

# **Отчёт по лабораторной работе №5**

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

Дарья Сергеевна Кочина

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Задание</b>	<b>5</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>6</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
<b>5</b>	<b>Выводы</b>	<b>25</b>

## Список иллюстраций

4.1	Копирование каталогов и файлов . . . . .	8
4.2	Перемещение и переименование файлов и каталогов . . . . .	9
4.3	Изменение прав доступа . . . . .	9
4.4	Копирование и создание каталогов . . . . .	10
4.5	Копирование и создание каталогов . . . . .	11
4.6	Создание файлов . . . . .	12
4.7	Определение опции <code>chmod</code> . . . . .	12
4.8	Копирование файлов . . . . .	13
4.9	Создание каталогов . . . . .	14
4.10	Изменение прав владельца . . . . .	14
4.11	Ввод команд в терминале . . . . .	16
4.12	<code>man mount</code> . . . . .	16
4.13	<code>man fsck</code> . . . . .	17
4.14	<code>man mkfs</code> . . . . .	17
4.15	<code>man kill</code> . . . . .	17

# 1 Цель работы

Целью данной лабораторной работы является ознакомление с файловой системой Linux, её структурой, именами и содержанием каталогов. А также 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

Ознакомиться с файловой системой Linux, её структурой, именами и содержанием каталогов.

## 3 Теоретическое введение

**Файловая система (file system, ФС)** — важная составляющая любой операционной системы (ОС), отвечающая за организацию, хранение, чтение, запись файлов. От ФС зависит физическая и логическая структура файлов, политика создания и управления ими, максимальный размер файла и длина его имени. Linux поддерживает множество разных file system, включая FAT, FAT32, NTFS из Windows. Но использовать рекомендуем «родные» системы: Ext3, Ext4, ReiserFS, XFS, Btrfs и пр. Особенно если вы намерены работать с облачной инфраструктурой.

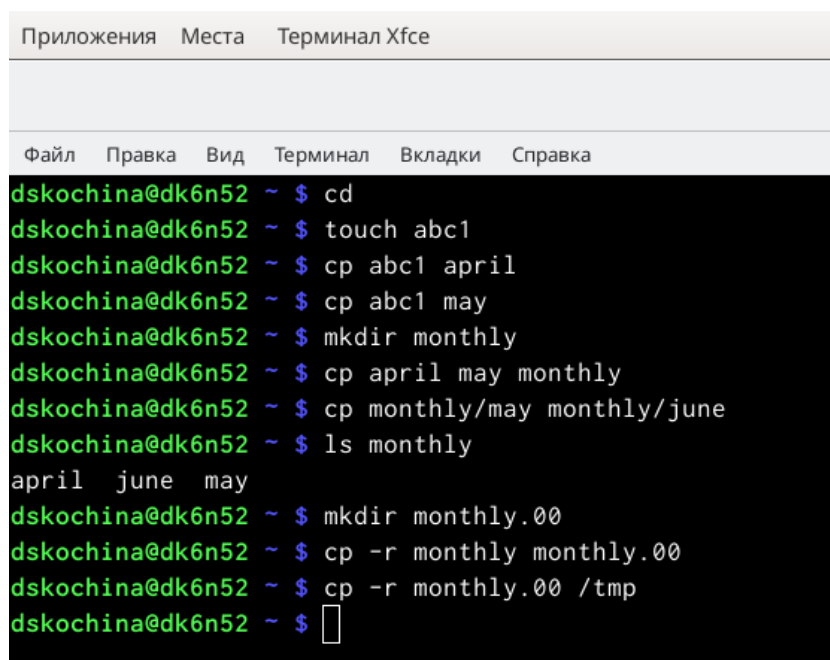
### **Организация файловой системы Linux**

Linux позволяет установить отдельную ФС для каждого раздела. Выбранная система определяет, как быстро будет выполняться работа с файлами, запись и чтение информации. Как в оперативной памяти будет храниться информация (и будет ли она вообще храниться), каким образом можно вносить изменения в конфигурацию ядра ОС — это тоже зависит от ФС.

File system Linux — пространство раздела, состоящее из кратных размеру сектора блоков. Обмен данными производится через VFS или с помощью драйверов. VFS (virtual file system) — это слой абстракции, необходимый для взаимодействия между ядром и софтом. VFS позволяет не думать о специфике работы той или иной ФС. Драйверы ФС обеспечивают взаимодействие между оборудованием (железом) и приложениями.

