

Отчет по Лабораторной работе №5

Дисциплина - Операционные системы

Газизова Регина

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	9
4	Выводы	15
5	Контрольные вопросы	16

Список иллюстраций

[illegible]

Список таблиц

1 Цель работы

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

2 Задание

- Выполните все примеры, приведённые в первой части описания лабораторной работы.
- Выполните следующие действия, зафиксировав в отчёте по лабораторной работе используемые при этом команды и результаты их выполнения:
 - Скопируйте файл `/usr/include/sys/io.h` в домашний каталог и назовите его `equipment`. Если файла `io.h` нет, то используйте любой другой файл в каталоге `/usr/include/sys/` вместо него.
 - В домашнем каталоге создайте директорию `~/ski.places`.
 - Переместите файл `equipment` в каталог `~/ski.places`.
 - Переименуйте файл `~/ski.places/equipment` в `~/ski.places/equiplist`.
 - Создайте в домашнем каталоге файл `abc1` и скопируйте его в каталог `~/ski.places`, назовите его `equiplist2`.
 - Создайте каталог с именем `equipment` в каталоге `~/ski.places`.
 - Переместите файлы `~/ski.places/equiplist` и `equiplist2` в каталог `~/ski.places/equipment`.
 - Создайте и переместите каталог `~/newdir` в каталог `~/ski.places` и назовите его `plans`.
- Определите опции команды `chmod`, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет:
 - `drwxr-r- ... australia`
 - `drwx-x-x ... play`

- `-r-xr-r- ... my_os`
 - `-rw-rw-r- ... feathers`
- При необходимости создайте нужные файлы.
- Прodelайте приведённые ниже упражнения, записывая в отчёт по лабораторной работе используемые при этом команды:
 - Просмотрите содержимое файла `/etc/passwd`.
 - Скопируйте файл `~/feathers` в файл `~/file.old`.
 - Переместите файл `~/file.old` в каталог `~/play`.
 - Скопируйте каталог `~/play` в каталог `~/fun`.
 - Переместите каталог `~/fun` в каталог `~/play` и назовите его `games`.
 - Лишите владельца файла `~/feathers` права на чтение.
 - Что произойдёт, если вы попытаетесь просмотреть файл `~/feathers` командой `cat`?
 - Что произойдёт, если вы попытаетесь скопировать файл `~/feathers`?
 - Дайте владельцу файла `~/feathers` право на чтение.
 - Лишите владельца каталога `~/play` права на выполнение.
 - Перейдите в каталог `~/play`. Что произошло?
 - Дайте владельцу каталога `~/play` право на выполнение.
- Прочитайте манпо командам `mount`, `fsck`, `mkfs`, `kill` и кратко их охарактеризуйте, приведя примеры. # Теоретическое введение
- Для создания текстового файла можно использовать команду `touch`
- Для просмотра файлов небольшого размера можно использовать команду `cat`.
- Команда `cp` используется для копирования файлов и каталогов.
- Команды `mv` и `mkdir` предназначены для перемещения и переименования файлов и каталогов.

- Права доступа к файлу или каталогу можно изменить, воспользовавшись командой `chmod`.

3 Выполнение лабораторной работы

Создаем файл abc1 и копируем его в файлы april и may с помощью команды cp (рис. 3.1)

```
rigazizova@dk6n60 ~ $ touch abc1
rigazizova@dk6n60 ~ $ cp abc1 april
rigazizova@dk6n60 ~ $ cp abc1 may
rigazizova@dk6n60 ~ $ ls
abc1  Architecture_PC  GNUstep  may  public_html  tmp  Видео  Загрузки  Музыка  'Рабочий стол'
april  bin               lab2.1.pub  public  R            work  Документы  Изображения  Общедоступные  Шаблоны
```

Рис. 3.1: 1

Создаем каталог monthly и копируем в него файлы april и may.(рис. 3.2)

```
rigazizova@dk6n60 ~ $ mkdir monthly
rigazizova@dk6n60 ~ $ cp april may monthly
rigazizova@dk6n60 ~ $ cd monthly
rigazizova@dk6n60 ~/monthly $ ls
april  may
```

