

# **Лабораторная работа 7**

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

Неустроева Ирина Николаевна

# Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	8
4	Контрольные вопросы	19
5	Вывод	26

## Список иллюстраций

3.1	Копирование файла в текущем каталоге . . . . .	8
3.2	Копирование нескольких файлов в каталог . . . . .	8
3.3	Копирование файлов в произвольном каталоге . . . . .	8
3.4	Копирование каталогов в текущем каталоге . . . . .	9
3.5	Копирование каталогов в произвольном каталоге . . . . .	9
3.6	Переименование файлов в текущем каталоге . . . . .	9
3.7	Перемещение файлов в другой каталог . . . . .	9
3.8	Переименование каталогов в текущем каталоге . . . . .	9
3.9	Переименование каталога, не являющегося текущим . . . . .	10
3.10	Создание файла с правами владельца . . . . .	10
3.11	Лишение прав на выполнение . . . . .	10
3.12	Запрет на чтение файла для группы . . . . .	10
3.13	Право записи для группы . . . . .	11
3.14	Проверка на целостность . . . . .	11
3.15	Копирование файла . . . . .	11
3.16	Создание директории и перемещение файла . . . . .	12
3.17	Переименование файла . . . . .	12
3.18	Копирование и переименовывание файла . . . . .	12
3.19	Создание и перемещение файла . . . . .	12
3.20	Создание, перемещение каталога . . . . .	13
3.21	Команда rm . . . . .	13
3.22	Изменение прав на файлы и каталоги . . . . .	14
3.23	Просмотр файла . . . . .	14
3.24	Копирование файла . . . . .	14
3.25	Перемещение файла, каталога . . . . .	15
3.26	Смена прав доступа у файла feathers . . . . .	15
3.27	Смена прав доступа у файла play . . . . .	16
3.28	mount . . . . .	16
3.29	fsck . . . . .	17
3.30	mkfs . . . . .	17
3.31	kill . . . . .	18

## **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:
  - 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.
  - В домашнем каталоге создайте директорию `~/ski.places`.
  - Переместите файл `equipment` в каталог `~/ski.places`.
  - Переименуйте файл `~/ski.places/equipment` в `~/ski.places/equiplist`.
  - Создайте в домашнем каталоге файл `abc1` и скопируйте его в каталог `~/ski.places`, назовите его `equiplist2`.
  - Создайте каталог с именем `equipment` в каталоге `~/ski.places`.
  - Переместите файлы `~/ski.places/equiplist` и `equiplist2` в каталог `~/ski.places/equipment`.
  - 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:
  - `drwxr-r- ... australia`

- `drwx-x-x ... play`
- `-r-xr-r- ... my_os`
- `-rw-rw-r- ... feathers`

4. Прodelайте приведённые ниже упражнения, записывая в отчёт по лабораторной работе используемые при этом команды:

- Просмотрите содержимое файла `/etc/password`.
- Скопируйте файл `~/feathers` в файл `~/file.old`.
- Переместите файл `~/file.old` в каталог `~/play`.
- Скопируйте каталог `~/play` в каталог `~/fun`.
- Переместите каталог `~/fun` в каталог `~/play` и назовите его `games`.
- Лишите владельца файла `~/feathers` права на чтение.
- Что произойдёт, если вы попытаетесь просмотреть файл `~/feathers` командой `cat`?
- Что произойдёт, если вы попытаетесь скопировать файл `~/feathers`?
- Дайте владельцу файла `~/feathers` право на чтение.
- Лишите владельца каталога `~/play` права на выполнение.
- Перейдите в каталог `~/play`. Что произошло?
- Дайте владельцу каталога `~/play` право на выполнение.

5. Прочитайте ман по командам `mount`, `fsck`, `mkfs`, `kill` и кратко их охарактеризуйте, приведя примеры.

### 3 Выполнение лабораторной работы

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

- Скопировали файл ~/abc1 в файл april и в файл may (рис. 3.1).

```
inneustroeva@inneustroeva:~$ cd
inneustroeva@inneustroeva:~$ touch abc1
inneustroeva@inneustroeva:~$ ls
abc1  'feh_002338_000001_capture_20240229232825886(1).bmp'  git-extended  work  Документы  Изображения  'Рабочий стол'
bin   'feh_006234_000002_filelist'                          newdir        Видео  Загрузки  Общедоступные  Шаблоны
inneustroeva@inneustroeva:~$ cp abc1 april
inneustroeva@inneustroeva:~$ cp abc1 may
```

