

# **Отчет по лабораторной №7**

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

Камалиева Лия Дамировна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Теоретическое введение</b>	<b>6</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
3.1	Контрольные вопросы . . . . .	16
<b>4</b>	<b>Выводы</b>	<b>21</b>
	<b>Список литературы</b>	<b>22</b>

## Список иллюстраций

3.1	рис.1.1	. . . . .	7
3.2	рис.1.2	. . . . .	8
3.3	рис.1.3	. . . . .	8
3.4	рис.1.4	. . . . .	8
3.5	рис.1.5	. . . . .	8
3.6	рис.1.6	. . . . .	8
3.7	рис.1.8	. . . . .	9
3.8	рис.1.8	. . . . .	9
3.9	рис.1.9	. . . . .	10
3.10	рис.1.9	. . . . .	11
3.11	рис.1.9	. . . . .	11
3.12	рис.1.9	. . . . .	11
3.13	рис.1.9	. . . . .	12
3.14	рис.1.13	. . . . .	12
3.15	рис.1.14	. . . . .	12
3.16	рис.1.15	. . . . .	13
3.17	рис.1.16	. . . . .	14
3.18	рис.1.17	. . . . .	15
3.19	рис.1.18	. . . . .	16

## **Список таблиц**

# 1 Цель работы

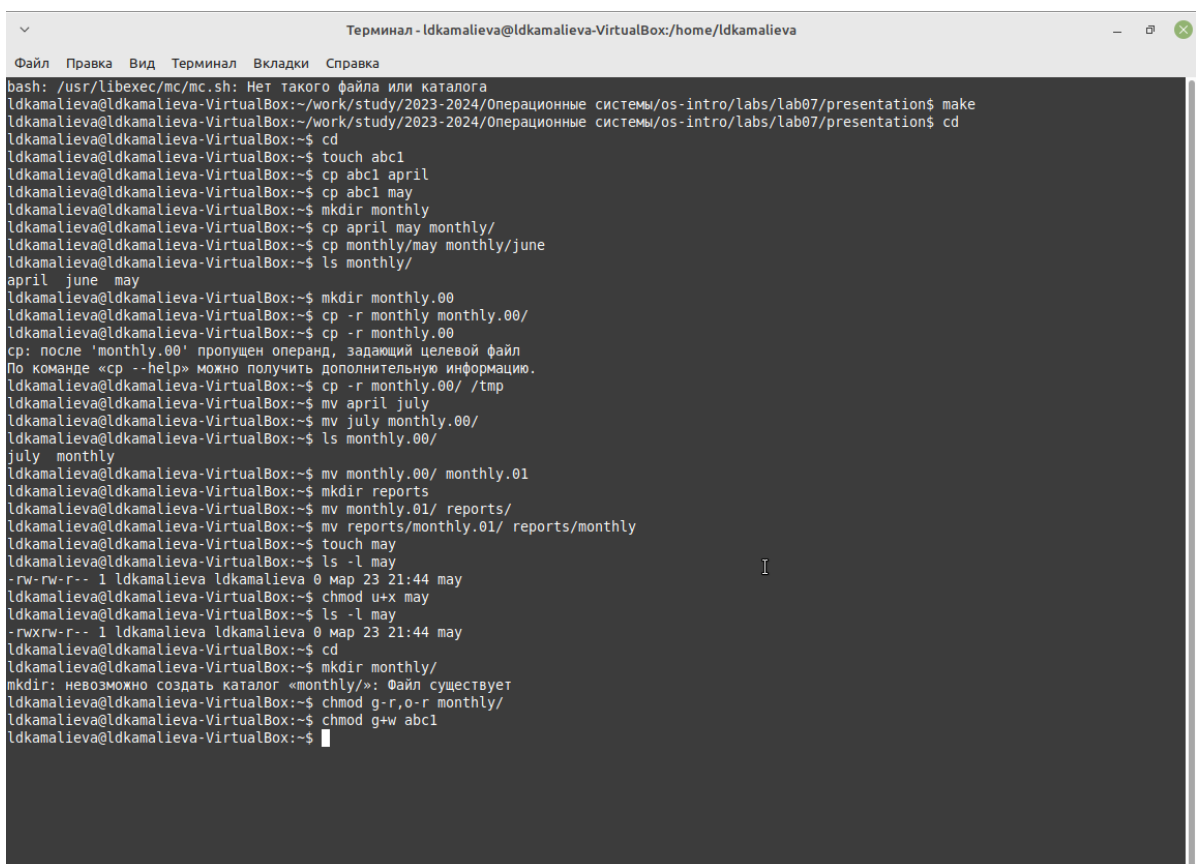
Ознакомление с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобретение практических навыков по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.

