МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ

УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА

Институт радиоэлектроники и информационных технологий

Кафедра «Вычислительные системы и технологии»

ОТЧЁТ

по лабораторной работе №3

«Работа пользователя в LINUX»

по дисциплине  
«Эксплуатация современных операционных систем»

РУКОВОДИТЕЛЬ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Кочешков А.А.

СТУДЕНТ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Сапожников В.О.

19-В-1

Работа защищена «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_

С оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

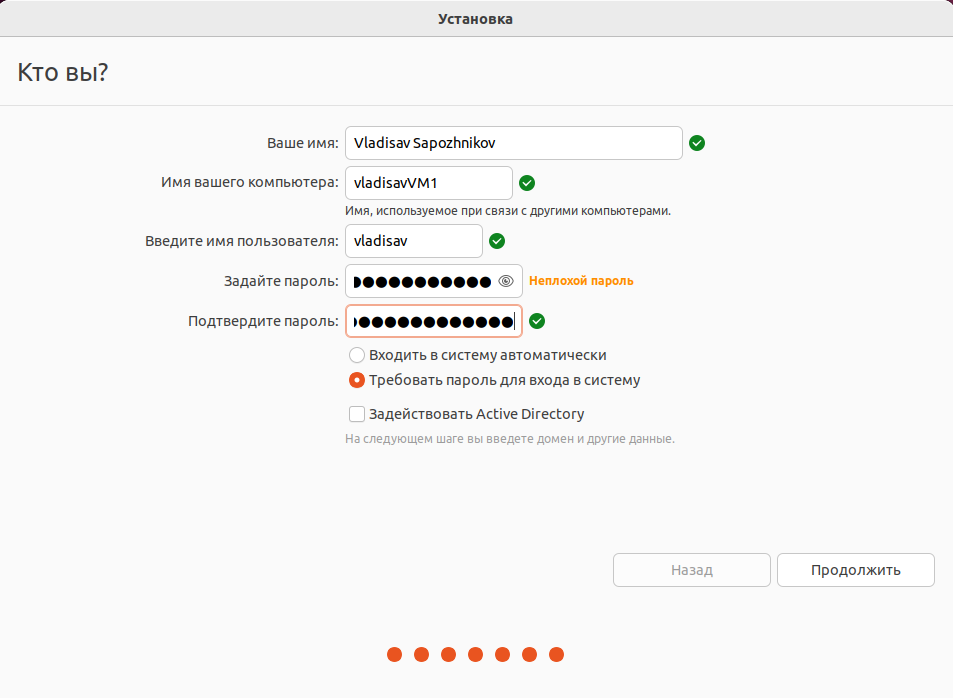
Нижний Новгород 2022г.

Для выполнения работы использовался дистрибутив **Ubuntu 2022.**

**1. Регистрация в системе. Зарегистрироваться в системе. Изучить правила входа в систему и завершения работы (команды login, logout,passwd, exit, shutdown,reboot, halt, Ctrl+Alt+Del, Alt+Fn)**

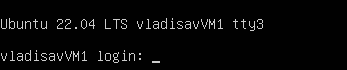
В Linux есть ровно один пользователь, полномочия которого в системе принципиально отличаются от полномочий остальных пользователей — это пользователь с идентификатором "0". Учётная запись пользователя с UID=0 называется root. Root — «администратор» системы Linux, учётная запись для root обязательно присутствует в любой системе Linux, даже если в ней нет никаких других учётных записей. Пользователю с таким UID разрешено выполнять любые действия в системе.

После установки операционной системы зарегистрируемся в системе



Так же при регистрации нам предлагается подключится к домену Active Directory. После подключения к домену будут применены групповые политики, которые определены в домене.

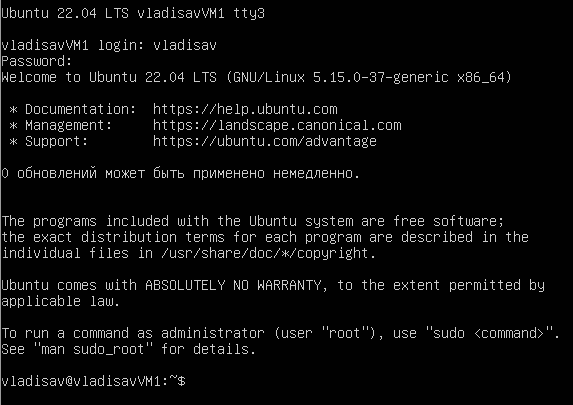
После регистрации в системе требуется перезагрузить компьютер. После перезагрузки переключаемся в текстовый режим комбинацией клавиш "Ctrl+Alt+F3"



