

Отчёт по лабораторной работе 5

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

Гисматуллин Артём Вадимович НПИбд-01-22

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	8
3.1	Права доступа к файлам	8
4	Выполнение лабораторной работы	10
5	Контрольные вопросы	19
6	Выводы	25

Список иллюстраций

4.1	Командная строка. Домашний каталог - создание и копирование файлов	10
4.2	Командная строка. Домашний каталог - перемещение и просмотр файлов	11
4.3	Командная строка. Домашний каталог - создание и перемещение файлов	11
4.4	Командная строка. Домашний каталог - создание ski.plases	12
4.5	Командная строка. Домашний каталог - изменение ski.plases . . .	12
4.6	Командная строка. ski.plases - создание equipment	13
4.7	Командная строка. ski.plases - создание plans	13
4.8	Командная строка. Создание файлов	14
4.9	Командная строка. Изменение прав файлов	14
4.10	Командная строка. Содержимое /etc/passwd	15
4.11	Командная строка. Изменение домашнего каталога	15
4.12	Командная строка. Изменение прав feathers	16
4.13	Командная строка. Изменение прав play	16
4.14	Командная строка. man mount	17
4.15	Командная строка. man fsck	17
4.16	Командная строка. man mkfs	18
4.17	Командная строка. man 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` При необходимости создайте нужные файлы.
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 Теоретическое введение

3.1 Права доступа к файлам

ОС GNU/Linux является многопользовательской операционной системой. И для обеспечения защиты данных одного пользователя от действий других пользователей существуют специальные механизмы разграничения доступа к файлам. Кроме ограничения доступа, данный механизм позволяет разрешить другим пользователям доступ данным для совместной работы.

Права доступа определяют набор действий (чтение, запись, выполнение), разрешённых для выполнения пользователям системы над файлами. Для каждого файла пользователь может входить в одну из трех групп: владелец, член группы владельца, все остальные. Для каждой из этих групп может быть установлен свой набор прав доступа. Владелцем файла является его создатель.

Набор прав доступа задается тройками битов и состоит из прав на чтение, запись и исполнение файла. В символьном представлении он имеет вид строк `gwx`, где вместо любого символа может стоять дефис. Всего возможно 8 комбинаций. Буква означает наличие права (установлен в единицу второй бит триады `r` — чтение, первый бит `w` — запись, нулевой бит `x` — исполнение), а дефис означает отсутствие права (нулевое значение соответствующего бита). Также права доступа могут быть представлены как восьмеричное число. Так, права доступа `rw` (чтение и запись, без исполнения) понимаются как три двоичные цифры `110` или как восьмеричная цифра `6`.

Для изменения прав доступа служит команда `chmod`, которая понимает как

символьное, так и числовое указание прав.

4 Выполнение лабораторной работы

1. Сперва необходимо выполнить все примеры. Для этого наем с создания, копирования каталогов (рис. [4.1]), (рис. [4.2]), (рис. [4.3]).

```
avgismatullin@dk4n62 ~$ cd
avgismatullin@dk4n62 ~$ touch abc1
avgismatullin@dk4n62 ~$ cp abc1 april
avgismatullin@dk4n62 ~$ cp abc1 may
avgismatullin@dk4n62 ~$ ls
abc1  bin      may      public  PycharmProjects  work  Документы  Изображения  Общедоступные  Шаблоны
april  GNUstep  pandoc-2.18  public_html  tmp              Видео  Загрузки    Музыка        'Рабочий стол'
avgismatullin@dk4n62 ~$ mkdir monthly
avgismatullin@dk4n62 ~$ cp april may monthly
avgismatullin@dk4n62 ~$ cp monthly/may monthly/june
avgismatullin@dk4n62 ~$ ls monthly
april  june  may
avgismatullin@dk4n62 ~$
```