## 2 Теоретическое введение

Права доступа Каждый файл или каталог имеет права доступа (табл. 5.1). В сведениях о файле или каталоге указываются: – тип файла (символ (-) обозначает файл, а символ (d) — каталог); – права для владельца файла (r — разрешено чтение, w — разрешена запись, x — разрешено выполнение, - — право доступа отсутствует); – права для членов группы (r — разрешено чтение, w — разрешена запись, x — разрешено выполнение, - — право доступа отсутствует); – права для всех остальных (r — разрешено чтение, w — разрешена запись, x — разрешено выполнение, - — право доступа отсутствует)

## 3 Выполнение лабораторной работы

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



```
bash: /usr/libexec/с/с.sh: Нет такого файла или каталога
ldkamaliev@ldkamaliev-VirtualBox:~/work/study/2023-2024/Операционные системы/os-intro/labs/lab07/presentation$ make
ldkamaliev@ldkamaliev-VirtualBox:~/work/study/2023-2024/Операционные системы/os-intro/labs/lab07/presentation$ cd
ldkamaliev@ldkamaliev-VirtualBox:~$ cd
ldkamaliev@ldkamaliev-VirtualBox:~$ touch abc1
ldkamaliev@ldkamaliev-VirtualBox:~$ cp abc1 april
ldkamaliev@ldkamaliev-VirtualBox:~$ cp abc1 may
ldkamaliev@ldkamaliev-VirtualBox:~$ mkdir monthly
ldkamaliev@ldkamaliev-VirtualBox:~$ cp april may monthly/
ldkamaliev@ldkamaliev-VirtualBox:~$ cp monthly/may monthly/june
ldkamaliev@ldkamaliev-VirtualBox:~$ ls monthly/
april  june  may
ldkamaliev@ldkamaliev-VirtualBox:~$ mkdir monthly.00
ldkamaliev@ldkamaliev-VirtualBox:~$ cp -r monthly monthly.00/
ldkamaliev@ldkamaliev-VirtualBox:~$ cp -r monthly.00
cp: после 'monthly.00' пропущен операнд, задающий целевой файл
По команде «cp --help» можно получить дополнительную информацию.
ldkamaliev@ldkamaliev-VirtualBox:~$ cp -r monthly.00/ /tmp
ldkamaliev@ldkamaliev-VirtualBox:~$ mv april july
ldkamaliev@ldkamaliev-VirtualBox:~$ mv july monthly.00/
ldkamaliev@ldkamaliev-VirtualBox:~$ ls monthly.00/
july  monthly
ldkamaliev@ldkamaliev-VirtualBox:~$ mv monthly.00/ monthly.01
ldkamaliev@ldkamaliev-VirtualBox:~$ mkdir reports
ldkamaliev@ldkamaliev-VirtualBox:~$ mv monthly.01/ reports/
ldkamaliev@ldkamaliev-VirtualBox:~$ mv reports/monthly.01/ reports/monthly
ldkamaliev@ldkamaliev-VirtualBox:~$ touch may
ldkamaliev@ldkamaliev-VirtualBox:~$ ls -l may
-rw-rw-r-- 1 ldkamaliev ldkamaliev 0 map 23 21:44 may
ldkamaliev@ldkamaliev-VirtualBox:~$ chmod u+x may
ldkamaliev@ldkamaliev-VirtualBox:~$ ls -l may
-rwxrwx-r-- 1 ldkamaliev ldkamaliev 0 map 23 21:44 may
ldkamaliev@ldkamaliev-VirtualBox:~$ cd
ldkamaliev@ldkamaliev-VirtualBox:~$ mkdir monthly/
mkdir: невозможно создать каталог «monthly/»: Файл существует
ldkamaliev@ldkamaliev-VirtualBox:~$ chmod g-r,o-r monthly/
ldkamaliev@ldkamaliev-VirtualBox:~$ chmod g+w abc1
ldkamaliev@ldkamaliev-VirtualBox:~$
```

