблоны
'3 pr3.png'      '7 pr 7.png'       reports  Загрузки
'4 pr 4.png'      '8 pr8.png'        ski.places  Изображения
'5 pr5 1part.png' '9 pr9.png'        snap      Музыка
'5 pr5 part2.png' equipment          work      Общедоступные
```

Рис. 3.9: 2.2

2.3 Далее с помощью `mv` переместила файл `equipment` в каталог `ski.places` (рис. 3.10).

`mv equipment ski.places`

```
[dlatihpova@fedora ~]$ mv equipment ski.places/
```

Рис. 3.10: 2.3

2.4 Используя все ту же команду `mv`, переименовала файл `equipment` в `equiplist` (рис. 3.11).

`mv /ski.places/equipment /ski.places/equiplist`

```
[dlatihpova@fedora ~]$ mv ~/ski.places/equipment ~/ski.places/equiplist
```

Рис. 3.11: 2.4

2.5 С помощью `mkdir` создала в домашнем каталоге файл `abc1` и затем скопировала (`cp`) его в каталог `ski.places`, назвала его `equiplist2`(рис. 3.12).

1 **`touch abc1`**

2 **`cp abc1 ski.places/equiplist2`**

```
[dlatihpova@fedora ~]$ touch abc1
[dlatihpova@fedora ~]$ cp abc1 ski.places/equiplist2
```

Рис. 3.12: 2.5

2.6 Перешла с помощью команды `cd` в каталог `ski.places` и создала в нем с помощью `mkdir` каталог с именем `equipment` (рис. 3.13).

1 **`cd ski.places`**

2 **`mkdir equipment`**

```
[dlatihpova@fedora ~]$ cd ski.places/
[dlatihpova@fedora ski.places]$ mkdir equipment
[dlatihpova@fedora ski.places]$ ls
equiplist  equiplist2  equipment
```

Рис. 3.13: 2.6

2.7 Перешла обратно в домашний каталог. Затем поочередно, используя команду `mv`, переместила файлы `equiplist` и `equiplist2` в каталог `equipment`(рис. 3.14).

1 **`cd`**

2 **`mv /ski.places/equiplist /ski.places/equipment`**

3 **`mv /ski.places/equiplist2 /ski.places/equipment`**

```
[dlatihpova@fedora ski.places]$ cd
[dlatihpova@fedora ~]$ mv ~/ski.places/equiplist ~/ski.places/equipment
[dlatihpova@fedora ~]$ mv
mv: пропущен операнд, задающий файл
По команде «mv --help» можно получить дополнительную информацию.
[dlatihpova@fedora ~]$ mv ~/ski.places/equiplist2 ~/ski.places/equipment
[dlatihpova@fedora ~]$ cd ski.places/
[dlatihpova@fedora ski.places]$ cd equipment/
[dlatihpova@fedora equipment]$ ls
equiplist  equiplist2
```

Рис. 3.14: 2.7

2.8 И наконец, создала в домашнем каталоге с помощью команды `mkdir` новый каталог `newdir`. И переместила с помощью `mv` его в каталог `ski.plases`, назвав его `plans`(рис. 3.15).

1 `mkdir newdir`

2 `mv newdir ski.plases/plans`

```
[dlatihpova@fedora equipment]$ cd  
[dlatihpova@fedora ~]$ mkdir newdir  
[dlatihpova@fedora ~]$ mv newdir ski.plases/plans  
[dlatihpova@fedora ~]$ ls ski.plases/  
equipment  plans
```

Рис. 3.15: 2.8

3. Далее я создала два каталога с помощью команды `mkdir` (`australia` и `play`) и два файла с помощью `touch` (`my_o` и `feathers`) в домашнем каталоге.

И настроила для них права доступа, определив опции команды `chmod`. (рис. 3.16). (рис. 3.17). А именно: - для *australia* `drwxr-r- 744`

- для *play* `drwx-x-x 711`
- для *my_o* `r-xr-r- 544`
- для *feathers* `-rw-rw-r- 664`

```
[dlatihpova@fedora ~]$ mkdir australia  
[dlatihpova@fedora ~]$ mkdir play  
[dlatihpova@fedora ~]$ touch my_os  
[dlatihpova@fedora ~]$ touch feathers  
[dlatihpova@fedora ~]$ chmod 744 australia/  
[dlatihpova@fedora ~]$ chmod 711 play/  
[dlatihpova@fedora ~]$ chmod 544 my_os  
[dlatihpova@fedora ~]$ chmod 664 feathers  
[dlatihpova@fedora ~]$ ls-l  
bash: ls-l: command not found...  
[dlatihpova@fedora ~]$ ls -l  
итого 436
```

Рис. 3.16: Настройка прав доступа