Рис. 4.1: Командная строка. Домашний каталог - создание и копирование файлов

```
avgismatullin@dk4n62 ~$ mkdir monthly.00
avgismatullin@dk4n62 ~$ ls
abc1  bin  may  monthly.00  public  PycharmProjects  work  Документы  Изображения  Общедоступные  Шаблоны
april  GNUstep  monthly  pandoc-2.18  public_html  tmp  Видео  Загрузки  Музыка  'Рабочий стол'
avgismatullin@dk4n62 ~$ cp -r monthly monthly.00
avgismatullin@dk4n62 ~$ cp -r monthly.00 /tmp
avgismatullin@dk4n62 ~$ ls /tmp
avgismatullin  monthly.00  systemd-private-e137d7c275ca488dbf766e4d2ccf163d-color.service-F0Rdgy  Temp-d8dac2b7-4b7d-49b4-b0ee-1b298bd302bb
krb5cc_4864_0FLA13  pulse-PKdhtXMmr18n  systemd-private-e137d7c275ca488dbf766e4d2ccf163d-systemd-logind.service-H7909i  tmux-0
krb5cc_4864_r185A3  root  systemd-private-e137d7c275ca488dbf766e4d2ccf163d-upower.service-faABZP
avgismatullin@dk4n62 ~$ ls montly.00
ls: невозможно получить доступ к 'montly.00': Нет такого файла или каталога
avgismatullin@dk4n62 ~$ ls monthly.00
monthly
avgismatullin@dk4n62 ~$
```

Рис. 4.2: Командная строка. Домашний каталог - перемещение и просмотр файлов

```
avgismatullin@dk4n62 ~$ cd
2 mv april july
bash: 2: команда не найдена
avgismatullin@dk4n62 ~$ mv april july
avgismatullin@dk4n62 ~$ mv july monthly.00
avgismatullin@dk4n62 ~$ ls monthly.00
july  monthly
avgismatullin@dk4n62 ~$ mv monthly.00 monthly.01
avgismatullin@dk4n62 ~$ mkdir reports
mv monthly.01 reports
avgismatullin@dk4n62 ~$ mv reports/monthly.01 reports/monthly
avgismatullin@dk4n62 ~$ ls reports
monthly
avgismatullin@dk4n62 ~$
```

Рис. 4.3: Командная строка. Домашний каталог - создание и перемещение файлов

2. Далее мы копируем файл `io.h` в `equipment`, создаем каталог `ski.plases` и перемещаем файл в этот каталог (рис. [4.4]):

```

avgismatullin@dk4n62 ~ $ cp /usr/include/sys/io.h equipment
avgismatullin@dk4n62 ~ $ ls equipment
equipment
avgismatullin@dk4n62 ~ $ ls
bin          GNUstep      public       PycharmProjects  work  Документы  Изображения  Общедоступные  Шаблоны
equipment    pandoc-2.18  public_html  tmp              Видео  Загрузки    Музыка        'Рабочий стол'
avgismatullin@dk4n62 ~ $ cd equipment
bash: cd: equipment: Это не каталог
avgismatullin@dk4n62 ~ $ mkdir ski.plases
avgismatullin@dk4n62 ~ $ ls
bin          pandoc-2.18  PycharmProjects  work  Загрузки  Общедоступные
equipment    public        ski.plases        Видео  Изображения  'Рабочий стол'
GNUstep      public_html   tmp              Документы  Музыка        Шаблоны
avgismatullin@dk4n62 ~ $ mv equipment ski.plases
avgismatullin@dk4n62 ~ $ ls ski.plases/
equipment
avgismatullin@dk4n62 ~ $

```

Рис. 4.4: Командная строка. Домашний каталог - создание ski.plases

Затем переименовываем файл equipment в equiplist, создаем файл abc1 -> equiplist2, перемещаем последний в ранее созданный каталог (рис. [4.5]).

```

avgismatullin@dk4n62 ~ $ mv ski.plases/equipment ski.plases/equiplist
avgismatullin@dk4n62 ~ $ ls ski.plases/
equiplist
avgismatullin@dk4n62 ~ $ touch abc1
avgismatullin@dk4n62 ~ $ cp abc1 ski.plases/
avgismatullin@dk4n62 ~ $ mv ski.plases/abc1 ski.plases/equiplist2
avgismatullin@dk4n62 ~ $ ls ski.plases/
equiplist  equiplist2
avgismatullin@dk4n62 ~ $