Рис. 3.1: рис.1.1

Шаг 2. В домашнем каталоге создайте директорию ~/ski.plases.

```
ldkamalieva@ldkamalieva-VirtualBox:~$ mkdir ~/ski.plases
ldkamalieva@ldkamalieva-VirtualBox:~$
```

Рис. 3.2: рис.1.2

Шаг 3. Переместите файл equipment в каталог ~/ski.plases

```
ldkamalieva@ldkamalieva-VirtualBox:~$ mv equipment ~/ski.plases
ldkamalieva@ldkamalieva-VirtualBox:~$
```

Рис. 3.3: рис.1.3

Шаг 4. Переименуйте файл ~/ski.plases/equipment в ~/ski.plases/equiplist

```
ldkamalieva@ldkamalieva-VirtualBox:~$ mv ~/ski.plases/equipment ~/ski.plases/equiplist
ldkamalieva@ldkamalieva-VirtualBox:~$
```

Рис. 3.4: рис.1.4

Шаг 5. Создайте в домашнем каталоге файл abc1 и скопируйте его в каталог ~/ski.plases, назовите его equiplist2.

```
ldkamalieva@ldkamalieva-VirtualBox:~$ mv equipment ~/ski.plases
ldkamalieva@ldkamalieva-VirtualBox:~$ mv ~/ski.plases/equipment ~/ski.plases/equiplist
```

Рис. 3.5: рис.1.5

Шаг 6. создайте каталог с именем equipment в каталоге ~/ski.plases.

```
ldkamalieva@ldkamalieva-VirtualBox:~$ mv ~/ski.plases/equipment ~/ski.plases/equiplist
ldkamalieva@ldkamalieva-VirtualBox:~$ touch abc1 cp abc1 ~/ski.plases/equiplist2
```

Рис. 3.6: рис.1.6

Шаг 7. Переместите файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment.



```
ldkamalieva@ldkamalieva-VirtualBox:~$ touch ~/abc1
ldkamalieva@ldkamalieva-VirtualBox:~$ cp ~/abc1 ~/ski.plases/equiplist2
ldkamalieva@ldkamalieva-VirtualBox:~$
```

Рис. 3.7: рис.1.8

Шаг 8.Создайте и переместите каталог ~/newdir в каталог ~/ski.plases и назовите его plans

```
ldkamalieva@ldkamalieva-VirtualBox:~$ mkdir ~/newdir
ldkamalieva@ldkamalieva-VirtualBox:~$ mv ~/newdir ~/ski.plases/plans
ldkamalieva@ldkamalieva-VirtualBox:~$
```