## 4 Выполнение лабораторной работы

1. Выполнила примеры, описанные в первой части лабораторной работы. (рис. [4.1]) Копирование файла в текущем каталоге. Скопировала файл ~/abc1 в файл april и в файл may: создала файл abc1 (команда “touch abc1”), копировала, используя команду “cp abc1 april” и “cp abc1 may”. Копирование нескольких файлов в каталог. Скопировала файлы april и may в каталог monthly. Копирование файлов в произвольном каталоге. Скопировала файл monthly/may в файл с именем june. Копирование каталогов в текущем каталоге. Скопировала каталог monthly в каталог monthly.00. Копирование каталогов в произвольном каталоге. Скопировала каталог monthly.00 в каталог /tmp.



```
Приложения  Места  Терминал Xfce

Файл  Правка  Вид  Терминал  Вкладки  Справка

dskochina@dk6n52 ~ $ cd
dskochina@dk6n52 ~ $ touch abc1
dskochina@dk6n52 ~ $ cp abc1 april
dskochina@dk6n52 ~ $ cp abc1 may
dskochina@dk6n52 ~ $ mkdir monthly
dskochina@dk6n52 ~ $ cp april may monthly
dskochina@dk6n52 ~ $ cp monthly/may monthly/june
dskochina@dk6n52 ~ $ ls monthly
april  june  may
dskochina@dk6n52 ~ $ mkdir monthly.00
dskochina@dk6n52 ~ $ cp -r monthly monthly.00
dskochina@dk6n52 ~ $ cp -r monthly.00 /tmp
dskochina@dk6n52 ~ $
```

Рис. 4.1: Копирование каталогов и файлов

2. Продолжила выполнять примеры. (рис. [4.2]) Переименование файлов в текущем каталоге. Изменила название файла `april` на `july` в домашнем каталоге. Перемещение файлов в другой каталог. Переместила файл `july` в каталог `monthly.00` (с помощью команды `mv`). Переименование каталогов в текущем каталоге. Переименовала каталог `monthly.00` в `monthly.01`. Перемещение каталога в другой каталог. Переместила каталог `monthly.01` в каталог `reports`. Переименование каталога, не являющегося текущим. Переименовала каталог `reports/monthly.01` в `reports/monthly`.



```

dskochina@dk6n52 ~ $ cd
dskochina@dk6n52 ~ $ mv april july
dskochina@dk6n52 ~ $ mv july monthly.00
dskochina@dk6n52 ~ $ ls monthly.00
july  monthly
dskochina@dk6n52 ~ $ mv monthly.00 monthly.01
dskochina@dk6n52 ~ $ mkdir reports
dskochina@dk6n52 ~ $ mv monthly.01 reports
dskochina@dk6n52 ~ $ mv reports/monthly.01 reports/monthly
dskochina@dk6n52 ~ $ clear

```

Рис. 4.2: Перемещение и переименование файлов и каталогов

3. Продолжила выполнять примеры. (рис. [4.3]) Создала файл /may с правом выполнения для владельца. Лишила владельца файла/мауправа на выполнение. Создала каталог monthly с запретом на чтение для членов группы и всех остальных пользователей. Создала файл ~/abc1 с правом записи для членов группы.

```

dskochina@dk6n52 ~ $ cd
dskochina@dk6n52 ~ $ touch may
dskochina@dk6n52 ~ $ ls -l may
-rw-r--r-- 1 dskochina studsci 0 map 6 17:26 may
dskochina@dk6n52 ~ $ chmod u-x may
dskochina@dk6n52 ~ $ ls -l may
-rw-r--r-- 1 dskochina studsci 0 map 6 17:26 may
dskochina@dk6n52 ~ $ chmod chmod u+x may
chmod: неверный режим: «chmod»
По команде «chmod --help» можно получить дополнительную информацию.
dskochina@dk6n52 ~ $ cd
dskochina@dk6n52 ~ $ mkdir monthly
mkdir: невозможно создать каталог «monthly»: Файл существует
dskochina@dk6n52 ~ $ chmod go-r monthly
dskochina@dk6n52 ~ $ cd
dskochina@dk6n52 ~ $ touch abc1
dskochina@dk6n52 ~ $ chmod g+w abc1
dskochina@dk6n52 ~ $ clear

```

Рис. 4.3: Изменение прав доступа

4. Выполнила следующие действия. (рис. [4.4], [4.5]) Скопировала файл /usr/include/io.h в домашний каталог (используя команду «cp/usr/include/io.h