```

Рис. 4.5: Командная строка. Домашний каталог - изменение ski.plases

Создаем внутри ski.plases директорию equipment и перемещаем туда все содержимое ski.plases (рис. [4.6]).

```

avgismatullin@dk4n62 ~/ski.plases $ ls
equiplist  equiplist2
avgismatullin@dk4n62 ~/ski.plases $ mkdir equipment
avgismatullin@dk4n62 ~/ski.plases $ mv equiplist equipment
avgismatullin@dk4n62 ~/ski.plases $ mv equiplist2 equipment
avgismatullin@dk4n62 ~/ski.plases $ ls equipment/
equiplist  equiplist2
avgismatullin@dk4n62 ~/ski.plases $

```

Рис. 4.6: Командная строка. ski.plases - создание equipment

После этого создаем newdir -> plans и перемещаем тоже в ski.plases (рис. [4.7]):

```

avgismatullin@dk4n62 ~/ski.plases $ cd ..
avgismatullin@dk4n62 ~ $ mkdir newdir
avgismatullin@dk4n62 ~ $ mv newdir ski.plases/
avgismatullin@dk4n62 ~ $ cd ski.plases/
avgismatullin@dk4n62 ~/ski.plases $ mv newdir plans
avgismatullin@dk4n62 ~/ski.plases $ ls
equipment  plans
avgismatullin@dk4n62 ~/ski.plases $

```

Рис. 4.7: Командная строка. ski.plases - создание plans

- Следующим шагом будет определение опций chmod. Для этого создадим файлы my_os, feathers и директории australia, play (рис. [4.8]) и наделим определенными правами их, как указано в задании (рис. [4.9]):

```
avgismatullin@dk4n62 ~ $ mkdir australia
avgismatullin@dk4n62 ~ $ mkdir play
avgismatullin@dk4n62 ~ $ touch my_os
avgismatullin@dk4n62 ~ $ touch feathers
avgismatullin@dk4n62 ~ $ ls
abc1      feathers  pandoc-2.18  public_html  tmp      Документы  Музыка      Шаблоны
australia GNUstep    play         PycharmProjects  work     Загрузки   Общедоступные
bin       my_os     public       ski.places      Видео    Изображения 'Рабочий стол'
```

Рис. 4.8: Командная строка. Создание файлов

```
avgismatullin@dk4n62 ~ $ chmod 744 australia
avgismatullin@dk4n62 ~ $ chmod 711 play
avgismatullin@dk4n62 ~ $ chmod 544 my_os
avgismatullin@dk4n62 ~ $ chmod 644 feathers
avgismatullin@dk4n62 ~ $ ls -l
итого 37
-rw-r--r-- 1 avgismatullin studsci  0 мар  9 15:27 abc1
drwxr--r-- 2 avgismatullin studsci 2048 мар  9 15:31 australia
drwxr-xr-x 2 avgismatullin studsci 2048 фев 21 13:03 bin
-rw-r--r-- 1 avgismatullin studsci  0 мар  9 15:32 feathers
drwxr-xr-x 3 avgismatullin studsci 2048 ноя 14 13:18 GNUstep
-r-xr--r-- 1 avgismatullin studsci  0 мар  9 15:31 my_os
drwxr-xr-x 4 avgismatullin studsci 2048 апр  4 2022 pandoc-2.18
drwx--x--x 2 avgismatullin studsci 2048 мар  9 15:31 play
drwxr-xr-x 3 avgismatullin root    2048 сен  2 2022 public
lrwxr-xr-x 1 avgismatullin root     18 мар  2 21:54 public_html -> public/public_html
drwxr-xr-x 3 avgismatullin studsci 2048 сен  8 15:20 PycharmProjects
drwxr-xr-x 4 avgismatullin studsci 2048 мар  9 15:30 ski.places
drwxr-xr-x 2 avgismatullin studsci 2048 фев 15 16:52 tmp
drwxr-xr-x 6 avgismatullin studsci 2048 фев 21 13:24 work
drwxr-xr-x 2 avgismatullin studsci 2048 сен  7 2022 Видео
drwxr-xr-x 2 avgismatullin studsci 2048 сен  7 2022 Документы
drwxr-xr-x 4 avgismatullin studsci 2048 фев 28 13:36 Загрузки
drwxr-xr-x 3 avgismatullin studsci 2048 ноя 14 13:55 Изображения
drwxr-xr-x 2 avgismatullin studsci 2048 сен  7 2022 Музыка
drwxr-xr-x 2 avgismatullin studsci 2048 сен  7 2022 Общедоступные
drwxr-xr-x 2 avgismatullin studsci 2048 сен  8 16:17 'Рабочий стол'
drwxr-xr-x 2 avgismatullin studsci 2048 сен  7 2022 Шаблоны
avgismatullin@dk4n62 ~ $
```