Рис. 3.1: Копирование файла в текущем каталоге

- Скопировали файлы april и may в каталог monthly (рис. 3.2).

```
inneustroeva@inneustroeva:~$ mkdir monthly
inneustroeva@inneustroeva:~$ ls
abc1  bin   april  'feh_002338_000001_capture_20240229232825886(1).bmp'  'feh_006234_000002_filelist'  may  newdir  Видео  Загрузки  Общедоступные  Шаблоны
git-extended  monthly  work  Документы  Изображения  'Рабочий стол'
inneustroeva@inneustroeva:~$ cp april may monthly
inneustroeva@inneustroeva:~$ cd monthly
bash: cd: monthly: Нет такого файла или каталога
inneustroeva@inneustroeva:~$ cd monthly
inneustroeva@inneustroeva:~/monthly$ ls
april  may
inneustroeva@inneustroeva:~/monthly$
```

Рис. 3.2: Копирование нескольких файлов в каталог

- Скопировали файл monthly/may в файл с именем june (рис. 3.3).

```
inneustroeva@inneustroeva:~$ cp monthly/may monthly/june
inneustroeva@inneustroeva:~$ ls monthly
april  june  may
inneustroeva@inneustroeva:~$
```

Рис. 3.3: Копирование файлов в произвольном каталоге

- Скопировали каталог monthly в каталог monthly.00 (рис. 3.4).



```
inneustroeva@inneustroeva:~$ mkdir monthly.00
inneustroeva@inneustroeva:~$ cp -r monthly monthly.00
inneustroeva@inneustroeva:~$ cd monthly.00
inneustroeva@inneustroeva:~/monthly.00$ ls
monthly
inneustroeva@inneustroeva:~/monthly.00$
```

Рис. 3.4: Копирование каталогов в текущем каталоге

- Скопировали каталог monthly.00 в каталог /tmp (рис. 3.5).

```
inneustroeva@inneustroeva:~$ cp -r monthly.00 /tmp
inneustroeva@inneustroeva:~$ ls /tmp
monthly.00
ssh-XXXXXXlKIWzQ
systemd-private-07a1b481523b4924893585fce9c6ac8f-chrond.service-aPhynk
systemd-private-07a1b481523b4924893585fce9c6ac8f-dbus-broker.service-0A5fvM
systemd-private-07a1b481523b4924893585fce9c6ac8f-ModemManager.service-0WC91R
```

Рис. 3.5: Копирование каталогов в произвольном каталоге

- Изменили название файла april на july в домашнем каталоге (рис. 3.6).

```
inneustroeva@inneustroeva:~$ mv april july
inneustroeva@inneustroeva:~$ ls
abc1 'feh_002838_000001_capture_20240229232825886(1).bmp' git-extended may monthly.00 work Документы Изображения
bin 'feh_006234_000002_filelist' july monthly newdir Видео Загрузки Общедоступные
```

Рис. 3.6: Переименование файлов в текущем каталоге

- Переместили файл july в каталог monthly.00 (рис. 3.7).

```
inneustroeva@inneustroeva:~$ mv july monthly.00
inneustroeva@inneustroeva:~$ ls monthly.00
july monthly
```

Рис. 3.7: Перемещение файлов в другой каталог

- Следующим шагом, переименовали каталог monthly.00 в monthly.01 (рис. 3.8).

```
inneustroeva@inneustroeva:~$ mv monthly.00 monthly.01
inneustroeva@inneustroeva:~$ ls
abc1 'feh_002838_000001_capture_20240229232825886(1).bmp' git-extended monthly newdir Видео Загрузки Общедоступные
bin 'feh_006234_000002_filelist' may monthly.01 work Документы Изображения 'Рабочий стол'
```

Рис. 3.8: Переименование каталогов в текущем каталоге

- Далее, переименовали каталог reports/monthly.01 в reports/monthly (рис. 3.9).

```
inneustroeva@inneustroeva:~$ mv reports/monthly.01 reports/monthly
inneustroeva@inneustroeva:~$ ls reports
monthly
```

Рис. 3.9: Переименование каталога, не являющегося текущим

- Создали файл ~/may с правом выполнения для владельца (рис. 3.10).

```
inneustroeva@inneustroeva:~$ touch may
inneustroeva@inneustroeva:~$ ls -l may
-rw-r--r--. 1 inneustroeva inneustroeva 0 мар 19 21:02 may
inneustroeva@inneustroeva:~$ chmod u+x may
inneustroeva@inneustroeva:~$ ls -l may
-rwxr--r--. 1 inneustroeva inneustroeva 0 мар 19 21:03 may
```

