Лабораторная работа №5

Дисциплина: Операционные системы

Старикова Евгения Дмитриевна

Содержание

# 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 исполь- зования диска и обслуживанию файловой системы

# Выполнение лабораторной работы

1. Выполнить все примеры, приведённые в вверзней части ЛР.

* изучаем команды для работы с файлами и каталогами (touch, cat, less, head, tail ) (рис.1 [-@fig:001])

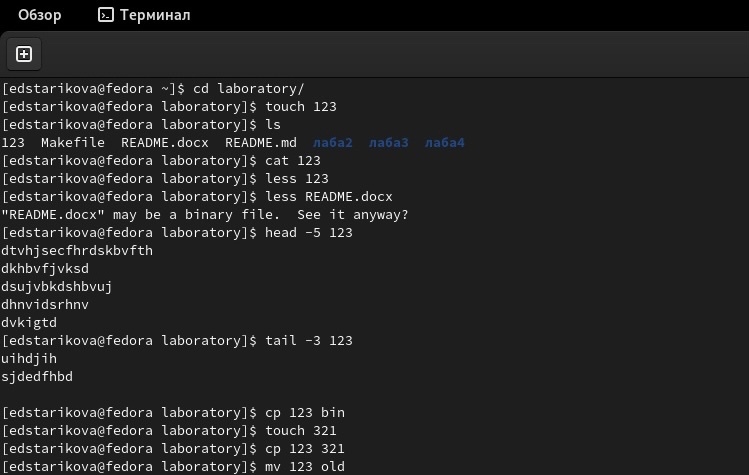