Рис. 4.9: Командная строка. Изменение прав файлов

Далее просматриваем содержимое /etc/passwd (рис. [4.10]):

```
avgismatullin@dk4n62 ~ $ cat /etc/passwd
root:x:0:0:System user; root:/root:/bin/bash
bin:x:1:1:bin:/bin:/bin/false
daemon:x:2:2:daemon:/sbin:/bin/false
adm:x:3:4:adm:/var/adm:/bin/false
lp:x:4:7:lp:/var/spool/lpd:/bin/false
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 program user:/var/spool/mail:/sbin/nologin
news:x:9:13:news:/usr/lib/news:/bin/false
uucp:x:10:14:uucp:/var/spool/uucppublic:/bin/false
operator:x:11:0:operator:/root:/bin/bash
man:x:13:15:System user; man:/dev/null:/sbin/nologin
postmaster:x:14:12:Postmaster user:/var/spool/mail:/sbin/nologin
cron:x:16:16:A user for sys-process/cronbase:/var/spool/cron:/sbin/nologin
ftp:x:21:21::/home/ftp:/bin/false
sshd:x:22:22:User for ssh:/var/empty:/sbin/nologin
at:x:25:25:at:/var/spool/cron/atjobs:/bin/false
squid:x:31:31:Squid:/var/cache/squid:/bin/false
gdm:x:32:32:User for running GDM:/var/lib/gdm:/sbin/nologin
xfs:x:33:33:X Font Server:/etc/X11/fs:/bin/false
games:x:35:35:games:/usr/games:/bin/bash
named:x:40:40:bind:/var/bind:/bin/false
mysql:x:60:60:MySQL program user:/dev/null:/sbin/nologin
postfix:x:70:70:Postfix SMTP program user:/var/lib/postfix:/bin/false
```

Рис. 4.10: Командная строка. Содержимое /etc/passwd

Затем по заданию: Создаем файлы, директории, перемещаем старые файлы в другие директории, проверяем изменения (рис. [4.11]):

```
avgismatullin@dk4n62 ~ $ cp ~/feathers file.old
avgismatullin@dk4n62 ~ $ ls file.old
file.old
avgismatullin@dk4n62 ~ $ cp file.old play
avgismatullin@dk4n62 ~ $ ls play
file.old
avgismatullin@dk4n62 ~ $ cp play fun
cp: не указан -r; пропускается каталог 'play'
avgismatullin@dk4n62 ~ $ cp -r play fun
avgismatullin@dk4n62 ~ $ ls fun
file.old
avgismatullin@dk4n62 ~ $ mv fun play
avgismatullin@dk4n62 ~ $ cd play
avgismatullin@dk4n62 ~/play $ mv fun games
avgismatullin@dk4n62 ~/play $ ls
file.old  games
avgismatullin@dk4n62 ~/play $
```

Рис. 4.11: Командная строка. Изменение домашнего каталога

После этого отнимем права на чтение feathers у владельца и попытаемся прочесть (рис. [4.12]):

```
avgismatullin@dk4n62 ~ $ chmod u-r feathers
avgismatullin@dk4n62 ~ $ cat feathers
cat: feathers: Отказано в доступе
avgismatullin@dk4n62 ~ $ cp feathers play
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
avgismatullin@dk4n62 ~ $ chmod u+r feathers
```

