

Лабораторная работа №5

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

Лебедева Ольга Андреевна

Содержание

1	Цель работы	6
2	Теоретическое введение	7
3	Ход работы	8
4	Задание 1	9
5	Задание 2	11
6	Задание 3	14
7	Задание 4	16
8	Вывод	21
9	Контрольные вопросы	22

Список иллюстраций

4.1	Команды touch, cat, tac	9
4.2	Команды head, tail	10
5.1	Проверили наличие файла io.h	11
5.2	Копировали файл io.h	11
5.3	Файл io.h в домашнем каталоге	11
5.4	Переименование файла	11
5.5	Создание директории	12
5.6	Перемещение файла	12
5.7	Переименование файла	12
5.8	Действия с файлом abc1	12
5.9	Переместили два файла	13
5.10	Проверка наличия файлов	13
5.11	Действия с каталогом	13
6.1	Создание необходимых файлов и каталогов	14
6.2	Добавление прав	14
6.3	Лишение прав	14
6.4	Итог	14
6.5	Итог	15
6.6	Меняем права	15
6.7	Итог	15
6.8	Итог	15
7.1	Просмотр файла	16
7.2	Копирование файла	16
7.3	Копирование файла	16
7.4	Переименование каталога	17
7.5	Лишение права на чтение	17
7.6	Отказано в доступе	17
7.7	Отказано в доступе	17
7.8	Дали право на чтение	17
7.9	Лишили права на выполнение	17
7.10	Перешли в каталог	18
7.11	Дали право на выполнение	18
7.12	man	18
7.13	mount	18
7.14	fsck	19

7.15 kill 19

7.16 mkfs 20

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

Для создания текстового файла можно использовать команду `touch`.

Для просмотра файлов небольшого размера можно использовать команду `cat`.

Для просмотра файлов постранично удобнее использовать команду `less`.

Следующие клавиши используются для управления процессом просмотра:

- `Space` — переход к следующей странице,
- `ENTER` — сдвиг вперёд на одну строку,
- `b` — возврат на предыдущую страницу,
- `h` — обращение за подсказкой,
- `q` — выход из режима просмотра файла.

3 Ход работы

4 Задание 1

Выполнили все примеры, приведённые в первой части описания лабораторной работы. (рис. 4.1) (рис. 4.2)

```
oalebedeva@dk3n57 ~ $ touch Hello
oalebedeva@dk3n57 ~ $ ls
Architecture_PC  GNUstep  public  temp
bin              Hello    public_html  tmp
oalebedeva@dk3n57 ~ $ cat Hello
привет, мир!
Погода в Мск +9 градусов

oooooooooooooooooooooooooooo
oalebedeva@dk3n57 ~ $ tac Hello
oooooooooooooooooooooooooooo

Погода в Мск +9 градусов
привет, мир!
```

Рис. 4.1: Команды touch, cat, tac

5 Задание 2

Скопировали файл `/usr/include/sys/io.h` в домашний каталог и назвали его `equipment`. (рис. 5.1) (рис. 5.2) (рис. 5.3) (рис. 5.4)

```
oalebedeva@dk3n57 /usr/include/sys $ ls
acct.h      gmon_out.h  procfs.h    signal.h     timex.h
acl.h       inotify.h   profil.h    single_threaded.h  ttychars.h
asoundlib.h ioctl.h     psx_syscall.h socket.h     ttydefaults.h
auxv.h      io.h        ptrace.h    socketvar.h  types.h
```

Рис. 5.1: Проверили наличие файла `io.h`

```
oalebedeva@dk3n57 /usr/include/sys $ cp -r io.h ~
```

Рис. 5.2: Копировали файл `io.h`

```
oalebedeva@dk3n57 /usr/include/sys $ cd
oalebedeva@dk3n57 ~ $ ls
Architecture_PC  io.h      tmp      Загрузки      'Рабочий стол'
bin              public    work     Изображения   Шаблоны
GNUstep          public_html Видео     Музыка
Hello            temp      Документы  Общедоступные
```

Рис. 5.3: Файл `io.h` в домашнем каталоге

```
oalebedeva@dk3n57 ~ $ mv io.h equipment
oalebedeva@dk3n57 ~ $ ls
Architecture_PC  Hello      tmp      Загрузки      'Рабочий стол'
bin              public     work     Изображения   Шаблоны
equipment        public_html Видео     Музыка
GNUstep          temp      Документы  Общедоступные
```