Рис. 3.2: 2

Переименовываем файл may в файл june с помощью той же команды cp.(рис. 3.3)

```
rigazizova@dk6n60 ~/monthly $ cd ..
rigazizova@dk6n60 ~ $ cp monthly/may monthly/june
rigazizova@dk6n60 ~ $ ls monthly
april  june  may
```

Рис. 3.3: 3

Создаем новый каталог и перемещаем в него уже существующий. (рис. 3.4)

```
rigazizova@dk6n60 ~ $ mkdir monthly.00
rigazizova@dk6n60 ~ $ cp -r monthly monthly.00
rigazizova@dk6n60 ~ $ cp -r monthly.00 /tmp
rigazizova@dk6n60 ~ $ ls monthly.00
monthly
```

Рис. 3.4: 4

Переместили файл `july` в каталог `monthly.00` (рис. 3.5)

```
rigazizova@dk6n60 ~ $ mv april july
rigazizova@dk6n60 ~ $ mv july monthly.00
rigazizova@dk6n60 ~ $ ls monthly.00
july  monthly
rigazizova@dk6n60 ~ $ ls -R monthly.00
monthly.00:
july  monthly

monthly.00/monthly:
april  june  may
```

Рис. 3.5: 5

Переименовали каталог `monthly.00` в `monthly.01`. Переместили каталог `monthly.01` в каталог `reports`. Переименовали каталог `reports/monthly.01` в `reports/monthly` (рис. 3.6)

```
rigazizova@dk6n60 ~ $ mv monthly.00 monthly.01
rigazizova@dk6n60 ~ $ mkdir reports
rigazizova@dk6n60 ~ $ mv monthly.01 reports
rigazizova@dk6n60 ~ $ mv reports/monthly.01 reports/monthly
```

Рис. 3.6: 6

Создали файл `~/may` с правом выполнения для владельца. Лишили владельца файла `~/may` права на выполнение (рис. 3.7)

```
rigazizova@dk6n60 ~ $ touch may
rigazizova@dk6n60 ~ $ ls -l may
-rw-r--r-- 1 rigazizova studsci 0 мая 5 15:38 may
rigazizova@dk6n60 ~ $ chmod u+x may
rigazizova@dk6n60 ~ $ ls -l may
-rwxr--r-- 1 rigazizova studsci 0 мая 5 15:38 may
rigazizova@dk6n60 ~ $ chmod u-x may
rigazizova@dk6n60 ~ $ ls -l may
-rw-r--r-- 1 rigazizova studsci 0 мая 5 15:38 may
```

Рис. 3.7: 7

Создали каталог monthly с запретом на чтение для членов группы и всех остальных пользователей (рис. 3.8)

```
rigazizova@dk6n60 ~ $ chmod o-r monthly
rigazizova@dk6n60 ~ $ chmod g-r monthly
rigazizova@dk6n60 ~ $ ls -l monthly
итого 0
-rw-r--r-- 1 rigazizova studsci 0 мая 5 15:30 april
-rw-r--r-- 1 rigazizova studsci 0 мая 5 15:31 june
-rw-r--r-- 1 rigazizova studsci 0 мая 5 15:30 may
```

Рис. 3.8: 8

Создали файл ~/abc1 с правом записи для членов группы (рис. 3.9)

```
rigazizova@dk6n60 ~ $ touch abc1
rigazizova@dk6n60 ~ $ chmod g+w abc1
```

Рис. 3.9: 9

Скопировали файл /usr/include/sys/io.h в домашний каталог и назвали его equipment. (рис. 3.10)

```
rigazizova@dk6n60 ~ $ cp /usr/include/sys/io.h equipment
rigazizova@dk6n60 ~ $ ls
abc1      equipment  may      public_html tmp      Документы  Музыка  Шаблоны
Architecture_PC  GNUstep  monthly  R        work     Загрузки  Общедоступные
bin       lab2_1.pub public   reports  Видео    Изображения 'Рабочий стол'
```

