

Лабораторная работа №7

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

Полякова Ю.А.

28 февраля 2007

Российский университет дружбы народов, Москва, Россия

Информация

- Полякова Юлия Александровна
- Студент
- Российский университет дружбы народов
- yulya.polyakova.07@mail.ru
- <https://github.com/JuliaMaffin123>



Вводная часть

- Важно иметь базовые знания работы с файлами и каталогами

- Файловая система Linux

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

- Файловая система Linux
- Команды для работы с файлами и каталогами
- Командная строка в терминале

Выполнение лабораторной работы

Копирование файлов и каталогов

Выполним примеры на копирование файлов и каталогов. 1) touch abc1; cp abc1 april; cp abc1 may - копирование файла в текущем каталоге. 2) mkdir monthly; cp april may monthly - копирование нескольких файлов в каталог. 3) cp monthly/may monthly/june; ls monthly - копирование файлов в произвольном каталоге. 4) mkdir monthly.00; cp -r monthly monthly.00 - копирование каталогов в текущем каталоге. 5) cp -r monthly.00 /tmp - копирование каталогов в произвольном каталоге.

```
yapolyakova1@yapolyakova1:~$ cd
yapolyakova1@yapolyakova1:~$ touch abc1
yapolyakova1@yapolyakova1:~$ cp abc1 april
yapolyakova1@yapolyakova1:~$ cp abc1 may
yapolyakova1@yapolyakova1:~$ ls
abc1  bin  Desktop  Downloads  JuliaWaffin123.github.io  LICENSE  Music  Pictures  Templates  work
april  Desktop  Downloads  JuliaWaffin123.github.io  may  newdir  Public  Videos
yapolyakova1@yapolyakova1:~$ mkdir monthly
yapolyakova1@yapolyakova1:~$ cp april may monthly
yapolyakova1@yapolyakova1:~$ ls monthly
april  may
yapolyakova1@yapolyakova1:~$ cp monthly/may monthly/june
yapolyakova1@yapolyakova1:~$ ls monthly
april  june  may
yapolyakova1@yapolyakova1:~$ mkdir monthly.00
yapolyakova1@yapolyakova1:~$ cp -r monthly monthly.00
yapolyakova1@yapolyakova1:~$ cp -r monthly.00 /tmp
yapolyakova1@yapolyakova1:~$
```

Рис. 1: Копирование файлов и каталогов

Перемещение и переименование файлов и каталогов

Выполним примеры на перемещение и переименование файлов и каталогов. 1) `mv april july` - переименование файлов в текущем каталоге. 2) `mv july monthly.00`; `ls monthly.00` - перемещение файлов в другой каталог. 3) `mv monthly.00 monthly.01` - переименование каталогов в текущем каталоге. 4) `mkdir reports`; `mv monthly.01 reports` - перемещение каталога в другой каталог. 5) `mv reports/monthly.01 reports/monthly` - переименование каталога, не являющегося текущим.

```
yapolyakova1@yapolyakova1:~$ cd
yapolyakova1@yapolyakova1:~$ mv april july
yapolyakova1@yapolyakova1:~$ ls
abc1 Desktop Downloads JuliaMeffin123.github.io LICENSE monthly Music Pictures Templates work
bin Documents git-extended july may monthly.00 newdir Public Videos
yapolyakova1@yapolyakova1:~$ mv july monthly.00
yapolyakova1@yapolyakova1:~$ ls monthly.00
july monthly
yapolyakova1@yapolyakova1:~$ mv monthly.00 monthly.01
yapolyakova1@yapolyakova1:~$ ls
abc1 Desktop Downloads JuliaMeffin123.github.io may monthly.01 newdir Public Videos
bin Documents git-extended LICENSE monthly Music Pictures Templates work
yapolyakova1@yapolyakova1:~$ mkdir reports
yapolyakova1@yapolyakova1:~$ mv monthly.01 reports
yapolyakova1@yapolyakova1:~$ ls reports
monthly.01
yapolyakova1@yapolyakova1:~$ mv reports/monthly.01 reports/monthly
yapolyakova1@yapolyakova1:~$ ls reports
monthly
yapolyakova1@yapolyakova1:~$
```