Рис. 5.4: Переименование файла

В домашнем каталоге создали директорию `~/ski.plases`. (рис. 5.5)

```
oalebedeva@dk3n57 ~ $ mkdir ski.plases
oalebedeva@dk3n57 ~ $ ls
Architecture_PC  Hello      temp      Документы  Общедоступные
bin              public    tmp       Загрузки   'Рабочий стол'
equipment        public_html work      Изображения Шаблоны
GNUstep          ski.plases Видео     Музыка
```

Рис. 5.5: Создание директории

Переместили файл equipment в каталог ~/ski.plases. (рис. 5.6)

```
oalebedeva@dk3n57 ~ $ mv equipment ski.plases
oalebedeva@dk3n57 ~ $ cd ski.plases/
oalebedeva@dk3n57 ~/ski.plases $ ls
equipment
oalebedeva@dk3n57 ~/ski.plases $
```

Рис. 5.6: Перемещение файла

Переименовали файл /ski.plases/equipment в ~/ski.plases/equiplist. (рис. 5.7)

```
oalebedeva@dk3n57 ~/ski.plases $ mv equipment equiplist
oalebedeva@dk3n57 ~/ski.plases $ ls
equiplist
```

Рис. 5.7: Переименование файла

Создали в домашнем каталоге файл abc1 и скопировали его в каталог ~/ski.plases, назвали его equiplist2. (рис. 5.8)

```
oalebedeva@dk3n57 ~ $ touch abc1
oalebedeva@dk3n57 ~ $ ls
abc1      Hello      temp      Документы  Общедоступные
Architecture_PC public    tmp       Загрузки   'Рабочий стол'
bin       public_html work      Изображения Шаблоны
GNUstep   ski.plases Видео     Музыка
oalebedeva@dk3n57 ~ $ cp abc1 ski.plases
oalebedeva@dk3n57 ~ $ cd ski.plases/
oalebedeva@dk3n57 ~/ski.plases $ ls
abc1  equiplist
oalebedeva@dk3n57 ~/ski.plases $ mv abc1 equiplist2
oalebedeva@dk3n57 ~/ski.plases $ ls
equiplist  equiplist2
```

Рис. 5.8: Действия с файлом abc1

Создали каталог с именем equipment в каталоге ~/ski.plases.

Создали каталог с именем equipment в каталоге ~/ski.plases.

Переместили файлы /ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment.

(рис. 5.9) (рис. 5.10)

```
oalebedeva@dk3n57 ~/ski.plases $ mv equiplist2 equiplist equipment
```

Рис. 5.9: Переместили два файла

```
oalebedeva@dk3n57 ~/ski.plases $ cd equipment/  
oalebedeva@dk3n57 ~/ski.plases/equipment $ ls  
equiplist  equiplist2
```

Рис. 5.10: Проверка наличия файлов

Создали и переместили каталог /newdir в каталог ~/ski.plases и назвали его plans. (рис. 5.11)

```
oalebedeva@dk3n57 ~ $ mkdir newdir  
oalebedeva@dk3n57 ~ $ ls  
abc1      Hello      ski.plases  Видео      Музыка  
Architecture_PC newdir     temp       Документы  Общедоступные  
bin        public     tmp        Загрузки   'Рабочий стол'  
GNUstep    public_html work       Изображения  Шаблоны  
oalebedeva@dk3n57 ~ $ mv newdir ski.plases  
oalebedeva@dk3n57 ~ $ cd ski.plases/  
oalebedeva@dk3n57 ~/ski.plases $ ls  
equipment newdir  
oalebedeva@dk3n57 ~/ski.plases $ mv newdir plans  
oalebedeva@dk3n57 ~/ski.plases $ ls  
equipment plans
```

Рис. 5.11: Действия с каталогом

6 Задание 3

Определили опции команды `chmod`, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет: (рис. 6.1)

```
oalebedeva@dk3n57 ~ $ mkdir australia
oalebedeva@dk3n57 ~ $ mkdir play
oalebedeva@dk3n57 ~ $ touch my_os
oalebedeva@dk3n57 ~ $ touch feathers
oalebedeva@dk3n57 ~ $ ls
abc1          GNUstep      public_html  Видео        Общедоступные
Architecture_PC Hello        ski.places   Документы    'Рабочий стол'
australia     my_os       temp         Загрузки     Шаблоны
bin           play        tmp          Изображения
feathers      public      work         Музыка
```