```
drwxr--r--. 1 dlatihpova dlatihpova 0 мая 4 09:15 australia
-rw-rw-r--. 1 dlatihpova dlatihpova 0 мая 4 09:16 feathers
-rw-rw-r--. 1 dlatihpova dlatihpova 0 мая 4 07:28 may
drwx-wx--x. 1 dlatihpova dlatihpova 24 мая 4 07:16 monthly
-r-xr--r--. 1 dlatihpova dlatihpova 0 мая 4 09:15 my_os
drwx--x--x. 1 dlatihpova dlatihpova 0 мая 4 09:15 play
```

Рис. 3.17: Проверка настройки прав доступа

4.Следующей моей задачей было проделать упражнения ниже:

4.1 Для начала просмотрела содержимое файла /etc/passwd(рис. 3.18).

cat /etc/passwd

```
[dlatihpova@fedora ~]$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:65534:65534:Kernel Overflow User:/sbin/nologin
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
dbus:x:81:81:System message bus:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/usr/sbin/nologin
systemd-oom:x:999:999:systemd Userspace OOM Killer:/usr/sbin/nologin
systemd-resolve:x:193:193:systemd Resolver:/usr/sbin/nologin
systemd-timesync:x:998:998:systemd Time Synchronization:/usr/sbin/nologin
systemd-coredump:x:997:997:systemd Core Dumper:/usr/sbin/nologin
tss:x:59:59:Account used for TPM access:/dev/null:/sbin/nologin
qemu:x:107:107:qemu user:/sbin/nologin
polkitd:x:996:996:User for polkitd:/sbin/nologin
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
unbound:x:995:994:Unbound DNS resolver:/etc/unbound:/sbin/nologin
dnsmasq:x:994:993:Dnsmasq DHCP and DNS server:/var/lib/dnsmasq:/sbin/nologin
nm-openconnect:x:993:991:NetworkManager user for OpenConnect:/sbin/nologin
usbmuxd:x:113:113:usbmuxd user:/sbin/nologin
gluster:x:992:990:GlusterFS daemon:/run/gluster:/sbin/nologin
```

Рис. 3.18: 4.1

4.2 Скопировала файл feathers в файл file.old(рис. 3.19).

cp feathers file.old

```
[dlatihpova@fedora ~]$ cp feathers file.old
```

Рис. 3.19: 4.2

4.3 Переместила файл file.old в каталог play(рис. 3.20).

mv file.old play

```
[dlatihpova@fedora ~]$ mv file.old play
```

Рис. 3.20: 4.3

4.4 Создала каталог fun. Скопировала в него каталог play(рис. 3.21).

1 mkdir fun

2 cp -r play fun/

```
[dlatihpova@fedora ~]$ mkdir fun
[dlatihpova@fedora ~]$ cp play fun/
cp: не указан -r; пропускается каталог 'play'
[dlatihpova@fedora ~]$ cp -r play fun/
```

Рис. 3.21: 4.4

4.5 Переместила каталог fun в каталог play, назвав его games(рис. 3.22).

mv fun play/games

```
[dlatihpova@fedora ~]$ mv fun play/games
```

Рис. 3.22: 4.5

4.6-4.8 Лишила владельца файла feathers права на чтение. После чего, попыталась просмотреть данный файл и скопировать его, однако, как и ожидалось, нам было отказано в доступе(рис. 3.23).

1 chmod -r feathers

2 cat feathers

3 cp feathers fun

```
[dlatihpova@fedora ~]$ chmod -r feathers
[dlatihpova@fedora ~]$ cat feathers
cat: feathers: Отказано в доступе
[dlatihpova@fedora ~]$ cp feathers fun
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
```

Рис. 3.23: 4.6-4.8

4.9 Я снова дала владельцу права на чтение файла feathers(рис. 3.24).

chmod +r feathers

```
[dlatihpova@fedora ~]$ chmod +r feathers  
[dlatihpova@fedora ~]$ cat feathers
```

Рис. 3.24: 4.9

4.10-4.11 Затем лишила владельца прав на выполнение каталога play. И попробовала перейти в него, как и ожидалось, нам было отказано в доступе(рис. 3.25).

1 chmod -x play

2 cd play

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

Рис. 3.25: 4.10-4.11

4.12 И наконец, снова дала владельцу права на выполнение каталога play(рис. 3.26).

chmod +x play