~»), и назвала его equipment (команда «mv io.h equipment»). В домашнем каталоге создала директорию ~/ski.plases (с помощью команды «mkdir ski.plases»). Переместила файл equipment в каталог ~/ski.plases(используем команду «mv equipment ski.plases»). Переименовала файл ~/ski.plases/equipment в ~/ski.plases/equiplist (с помощью команды «mvski.plases/equipment ski.plases/equiplist»). Создала в домашнем каталоге файл abc1 (команда «touchabc1») и скопировала его в каталог ~/ski.plases (команда «cp abc1 ski.plases»), назвала его equiplist2 (с помощью команды «mv ski.plases/abc1 ski.plases/equiplist2»). Создала каталог с именем equipment в каталоге ~/ski.plases (команда «mkdir ski.plases/equipment»). Переместила файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment. Создала (команда «mkdir newdir») и переместила каталог ~/newdir в каталог ~/ski.plases (команда «mv newdir ski.plases») и называла его plans (команда «mv ski.plases/newdir ski.plases/plans»).

```
dskochina@dk6n52 ~ - $ cp /usr/include/sys/io.h ~
dskochina@dk6n52 ~ - $ ls
abc1  blog  io.h  monthly  public_html  tmp  Видео  Загрузки  Музыка  'Рабочий стол'
bin  GNUstep  may  public  reports  work  Документы  Изображения  Общедоступные  Шаблоны
dskochina@dk6n52 ~ - $ mv io.h equipment
dskochina@dk6n52 ~ - $ ls
abc1  blog  GNUstep  monthly  public_html  tmp  Видео  Загрузки  Музыка  'Рабочий стол'
bin  equipment  may  public  reports  work  Документы  Изображения  Общедоступные  Шаблоны
dskochina@dk6n52 ~ - $ mkdir ski.plases
dskochina@dk6n52 ~ - $ cp equipment ski.plases
dskochina@dk6n52 ~ - $ mv equipment ~/ski.plases
dskochina@dk6n52 ~ - $ ls ski.plases
equipment
dskochina@dk6n52 ~ - $ mv ~/ski.plases/equipment ~/ski.plases/equiplist
dskochina@dk6n52 ~ - $ ls ski.plases
equiplist
dskochina@dk6n52 ~ - $ touch abc1
dskochina@dk6n52 ~ - $ mv abc1 ski.plases
dskochina@dk6n52 ~ - $ mv ski.plases/abc1 ski.plases/equiplist2
dskochina@dk6n52 ~ - $ mkdir ski.plases/equipment
dskochina@dk6n52 ~ - $ mv ski.plases/equiplist ski.plases/equipment
dskochina@dk6n52 ~ - $ mv ski.plases/equiplist2 ski.plases/equipment
dskochina@dk6n52 ~ - $ mkdir newdir
dskochina@dk6n52 ~ - $
```

Рис. 4.4: Копирование и создание каталогов

```

dskochina@dk6n52 ~ $ mkdir newdir
dskochina@dk6n52 ~ $ mv newdir plans
bash: mvnewdir: команда не найдена
dskochina@dk6n52 ~ $ mv newdir plans
dskochina@dk6n52 ~ $ mv plans ski.places
dskochina@dk6n52 ~ $ ls
bin  GNUstep  monthly  public_html  ski.places  work  Документы  Изображения  Общедоступные  Шаблоны
blog  may     public   reports      tmp         Видео  Загрузки  Музыка      'Рабочий стол'
dskochina@dk6n52 ~ $ ls ski.places
equipment  plans
dskochina@dk6n52 ~ $

```

Рис. 4.5: Копирование и создание каталогов

5. Определила опции команды `chmod`, необходимые для того, чтобы присвоить соответствующим файлам выделенные права доступа, считая, что в начале таких прав нет. Предварительно создала необходимые файлы с помощью команд: «`mkdir australia`», «`mkdir play`», «`touch my_os`», «`touch feathers`». (рис. [4.6], [4.7])
  - 1) `drwxr-r...` `australia`: команда «`chmod 744 australia`» (это каталог, владелец имеет право на чтение, запись и выполнение, группа владельца и остальные – только чтение).
  - 2) `drwx-x-x ...` `play`: команда «`chmod 711 play`» (это каталог, владелец имеет право на чтение, запись и выполнение, группа владельца и остальные – только выполнение).
  - 3) `-r-xr-r...` `my_os`: команды «`chmod 544 my_os`» (это файл, владелец имеет право на чтение и выполнение, группа владельца и остальные – только чтение).
  - 4) `-rw-rw-r...` `feathers`: команды «`chmod 664 feathers`» (это файл, владелец и группа владельца имеют право на чтение и запись, остальные – только чтение). Командой «`ls-l`» проверяем правильность выполненных действий.

```

dskochina@dk6n52 ~ $ mkdir australia
dskochina@dk6n52 ~ $ mkdir play
dskochina@dk6n52 ~ $ touch my_os
dskochina@dk6n52 ~ $ touch feathers
dskochina@dk6n52 ~ $ chmod 744 australia
dskochina@dk6n52 ~ $ chmod 711 play
dskochina@dk6n52 ~ $ chmod 544 my_os
dskochina@dk6n52 ~ $ chmod 664 feathers
dskochina@dk6n52 ~ $ ls -l