Рис. 6.1: Создание необходимых файлов и каталогов

3.1. `drwxr--r--` ... `australia` (рис. 6.2) (рис. 6.3) (рис. 6.4)

```
oalebedeva@dk3n57 ~ $ chmod u+r australia
oalebedeva@dk3n57 ~ $ chmod u+w australia
oalebedeva@dk3n57 ~ $ chmod u+x australia
```

Рис. 6.2: Добавление прав

```
oalebedeva@dk3n57 ~ $ chmod g-x australia
oalebedeva@dk3n57 ~ $ chmod o-x australia
```

Рис. 6.3: Лишение прав

```
drwxr--r--  2 oalebedeva studsci  2048 мая  4 13:41  australia
```

Рис. 6.4: Итог

3.2. drwx-x-x ... play (рис. 6.5)

```
drwx--x--x  2 oalebedeva studsci  2048 мая  4 13:41  play
```

Рис. 6.5: Итог

3.3. -r-xr-r- ... my_os (рис. 6.7) (рис. 6.6)

```
oalebedeva@dk3n57 ~ $ chmod u-w my_os  
oalebedeva@dk3n57 ~ $ chmod u+x my_os
```

Рис. 6.6: Меняем права

```
-r-xr--r--  1 oalebedeva studsci    0 мая  4 13:41  my_os
```

Рис. 6.7: Итог

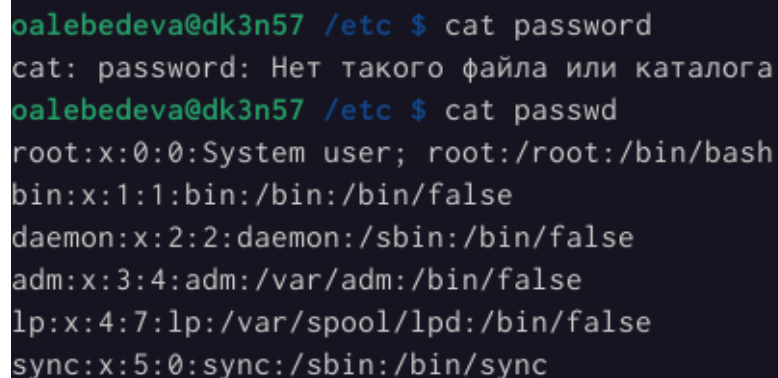
3.4. -rw-rw-r- ... feathers (рис. 6.8)

```
-rw-rw-r--  1 oalebedeva studsci    0 мая  4 13:42  feathers
```

Рис. 6.8: Итог

7 Задание 4

Просмотрели содержимое файла /etc/passwd. (рис. 7.1)



```
oalebedeva@dk3n57 /etc $ cat passwd
cat: passwd: Нет такого файла или каталога
oalebedeva@dk3n57 /etc $ cat 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
```

Рис. 7.1: Просмотр файла

Скопировали файл /feathers в файл ~/file.old. (рис. 7.2)



```
oalebedeva@dk3n57 ~ $ cp feathers file.old
```

Рис. 7.2: Копирование файла

Переместили файл /file.old в каталог ~/play. (рис. 7.3)



```
oalebedeva@dk3n57 ~ $ mv file.old play/
```

Рис. 7.3: Копирование файла

Скопировали каталог /play в каталог /fun. Переместили каталог /fun в каталог ~/play и назвали его games.(рис. 7.4)


```
oalebedeva@dk3n57 ~ $ mv fun play/games
```

Рис. 7.4: Переименование каталога

Лишили владельца файла ~/feathers права на чтение. (рис. 7.5)

```
oalebedeva@dk3n57 ~ $ chmod u-r feathers
```

Рис. 7.5: Лишение права на чтение

Что произойдёт, если вы попытаетесь просмотреть файл ~/feathers командой cat? (рис. 7.6)

```
oalebedeva@dk3n57 ~ $ cat feathers  
cat: feathers: Отказано в доступе
```

Рис. 7.6: Отказано в доступе

Что произойдёт, если вы попытаетесь скопировать файл ~/feathers? (рис. 7.7)

```
oalebedeva@dk3n57 ~ $ cp feathers fun/  
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
```

Рис. 7.7: Отказано в доступе

Дали владельцу файла ~/feathers право на чтение. (рис. 7.8)

```
oalebedeva@dk3n57 ~ $ chmod u+r feathers
```

Рис. 7.8: Дали право на чтение

Лишили владельца каталога ~/play права на выполнение. (рис. 7.9)

```
oalebedeva@dk3n57 ~ $ chmod u-x play
```