```
[dlatihpova@fedora ~]$ chmod +x play  
[dlatihpova@fedora ~]$ cd play
```

Рис. 3.26: 4.12

5. Последним заданием было прочитать каждую из нижеперечисленных команд с помощью man и кратко охарактеризировать(рис. 3.27).

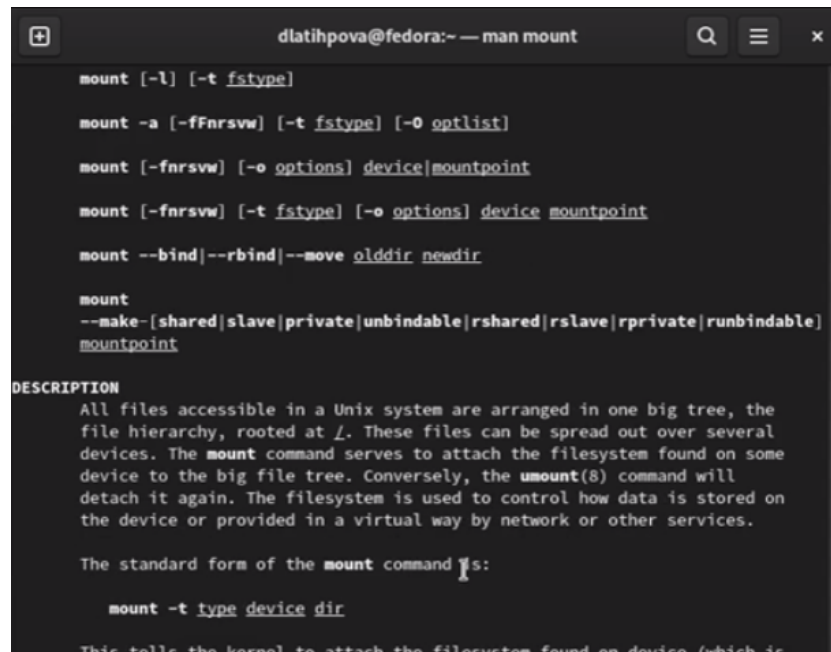
```
[dlatihpova@fedora play]$ cd  
[dlatihpova@fedora ~]$ man mount  
[dlatihpova@fedora ~]$ man fsck  
[dlatihpova@fedora ~]$ man mkfs  
[dlatihpova@fedora ~]$ man kill
```

Рис. 3.27: Характеристика

- mount монтирует файловую систему(рис. 3.28)

mount -h выводит на экран помощь-подсказку программы

mount -bind olddir newdir после выполнения этой команды одно и тот же содержимое становится доступным из двух точек.



```

dlatihpova@fedora:~ — man mount

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

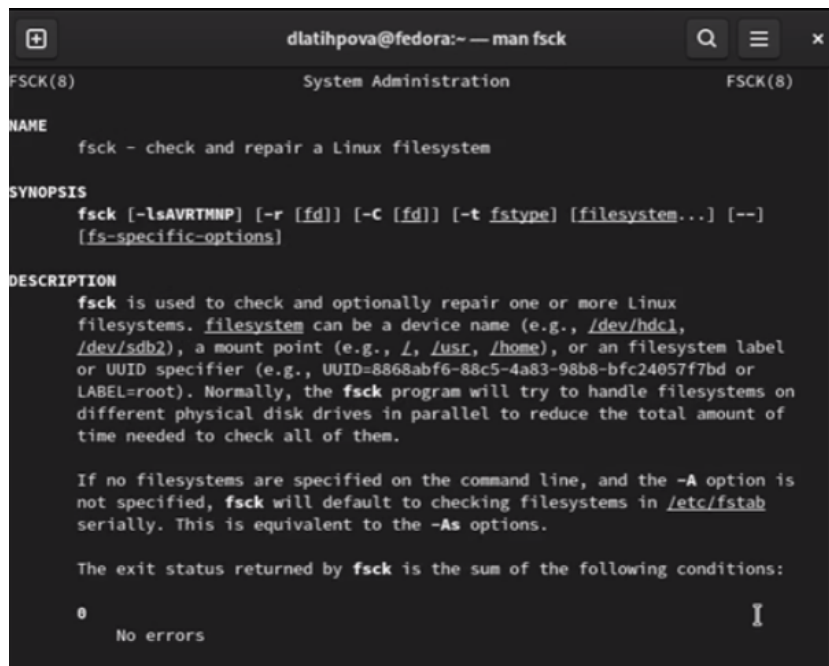
```

Рис. 3.28: mount

- fsck предназначена для проверки и исправления ошибок файловой системы Linux. (рис. 3.29)

Perform fsck dry run — это выполнение тестового запуска.

fsck -AR -запуск fsck во всех файловых системах



```

dlatihpova@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 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                                I
    No errors

```

Рис. 3.29: fsck

- mkfs создаёт новую файловую систему Linux (рис. 3.30)

опция -с перед созданием файловой системы проверяет наличие сбойных блоков на устройстве.

опция -v подробно комментирует происходящее.