Рис. 3.8: рис.1.8

Шаг 9. я определила опции команды chmod, необходимые для того, чтобы присвоить перечис- ленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет

```
Терминал - ldkamalieva@ldkamalieva-VirtualBox:/home/ldkamalieva
Файл Правка Вид Терминал Вкладки Справка
ldkamalieva@ldkamalieva-VirtualBox:~$ mkdir australia play
ldkamalieva@ldkamalieva-VirtualBox:~$ touch my_os feathers
ldkamalieva@ldkamalieva-VirtualBox:~$ chmod 744 australia/
ldkamalieva@ldkamalieva-VirtualBox:~$ chmod 711 play/
ldkamalieva@ldkamalieva-VirtualBox:~$ chmod 544 my_os
ldkamalieva@ldkamalieva-VirtualBox:~$ chmod 664 feathers

Команда «chmod» не найдена. Возможно, вы имели в виду:

  command 'chmod' from deb coreutils (8.30-3ubuntu2)

Try: sudo apt install <deb name>

ldkamalieva@ldkamalieva-VirtualBox:~$ chmod 664 feathers
ldkamalieva@ldkamalieva-VirtualBox:~$ ls -l
итого 43120
-rw-rw-r-- 1 ldkamalieva ldkamalieva      0 мар 23 22:01 abc1
drwxr-xr-x 2 ldkamalieva ldkamalieva    4096 мар 23 22:11 australia
drwxr-xr-x 2 ldkamalieva ldkamalieva    4096 мар 16 13:51 bin
drwxr-xr-x 3 ldkamalieva ldkamalieva    4096 мар 12 20:27 blog
-rw-rw-r-- 1 ldkamalieva ldkamalieva      9 фев 23 20:54 COURSE
-rw-rw-r-- 1 ldkamalieva ldkamalieva      0 мар 23 21:56 cp
drwxr-xr-x 2 ldkamalieva ldkamalieva    4096 мар 21 13:38 Desktop
drwxr-xr-x 5 ldkamalieva ldkamalieva    4096 мар 16 13:55 dotfiles
-rw-rw-r-- 1 ldkamalieva ldkamalieva      0 мар 23 22:11 feathers
drwxr-xr-x 3 ldkamalieva ldkamalieva    4096 мар 7 14:18 git-extended
drwxr-xr-x 3 ldkamalieva ldkamalieva    4096 мар 12 20:58 go
-rw-rw-r-- 1 ldkamalieva ldkamalieva 22010269 мар 1 18:28 hugo_extended_0.123.6_Linux-64bit.tar.gz
drwxr-xr-x 2 ldkamalieva ldkamalieva    4096 мар 14 14:28 hugo_extended_0.123.8_Linux-64bit
-rw-rw-r-- 1 ldkamalieva ldkamalieva 22010952 мар 14 14:26 hugo_extended_0.123.8_Linux-64bit.tar.gz
-rw-rw-r-- 1 ldkamalieva ldkamalieva    18657 мар 16 13:59 LICENSE
-rwxr-xr-x 1 ldkamalieva ldkamalieva      0 мар 23 21:44 may
drwx-wx--x 2 ldkamalieva ldkamalieva    4096 мар 23 21:41 monthly
-r-xr--r-- 1 ldkamalieva ldkamalieva      0 мар 23 22:11 my_os
-rw-rw-r-- 1 ldkamalieva ldkamalieva     398 мар 6 19:47 package.json
drwx--x--x 2 ldkamalieva ldkamalieva    4096 мар 23 22:11 play
-rw-r--r-- 1 ldkamalieva ldkamalieva   12891 фев 28 21:28 README.md
drwxr-xr-x 3 ldkamalieva ldkamalieva    4096 мар 23 21:44 reports
drwxr-xr-x 4 ldkamalieva ldkamalieva    4096 мар 23 22:05 ski.places
drwxr-xr-x 6 ldkamalieva ldkamalieva    4096 мар 14 14:43 work
drwxr-xr-x 2 ldkamalieva ldkamalieva    4096 окт 23 19:21 Видео
drwxr-xr-x 2 ldkamalieva ldkamalieva    4096 окт 23 19:21 Документы
drwxr-xr-x 3 ldkamalieva ldkamalieva    4096 мар 16 00:46 Загрузки
drwxr-xr-x 2 ldkamalieva ldkamalieva    4096 окт 23 19:21 Изображения
drwxr-xr-x 2 ldkamalieva ldkamalieva    4096 окт 23 19:21 Музыка
```