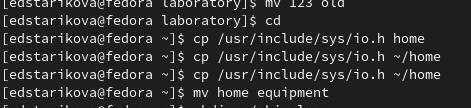
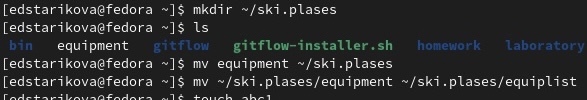
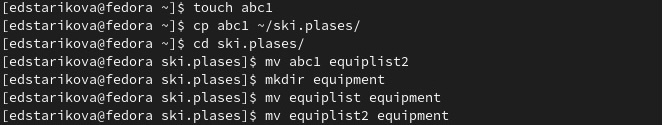
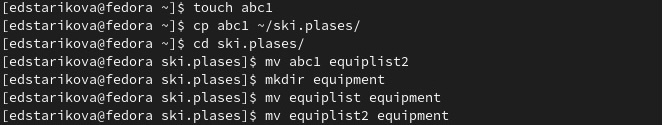
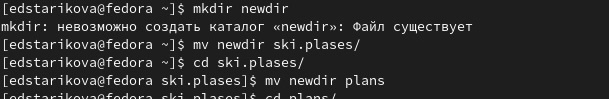
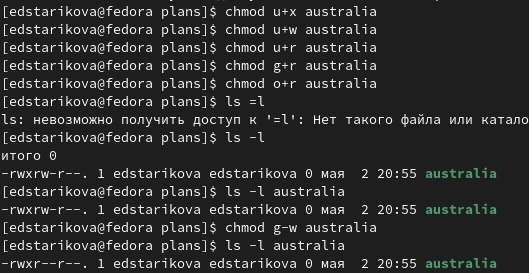
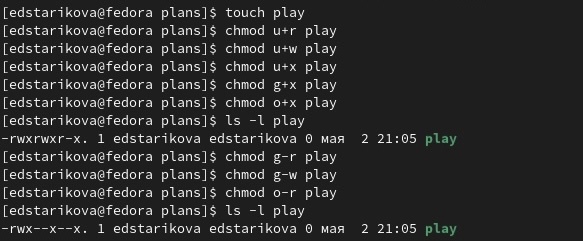
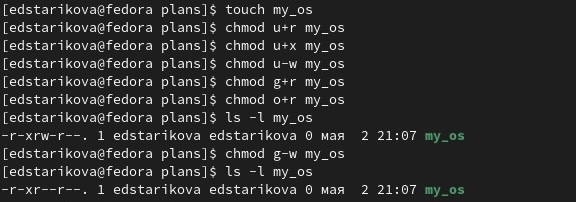
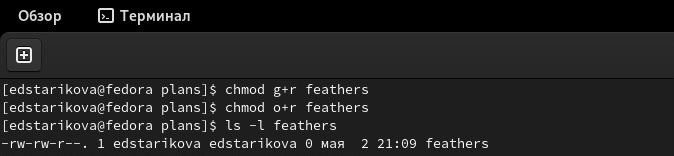
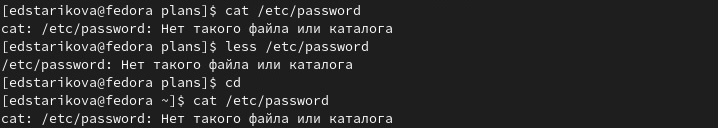