Рис. 3.10: Создание файла с правами владельца

- Лишили владельца файла ~/may права на выполнение (рис. 3.11).

```
inneustroeva@inneustroeva:~$ chmod u-x may
inneustroeva@inneustroeva:~$ ls -l may
-rw-r--r--. 1 inneustroeva inneustroeva 0 мар 19 21:02 may
inneustroeva@inneustroeva:~$
```

Рис. 3.11: Лишение прав на выполнение

- Создали каталог monthly с запретом на чтение для членов группы и всех остальных пользователей:(рис. 3.12).

```
inneustroeva@inneustroeva:~$ mkdir monthly
mkdir: невозможно создать каталог «monthly»: Файл существует
inneustroeva@inneustroeva:~$ chmod g-r, o-r monthly
chmod: неверный режим: «g-r,»
По команде «chmod --help» можно получить дополнительную информацию.
inneustroeva@inneustroeva:~$ chmod g-r,o-r monthly
```

Рис. 3.12: Запрет на чтение файла для группы

Создали файл ~/abc1 с правом записи для членов группы рис. 3.13).

```

inneustroeva@inneustroeva:~$ ls
abc1  'feh_002838_000001_capture_20240229232825886(1).bmp'  git-extensions
bin   feh_006234_000002_filelist                             may
inneustroeva@inneustroeva:~$ chmod g+w abc1

```

Рис. 3.13: Право записи для группы

С помощью команды `fsck` проверяем целостность файловой системы: (рис. 3.14).

```

inneustroeva@inneustroeva:~$ sudo -i
root@inneustroeva:~# fsck /dev/sda1
fsck from util-linux 2.39.3
e2fsck 1.47.0 (5-Feb-2023)
ext2fs_open2: Bad magic number in super-block
fsck.ext2: Superblock invalid, trying backup blocks...
fsck.ext2: Bad magic number in super-block while trying to open /dev/sda1

The superblock could not be read or does not describe a valid ext2/ext3/ext4
filesystem. If the device is valid and it really contains an ext2/ext3/ext4
filesystem (and not swap or ufs or something else), then the superblock
is corrupt, and you might try running e2fsck with an alternate superblock:
    e2fsck -b 8193 <device>
or
    e2fsck -b 32768 <device>

```

Рис. 3.14: Проверка на целостность

## 2. Выполните следующие действия

- Скопировали файл `/usr/include/sys/io.h` в домашний каталог и назвали его `equipment`(рис. 3.15).

```

inneustroeva@inneustroeva:~$ mkdir equipment
inneustroeva@inneustroeva:~$ cp /usr/include/sys/io.h ~/equipment
inneustroeva@inneustroeva:~$ cd equipment
inneustroeva@inneustroeva:~/equipment$ ls
io.h
inneustroeva@inneustroeva:~/equipment$ cd
inneustroeva@inneustroeva:~$ 

```

Рис. 3.15: Копирование файла

- В домашнем каталоге создали директорию `~/ski.places`. Далее переместили файл `equipment` в каталог `~/ski.places`.(рис. 3.16)

```

inneustroeva@inneustroeva:~$ mv equipment ~/ski.plases
inneustroeva@inneustroeva:~$ ls equipment
ls: невозможно получить доступ к 'equipment': Нет такого файла или каталога
inneustroeva@inneustroeva:~$ ls ski.plases
equipment
inneustroeva@inneustroeva:~$

```

Рис. 3.16: Создание директории и перемещение файла

- Переименовали файл ~/ski.plases/equipment в ~/ski.plases/equiplist(рис. 3.17)

```

inneustroeva@inneustroeva:~$ mv ski.plases/equipment ski.plases/equiplist
inneustroeva@inneustroeva:~$ ls
abc1  'feh_002838_000001_capture_20240229232825886(1).bmp'  git-extended  monthly  reports  work  Документы  Изображ
bin   feh_006234_000002_filelist                          may           newdir   ski.plases  Видео  Загрузки  Общедос

```

Рис. 3.17: Переименование файла

- Создали в домашнем каталоге файл abc1 и скопировали его в каталог ~/ski.plases, назвали его equiplist2.(рис. 3.18)

```

inneustroeva@inneustroeva:~$ cp abc1 ski.plases
inneustroeva@inneustroeva:~$ ls
abc1  'feh_002838_000001_capture_20240229232825886(1).bmp'  git-extended  monthly  reports  work  Документы  Изображения
bin   feh_006234_000002_filelist                          may           newdir   ski.plases  Видео  Загрузки  Общедоступные
inneustroeva@inneustroeva:~$ cd ski.plases
inneustroeva@inneustroeva:~/ski.plases$ ls
abc1  equiplist  equiplist2
inneustroeva@inneustroeva:~/ski.plases$ mv abc1 equiplist2
inneustroeva@inneustroeva:~/ski.plases$ ls equiplist2
equiplist2