Рис. 3.10: 10

В домашнем каталоге создали директорию ~/ski.plases (рис. 3.11)

```
rigazizova@dk6n60 ~ $ mkdir ~/ski.plases
rigazizova@dk6n60 ~ $ ls
abc1      equipment  may      public_html ski.plases Видео  Изображения 'Рабочий стол'
Architecture_PC  GNUstep  monthly  R        tmp     Документы  Музыка  Шаблоны
bin       lab2_1.pub public   reports  work    Загрузки  Общедоступные
```

Рис. 3.11: 11

Переместили файл equipment в каталог ~/ski.plases. Переименовали файл ~/ski.plases/equipment в ~/ski.plases/equiplist (рис. 3.12)

```
rigazizova@dk6n60 ~ $ mv equipment ~/ski.plases
rigazizova@dk6n60 ~ $ mv ~/ski.plases/equipment ~/ski.plases/equiplist
rigazizova@dk6n60 ~ $ cd ~/ski.plases
rigazizova@dk6n60 ~/ski.plases $ ls
equiplist
```

Рис. 3.12: 12

Создали в домашнем каталоге файл abc1 и скопировали его в каталог ~/ski.plases, назвали его equiplist2. (рис. 3.13)

```
rigazizova@dk6n60 ~/ski.plases $ mkdir equipment
rigazizova@dk6n60 ~/ski.plases $ cd ..
rigazizova@dk6n60 ~ $ touch abc1
rigazizova@dk6n60 ~ $ cp abc1 ~/ski.plases/equiplist2
rigazizova@dk6n60 ~ $ cd ~/ski.plases
rigazizova@dk6n60 ~/ski.plases $ ls
equiplist equiplist2 equipment
```

Рис. 3.13: 13

Переместили файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment. (рис. 3.14)

```
rigazizova@dk6n60 ~/ski.plases $ mv equiplist equiplist2 equipment
rigazizova@dk6n60 ~/ski.plases $ ls
equipment
rigazizova@dk6n60 ~/ski.plases $ cd equipment
rigazizova@dk6n60 ~/ski.plases/equipment $ ls
equiplist equiplist2
```

Рис. 3.14: 14

Создали и переместили каталог ~/newdir в каталог ~/ski.plases и назвали его plans (рис. 3.15)

```
rigazizova@dk6n60 ~ $ mkdir ~/newdir
rigazizova@dk6n60 ~ $ mv ~/newdir ~/ski.plases/plans
rigazizova@dk6n60 ~ $ cd ~/ski.plases
rigazizova@dk6n60 ~/ski.plases $ ls
equipment plans
```

Рис. 3.15: 15

Создали нужные файлы - australia, play, my_os, feathers. Определили опции команды chmod, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет (744, 711, 544, 664) (рис. 3.16, 3.17, 3.18)

```
rigazizova@dk6n60 ~ $ touch australia play my_os feathers
rigazizova@dk6n60 ~ $ ls
abc1      bin      lab2_1.pub  my_os  public_html  ski.plases  Видео  Изображения  'Рабочий стол'
Architecture_PC  feathers  may        play   R            tmp         Документы  Музыка  Шаблоны
australia  GNUstep  monthly   public reports    work       Загрузки  Общедоступные
```

Рис. 3.16: 16

```
rigazizova@dk6n60 ~ $ chmod 744 australia
rigazizova@dk6n60 ~ $ chmod 711 play
rigazizova@dk6n60 ~ $ chmod 544 my_os
rigazizova@dk6n60 ~ $ chmod 664 feathers
```