```

Рис. 4.6: Создание файлов

```

итого 39
drwxr--r--  2 dskochina studsci 2048 мар  6 19:05  australia
drwxr-xr-x  2 dskochina studsci 2048 фев 22 10:22  bin
drwxr-xr-x 12 dskochina studsci 2048 фев 22 11:07  blog
-rw-rw-r--  1 dskochina studsci   0 мар  6 19:06  feathers
drwxr-xr-x  3 dskochina studsci 2048 окт 26 17:53  GNUstep
-rw-r--r--  1 dskochina studsci   0 мар  6 17:26  may
drwx--x--x  2 dskochina studsci 2048 мар  6 17:19  monthly
-r-xr--r--  1 dskochina studsci   0 мар  6 19:06  my_os
drwx--x--x  2 dskochina studsci 2048 мар  6 19:05  play
drwxr-xr-x  3 dskochina root    2048 сен  1 2022  public
lrwxr-xr-x  1 dskochina root    18 мар  2 22:26  public_html -> public/public_html
drwxr-xr-x  3 dskochina studsci 2048 мар  6 17:22  reports
drwxr-xr-x  4 dskochina studsci 2048 мар  6 19:04  ski.places
drwxr-xr-x  2 dskochina studsci 2048 фев 22 14:24  tmp
drwxr-xr-x  5 dskochina studsci 2048 фев 22 15:01  work
drwxr-xr-x  2 dskochina studsci 2048 сен  8 11:07  Видео
drwxr-xr-x  2 dskochina studsci 2048 сен  8 11:07  Документы
drwxr-xr-x  2 dskochina studsci 2048 мар  2 15:13  Загрузки
drwxr-xr-x  3 dskochina studsci 2048 сен 15 11:38  Изображения
drwxr-xr-x  2 dskochina studsci 2048 сен  8 11:07  Музыка
drwxr-xr-x  2 dskochina studsci 2048 сен  8 11:07  Общедоступные
drwxr-xr-x  2 dskochina studsci 2048 дек 22 12:03  'Рабочий стол'
drwxr-xr-x  2 dskochina studsci 2048 сен  8 11:07  Шаблоны
dskochina@dk6n52 ~ $ █

```

Рис. 4.7: Определение опции chmod

6. Выполнила следующие действия. (рис. [4.8], [4.9], [4.10]) Просмотрела содержимое файла /etc/passwd (команда «cat/etc/passwd»). Скопировала файл ~/feathers в файл ~/file.old (используем команду «cp feathers file.old»). Переместила файл ~/file.old в каталог ~/play (с помощью команды «mv file.ordplay»). Скопировала каталог ~/play в каталог ~/fun (команда «cp - r play fun»). Переместила каталог ~/fun в каталог ~/play (используем

команда «mv fun play») и назвала его games (используем команда «mv play/funplay/games»). Лишила владельца файла ~/feathers права на чтение (команда «chmod u-r feathers»). Если мы попытаемся просмотреть файл ~/feathers командой cat, то получим отказ в доступе, т.к. в предыдущем пункте лишили владельца права на чтение данного файла. Если мы попытаемся скопировать файл ~/feathers, например, в каталог monthly, то получим отказ в доступе, по причине, описанной в предыдущем пункте. Дала владельцу файла ~/feathers право на чтение (команда «chmod u+r feathers»). Лишила владельца каталога ~/play права на выполнение (с помощью команды «chmod u-x play»). Перешла в каталог ~/play (команда «cd play»). Получила отказ в доступе, т.к. в предыдущем пункте лишила владельца права на выполнение данного каталога. Дала владельцу каталога ~/play право на выполнение (команда «chmod u+ xplay»).

```
dskochina@dk6n52 ~ $ 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
```