Рис. 2: Перемещение и переименование файлов и каталогов

Изменение прав доступа к файлам и каталогам 1-3

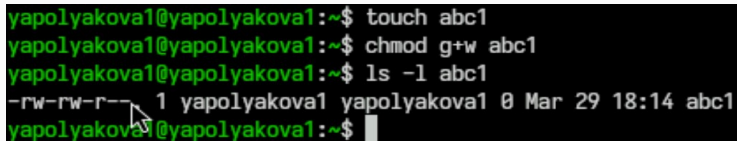
Выполним примеры изменения прав доступа к файлам и каталогам. 1) `touch may; ls -l may;`
`chmod u+x may; ls -l may` - создаем файл `~/may` с правом выполнения для владельца. 2)
`chmod u-x may; ls -l may` - лишаем владельца файла `~/may` права на выполнение. 3) `mkdir`
`monthly; chmod g-r,o-r monthly` - создаем каталог `monthly` с запретом на чтение для членов
группы и всех остальных пользователей.

```
yapolyakova@yapolyakova:~$ mkdir monthly
mkdir: cannot create directory 'monthly': File exists
yapolyakova@yapolyakova:~$ chmod g-r,o-r monthly
yapolyakova@yapolyakova:~$ ls -l monthly
total 0
-rw-r--r--. 1 yapolyakova1 yapolyakova1 0 Mar 29 17:57 april
-rw-r--r--. 1 yapolyakova1 yapolyakova1 0 Mar 29 17:58 june
-rw-r--r--. 1 yapolyakova1 yapolyakova1 0 Mar 29 17:57 may
yapolyakova@yapolyakova:~$ ls -l monthly/
total 0
-rw-r--r--. 1 yapolyakova1 yapolyakova1 0 Mar 29 17:57 april
-rw-r--r--. 1 yapolyakova1 yapolyakova1 0 Mar 29 17:58 june
-rw-r--r--. 1 yapolyakova1 yapolyakova1 0 Mar 29 17:57 may
yapolyakova@yapolyakova:~$ ls -l
total 28
-rw-r--r--. 1 yapolyakova1 yapolyakova1 0 Mar 29 17:56 abc1
drwxr-xr-x. 1 yapolyakova1 yapolyakova1 38 Mar 14 19:40 bin
drwxr-xr-x. 1 yapolyakova1 yapolyakova1 0 Mar 4 20:17 Desktop
drwxr-xr-x. 1 yapolyakova1 yapolyakova1 0 Mar 4 20:17 Document
drwxr-xr-x. 1 yapolyakova1 yapolyakova1 1512 Mar 22 19:42 Download
drwxr-xr-x. 1 yapolyakova1 yapolyakova1 74 Mar 7 22:21 git-exte
drwxr-xr-x. 1 yapolyakova1 yapolyakova1 388 Mar 15 19:54 JuliaMed
-rw-r--r--. 1 yapolyakova1 yapolyakova1 18657 Mar 14 19:44 LICENSE
-rw-r--r--. 1 yapolyakova1 yapolyakova1 0 Mar 29 18:08 may
drwxr-xr-x. 1 yapolyakova1 yapolyakova1 24 Mar 29 17:58 monthly
```

Рис. 3: Изменение прав доступа к файлам и каталогам 1-3

Изменение прав доступа к файлам и каталогам 4

Пример 4) `touch abc1`; `chmod g+w abc1` - создаем файл `~/abc1` с правом записи для членов группы.



```
yapolyakova1@yapolyakova1:~$ touch abc1
yapolyakova1@yapolyakova1:~$ chmod g+w abc1
yapolyakova1@yapolyakova1:~$ ls -l abc1
-rw-rw-r-- 1 yapolyakova1 yapolyakova1 0 Mar 29 18:14 abc1
yapolyakova1@yapolyakova1:~$
```

Рис. 4: Изменение прав доступа к файлам и каталогам 4

Задания 2.1. - 2.4.