```

Рис. 3.18: Копирование и переименовывание файла

- Создали каталог с именем equipment в каталоге ~/ski.plases. Переместили файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment(рис. 3.19)

```

inneustroeva@inneustroeva:~/ski.plases$ mkdir equipment
inneustroeva@inneustroeva:~/ski.plases$ ls
equiplist  equiplist2  equipment
inneustroeva@inneustroeva:~/ski.plases$ cd
inneustroeva@inneustroeva:~$ mv ski.plases/equiplist ski.plases/equipment
inneustroeva@inneustroeva:~$ mv ski.plases/equiplist2 ski.plases/equipment
inneustroeva@inneustroeva:~$ ls equipment
ls: невозможно получить доступ к 'equipment': Нет такого файла или каталога
inneustroeva@inneustroeva:~$ cd equipment
bash: cd: equipment: Нет такого файла или каталога
inneustroeva@inneustroeva:~$ cd ski.plases/equipment
inneustroeva@inneustroeva:~/ski.plases/equipment$ ls
equiplist  equiplist2

```

Рис. 3.19: Создание и перемещение файла

- Создали и переместили каталог ~/newdir в каталог ~/ski.places и назвали его plans (рис. 3.20).

```
inneustroeva@inneustroeva:~$ ls
abc1  'feh_002838_000001_capture_20240229232825886(1).bmp'  git-extended  monthly  reports  work
bin   feh_006234_000002_filelist                          may           newdir   ski.places  Видео
inneustroeva@inneustroeva:~$ mv newdir ski.places
inneustroeva@inneustroeva:~$ cd ski.places
inneustroeva@inneustroeva:~/ski.places$ ls
equipment  newdir
inneustroeva@inneustroeva:~/ski.places$ mv newdir plans
inneustroeva@inneustroeva:~/ski.places$ ls
equipment  plans
```

Рис. 3.20: Создание, перемещение каталога

3. Определите опции команды chmod, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет

- Создали нужные каталоги australia, play, my\_os, feathers (рис. 3.21)

```
inneustroeva@inneustroeva:~/ski.places$ cd
inneustroeva@inneustroeva:~$ mkdir australia
inneustroeva@inneustroeva:~$ mkdir play
inneustroeva@inneustroeva:~$ touch my_os
inneustroeva@inneustroeva:~$ touch feathers
```

Рис. 3.21: Команда rm

- Далее, изменили необходимые права доступа для созданных файлов на требуемые: drwxr-r- ... australia, drwx-x-x ... play, -r-xr-r- ... my\_os, -rw-rw-r- ... feathers, с помощью команды chmod (рис. 3.22).

```

inneustroeva@inneustroeva:~$ chmod u+x my_os
inneustroeva@inneustroeva:~$ ls -l
итого 288
-rw-rw-r--. 1 inneustroeva inneustroeva 0 мар 19 20:35 abc1
drwxr--r--. 1 inneustroeva inneustroeva 0 мар 19 22:02 australia
drwxr-xr-x. 1 inneustroeva inneustroeva 8 мар 1 18:55 bin
-rw-rw-r--. 1 inneustroeva inneustroeva 0 мар 19 22:03 feathers
-rw-rw-r--. 1 inneustroeva inneustroeva 287554 фев 29 23:59 'feh_002838_000001_capture_20240229232825886(1).bmp'
-rw-rw-r--. 1 inneustroeva inneustroeva 66 фев 22 19:04 feh_006234_000002_filelist
drwxr-xr-x. 1 inneustroeva inneustroeva 74 мар 9 16:09 git-extended
-rw-rw-r--. 1 inneustroeva inneustroeva 0 мар 19 21:02 may
drwxr-xr-x. 1 inneustroeva inneustroeva 24 мар 19 20:42 monthly
-rwxr--r--. 1 inneustroeva inneustroeva 0 мар 19 22:02 my_os
drwxr-xr-x. 1 inneustroeva inneustroeva 0 мар 19 22:02 play
drwxr-xr-x. 1 inneustroeva inneustroeva 14 мар 19 20:57 reports
drwxr-xr-x. 1 inneustroeva inneustroeva 28 мар 19 21:58 ski_places
drwxr-xr-x. 1 inneustroeva inneustroeva 62 мар 2 18:42 work
drwxr-xr-x. 1 inneustroeva inneustroeva 0 фев 9 20:47 Видео
drwxr-xr-x. 1 inneustroeva inneustroeva 0 фев 9 20:47 Документы
drwxr-xr-x. 1 inneustroeva inneustroeva 378 мар 14 01:35 Загрузки
drwxr-xr-x. 1 inneustroeva inneustroeva 0 фев 28 19:42 Изображения
drwxr-xr-x. 1 inneustroeva inneustroeva 0 фев 9 20:47 Общедоступные
drwxr-xr-x. 1 inneustroeva inneustroeva 0 фев 9 20:47 'Рабочий стол'