Рис. 4.8: Копирование файлов

```

dskochina@dk6n52 ~ $ cp feathers file.old
dskochina@dk6n52 ~ $ mv file.old play
dskochina@dk6n52 ~ $ cp -r play fun
dskochina@dk6n52 ~ $ mv fun play
dskochina@dk6n52 ~ $ mv play/games
dskochina@dk6n52 ~ $ mv fun play
mv: не удалось выполнить stat для 'fun': Нет такого файла или каталога
dskochina@dk6n52 ~ $ ls
australia  blog      games     may       my_os     public    reports   tmp        Видео   Загрузки   Музыка   'Рабочий стол'
bin        feathers  GNUstep  monthly   play      public_html  ski.places  work      Документы  Изображения  Общедоступные  Шаблоны
dskochina@dk6n52 ~ $ ls play
file.old
dskochina@dk6n52 ~ $ ls fun
ls: невозможно получить доступ к 'fun': Нет такого файла или каталога
dskochina@dk6n52 ~ $ grep fun
^?
[1]+  Остановлен      grep --colour=auto fun
dskochina@dk6n52 ~ $ cp -r play fun
dskochina@dk6n52 ~ $ ls
australia  blog      fun       GNUstep  monthly   play      public_html  ski.places  work      Документы  Изображения  Общедоступные  Шаблоны
bin        feathers  games     may       my_os     public    reports   tmp        Видео   Загрузки   Музыка   'Рабочий стол'
dskochina@dk6n52 ~ $ mv play/fun play/games
dskochina@dk6n52 ~ $ chmod u-r feathers
dskochina@dk6n52 ~ $ cat feathers
cat: feathers: Отказано в доступе
dskochina@dk6n52 ~ $ cp feathers monthly
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
dskochina@dk6n52 ~ $ chmod u-r feather[.]

```

Рис. 4.9: Создание каталогов

```

dskochina@dk6n52 ~ $ cp feathers monthly
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
dskochina@dk6n52 ~ $ chmod u+r feathers
dskochina@dk6n52 ~ $ chmod u-r play
dskochina@dk6n52 ~ $ cd play
dskochina@dk6n52 ~/play $ chmod u-x play
chmod: невозможно получить доступ к 'play': Нет такого файла или каталога
dskochina@dk6n52 ~/play $ cd ~
dskochina@dk6n52 ~ $ chmod u-x play
dskochina@dk6n52 ~ $ cd play
dskochina@dk6n52 ~/play $ cd ~
dskochina@dk6n52 ~ $ chmod u+x play
dskochina@dk6n52 ~ $ 

```

Рис. 4.10: Изменение прав владельца

7. Используя команды «man mount», «man fsck», «man mkfs», «man kill», получила информацию о соответствующих командах. (рис. [4.11], [4.12], [4.13], [4.14], [4.15])

1) Команда mount: предназначена для монтирования файловой системы. Все файлы, доступные в Unix системах, составляют иерархическую файловую структуру, которая имеет ветки (каталоги) и листья (файлы в каталогах). Корень этого дерева обозначается как слеш. Физически файлы могут располагаться на различных устройствах. Команда mount служит для подключения файловых систем разных устройств к этому большому дереву. Наиболее часто встречающаяся форма команды mount выглядит следующим образом:

«mount -t vfstype device dir». Такая команда предлагает ядру смонтировать (подключить) файловую систему указанного типа vfstype, расположенную на устройстве device, к заданному каталогу dir, который часто называют точкой монтирования.

- 2) Команда fsck: это утилита командной строки, которая позволяет выполнять проверки согласованности и интерактивное исправление в одной или нескольких файловых системах Linux. Он использует программы, специфичные для типа файловой системы, которую он проверяет. У команды fsck следующий синтаксис: fsck [параметр] –[параметры ФС] [ . . .]. Например, если нужно восстановить («починить») файловую систему на некотором устройстве /dev/sdb2, следует воспользоваться командой: «sudo fsck -y /dev/sdb2». Опция -y необходима, т.к. при её отсутствии придётся слишком часто давать подтверждение.
- 3) Команда mkfs: создаёт новую файловую систему Linux. Имеет следующий синтаксис: mkfs[-V] [-tfstype] [fs-options] filesys [blocks] mkfs используется для создания файловой системы Linux на некотором устройстве, обычно в разделе жёсткого диска. В качестве аргумента file sys для файловой системы может выступать или название устройства (например, /dev/hda1, /dev/sdb2) или точка монтирования (например, /, /usr, /home). Аргументом blocks указывается количество блоков, которые выделяются для использования этой файловой системой. По окончании работы mkfs возвращает 0 - в случае успеха, а 1 при неудачной операции. Например, команда «mkfs -t ext2 /dev/hdb1» создаёт файловую систему типа ext 2 в разделе /dev/hdb1 (второй жёсткий диск).
- 4) Команда kill: посылает сигнал процессу или выводит список допустимых сигналов. Имеет следующий синтаксис: kill [опции] PID, где PID – это PID (числовой идентификатор) процесса или несколько PID процессов, если требуется послать сигнал сразу нескольким процессам. Например, команда