Рис. 4.12: Командная строка. Изменение прав feathers

Схожим образом действуем с play - отнимаем право на исполнение. В таком случае мы не должны переходить в него, но доступ по-прежнему открыт (рис. [4.13]):

```
avgismatullin@dk4n62 ~ $ chmod o-x play
avgismatullin@dk4n62 ~ $ chmod g-x play
avgismatullin@dk4n62 ~ $ chmod u-x play
avgismatullin@dk4n62 ~ $ cd play
avgismatullin@dk4n62 ~/play $ cd ..
avgismatullin@dk4n62 ~ $ ls -l
итого 37
-rw-r--r-- 1 avgismatullin studsci  0 мар  9 15:27 abc1
drwxr--r-- 2 avgismatullin studsci 2048 мар  9 15:31 australia
drwxr-xr-x 2 avgismatullin studsci 2048 фев 21 13:03 bin
-rw-r--r-- 1 avgismatullin studsci  0 мар  9 15:32 feathers
-rw-r--r-- 1 avgismatullin studsci  0 мар  9 15:37 file.old
drwxr-xr-x 3 avgismatullin studsci 2048 ноя 14 13:18 GNUstep
-r-xr--r-- 1 avgismatullin studsci  0 мар  9 15:31 my_os
drwxr-xr-x 4 avgismatullin studsci 2048 апр  4 2022 pandoc-2.18
drw----- 3 avgismatullin studsci 2048 мар  9 15:38 play
drwxr-xr-x 3 avgismatullin root    2048 сен  2 2022 public
lrwxr-xr-x 1 avgismatullin root    18 мар  2 21:54 public_html -> public/public_html
drwxr-xr-x 3 avgismatullin studsci 2048 сен  8 15:20 PycharmProjects
drwxr-xr-x 4 avgismatullin studsci 2048 мар  9 15:30 ski.plases
drwxr-xr-x 2 avgismatullin studsci 2048 фев 15 16:52 tmp
drwxr-xr-x 6 avgismatullin studsci 2048 фев 21 13:24 work
drwxr-xr-x 2 avgismatullin studsci 2048 сен  7 2022 Видео
drwxr-xr-x 2 avgismatullin studsci 2048 сен  7 2022 Документы
drwxr-xr-x 4 avgismatullin studsci 2048 фев 28 13:36 Загрузки
drwxr-xr-x 3 avgismatullin studsci 2048 ноя 14 13:55 Изображения
drwxr-xr-x 2 avgismatullin studsci 2048 сен  7 2022 Музыка
drwxr-xr-x 2 avgismatullin studsci 2048 сен  7 2022 Общедоступные
drwxr-xr-x 2 avgismatullin studsci 2048 сен  8 16:17 'Рабочий стол'
```

Рис. 4.13: Командная строка. Изменение прав play

5. Финальным шагом будет прочтение с помощью команды man следующих команд: mount, fsck, mkfs, kill. Кратко охарактеризуем эти команды.

Для просмотра используемых в операционной системе файловых систем используется команда mount (рис. [4.14]).


```
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|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:

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

Рис. 4.14: Командная строка. man mount

С помощью команды `fsck` можно проверить (а в ряде случаев восстановить) целостность файловой системы (рис. [4.15]).