```

dlatihpova@fedora:~ — man mkfs
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

```

Рис. 3.30: mkfs

- kill посылает сигнал процессу. (рис. 3.31)

kill -L в результате получим список сигналов и их числовые значения

```

dlatihpova@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.

```

Рис. 3.31: kill

4 Контрольные вопросы

1. Дайте характеристику каждой файловой системе, существующей на жёстком диске компьютера, на котором вы выполняли лабораторную работу.

Если при загрузке чистый бит не установлен, то ОС запускает средство проверки файловой системы. В Linux программа - `fsck`. Программа проверяют целостность файловой системы.

В ФайловОЙ системЕ Ext2 данные сначала кэшируются и только потом записываются на диск, чем достигается высокая производительность.

Файловая система Extfs (Extended File System) расширенная файловая система - первая файловая система, разработанная специально для ОС на ядре Linux. Наибольший возможный размер раздела файла - 2 Гб. Максимальная длина имени файла - 255 символов. Вытеснена файловыми системами Ext2 и Ext3.

2. Приведите общую структуру файловой системы и дайте характеристику каждой директории первого уровня этой структуры.

Файловая система в Linux определяет также организацию расположения файлов, по сути представляя собой иерархическую структуру «дерева»: начинается с корневого каталога «/» и разрастается ветвями в зависимости от работы системы.

ФС также характерно понятие целостности: в такой системе изменения, внесенные в один файл, не приведут к изменению другого файла, не связанного с первым. У всех данных есть собственная физическая память. В Linux целостность ФС проверяется специальной командой — `fsck`.

Первая группа ФС — Extended Filesystem (Ext2, Ext3, Ext4) — является стандартом для Linux. Как следствие, это самые распространенные системы. Они редко обновляются, но зато стабильны. Ext2 создавалась специально под Linux (изначально Extended Filesystem делали еще под Minix).

Эта группа ФС поддерживает наибольшее количество доступных функций из всех, предложенных на рынке. Ext3 (2001 г.) стала еще более стабильной, чем ее предшественница, за счет использования журналирования, а версия Ext3 (2006 г.) популярна и среди современных пользователей. Среди улучшений: увеличение максимального размера раздела до 1 Эксабайта.

4. Назовите основные причины нарушения целостности файловой системы.

Содержимое отдельных файлов не вносит существенных нарушений в целостность файловой системы.

Другое дело, когда подобные несоответствия затрагивают метаданные файла или другую управляющую информацию файловой системы, например, суперблок. Многие файловые операции затрагивают сразу несколько объектов файловой системы, и если на диске будут сохранены изменения только для части этих объектов, целостность файловой системы может быть существенно нарушена.

5. Как создаётся файловая система?

В общем виде команда выглядит так:

mkfs -t ext2 /dev/hda5

Для создания FAT32 можно воспользоваться такой командой:

mkfs -t vfat /dev/hda5

В Debian, чтобы было доступно создание файловых систем FAT и FAT32 требуется установить пакет dosfstools.

6. Дайте характеристику командам для просмотра текстовых файлов.

Cat просто печатает содержимое файла на стандартном экране, т.е. на экране.

Команда `nl` почти как команда `cat`. Разница лишь в том, что она добавляет номера строк при отображении текста в терминале.

Команда `Less` просматривает файл по одной странице за раз. Лучше всего, чтобы выходить меньше (нажимая `q`), на экране не отображаются строки. Терминал останется чистым и нетронутым.

7. Приведите основные возможности команды `cp` в Linux.

`cp` копирует файлы и директории из одного места в другое.

Опция `-r` сохраняет время модификации файла и максимально возможные полномочия. Без этой опции для нового файла задаются полномочия, соответствующие полномочиям запустившего команду пользователя

`-d`, если задать эту опцию, то символические ссылки будут оставаться ссылками (а иначе вместо ссылки копируется файл, на который дается ссылка)

`-f` Перезапись файлов при копировании (если такие уже есть) без дополнительных предупреждений

8. Приведите основные возможности команды `mv` в Linux.

`mv` используется для перемещения одного или нескольких файлов (или директорий) в другую директорию, а также для переименования файлов и директорий.

`-f` или `-force` -при перезаписи существующих файлов не спрашивать разрешения пользователя.

`-i` или `-interactive` -при перезаписи существующих файлов спрашивать разрешение пользователя на подтверждение операции.

`-n` или `-no-clobber` -не перезаписывать существующие файлы.

9. Что такое права доступа? Как они могут быть изменены? При ответах на вопросы используйте дополнительные источники информации по теме.

Права доступа указывают для других пользователей системы, могут ли они читать, изменять и исполнять файл. Права доступа к файлу могут быть изменены

только владельцем файла или пользователем root. Для каждого класса пользователей система устанавливает права доступа в виде набора разрешений.

chmod - это команда для изменения прав доступа к файлам или каталогам в Linux / Unix. Изменяя разрешения, указанный человек может сделать файл читаемым, доступным для записи и запуска, что в значительной степени гарантирует безопасность данных.

5 Выводы

Я ознакомилась с файловой системой Linux, её структурой, именами и содержанием каталогов. Кроме того, 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.