«kill -KILL 3121» посылает сигнал KILL процессу с PID 3121, чтобы принудительно завершить процесс.

```
dskochina@dk6n52 ~ $ man mount
[2]+  Остановлен      man mount
dskochina@dk6n52 ~ $ man fsck

[3]+  Остановлен      man fsck
dskochina@dk6n52 ~ $ man mkfs

[4]+  Остановлен      man mkfs
dskochina@dk6n52 ~ $ man kill

[5]+  Остановлен      man kill
dskochina@dk6n52 ~ $
```

Рис. 4.11: Ввод команд в терминале

```
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
```

Рис. 4.12: man mount



```

FCK(8)

NAME
    fsck - check and repair a Linux filesystem

SYNOPSIS
    fsck [-lsAVRTMNP] [-r [fd]] [-C [fd]] [-t fstype] [filesystem...]

DESCRIPTION
    fsck is used to check and optionally repair one or more Linux files
    an filesystem label or UUID specifier (e.g., UUID=8868abf6-88c5-4a8
    physical disk drives in parallel to reduce the total amount of time

```

Рис. 4.13: man fsck

```

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 m

    mkfs is used to build a Linux filesystem on a device, usually a har
    file that shall contain the filesystem. The size argument is the nu

```

Рис. 4.14: man mkfs

```

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 availa
    specified in three ways: -9, -SIGKILL or -KILL. Negative PID value
    cial; it indicates all processes except the kill process itself and

```

Рис. 4.15: man kill

## Ответы на контрольные вопросы

1. Чтобы узнать, какие файловые системы существуют на жёстком диске моего компьютера, использую команду «df-Th». Из рисунка видно, что на моем компьютере есть следующие файловые системы: dev tmpfs,tmpfs,ext4,iso9660. dev tmpfs позволяет ядру создать экземпляр tmpfs с именем devtmpfs при инициализации ядра, прежде чем регистрируется какое-либо устройство с драйверами. Каждое устройство с майором / минором будет предоставлять узел устройства в devtmpfs.devtmpfs монтируется на /dev и содержит специальные файлы устройств для всех устройств.tmpfs–временное файловое хранилище во многих Unix-подобных ОС. Предназначена для монтирования файловой системы, но размещается в ОЗУ вместо ПЗУ. Подобная конструкция является RAM диском. Данная файловая система также предназначена для быстрого и ненадёжного хранения временных данных. Хорошо подходит для /tmp и массовой сборки пакетов/образов. Предполагает наличие достаточного объёма виртуальной памяти. Файловая система tmpfs предназначена для того, чтобы использовать часть физической памяти сервера как обычный дисковый раздел, в котором можно сохранять данные (чтение и запись). Поскольку данные размещены в памяти, то чтение или запись происходят во много раз быстрее, чем с обычного HDD диска.ext4– имеет обратную совместимость с предыдущими версиями ФС. Эта версия была выпущена в 2008 году. Является первой ФС из «семейства»Ext, использующая механизм «extentfile system», который позволяет добиться меньшей фрагментации файлов и увеличить общую производительность файловой системы. Кроме того, вExt4реализован механизм отложенной записи (delayed allocation –delalloc), который так же уменьшает фрагментацию диска и снижает нагрузку на CPU. С другой стороны, хотя механизм отложенной записи и используется во многих ФС, но в силу сложности своей реализации он повышает вероятность утери данных. Характеристики: максимальный размер файла: 16 TB; максимальный размер раздела: 16TB; максимальный

размер имени файла: 255 символов. Рекомендации по использованию: наилучший выбор для SSD; наилучшая производительность по сравнению с предыдущими Ext-системами; она так же отлично подходит в качестве файловой системы для серверов баз данных, хотя сама система и моложе Ext3. ISO 9660 – стандарт, выпущенный Международной организацией по стандартизации, описывающий файловую систему для дисков CD-ROM. Также известен как CDFS (Compact Disc File System). Целью стандарта является обеспечить совместимость носителей под разными операционными системами, такими, как Unix, Mac OS, Windows.

2. Файловая система Linux/UNIX физически представляет собой пространство раздела диска, разбитое на блоки фиксированного размера, кратные размеру сектора – 1024, 2048, 4096 или 8120 байт. Размер блока указывается при создании файловой системы. В файловой структуре Linux имеется один корневой раздел `/` (он же `root`, корень). Все разделы жесткого диска (если их несколько) представляют собой структуру подкаталогов, “примонтированных” к определенным каталогам.