Рис. 7.9: Лишили права на выполнение

Перейдите в каталог ~/play. Что произошло? (рис. 7.10)

```
oalebedeva@dk3n57 ~ $ chmod u-x play
oalebedeva@dk3n57 ~ $ cd play
oalebedeva@dk3n57 ~/play $
```

Рис. 7.10: Перешли в каталог

Дали владельцу каталога ~/play право на выполнение. (рис. 7.11)

```
oalebedeva@dk3n57 ~ $ chmod u+x play
```

Рис. 7.11: Дали право на выполнение

Прочитали man по командам mount, fsck, mkfs, kill и кратко их охарактеризовали, приведя примеры. (рис. 7.13) (рис. 7.14) (рис. 7.15) (рис. 7.16) (рис. 7.12)

```
oalebedeva@dk3n57 ~ $ man mount
oalebedeva@dk3n57 ~ $ man fsck
oalebedeva@dk3n57 ~ $ man kill
oalebedeva@dk3n57 ~ $ man mkfs
```

Рис. 7.12: man

```
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|rpri-
    vate|runbindable} mountpoint
```

Рис. 7.13: mount

```

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. 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). Nor-
    mally, 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.

```

Рис. 7.14: fsck

```

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 avail-
    able 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 out-
    put. 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.

```

Рис. 7.15: kill

```
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 status returned by mkfs is 0 on success and 1 on failure.
```

Рис. 7.16: mkfs

8 Вывод

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.

9 Контрольные вопросы

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

- Tmpfs — временное файловое хранилище в Unix . Предназначена для монтирования файловой системы, но размещается в ОЗУ вместо физического диска. Подобная конструкция является RAM диском. Все данные в Tmpfs являются временными, в том смысле, что ни одного файла не будет создано на жёстком диске. После перезагрузки все данные, содержащиеся в Tmpfs, будут утеряны.

- Fourth extended file system , сокр. ext4, или ext4fs — журналируемая ФС, используемая в ОС с ядром Linux. Основана на ФС ext3, ранее использовавшейся по умолчанию во многих дистрибутивах GNU/Linux.

- Распределенная сетевая файловая система AFS (Andrew File System) внедрена с целью создания единого файлового пространства пользователей при работе на различных фермах или рабочих станциях ОИЯИ, где домашний каталог пользователя определен как каталог в -AFS. Использование системы AFS позволяет пользователям осуществлять прямой доступ к файловому пространству других организаций, где эта система используется и где пользователь имеет регистрацию в AFS.

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

Все каталоги можно разделить на две группы: для статической (редко меняющейся) информации – /bin, /usr и динамической (часто меняющейся) информации – /var, /tmp. Исходя из этого администраторы могут разместить каждый

из этих каталогов на собственном носителе, обладающем соответствующими характеристиками.

- Корневой каталог. Корневой каталог / является основой любой ФС UNIX. Все остальные каталоги и файлы располагаются в рамках структуры (дерева), порождённой корневым каталогом, независимо от их физического местонахождения.

- /bin. В этом каталоге находятся часто употребляемые команды и утилиты системы общего пользования. Сюда входят все базовые команды, доступные даже если была примонтирована только корневая файловая система. Примерами таких команд являются: `ls`, `cp` и т.д.

- /boot. Директория содержит всё необходимое для процесса загрузки операционной системы: программу-загрузчик, образ ядра операционной системы и т.п..

- /dev. Каталог содержит специальные файлы устройств, являющиеся интерфейсом доступа к периферийным устройствам. Наличие такого каталога не означает, что специальные файлы устройств нельзя создавать в другом месте, просто достаточно удобно иметь один каталог для всех файлов такого типа.

- /etc. В этом каталоге находятся системные конфигурационные файлы. В качестве примеров можно привести файлы `/etc/fstab`, содержащий список монтируемых файловых систем, и `/etc/resolv.conf`, который задаёт правила составления локальных DNS-запросов. Среди наиболее важных файлов – скрипты инициализации и деинициализации системы. В системах, наследующих особенности UNIX System V, для них отведены каталоги с `/etc/rc0.d` по `/etc/rc6.d` и общий для всех файл описания – `/etc/inittab`.

- /home (необязательно). Директория содержит домашние директории пользователей. Её существование в корневом каталоге не обязательно и её содержимое зависит от особенностей конкретной UNIX-подобной операционной системы.

- /lib. Каталог для статических и динамических библиотек, необходимых для запуска программ, находящихся в директориях `/bin`, `/sbin`.