```

Рис. 3.22: Изменение прав на файлы и каталоги

#### 4. Проделайте приведённые ниже упражнения

- Пытались просмотрели содержимое файла /etc/passwd, но его у нас нет (рис. 3.23)

```

inneustroeva@inneustroeva:~$ cat /etc/passwd
cat: /etc/passwd: Нет такого файла или каталога

```

Рис. 3.23: Просмотр файла

- Скопировали файл ~/feathers в файл ~/file.old (рис. 3.24)

```

inneustroeva@inneustroeva:~$ cd /etc
inneustroeva@inneustroeva:~$ cp feathers file.old
inneustroeva@inneustroeva:~$ file.old ls
bash: file.old: команда не найдена

```

Рис. 3.24: Копирование файла

- Переместили файл ~/file.old в каталог ~/play. Скопировали каталог ~/play в каталог ~/fun. Переместили каталог ~/fun в каталог ~/play и назвали его games. (рис. 3.25)



```

inneustroeva@inneustroeva:~$ mv file.old play
inneustroeva@inneustroeva:~$ ls play
file.old
inneustroeva@inneustroeva:~$ cp -r play fun
inneustroeva@inneustroeva:~$ cd fun
inneustroeva@inneustroeva:~/fun$ ls
file.old
inneustroeva@inneustroeva:~/fun$ cd
inneustroeva@inneustroeva:~$ cp -r play fun
inneustroeva@inneustroeva:~$ cd fun
inneustroeva@inneustroeva:~/fun$ ls
file.old  play
inneustroeva@inneustroeva:~/fun$ cd
inneustroeva@inneustroeva:~$ mv fun play
inneustroeva@inneustroeva:~$ cd play
inneustroeva@inneustroeva:~/play$ ls
file.old  fun
inneustroeva@inneustroeva:~/play$ mv fun game
inneustroeva@inneustroeva:~/play$ ls
file.old  game

```

Рис. 3.25: Перемещение файла, каталога

Лишили владельца файла ~/feathers права на чтение. Далее, попытались посмотреть файл ~/feathers командой cat, нам было отказанно в доступе, потом попытались скопировать файл ~/feathers, тоже было отказанно в доступе. Далее вернули владельцу файла ~/feathers право на чтение (это можно посмотреть на следующей картинке) (рис. 3.26)

```

inneustroeva@inneustroeva:~$ ls -l feathers
-rw-rw-r--. 1 inneustroeva inneustroeva 0 map 19 22:03 feathers
inneustroeva@inneustroeva:~$ chmod u-r feathers
inneustroeva@inneustroeva:~$ ls -l feathers
--w-rw-r--. 1 inneustroeva inneustroeva 0 map 19 22:03 feathers
inneustroeva@inneustroeva:~$ cat feathers
cat: feathers: Отказано в доступе
inneustroeva@inneustroeva:~$ cp feathers
cp: после 'feathers' пропущен операнд, задающий целевой файл
По команде «cp --help» можно получить дополнительную информацию.
inneustroeva@inneustroeva:~$ cp feathers fun
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе

```

Рис. 3.26: Смена прав доступа у файла feathers

Лишили владельца каталога ~/play права на выполнение. Перешли в каталог ~/play, но было отказанно в доступе. Далее вернули владельцу каталога ~/play право на выполнение. (рис. 3.27)

```

inneustroeva@inneustroeva:~$ chmod u+x play
inneustroeva@inneustroeva:~$ 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
-rw-rw-r--r-- 1 inneustroeva inneustroeva 0 map 19 20:35 abc1
drwxr-xr-x 1 inneustroeva inneustroeva 0 map 19 22:02 australia
drwxr-xr-x 1 inneustroeva inneustroeva 8 map 1 18:55 bin
-rw-rw-r--r-- 1 inneustroeva inneustroeva 0 map 19 22:03 feathers
-rw-rw-r--r-- 1 inneustroeva inneustroeva 287554 feb 29 23:59 'feh_002838_000001_capture_20240229232825886(1).bmp'
-rw-rw-r--r-- 1 inneustroeva inneustroeva 66 feb 22 19:04 feh_006234_000002_filelist
drwxr-xr-x 1 inneustroeva inneustroeva 74 map 9 16:09 git-extended
-rw-rw-r--r-- 1 inneustroeva inneustroeva 0 map 19 21:02 may
drwxr-xr-x 1 inneustroeva inneustroeva 24 map 19 20:42 monthly
-rw-rw-r--r-- 1 inneustroeva inneustroeva 0 map 19 22:02 my_os
drwxr-xr-x 1 inneustroeva inneustroeva 26 map 19 22:51 play

```

Рис. 3.27: Смена прав доступа у файла play

5. Прочитайте ман по командам mount, fsck, mkfs, kill и кратко их охарактеризуйте, приведя примеры.