```
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      No errors

    1      Filesystem errors corrected

    2      System should be rebooted

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

Рис. 4.15: Командная строка. man fsck

`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`) (рис. [4.16]).

```
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/ndai, /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.

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

Рис. 4.16: Командная строка. man mkfs

Утилита kill отправляет сигнал процессу(-ам), указанному с помощью каждого из операндов идентификатор_процесса. По умолчанию утилита kill отправляет сигнал SIGTERM, но эту настройку по умолчанию можно переопределить путем определения имени сигнала для отправки (рис. [4.17]).

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

NAME
    kill - send a signal to a process

SYNOPSIS
    kill [options] <pid> [...]

DESCRIPTION
    The default signal for kill is TERM. Use -l or -L to list available signals. Particularly useful signals include HUP, INT, KILL, STOP, CONT, and 0. Alternate signals may be specified in three ways: -9, -SIGKILL or -KILL. Negative PID values may be used to choose whole process groups; see the PGID column in ps command output. A PID of -1 is special; it indicates all processes except the kill process itself and init.

OPTIONS
    <pid> [...]
        Send signal to every <pid> listed.

    -<signal>
    -s <signal>
    --signal <signal>
        Specify the signal to be sent. The signal can be specified by using name or number. The behavior of signals is explained in signal(7) manual page.

    -q, --queue value
        Use sigqueue(3) rather than kill(2) and the value argument is used to specify an integer to be sent with the signal. If the receiving process has installed a handler for this signal using the SA_SIGINFO flag to sigaction(2), then it can obtain this data via the si_value field of the siginfo_t

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

Рис. 4.17: Командная строка. man kill

5 Контрольные вопросы

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

Ext2, Ext3, Ext4 или Extended Filesystem - это стандартная файловая система для Linux. Она была разработана еще для Minix. Она самая стабильная из всех существующих, кодовая база изменяется очень редко и эта файловая система содержит больше всего функций. Версия ext2 была разработана уже именно для Linux и получила много улучшений. В 2001 году вышла ext3, которая добавила еще больше стабильности благодаря использованию журналирования. В 2006 была выпущена версия ext4, которая используется во всех дистрибутивах Linux до сегодняшнего дня. В ней было внесено много улучшений, в том числе увеличен максимальный размер раздела до одного экзбайта.

JFS или Journaled File System была разработана в IBM для AIX UNIX и использовалась в качестве альтернативы для файловых систем ext. Сейчас она используется там, где необходима высокая стабильность и минимальное потребление ресурсов. При разработке файловой системы ставилась цель создать максимально эффективную файловую систему для многопроцессорных компьютеров. Также как и ext, это журналируемая файловая система, но в журнале хранятся только метаданные, что может привести к использованию старых версий файлов после сбоев.

ReiserFS - была разработана намного позже, в качестве альтернативы ext3 с улучшенной производительностью и расширенными возможностями. Она была разработана под руководством Ганса Райзера и поддерживает только Linux.

Из особенностей можно отметить динамический размер блока, что позволяет упаковывать несколько небольших файлов в один блок, что предотвращает фрагментацию и улучшает работу с небольшими файлами. Еще одно преимущество - в возможности изменять размеры разделов на лету. Но минус в некоторой нестабильности и риске потери данных при отключении энергии. Раньше ReiserFS применялась по умолчанию в SUSE Linux, но сейчас разработчики перешли на Btrfs.

XFS - это высокопроизводительная файловая система, разработанная в Silicon Graphics для собственной операционной системы еще в 2001 году. Она изначально была рассчитана на файлы большого размера, и поддерживала диски до 2 Терабайт. Из преимуществ файловой системы можно отметить высокую скорость работы с большими файлами, отложенное выделение места, увеличение разделов на лету и незначительный размер служебной информации.

XFS - журналируемая файловая система, однако в отличие от ext, в журнал записываются только изменения метаданных. Она используется по умолчанию в дистрибутивах на основе Red Hat. Из недостатков - это невозможность уменьшения размера, сложность восстановления данных и риск потери файлов при записи, если будет неожиданное отключение питания, поскольку большинство данных находится в памяти.