Рис. 3.9: рис.1.9

Шаг 10. Скопируйте файл ~/feathers в файл ~/file.old

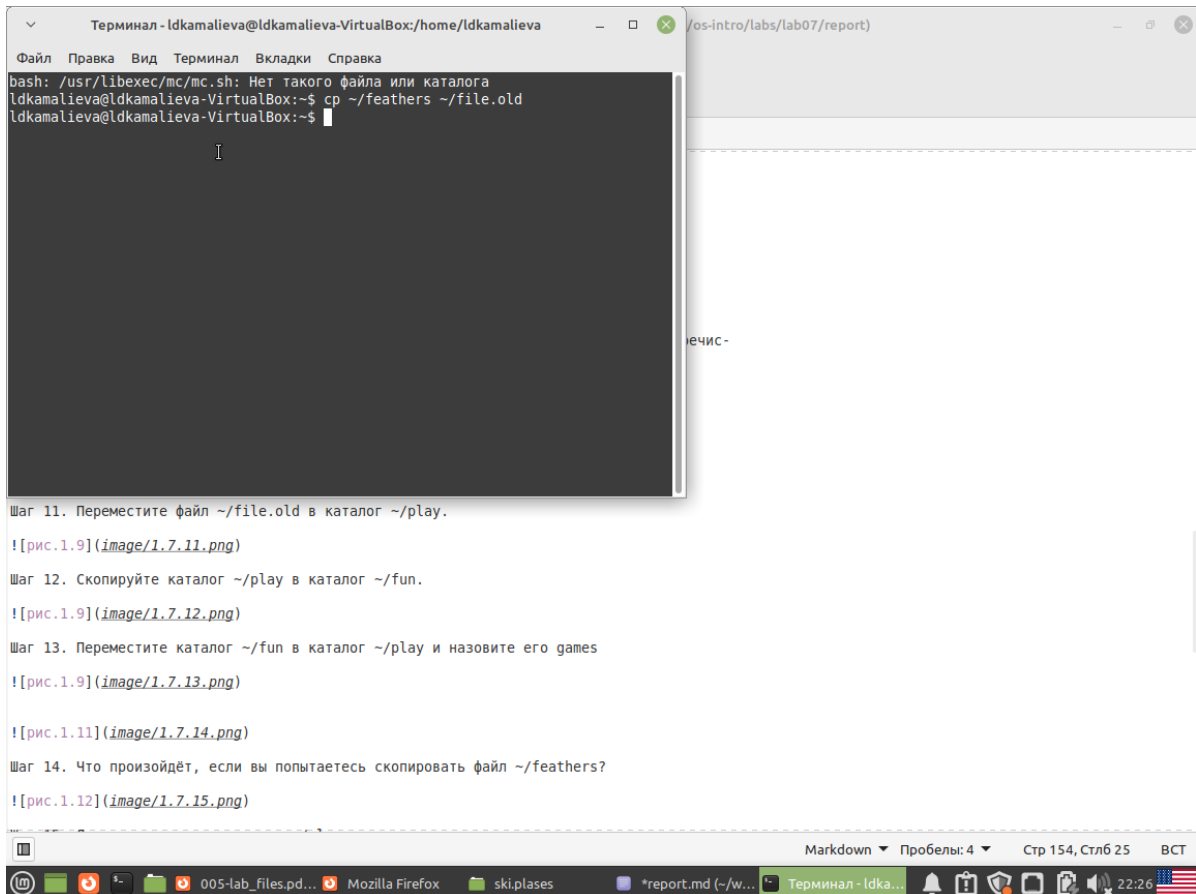


Рис. 3.10: рис.1.9

Шаг 11. Переместите файл ~/file.old в каталог ~/play.

```
ldkamalieva@ldkamalieva-VirtualBox:~$ mv ~/file.old ~/play
ldkamalieva@ldkamalieva-VirtualBox:~$
```

Рис. 3.11: рис.1.9

Шаг 12. Скопируйте каталог ~/play в каталог ~/fun.

```
ldkamalieva@ldkamalieva-VirtualBox:~$ cp -r ~/play ~/fun
ldkamalieva@ldkamalieva-VirtualBox:~$
```

Рис. 3.12: рис.1.9

Шаг 13. Переместите каталог ~/fun в каталог ~/play и назовите его games

```
ldkamalieva@ldkamalieva-VirtualBox:~$ mv ~/fun ~/play/games  
ldkamalieva@ldkamalieva-VirtualBox:~$
```

Рис. 3.13: рис.1.9

Шаг 15. Лишите владельца каталога ~/play права на выполнение.

```
ldkamalieva@ldkamalieva-VirtualBox:~$ mv ~/fun ~/play/games  
ldkamalieva@ldkamalieva-VirtualBox:~$
```

Рис. 3.14: рис.1.13

Шаг 16. Перейдите в каталог ~/play. Что произошло?