- ман по команде mount (рис. 3.28)

```

MOUNT(8)                                     System Administration
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 glodir newdir

    mount --make-[shared|slave|private|unbindable|rshared|rslave|zprivate|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
    to attach the filesystem found on some device to the big file tree. Conversely, the umount(8) command will detach it again. The filesy
    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 opti
    detect a filesystem. The root permissions are necessary to mount a filesystem by default. See section "Non-superuser mounts" below for
    any) and owner and mode of dir become invisible, and as long as this filesystem remains mounted, the pathname dir refers to the root o
    If only the directory or the device is given, for example:

        mount /dir

```

Рис. 3.28: mount

Команда используется для просмотра используемых в операционной системе файловых систем

- ман по команде fsck (рис. 3.29)



```

fscck(8)                                     System Administration

NAME
     fscck - check and repair a Linux filesystem

SYNOPSIS
     fscck [-lsAVRTMNP] [-r [fd]] [-C [fd]] [-t fstype] [filesystem...] [--] [fs-specific-options]

DESCRIPTION
     fscck is used to check and optionally repair one or more linux filesystems. filesystem can be a device name (e.g., /dev/hdc1, /dev/sda1), a filesystem label or UUID specifier (e.g., UUID=8868abf6-88c5-4a83-90b8-bfc24057f7bd or LABEL=root). Normally, the fscck program checks all 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, fscck will default to checking filesystems on all physical disk drives.

     The exit status returned by fscck 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      I

    16      Usage or syntax error

    32      Checking canceled by user request

   128     Shared-library error

```

Рис. 3.29: fsck

Команда нужна для проверки целостности файловой системы

- man по команде mkfs (рис. 3.30)

```

mkfs(8)                                     System Administration

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/hdc1, /dev/sda1) or a 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 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 output. 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 a normal option.)

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

```

Рис. 3.30: mkfs

Команда используется для создания файловой системы на некотором устройстве.

- man по команде kill (рис. 3.31)

```
KILL(1) User Commands
NAME
    kill - terminate a process

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

    kill -l [number] | -t

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 shows
    (number 9), since a process may install a handler for the TERM signal in order to perform clean-up steps before terminating in an or
    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
    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,
    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
    signal must be specified first, or the argument must be preceded by a '--' option, otherwise it will be taken as the signal
```

Рис. 3.31: kill

Команда передаёт сигнал для прекращения процесса.

## 4 Контрольные вопросы

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

Долгие годы ext2 была файловой системой по умолчанию в GNU/Linux. Ext2 заменила собой Extended File System (вот откуда появилось “Second” в названии). В “новой” файловой системе были исправлены некоторые проблемы, а также убраны ограничения. Отличная стабильность, комплексные инструментальные средства для спасения удаленных файлов, очень долгое время перезагрузки после аварии, есть вероятность частичной или полной потери данных после аварии. Одним из главных недостатков “традиционных” файловых систем, подобных Ext2FS, является низкая сопротивляемость к резким системным сбоям (сбой питания или авария программного обеспечения)

Ext3 (Расширенная файловая система номер три) - является наследником файловой системы Ext2FS. Ext3 совместима с Ext2, но обладает одной новой и очень интересной особенностью – запись. Процесс сохранения объекта происходит прежде чем запись в журнал. В результате мы получаем всегда последовательную файловую систему. Это приводит к тому, что при появлении проблем, проверка и восстановление происходят очень быстро. Время, потраченное на то, чтобы проверить файловую систему таким образом, пропорционально его фактическому использованию и не больше его размера. ReiserFS (Это тоже журналируемая файловая система подобно Ext3FS, но их внутренняя структура радикально отличается. В ReiserFS используется концепция бинарных деревьев (binary-tree), позаимствованная из программного обеспечения баз данных. JFS

(журналируемая файловая система). JFS была разработана и использовалась IBM. Вначале JFS была закрытой системой, но недавно IBM решила открыть доступ для движения свободного программного обеспечения. Внутренняя структура JFS близка к ReiserFS. Средняя стабильность, нет комплексных инструментальных средств для спасения удаленных файлов, очень быстрая перезагрузка после аварии, очень хорошее восстановление данных после аварии.