Задание 2.1. Скопируйте файл `/usr/include/sys/io.h` в домашний каталог и назовите его `equipment`. Команда: `cp /usr/include/sys/io.h; mv io.h equipment`. Задание 2.2. В домашнем каталоге создайте директорию `~/ski.places`. Команда: `mkdir ~/ski.places` Задание 2.3. Переместите файл `equipment` в каталог `~/ski.places`. Команда: `mv equipment ~/ski.places` Задание 2.4. Переименуйте файл `~/ski.places/equipment` в `~/ski.places/equiplist`. Команда: `mv ~/ski.places/equipment ~/ski.places/equiplist`

```
yapolyakova1@yapolyakova1:~$ cp /usr/include/sys/io.h ~/
yapolyakova1@yapolyakova1:~$ mv io.h equipment
yapolyakova1@yapolyakova1:~$ ls
abc1 Desktop Downloads git-extended LICENSE monthly newdir Public Templates work
bin Documents equipment JuliaMaffin123.github.io may Music Pictures reports Videos
yapolyakova1@yapolyakova1:~$ mkdir ~/ski.places
yapolyakova1@yapolyakova1:~$ mv equipment ~/ski.places
yapolyakova1@yapolyakova1:~$ ls ski.places/
equipment
yapolyakova1@yapolyakova1:~$ mv ~/ski.places/equipment ~/ski.places/equiplist
yapolyakova1@yapolyakova1:~$ ls ski.places/
equiplist
yapolyakova1@yapolyakova1:~$
```

Рис. 5: Задания 2.1. - 2.4.

Задания 2.5. - 2.8.

Задание 2.5. Создайте в домашнем каталоге файл abc1 и скопируйте его в каталог ~/ski.plases, назовите его equiplist2. Команда: touch abc1; cp abc1 ~/ski.plases; mv ~/ski.plases/abc1 ~/ski.plases/equiplist2

Задание 2.6. Создайте каталог с именем equipment в каталоге ~/ski.plases. Команда: mkdir ~/ski.plases/equipment

Задание 2.7. Переместите файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment. Команда: mv ~/ski.plases/equiplist ~/ski.plases/equiplist2 ~/ski.plases/equipment/

Задание 2.8. Создайте и переместите каталог ~/newdir в каталог ~/ski.plases и назовите его plans. Команда: mkdir newdir; mv newdir ~/ski.plases/plans

```
equiplist@equiplist:~$ cp /usr/include/sys/in.h ~/
equiplist@equiplist:~$ mv /usr/include/sys/in.h ~/
equiplist@equiplist:~$ ls
abc1 backup bin Documents equipment /usr/include/sys/in.h .LINUX monthly newdir Public Templates work
equiplist@equiplist:~$ mkdir ~/ski.plases
equiplist@equiplist:~$ mv ~/ski.plases/abc1 ~/ski.plases/equiplist2
equiplist@equiplist:~$ mv ~/ski.plases/equiplist ~/ski.plases/equipment
equiplist@equiplist:~$ mv ~/ski.plases/equiplist2 ~/ski.plases/equipment
equiplist@equiplist:~$ mv ~/ski.plases/newdir ~/ski.plases/plans
```

Рис. 6: Задания 2.5. - 2.8.

Задания 3.1. - 3.4.