`/` – корень. Это главный каталог в системе Linux. По сути, это и есть файловая система Linux. Адреса всех файлов начинаются с корня, а дополнительные разделы, флешки или оптические диски подключаются в папки корневого каталога. Только пользователь `root` имеет право читать и изменять файлы в этом каталоге. `/BIN` – бинарные файлы пользователя. Этот каталог содержит исполняемые файлы. Здесь расположены программы, которые можно использовать в однопользовательском режиме или режиме восстановления. `/SBIN` – системные исполняемые файлы. Так же как и `/bin`, содержит двоичные исполняемые файлы, которые доступны на ранних этапах загрузки, когда не примонтирован каталог `/usr`. Но здесь находятся программы, которые можно выполнять только с правами суперпользователя. `/ETC` – конфигурационные файлы. В этой папке содержатся конфигурационные файлы всех программ, установленных в

системе. Кроме конфигурационных файлов, в системе инициализации Init Scripts, здесь находятся скрипты запуска и завершения системных демонов, монтирования файловых систем и автозагрузки программ. “/DEV” – файлы устройств. В Linux все, в том числе внешние устройства являются файлами. Таким образом, все подключенные флешки, клавиатуры, микрофоны, камеры – это просто файлы в каталоге /dev/. Выполняется сканирование всех подключенных устройств и создание для них специальных файлов. “/PROC” – информация о процессах. По сути, это псевдофайловая система, содержащая подробную информацию о каждом процессе, его Pid, имя исполняемого файла, параметры запуска, доступ к оперативной памяти и так далее. Также здесь можно найти информацию об использовании системных ресурсов. “/VAR” – переменные файлы. Название каталога “/var” говорит само за себя, он должен содержать файлы, которые часто изменяются. Размер этих файлов постоянно увеличивается. Здесь содержатся файлы системных журналов, различные кешы, базы данных и так далее. “/TMP” – временные файлы. В этом каталоге содержатся временные файлы, созданные системой, любыми программами или пользователями. Все пользователи имеют право записи в эту директорию. “/USR” – программы пользователя. Это самый большой каталог с большим количеством функций. Здесь находятся исполняемые файлы, исходники программ, различные ресурсы приложений, картинки, музыку и документацию. “/HOME” – домашняя папка. В этой папке хранятся домашние каталоги всех пользователей. В них они могут хранить свои личные файлы, настройки программ и т.д. “/BOOT” – файлы загрузчика. Содержит все файлы, связанные с загрузчиком системы. Это ядро vmlinuz, образ initrd, а также файлы загрузчика, находящиеся в каталоге /boot/grub. “/LIB” – системные библиотеки. Содержит файлы системных библиотек, которые используются исполняемыми файлами в каталогах /bin и /sbin. “/OPT” – дополнительные программы. В эту папку устанавливаются проприетарные программы, игры или драйвера. Это программы созданные

в виде отдельных исполняемых файлов самими производителями. “/MNT” – монтирование. В этот каталог системные администраторы могут монтировать внешние или дополнительные файловые системы. “/MEDIA” – съемные носители. В этот каталог система монтирует все подключаемые внешние накопители – USB флешки, оптические диски и другие носители информации. “/SRV” – сервер. В этом каталоге содержатся файлы серверов и сервисов. “/RUN” – процессы. Каталог, содержащий PID файлы процессов, похожий на “/var/run”, но в отличие от него, он размещен в TMPFS, а поэтому после перезагрузки все файлы теряются.

3. Чтобы содержимое некоторой файловой системы было доступно операционной системе необходимо воспользоваться командой mount.
4. Целостность файловой системы может быть нарушена из-за перебоев в питании, неполадок в оборудовании или из-за некорректного/внезапного выключения компьютера. Чтобы устранить повреждения файловой системы необходимо использовать команду fsck.
5. Файловую систему можно создать, используя команду mkfs. Ее краткое описание дано в пункте 5) в ходе выполнения заданий лабораторной работы.
6. Для просмотра текстовых файлов существуют следующие команды:

cat. Задача команды cat очень проста – она читает данные из файла или стандартного ввода и выводит их на экран. Синтаксис утилиты: cat [опции] файл1 файл2 ... Основные опции: -b – нумеровать только непустые строки -E – показывать символ \$ в конце каждой строки -n – нумеровать все строки -s – удалять пустые повторяющиеся строки -T – отображать табуляции в виде ^I -h – отобразить справку -v – версия утилиты nl. Команда nl действует аналогично команде cat, но выводит еще номера строк в столбце слева. less. Существенно более развитая команда для пролистывания текста. При чтении данных со стандартного ввода она создает буфер, который позволяет