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

Загрузочный блок занимает первый блок файловой системы. Только корневая файловая система имеет активный загрузочный блок, хотя место для него резервируется в каждой файловой системе. Суперблок располагается непосредственно за загрузочным блоком и содержит самую общую информацию о ФС (размер ФС, размер области индексных дескрипторов, их число, список свободных блоков, свободные индексные дескрипторы и т. д.). Суперблок всегда находится в оперативной памяти. Различные версии ОС Unix способны поддерживать разные типы файловых систем. Поэтому у структуры суперблока могут быть варианты (сведения о свободных блоках, например, часто хранятся не как список, а как шкала бит), но суперблок всегда располагается за загрузочным блоком. При монтировании файловой системы в оперативной памяти создается копия ее суперблока. Все последующие операции по созданию и удалению файлов влекут изменения копии суперблока в оперативной памяти. Эта копия периодически записывается на магнитный диск. Обычно причиной повреждения файловой системы является отключение электропитания (или зависание ОС) в тот момент, когда система производит копирование суперблока из оперативной памяти на магнитный диск. - Область индексных дескрипторов содержит описатели файлов (inode). С каждым файлом связан один inode, но одному inode может соответствовать несколько файлов. Binode хранится вся информация о файле, кроме его имени. Область индексных дескрипторов имеет фиксированный формат

и располагается непосредственно за суперблоком. Общее число описателей и, следовательно, максимальное число файлов задается в момент создания файловой системы. Описатели нумеруются натуральными числами. Первый описатель используется ОС для описания специального файла (файла «Плохих блоков»). То есть поврежденные блоки раздела рассматриваются ОС как принадлежащие к специальному файлу и поэтому считаются «занятыми». Второй – описывает корневой каталог файловой системы. - В области данных расположены как обычные файлы, так и файлы каталогов (в том числе корневой каталог). Специальные файлы представлены в ФС только записями в соответствующих каталогах и индексными дескрипторами специального формата, т. е. места в области памяти не занимают.

3. Какая операция должна быть выполнена, чтобы содержимое некоторой файловой системы было доступно операционной системе?

Команда `cat` - позволяет вывести на экран содержимое любого файла, однако в таком виде эта команда практически не используется. Если файл слишком большой, то его содержимое пролистается на экране, а Вы увидите только последние строки файла. С помощью этой команды можно комбинировать и объединять копии файлов, а также создавать новые файлы. Если набрать просто в командной строке `cat` и нажать `Enter`, то можно вводить (и соответственно видеть) текст на экране. Повторное нажатие клавиши `Enter` удвоит строку и позволит начать следующую. Когда текст набран, следует одновременно нажать клавиши `Ctrl` и `d`.

4. Назовите основные причины нарушения целостности файловой системы. Как устранить повреждения файловой системы?

Некорректность файловой системы может возникать: В результате насильственного прерывания операций ввода-вывода, выполняемых непосредственно с диском. В результате нарушения работы дискового кэша. Кэширование

данных с диска предполагает, что в течение некоторого времени результаты операций ввода-вывода никак не сказываются на содержимом диска — все изменения происходят с копиями блоков диска, временно хранящихся в буферах оперативной памяти (в этих буферах оседают данные из пользовательских файлов и служебная информация файловой системы, такая как каталоги, индексные дескрипторы, списки свободных, занятых и поврежденных блоков и т. п.)

## 5. Как создаётся файловая система?

Общее дерево файлов и каталогов системы Linux формируется из отдельных “ветвей”, соответствующих различным физическим носителям. В UNIX нет понятия “форматирования диска” (и команды форматирования), а используется понятие “создание файловой системы”. Когда мы получаем новый носитель, например, жесткий диск, мы должны создать на нем файловую систему. То есть каждому носителю ставится в соответствие отдельная файловая система. Чтобы эту файловую систему использовать для записи в нее файлов, надо ее вначале подключить в общее дерево каталогов (“смонтировать”). Вот и получается, что можно говорить о монтировании файловых систем или о монтировании носителей (с созданными на них файловыми системами). Например, создается файловая система типа ext2fs. Создание файловой системы типа ext2fs подразумевает создание в данном разделе на диске суперблока, таблицы индексных дескрипторов и совокупности блоков данных. Делается все это все с помощью команды mkfs. В простейшем случае достаточно дать эту команду в следующем формате:

[root]# mkfs -t ext2 /dev/hda5, где /dev/hda5 надо заменить указанием на соответствующее устройство или раздел. Например, если вы хотите создать файловую систему на дискете, то команда примет вид:

```
[root]# mkfs -t ext2 /dev/fd0
```