Определяем опции команды `chmod`, необходимые для того, чтобы присвоить перечисленным в условии файлам выделенные права доступа, считая, что в начале таких прав нет: 3.1. `g-x,o-x`; 3.2. `g-r,o-r`; 3.3. `544`; 3.4. `g+w`

```
yapolyakova1@yapolyakova1:~$ mkdir australia; ls -l australia
total 0
yapolyakova1@yapolyakova1:~$ ls -ld australia
drwxr-xr-x. 1 yapolyakova1 yapolyakova1 0 Mar 29 18:27 australia
yapolyakova1@yapolyakova1:~$ chmod g-x,o-x australia
yapolyakova1@yapolyakova1:~$ ls -ld australia
drwxr--r--. 1 yapolyakova1 yapolyakova1 0 Mar 29 18:27 australia
yapolyakova1@yapolyakova1:~$ mkdir play; ls -ld play
drwxr-xr-x. 1 yapolyakova1 yapolyakova1 0 Mar 29 18:31 play
yapolyakova1@yapolyakova1:~$ chmod g-r,o-r play
yapolyakova1@yapolyakova1:~$ ls -ld play
drwx--x--x. 1 yapolyakova1 yapolyakova1 0 Mar 29 18:31 play
yapolyakova1@yapolyakova1:~$ touch my_os; ls -l my_os
-rw-r--r--. 1 yapolyakova1 yapolyakova1 0 Mar 29 18:32 my_os
yapolyakova1@yapolyakova1:~$ chmod 544 my_os
yapolyakova1@yapolyakova1:~$ ls -l my_os
-r-xr--r--. 1 yapolyakova1 yapolyakova1 0 Mar 29 18:32 my_os
yapolyakova1@yapolyakova1:~$ touch feathers; ls -l feathers
-rw-r--r--. 1 yapolyakova1 yapolyakova1 0 Mar 29 18:34 feathers
yapolyakova1@yapolyakova1:~$ chmod g+w my_os; ls -l feathers
-rw-r--r--. 1 yapolyakova1 yapolyakova1 0 Mar 29 18:34 feathers
yapolyakova1@yapolyakova1:~$ chmod g+w feathers; ls -l feathers
-rw-rw-r--. 1 yapolyakova1 yapolyakova1 0 Mar 29 18:34 feathers
yapolyakova1@yapolyakova1:~$
```

Рис. 7: Задания 3.1. - 3.4.

Задания 4.1. - 4.5.

Задание 4.1. Просмотрите содержимое файла /etc/passwd. Команда: cat /etc/passwd

Задание 4.2. Скопируйте файл ~/feathers в файл ~/file.old. Команда: cp ~/feathers ~/file.old

Задание 4.3. Переместите файл ~/file.old в каталог ~/play. Команда: mv ~/file.old ~/play

Задание 4.4. Скопируйте каталог ~/play в каталог ~/fun. Команда: cp -r ~/play ~/fun/

Задание 4.5. Переместите каталог ~/fun в каталог ~/play и назовите его games. Команда: mv ~/fun ~/play/games

```
yapolyakova1@yapolyakova1:~$ cat /etc/passwd
cat: /etc/passwd: No such file or directory
yapolyakova1@yapolyakova1:~$ cp ~/feathers ~/file.old
yapolyakova1@yapolyakova1:~$ ls
abc1  bin  Documents  feathers  git-extended  LICENSE  monthly  my_os  play  reports  Templates  work
yapolyakova1@yapolyakova1:~$ mv ~/file.old ~/play
yapolyakova1@yapolyakova1:~$ ls play
file.old
yapolyakova1@yapolyakova1:~$ cp -r ~/play ~/fun/
yapolyakova1@yapolyakova1:~$ ls fun
file.old
yapolyakova1@yapolyakova1:~$ mv -r ~/fun ~/play/games
mv: invalid option -- 'r'
Try 'mv --help' for more information.
yapolyakova1@yapolyakova1:~$ mv ~/fun ~/play/games
yapolyakova1@yapolyakova1:~$ ls
abc1  bin  Documents  feathers  JuliaHaffin123.github.io  may  Music  Pictures  Public  ski_places  Videos
yapolyakova1@yapolyakova1:~$ ls play
file.old  games
yapolyakova1@yapolyakova1:~$
```

Рис. 8: Задания 4.1. - 4.5.