- /mnt. Стандартный каталог для временного монтирования файловых систем

– например, гибких и флэш-дисков, компакт-дисков и т. п.

- /root (необязательно). Директория содержит домашнюю директорию суперпользователя. Её существование в корневом каталоге не обязательно.

- /sbin. В этом каталоге находятся команды и утилиты для системного администратора. Примерами таких команд являются: route, halt, init и др. Для аналогичных целей применяются директории /usr/sbin и /usr/local/sbin.

- /usr. Эта директория повторяет структуру корневой директории – содержит каталоги /usr/bin, /usr/lib, /usr/sbin, служащие для аналогичных целей. Каталог /usr/include содержит заголовочные файлы языка С для всевозможных библиотек, расположенных в системе.

- /usr/local является следующим уровнем повторения корневого каталога и служит для хранения программ, установленных администратором в дополнение к стандартной поставке операционной системы.

- /usr/share хранит неизменяющиеся данные для установленных программ. Особый интерес представляет каталог /usr/share/doc, в который добавляется документация ко всем установленным программам.

- /var, /tmp. Используются для хранения временных данных процессов – системных и пользовательских соответственно.

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

С помощью команды cd мы переходим в каталог, в котором находится файл. С помощью less мы открываем этот файл.

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

Основные причины нарушения целостности файловой системы:

- Из-за прерывания операций ввода-вывода выполняемых непосредственно с диском;
- Сбоя питания;

- Краха ОС;
- Нарушения работы дискового КЭШа;

Устранение поврежденных файлов: В большинстве случаев, проверка файловой системы способна обнаружить и выполнить ремонт такой ошибки автоматически, и после завершения процесс начальной загрузки продолжится как обычно.

Если проблема файловой системы более серьезна, проверка файловой системы не может решить проблему автоматически. В этом случае процесс надо будет запустить вручную.

5. Как создаётся файловая система?

Обычно при установке Linux создание файловых систем - компетенция инсталлятора, который осуществляет его с некоторыми опциями по умолчанию. Изменить характеристики, определенные для файловой системы при ее создании, невозможно без повторного выполнения этого процесса.

Файловая система Ext2fs может быть создана любой из следующих команд - /sbin/mke2fs, /sbin/mkfs, /sbin/mkfs.ext2 с указанием файла устройства в качестве аргумента.

Для создания XFS -mkfs.xfs (из пакета xfsprogs).

Для создания файловой системы ext3fs -mke2fs с опцией j.

Файловая система ReiserFS - /sbin/mkreiserfs из пакета reiserfsprogs.

6. Дайте характеристику командам для просмотра текстовых файлов.

- Для просмотра небольших файлов -cat. (cat имя-файла)
- Для просмотра больших файлов-less . (less имя-файла)
- Для просмотра начала файла-head. По умолчанию она выводит первые 10 строк файла. (head [-n] имя-файла), n — количество выводимых строк.
- Команда tail . выводит несколько (по умолчанию 10) последних строк файла. (tail [-n] имя-файла),n — количество выводимых строк.

7. Приведите основные возможности команды cp в Linux.

При помощи команды `cp` осуществляется копирование файлов и каталогов (`cp`[-опции] исходный_файл целевой_файл) Возможности команды `cp`:

- копирование файла в текущем каталоге
- копирование нескольких файлов в каталог
- копирование файлов в произвольном каталоге
- опция `i` в команде `cp` поможет избежать уничтожения информации в случае, если на место целевого файла вы поставите имя уже существующего файла(т.е. система попросит подтвердить, что вы хотите перезаписать этот файл)
- Команда `cp` с опцией `r` (`recursive`) позволяет копировать каталоги вместе с входящими в них файлами и каталогами.

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

Команды `mv` и `mkdir` предназначены для перемещения и переименования файлов и каталогов. (`mv` [-опции] старый_файл новый_файл)

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

Права доступа - совокупность правил, регламентирующих порядок и условия доступа субъекта к объектам информационной системы (информации, её носителям, процессам и другим ресурсам) установленных правовыми документами или собственником, владельцем информации. Для изменения прав доступа к файлу или каталогу используется команда `chmod`. (`chmod` режим имя_файла).

Права доступа к файлу может поменять только владелец и администратор.

Режим (в формате команды) имеет следующую структуру и способ записи:

- `=` установить право
- `-` лишить права
- `+` дать право
- `r` чтение
- `w` запись
- `x` выполнение

- u (user) владелец файла
- g (group) группа, к которой принадлежит владелец файла
- (others) все остальные.