После выполнения команды mkfs в указанном разделе будет создана файловая система ext2fs. В новой файловой системе автоматически создается один каталог с именем lost+found. Он используется в экстренных случаях программой

fsck, поэтому не удаляйте его. Для того, чтобы начать работать с новой файловой системой, необходимо подключить ее в общее дерево каталогов, что делается с помощью команды `mount`. В качестве параметров команде `mount` надо, как минимум, указать устройство и “точку монтирования”. Точкой монтирования называется тот каталог в уже существующем и известном системе дереве каталогов, который будет теперь служить корневым каталогом для подключаемой файловой системы. После монтирования файловой системы в каталог `/mnt/disk2` прежнее содержимое этого каталога станет для вас недоступно до тех пор, пока вы не размонтируете вновь подключенную файловую систему. Прежнее содержимое не уничтожается, а просто становится временно недоступным. Поэтому в качестве точек монтирования лучше использовать пустые каталоги (заранее заготовленные).

6. Дайте характеристику командам, которые позволяют просмотреть текстовые файлы.

Для просмотра небольших файлов удобно пользоваться командой `cat`. Формат команды: `cat имя-файла`

Для просмотра больших файлов используйте команду `less` — она позволяет осуществлять постраничный просмотр файлов (длина страницы соответствует размеру экрана). Формат команды: `less имя-файла`.

Для просмотра начала файла можно воспользоваться командой `head`. По умолчанию она выводит первые 10 строк файла. Формат команды: `head [-n] имя-файла`, где `n` — количество выводимых строк.

Команда `tail` выводит несколько (по умолчанию 10) последних строк файла. Формат команды: `tail [-n] имя-файла`, где `n` — количество выводимых строк.

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

**Копирование отдельных файлов** Для копирования файла следует использовать утилиту `cp` с аргументами, представленными путями к исходному и целевому файлам.

Копирование файлов в другую директорию В том случае, если в качестве пути к целевому файлу используется путь к директории, исходные файлы будут скопированы в эту целевую директорию.

Команда `cp -r` Для копирования директорий целиком следует использовать команду `cp -r` (параметр `-r` позволяет осуществлять рекурсивное копирование всех файлов из всех поддиректорий).

Копирование множества файлов в директорию Вы также можете использовать утилиту `cp` для копирования множества файлов в одну директорию. В этом случае последний аргумент (аргумент, указывающий на цель) должен быть представлен путем к директории.

Команда `cp -i` Для предотвращения перезаписи существующих файлов в ходе использования утилиты `cp` следует использовать параметр `-i` (для активации интерактивного режима копирования).

#### 8. Назовите и дайте характеристику командам перемещения и переименования файлов и каталогов.

Команды `mv` и `mkdir` предназначены для перемещения и переименования файлов и каталогов. Формат команды `mv`: `mv [-опции] старый_файл новый_файл` Примеры: \* Переименование файлов в текущем каталоге. Изменить название файла `april` на `july` в домашнем каталоге: `cd mv april july` \* Перемещение файлов в другой каталог. Переместить файл `july` в каталог `monthly.00`: `mv july monthly.00 ls monthly.00` Результат: `april july june may`. Если необходим запрос подтверждения о перезаписи файла, то нужно использовать опцию `i`. \* Переименование каталогов в текущем каталоге. Переименовать каталог `monthly.00` в `monthly.01` `mv monthly.00 monthly.01` \* Перемещение каталога в другой каталог. Переместить каталог `monthly.01` в каталог `reports`: `mkdir reports mv monthly.01 reports` \* Переименование каталога, не являющегося текущим. Переименовать каталог `reports/monthly.01` в `reports/monthly`: `mv reports/monthly.01 reports/monthly`



9. Что такое права доступа? Как они могут быть изменены? Права доступа — совокупность правил, регламентирующих порядок и условия доступа субъекта к объектам информационной системы (информации, её носителям, процессам и другим ресурсам). Права доступа к файлу или каталогу можно изменить, воспользовавшись командой `chmod`. Сделать это может владелец файла (или каталога) или пользователь с правами администратора. Формат команды: `chmod режим имя_файла` Режим (в формате команды) имеет следующие компоненты структуры и способ записи: = установить право - лишить права + дать право r чтение w запись x выполнение u (user) владелец файла g (group) группа, к которой принадлежит владелец файла o (others) все остальные В работе с правами доступа можно использовать их цифровую запись (восьмеричное значение) вместо символьной

## 5 Вывод

В ходе выполнения лабораторной работы мы ознакомились с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобрели практические навыки по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.