Задания 4.6. - 4.9.

Задание 4.6. Лишите владельца файла ~/feathers права на чтение. Команда: `chmod u-r ~/feathers`

Задание 4.7. Что произойдёт, если вы попытаетесь просмотреть файл ~/feathers командой `cat`? Ответ: доступ будет запрещен

Задание 4.8. Что произойдёт, если вы попытаетесь скопировать файл ~/feathers? Ответ: доступ будет запрещен

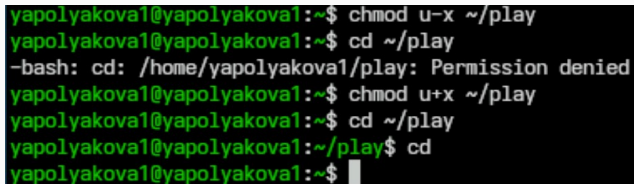
Задание 4.9. Дайте владельцу файла ~/feathers право на чтение. Команда: `chmod u+r ~/feathers`

```
yapolyakova1@yapolyakova1:~$ chmod u-r ~/feathers
yapolyakova1@yapolyakova1:~$ cat ~/feathers
cat: /home/yapolyakova1/feathers: Permission denied
yapolyakova1@yapolyakova1:~$ cp feathers feathers1
cp: cannot open 'feathers' for reading: Permission denied
yapolyakova1@yapolyakova1:~$ chmod u+r ~/feathers
yapolyakova1@yapolyakova1:~$ cat ~/feathers
yapolyakova1@yapolyakova1:~$
```

Рис. 9: Задания 4.6. - 4.9.

Задания 4.10. - 4.12.

Задание 4.10. Лишите владельца каталога ~/play права на выполнение. Команда: `chmod u-x ~/play`
Задание 4.11. Перейдите в каталог ~/play. Что произошло? Команда: `cd ~/play`. Доступ был запрещен.
Задание 4.12. Дайте владельцу каталога ~/play право на выполнение. Команда: `chmod u+x ~/play`



```
yapolyakova1@yapolyakova1:~$ chmod u-x ~/play
yapolyakova1@yapolyakova1:~$ cd ~/play
-bash: cd: /home/yapolyakova1/play: Permission denied
yapolyakova1@yapolyakova1:~$ chmod u+x ~/play
yapolyakova1@yapolyakova1:~$ cd ~/play
yapolyakova1@yapolyakova1:~/play$ cd
yapolyakova1@yapolyakova1:~$
```

Рис. 10: Задания 4.10. - 4.12.

Команда `man mount`, `mount` - команда для монтирования файловых систем. Когда файловая система монтируется, она становится доступной для чтения и записи в указанной точке монтирования. Пример: `sudo mount /dev/sda1 /mnt` - монтирует файловую систему на устройстве `/dev/sda1` в точку `/mnt`

```
NAME
    mount - mount a filesystem

SYNOPSIS
    mount [-h-V]

    mount [-l] [-t fstype]

    mount -s [-ffrswv] [-t fstype] [-O optlist]

    mount [-frswv] [-o options] device mountpoint

    mount [-frswv] [-t fstype] [-o options] device mountpoint

    mount --bind|--rbind|--move olddir newdir

    mount --make-shared|slave|private|unbindable|rsshared|rslave|rprivate|rbindable 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(s) command will detach it again. The filesystem is used to control how data is stored on the device or provided in a virtual way by network or other services.

    The standard form of the mount command is:

        mount -t type device dir

    This tells the kernel to attach the filesystem found on device (which is 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 section "Non-superuser mounts" below for more details. The previous contents (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.
```