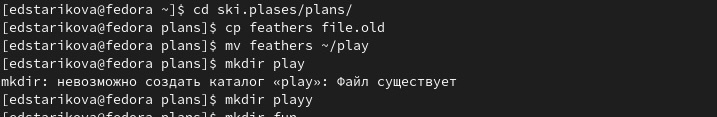
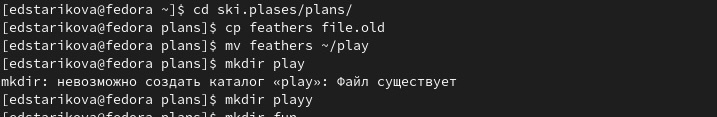
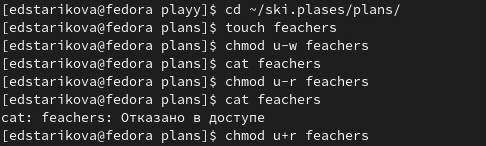
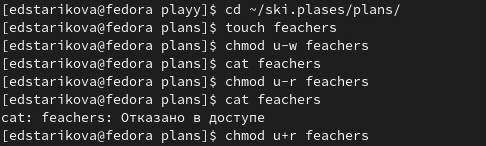
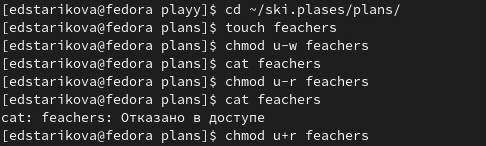
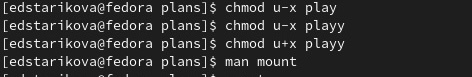
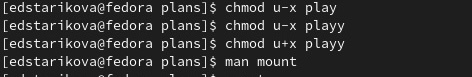
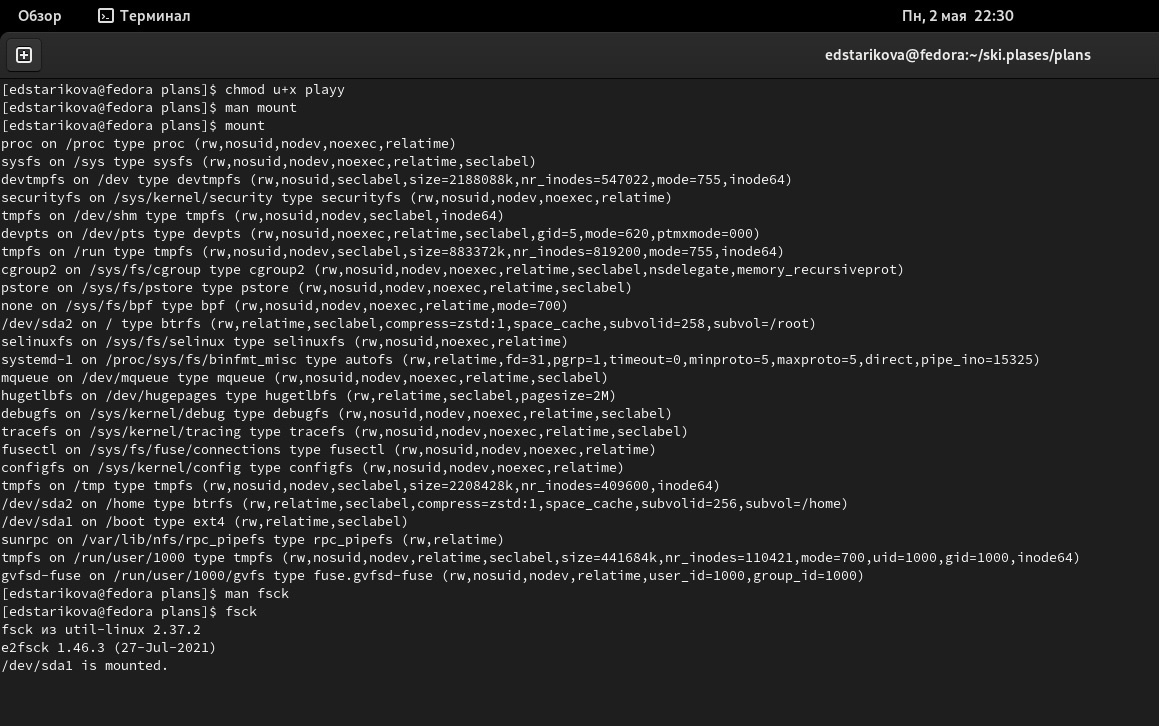
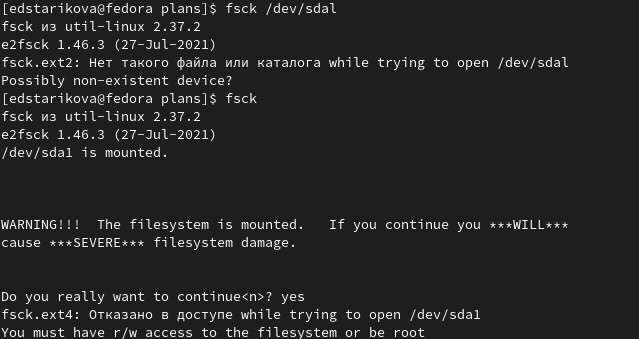
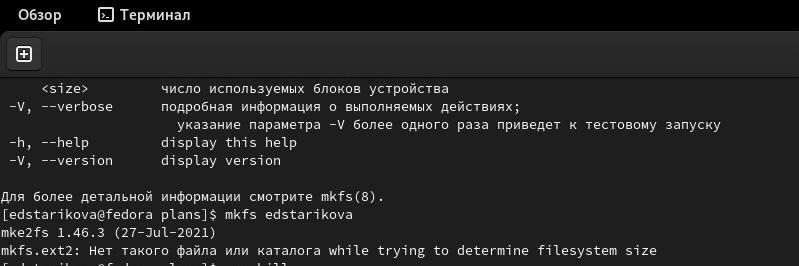
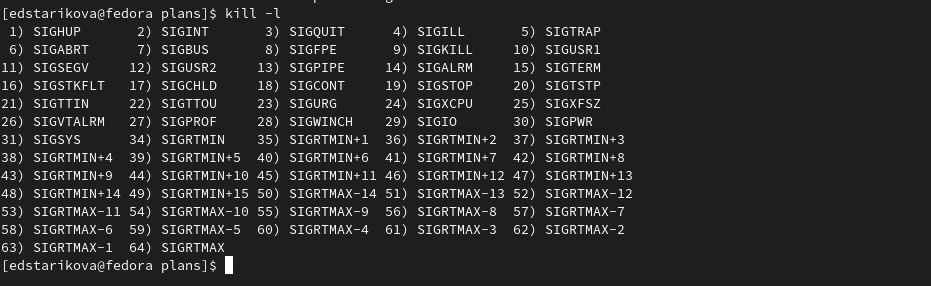