будет выведено сообщение об ошибке

Шаг 17. Дайте владельцу каталога ~/play право на выполнение.

```
ldkamalieva@ldkamalieva-VirtualBox:~$ chmod u+x ~/play  
ldkamalieva@ldkamalieva-VirtualBox:~$
```

Рис. 3.15: рис.1.14

Шаг 18. `man mount`. команда для подключения файловых систем к директории в иерархии файловой системы Linux. Пример: `mount /dev/sdb1 /mnt/usb` - подключение USB устройства к директории /mnt/usb

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

NAME
    mount - mount a filesystem

SYNOPSIS
    mount [-l|-h|-V]

    mount -a [-fFnrsvw] [-t fstype] [-O optlist]

    mount [-fnrsvw] [-o options] device|dir

    mount [-fnrsvw] [-t fstype] [-o options] device dir

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 sev-
    eral 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 another
    services.

    The standard form of the mount command is:

        mount -t type device dir

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

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

        mount /dir

    then mount looks for a mountpoint (and if not found then for a device)
    in the /etc/fstab file. It's possible to use the --target or --source
    options to avoid ambivalent interpretation of the given argument. For
    example:

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

Рис. 3.16: рис.1.15

Шаг 19. **man fsck**. команда для проверки и восстановления целостности фай-  
ловой системы. Пример: **fsck /dev/sda1** - проверка и восстановление файловой

системы на разделе /dev/sda1.

```
FCK(8)                                System Administration                                FCK(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. filesystems 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 code returned by fsck is the sum of the following conditions:

        0      No errors
        1      Filesystem errors corrected
        2      System should be rebooted
        4      Filesystem errors left uncorrected
        8      Operational error
        16     Usage or syntax error
        32     Checking canceled by user request
        128    Shared-library error

    The exit code returned when multiple filesystems are checked is the bit-wise OR of the exit codes for each filesystem that is checked.

    In actuality, fsck is simply a front-end for the various filesystem checkers (fsck.fstype) available under Linux. The filesystem-specific checker is searched for in the PATH environment variable. If the PATH is undefined then fallback to "/sbin".

    Please see the filesystem-specific checker manual pages for further details.

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

Рис. 3.17: рис.1.16

Шаг 19. mkfs - команда для создания новой файловой системы на указанном устройстве. Пример: mkfs.ext4 /dev/sdb1 - создание файловой системы ext4 на USB устройстве /dev/sdb1.

```
MKFS(8)                                System Administration                                MKFS(8)

NAME
    mkfs - build a Linux filesystem

SYNOPSIS
    mkfs [options] [-t type] [fs-options] device [size]

DESCRIPTION
    This mkfs frontend is deprecated in favour of filesystem specific
    mkfs.<type> utils.

    mkfs is used to build a Linux filesystem on a device, usually a hard
    disk partition. The device argument is either the device name (e.g.
    /dev/hda1, /dev/sdb2), or a regular file that shall contain the
    filesystem. The size argument is the number of blocks to be used for
    the filesystem.

    The exit code 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
    Manual page mkfs(8) line 1 (press h for help or a to quit)
```

Рис. 3.18: рис.1.17

Шаг 19. команда для завершения процессов в Linux. Пример: kill PID - отправка сигнала завершения процессу с указанным идентификатором PID.

```
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>

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

Рис. 3.19: рис.1.18

## 3.1 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 экзабайта. 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 восстановле-

ния данных. Файловая система объединяет в себе очень много новых интересных возможностей, таких как размещение на нескольких разделах, поддержка подто- мов, изменение размера не лету, создание мгновенных снимков, а также высокая производительность. Но многими пользователями файловая система Btrfs счита- ется 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 каж- дой 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.

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. `Мv` - 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.

## 4 Выводы

я научилась работать с файловой системой линукс

## **Список литературы**