Рис. 11: Документация mount

Команда `man fsck`, `fsck` проверяет целостность файловой системы и исправляет обнаруженные ошибки. Пример: `fsck -A` - проверка ошибок во всех разделах системы

```
fsck(8)                                System Administration                fsck(8)

NAME
    fsck - check and repair a Linux filesystem

SYNOPSIS
    fsck [-lAaRTnmp] [-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 a filesystem label or UUID specifier (e.g., UUID=083046f6-08c5-4a83-98d3-bfc2485777bd or
    LABEL=roots). 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.

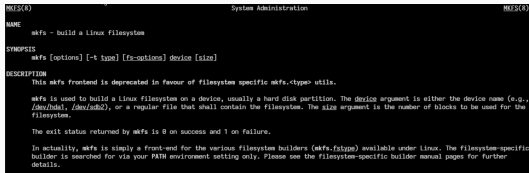
    If no filesystems are specified on the command line, and the -A option is not specified, fsck will default to checking filesystems in
    /etc/fstab serially. This is equivalent to the -As options.

    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
    4      Filesystem errors left uncorrected
    8      Operational error
    16
```

Рис. 12: Документация fsck

Команда `man mkfs`, `mkfs` создает файловую систему. Пример: `mkfs -t ext4 /dev/sda1` - создает файловую систему `ext4` в разделе `/dev/sda1`



```
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/sda2), 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.
```

Рис. 13: Документация `mkfs`

Команда `man kill`, `kill` - для завершения процессов вручную, посылает сигнал завершения выбранному процессу. Пример: `kill 279` - чтобы остановить процесс с идентификатором 279

```
KILL(1)                                user Commands                                KILL(1)

NAME
    kill - terminate a process

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

    kill -l {number} | -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 should be used in preference to the KILL signal (number 9), since a process may install a handler for the TERM signal in order to perform clean-up steps before terminating in an orderly fashion. If a process does not terminate after a TERM signal has been sent, then the KILL signal may be used; be aware that the latter signal cannot be caught, and so does not give the target process the opportunity to perform any clean-up before terminating.

    Most modern shells have a builtin kill command, with a usage rather similar to that of the command described here. The --all, --pid, and --queue options, and the possibility to specify processes by command name, are local extensions.

    If signal is 0, then no actual signal is sent, but error checking is still performed.
```

Рис. 14: Документация kill

Контрольные вопросы

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

Это btrfs и ext4. Файловая система BTRFS (B-Tree Filesystem) — это современная файловая система, разработанная для использования в операционных системах Linux. Она построена по принципу копирования при записи (Copy on Write): при чтении области данных используется общая копия, в случае изменения данных — создаётся новая копия. Файловая система Ext4 — это собственная файловая система Linux, созданная для преодоления проблем Ext3. Она использует базовые принципы Ext3, но имеет ряд нововведений, например, увеличена скорость работы, увеличен максимальный размер файла и добавлена поддержка накопителей объёмом до 1 Экзабайта.

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

Btrfs: на низком уровне располагаются блочные устройства, далее идут метаданные и экстенды, адресация которых происходит уже на логическом уровне. На этом же уровне реализуется избыточность (аналог RAID 0, 1, 5, 6, 10). Предпоследний уровень — субтома (Subvolumes). На верхнем уровне находятся данные пользователя в обычном виде. Btrfs не использует суперблок и другие привычные элементы файловой системы. Ext4: 1) Суперблок. Находится в самом начале файловой системы. 2) Глобальная таблица дескрипторов группы блоков. В ней описаны первый и последний блоки для каждой группы блоков. 3) Битовая карта блока. Это специальная таблица, где указано, какие блоки в группе использованы, а какие свободны. 4) Битовая карта inodes. Аналогична битовой карте блока, но с информацией о свободных inodes, которые могут быть использованы для записи новых файлов. 5) Блоки данных. Физические блоки памяти, с данными пользователя.

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

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

Некоторые основные причины нарушения целостности файловой системы в Linux: Случайное удаление файлов. Неисправности оборудования. Вредоносные программы и вирусы. Ошибка обновлений и настройка системы. Проблемы с логическими разделами и форматирование. Сбои в работе RAID-массивов. Ошибки пользователей при выполнении административных задач.