рис. 1

1. Копируем файл /usr/include/sys/io.h в домашний каталоги назовите его equipment. (рис.2 [-@fig:002]) 
2. В домашнем каталоге создаём директорию ~/ski.plases. Перемещаем файл equipment в каталог ~/ski.plases. Переименновываем файл ~/ski.plases/equipment в ~/ski.plases/equiplist. (рис.3 [-@fig:003]) 
3. Создаём в домашнем каталоге файл abc1 и скопируем его в каталог ~/ski.plases,называем его equiplist2 (рис.4 [-@fig:004]) 
4. Создаём каталог с именем equipment в каталоге ~/ski.plases Переместим файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment. (рис.5 [-@fig:005]) 
5. Создаём и переместим каталог ~/newdir в каталог ~/ski.plases и назовите его plans. (рис.6 [-@fig:006]) 
6. Определяем опции команды chmod,необходимые длятого,чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет:

* drwxr–r– … australia (рис.7 [-@fig:007]) 
* drwx–x–x … play (рис.8 [-@fig:008]) 
* -r-xr–r– … my\_os (рис.9 [-@fig:009]) 
* -rw-rw-r– … feathers (рис.10 [-@fig:010]) 

1. Чтобы просмотреть содержимое файла /etc/password используе команду cat (рис.11 [-@fig:011]) 
2. Чтобы скопировать файл ~/feathers в файл ~/file.old используем команду ср (рис.12 [-@fig:012]) 
3. Чтобы переместить файл ~/file.old в каталог ~/play используем команду mv (рис.13 [-@fig:013]) 
4. Чтобы скопировать каталог ~/play в каталог ~/fun mvdir (рис.14 [-@fig:014]) 
5. Чтобы переместить каталог ~/fun в каталог ~/play и назовите его games используем команду mv (рис.15 [-@fig:015]) 
6. Чтобы лишить владельца файла ~/feathers права на чтение используем u-r (рис.16 [-@fig:016]) 
7. Что произойдёт,если вы попытаетесь просмотреть файл ~/feathers командой cat? (рис.17 [-@fig:017])  { #fig:017 width=70% }
8. Чтобы дать владельцу файла ~/feathers право на чтение используем u+r (рис.18 [-@fig:018])  { #fig:018 width=70% }
9. Чтобы лишить владельца каталога ~/play права на выполнение используем u-x (рис.19 [-@fig:019])  { #fig:019 width=70% }
10. Чтобы дать владельцу каталога ~/play право на выполнение используем u+x (рис.20 [-@fig:020])  { #fig:020 width=70% }
11. Прочитайте man по командам mount,fsck,mkfs,kill и кратко их охарактеризуйте, приведя примеры. (рис.21 [-@fig:021]) (рис.22 [-@fig:022]) (рис.23 [-@fig:023]) (рис.24 [-@fig:024])  { #fig:021 width=70% }  { #fig:022 width=70% }  { #fig:023 width=70% }  { #fig:024 width=70% }

# Выводы

Я ознакомилась с файловой системой, её структурой, именами и соделжанием каталогов. Приобрела навыки по выполнению команд.

# Контрольные вопросы

1. Чтобы узнать, какие файловые системы существуют на жёстком диске моего компьютера, использую команду «df -Th». 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 предназначенная 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 Etx-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.

1. 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 испольняемые 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 кеши, 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, находящие 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.

1. Чтобы содержимое некоторой файловой системы было доступно операционной системе необходимо воспользоваться командой mount.
2. 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.
3. Файловую систему можно создать, используя команду mkfs. Ее краткое описание дано в пункте 5 в ходе выполнения заданий лабораторной работы.
4. 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 Cущественно 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

1. Утилита cp позволяет полностью копировать файлы и директории. Cинтаксис: 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
2. 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
3. Права доступа − совокупность правил, регламентирующих порядок и условия доступа субъекта к объектам информационной системы (информации, её носителям, процессам и другим ресурсам) установленных правовыми документами или собственником, владельцем информации. Права доступа к файлу или каталогу можно изменить, воспользовавшись командой chmod. Сделать это может владелец файла (или каталога) или пользователь с правами администратора. Синтаксис команды: chmod режим имя\_файла Режим имеет следующие компоненты структуры и способ записи:

* = установить право
* ‘-’ лишить права
* ’\*’ дать право
* r чтение
* w запись
* x выполнение
* u (user) владелец файла
* g (group) группа, к которой принадлежит владелец файла
* o (others) все остальные