Рис. 3.17: 17

```
rigazizova@dk6n60 ~ $ ls -l play australia my_os feathers
-rwxr--r-- 1 rigazizova studsci 0 мая 5 16:02 australia
-rw-rw-r-- 1 rigazizova studsci 0 мая 5 16:02 feathers
-r-xr--r-- 1 rigazizova studsci 0 мая 5 16:02 my_os
-rwx--x--x 1 rigazizova studsci 0 мая 5 16:02 play
```

Рис. 3.18: 18

Скопировали файл ~/feathers в файл ~/file.old. Переместили файл ~/file.old в каталог ~/play. Скопировали каталог ~/play в каталог ~/fun. (рис. 3.19)

```
rigazizova@dk6n60 ~ $ cp ~/feathers ~/file.old
rigazizova@dk6n60 ~ $ mv ~/file.old ~/play
rigazizova@dk6n60 ~ $ cp ~/play ~/fun
```

Рис. 3.19: 19

Лишили владельца файла ~/feathers права на чтение. Попытались просмотреть файл ~/feathers командой cat - не получилось (отказано в доступе). Попытались скопировать файл ~/feathers - тоже не вышло. (рис. 3.20)

```
rigazizova@dk6n60 ~ $ chmod u-r ~/feathers
rigazizova@dk6n60 ~ $ cat ~/feathers
cat: /afs/.dk.sci.pfu.edu.ru/home/r/i/rigazizova/feathers: Отказано в доступе
rigazizova@dk6n60 ~ $ cp ~/feathers ~/fun
cp: невозможно открыть '/afs/.dk.sci.pfu.edu.ru/home/r/i/rigazizova/feathers' для чтения: Отказано в доступе
```

Рис. 3.20: 20

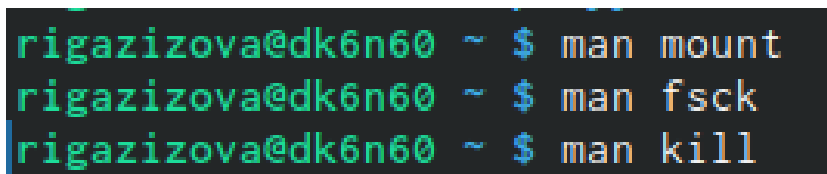
Лишили владельца каталога ~/play всех прав. Перешли в каталог ~/play. Проблем не возникло. (рис. 3.21)

```
rigazizova@dk6n60 ~ $ chmod u-xwr ~/playy
rigazizova@dk6n60 ~ $ cd ~/playy
rigazizova@dk6n60 ~/playy $ cd ..
```

Рис. 3.21: 21

Прочитали man по командам mount - служит для подключения файловых систем разных устройств к этому большому дереву, fsck - используется для проверки

и произвольного восстановления одной или более файловых систем, `mkfs` - используется для создания файловой системы Linux на некотором устройстве, `kill` - может быть использован для посылки какого-либо сигнала какому-либо процессу или группе процесса. (рис. 3.22)

A terminal window with a dark background and light green text. It shows three lines of commands entered by a user named 'rigazizova' on a machine named 'dk6n60'. The commands are 'man mount', 'man fsck', and 'man kill', each preceded by a '\$' prompt character. The output of these commands is not visible.

```
rigazizova@dk6n60 ~ $ man mount
rigazizova@dk6n60 ~ $ man fsck
rigazizova@dk6n60 ~ $ man kill
```

Рис. 3.22: 22

4 Выводы

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

5 Контрольные вопросы

1. Чтобы узнать, какие файловые системы существуют на жёстком диске моего компьютера, использую команду «df -Th». После выполнения команды видно, что на моем компьютере есть следующие файловые системы: devtmpfs, tmpfs, ext4, iso9660. devtmpfs 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 «extent file system», который позволяет добиться меньшей фрагментации файлов и увеличить общую

производительность файловой системы. Кроме того, в Ext4 реализован 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: 16 TB; -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`.
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 и 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 установленных носителям, правовыми процессам и документами другим или ресурсам) 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 r чтение w запись x выполнение u (user) владелец файла g (group) группа, к которой принадлежит владелец файла o (others) все остальные