Для устранения повреждений файловой системы в Linux можно использовать утилиту fsck (File System Check). Она проверяет целостность файловой системы и исправляет обнаруженные ошибки. Некоторые рекомендации по использованию fsck: Делать резервные копии перед проверкой. Не запускать fsck на смонтированных разделах. Использовать флаг -n для первичного анализа.

5. Как создаётся файловая система? Ответ: Файловая система в Linux создаётся с помощью команды `mkfs`. Она организует иерархию каталогов, подкаталогов и файлов на отформатированном запоминающем устройстве (разделе на жёстком диске, USB-накопителе или другом носителе информации).

6. Дайте характеристику командам для просмотра текстовых файлов.

Некоторые команды для просмотра текстовых файлов в Linux и их характеристика: 1) `cat`. Выводит содержимое файла на экран. 2) `nl`. Действует аналогично команде `cat`, но выводит ещё и номера строк в столбце слева. 3) `less`. Выводит содержимое файла, но отображает его только в рамках текущего окна в режиме просмотра. Можно прокручивать текст файла клавишами стрелок или перелистывать страницы клавишами `w` и `z`. Чтобы выйти из режима просмотра используется клавиша `q`. 4) `head`. Выводит на экран только первые 10 строк файла. Используя опцию `-n`, можно задать количество строк, которое нужно вывести. 5) `tail`. Аналогична команде `head`, но выводит последние 10 строк файла. С помощью команды `tail` можно увидеть изменения, сделанные в файле в реальном времени.

7. Приведите основные возможности команды `cp` в Linux. Ответ: 1) Рекурсивное копирование директорий со всем содержимым (опция `-r` или `-recursive`). 2) Подробный вывод информации о процессе копирования (опция `-v` или `-verbose`). 3) Запрос подтверждения перед копированием файлов (опция `-i` или `-interactive`). 4) Копирование только новых или обновлённых файлов (опция `-u` или `-update`). 5) Сохранение всех атрибутов файлов (владелец, права, временные метки) (опция `-p` или `-preserve`). 6) Создание резервной копии для существующих файлов (опция `-b` или `-backup`). 7) Принудительная перезапись существующих файлов назначения (опция `-f`). 8) Обработка разреженных файлов. По умолчанию `cp` создаёт разреженные файлы только если файл-источник разрежен, но можно настроить это поведение так, чтобы все файлы, содержащие последовательности нулей, делались разреженными (опция `-sparse`).

8. Приведите основные возможности команды `mv` в Linux. Ответ: 1) Перемещение файлов и каталогов из одного места в другое. Можно перемещать несколько файлов одновременно. 2) Переименование файлов и каталогов. Если в качестве аргументов заданы имена двух файлов, то имя первого файла будет изменено на имя второго. 3) Создание резервных копий существующих файлов, которые будут перезаписаны или удалены. 4) Перемещение отсутствующих в целевой директории файлов. Если в определённую директорию нужно переместить ряд файлов, однако некоторые из них уже имеются в данном каталоге, используется опция `-n`. 5) Подробный вывод информации о перемещаемых или переименовываемых файлах. Для этого используется опция `-v`. 6) Перемещение файлов на основе шаблона. С помощью команды `mv` можно перемещать файлы на основе расширения файлов или части имени файла.

9. Что такое права доступа? Как они могут быть изменены? Ответ: Права доступа в Linux — это система, которая регулирует взаимодействие пользователей с файлами и папками. Они устанавливают, кто имеет возможность читать (просматривать), изменять (записывать), выполнять (исполнять) файлы и программы. Чтобы изменить права доступа к файлу или папке в Linux, используется команда `chmod`. Она позволяет устанавливать или изменять права чтения, записи и выполнения для владельца файла, группы и остальных пользователей.

Мы ознакомились с файловой системой Linux, её структурой, именами и содержанием каталогов. Также 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.