Btrfs или B-Tree File System - это совершенно новая файловая система, которая сосредоточена на отказоустойчивости, легкости администрирования и восстановления данных. Файловая система объединяет в себе очень много новых интересных возможностей, таких как размещение на нескольких разделах, поддержка подтомов, изменение размера на лету, создание мгновенных снимков, а также высокая производительность. Но многими пользователями файловая система Btrfs считается нестабильной. Тем не менее, она уже используется как файловая система по умолчанию в OpenSUSE и SUSE Linux.

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

/ — root каталог. Содержит в себе всю иерархию системы;

/bin — здесь находятся двоичные исполняемые файлы. Основные общие команды, хранящиеся отдельно от других программ в системе (прим.: pwd, ls, cat, ps);

/boot — тут расположены файлы, используемые для загрузки системы (образ initrd, ядро vmlinuz);

/dev — в данной директории располагаются файлы устройств (драйверов). С помощью этих файлов можно взаимодействовать с устройствами. К примеру, если это жесткий диск, можно подключить его к файловой системе. В файл принтера же можно написать напрямую и отправить задание на печать;

/etc — в этой директории находятся файлы конфигураций программ. Эти файлы позволяют настраивать системы, сервисы, скрипты системных демонов;

/home — каталог, аналогичный каталогу Users в Windows. Содержит домашние каталоги учетных записей пользователей (кроме root). При создании нового пользователя здесь создается одноименный каталог с аналогичным именем и хранит личные файлы этого пользователя;

/lib — содержит системные библиотеки, с которыми работают программы и модули ядра;

/lost+found — содержит файлы, восстановленные после сбоя работы системы. Система проведет проверку после сбоя и найденные файлы можно будет посмотреть в данном каталоге;

/media — точка монтирования внешних носителей. Например, когда вы вставляете диск в дисковод, он будет автоматически смонтирован в директорию /media/cdrom;

/mnt — точка временного монтирования. Файловые системы подключаемых устройств обычно монтируются в этот каталог для временного использования;

/opt — тут расположены дополнительные (необязательные) приложения. Такие программы обычно не подчиняются принятой иерархии и хранят свои файлы в одном подкаталоге (бинарные, библиотеки, конфигурации);

/proc — содержит файлы, хранящие информацию о запущенных процессах и о состоянии ядра ОС;

/root — директория, которая содержит файлы и личные настройки суперпользователя;

/run — содержит файлы состояния приложений. Например, PID-файлы или UNIX-сокеты;

/sbin — аналогично /bin содержит бинарные файлы. Утилиты нужны для настройки и администрирования системы суперпользователем;

/srv — содержит файлы сервисов, предоставляемых сервером (прим. FTP или Apache HTTP);

/sys — содержит данные непосредственно о системе. Тут можно узнать информацию о ядре, драйверах и устройствах;

/tmp — содержит временные файлы. Данные файлы доступны всем пользователям на чтение и запись. Стоит отметить, что данный каталог очищается при перезагрузке;

/usr — содержит пользовательские приложения и утилиты второго уровня, используемые пользователями, а не системой. Содержимое доступно только для чтения (кроме root). Каталог имеет вторичную иерархию и похож на корневой;

/var — содержит переменные файлы. Имеет подкаталоги, отвечающие за отдельные переменные. Например, логи будут храниться в /var/log, кэш в /var/cache, очереди заданий в /var/spool/ и так далее.

- 2023-03-09 16-06-01
3. Какая операция должна быть выполнена, чтобы содержимое некоторой файловой системы было доступно операционной системе?

Монтирование тома.

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

Отсутствие синхронизации между образом файловой системы в памяти и ее

данными на диске в случае аварийного останова может привести к появлению следующих ошибок:

- Один блок адресуется несколькими `inode` (принадлежит нескольким файлам).
- Блок помечен как свободный, но в то же время занят (на него ссылается `onode`).
- Блок помечен как занятый, но в то же время свободен (ни один `inode` на него не ссылается).
- Неправильное число ссылок в `inode` (недостаток или избыток ссылающихся записей в каталогах).
- Несовпадение между размером файла и суммарным размером адресуемых `inode` блоков.
- Недопустимые адресуемые блоки (например, расположенные за пределами файловой системы).
- “Потерянные” файлы (правильные `inode`, на которые не ссылаются записи каталогов).
- Недопустимые или неразмещенные номера `inode` в записях каталогов.

5. Как создаётся файловая система?

`mkfs` - позволяет создать файловую систему Linux.

6. Дайте характеристику командам для просмотра текстовых файлов.

`Cat` - выводит содержимое файла на стандартное устройство вывода

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

`Ср` – копирует или перемещает директорию, файлы.

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

`Mv` - переименовать или переместить файл или директорию

9. Что такое права доступа? Как они могут быть изменены?

Права доступа к файлу или каталогу можно изменить, воспользовавшись командой `chmod`. Сделать это может владелец файла (или каталога) или пользователь с правами администратора.

6 Выводы

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.