листать текст как вперед, так и назад, а также искать как по направлению к концу, так и по направлению к началу текста. Синтаксис аналогичный синтаксису команды `cat`. Некоторые опции: `-g` – при поиске подсвечивать только текущее найденное слово (по умолчанию подсвечиваются все вхождения) `-N` – показывать номера строк `head`. Команда `head` выводит начальные строки (по умолчанию – 10) из одного или нескольких документов. Также она может показывать данные, которые передает на вывод другая утилита. Синтаксис аналогичный синтаксису команды `cat`. Основные опции: `-c` (`-bytes`) – позволяет задавать количество текста не в строках, а в байтах `-n` (`-lines`) – показывает заданное количество строк вместо 10, которые выводятся по умолчанию `-q` (`-quiet`, `-silent`) – выводит только текст, не добавляя к нему название файла `-v` (`-verbose`) – перед текстом выводит название файла `-z` (`-zero-terminated`) – символы перехода на новую строку заменяет символами завершения строк `tail` Эта команда позволяет выводить заданное количество строк с конца файла, а также выводить новые строки в интерактивном режиме. Синтаксис аналогичный синтаксису команды `cat`. Основные опции: `-c` – выводить указанное количество байт с конца файла `-f` – обновлять информацию по мере появления новых строк в файле `-n` – выводить указанное количество строк из конца файла `-pid` – используется с опцией `-f`, позволяет завершить работу утилиты, когда завершится указанный процесс `-q` – не выводить имена файлов `-retry` – повторять попытки открыть файл, если он недоступен `-v` – выводить подробную информацию о файле.

7. Утилита `cp` позволяет полностью копировать файлы и директории. Синтаксис: `cp [опции] файл-источник файл-приемник` После выполнения команды файл-источник будет полностью перенесен в файл-приемник. Если в конце указан слэш, файл будет записан в заданную директорию с оригинальным именем. Основные опции: `-attributes-only` – не копировать содержимое файла, а только флаги доступа и владельца `-f`, `-force` – перезаписывать существующие файлы `-i`, `-interactive` – спрашивать, нужно ли перезаписывать

существующие файлы-L –копировать не символические ссылки, а то, на что они указывают -n –не перезаписывать существующие файлы-P –не следовать символическим ссылкам-r –копировать папку Linux рекурсивно-s –не выполнять копирование файлов в Linux, а создавать символические ссылки-u –скопировать файл, только если он был изменён-x –не выходить за пределы этой файловой системы-p –сохранять владельца, временные метки и флаги доступа при копировании-t –считать файл-приемник директорией и копировать файл-источник в эту директорию.

8. Команда mv используется для перемещения одного или нескольких файлов (или директорий) в другую директорию, а также для переименования файлов и директорий. Синтаксис: mv [-опции] старый\_файл новый\_файл Основные опции: -help –выводит на экран официальную документацию об утилите -version –отображает версию mv -b –создает копию файлов, которые были перемещены или перезаписаны -f –при активации не будет спрашивать разрешение у владельца файла, если речь идет о перемещении или переименовании файла -i –наоборот, будет спрашивать разрешение у владельца -n –отключает перезапись уже существующих объектов -strip-trailing-slashes –удаляет завершающий символ / у файла при его наличии -t [директория] –перемещает все файлы в указанную директорию -u –осуществляет перемещение только в том случае, если исходный файл новее объекта назначения -v –отображает сведения о каждом элементе во время обработки команды Команда rename также предназначена, чтобы переименовать файл. Синтаксис: rename [опции] старое\_имя новое\_имя файлы. Основные опции: -v –вывести список обработанных файлов -n –тестовый режим, на самом деле никакие действия выполнены не будут -f –принудительно перезаписывать существующие файлы.

9. Права доступа– совокупность правил, регламентирующих порядок и условия доступа субъекта к объектам информационной системы (информации,

её носителям, процессам и другим ресурсам) установленных правовыми документами или собственником, владельцем информации. Права доступа к файлу или каталогу можно изменить, воспользовавшись командой `chmod`. Сделать это может владелец файла (или каталога) или пользователь с правами администратора. Синтаксис команды: `chmod режим имя_файла-Режим` имеет следующие компоненты структуры и способ записи: `=` установить право-лишить права+ дать право чтение записей выполнения (user) владелец файла (group) группа, к которой принадлежит владелец файла (others) все остальные.



## 5 Выводы

В ходе выполнения данной лабораторной работы я ознакомилась с файловой системой Linux, её структурой, именами и содержанием каталогов. А также 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.