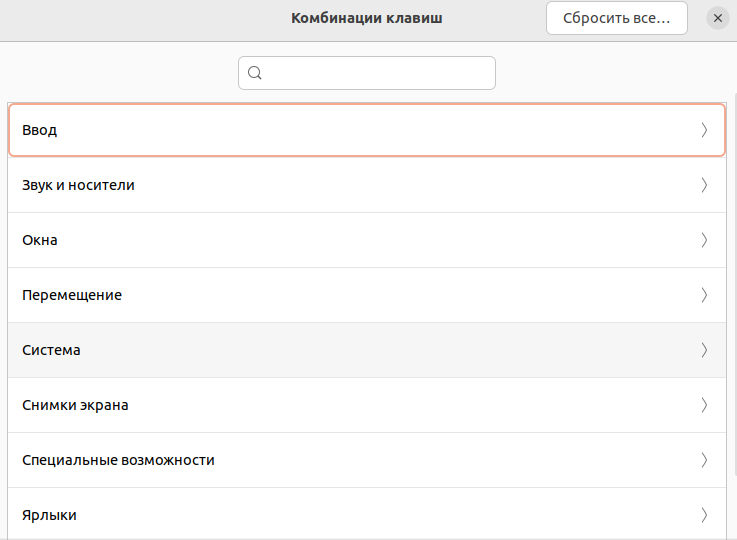
Системы просит пользователя ввести данные одной из учетных записей. Работа незарегистрированного пользователя невозможна в ОС Linux.

В текстовом режиме недоступны возможности графических интерфейсов. В различных реализациях Linux работа в графическом режиме может выглядеть по-разному. Текстовый режим доступен в любой реализации Linux и всегда выглядит практически одинаково.

Введем данные ранее зарегистрированного пользователя:



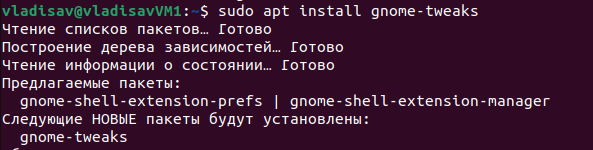
В данном дистрибутиве Linux в графическом интерфейсе существует настройка комбинаций клавиш. Найти ее можно по следующему пути: Настройки → Клавиатура → Комбинации клавиш.



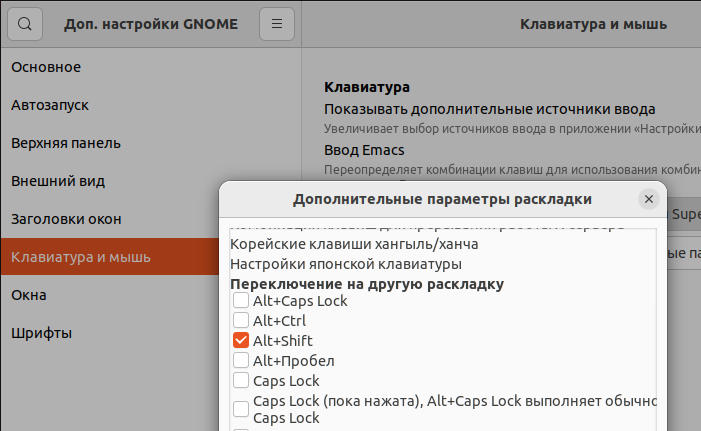
Здесь можно настроить много параметров, которые содержатся в следующих группах:

* Ввод
* Запуск приложений
* Звук и носители
* Окна
* Перемещение
* Система
* Снимки экрана
* Специальные возможности

Переключение раскладки клавиатуры в Ubuntu по умолчанию настроено на комбинации клавиш Super + пробел, что является не удобным для пользователей Windows. Вернуть привычную всем “виндузятникам” комбинацию клавиш Alt + Shift в простом меню комбинации клавиш нельзя. Для этого необходимо поставить специальный пакет: gnome – tweaks



При установке данного пакета нам станут доступны дополнительные настройки оболочки Gnome. В меню Клавиатура → Дополнительные параметры клавиатуры → Переключение на другую раскладку мы можем выбрать любое всеми сочетание клавиш Alt+Shift (или другие).



Для входа в систему и завершения работы используются следующие команды:

* login – войти в систему. login — это механизм идентификации и проверки идентичности. Он обеспечивает средства идентификации пользователей в системе и проверки идентичности пользователя. При вызове login появляется приглашение. В ответ на приглашение необходимо ввести имя пользователя, для которого есть соответствующая учетная запись в системе.
* logout – завершить работу с командной оболочкой.





В ответ на эту команду возобновляется приглашение к регистрации в системе. На этой консоли работа завершена, и теперь здесь снова может зарегистрироваться любой пользователь. Эта команда схожа со сменой пользователя в Windows.

* passwd – изменить пароль пользователя

При смене пароля может возникнуть проблема, когда не удается сменить пароль, т.к. новый пароль оказался слишком простым (не соответствует политике безопасности системы). В таком случае необходимо ввести более сложный пароль. При корректном вводе старого и нового пароля запрашивается ввод нового пароля еще раз, чтобы убедиться, что в первый раз не было опечатки. Если все в порядке, то выводится сообщение о том, что смена пароля прошла успешно, а затем произойдет завершение работы, вернув приглашение.

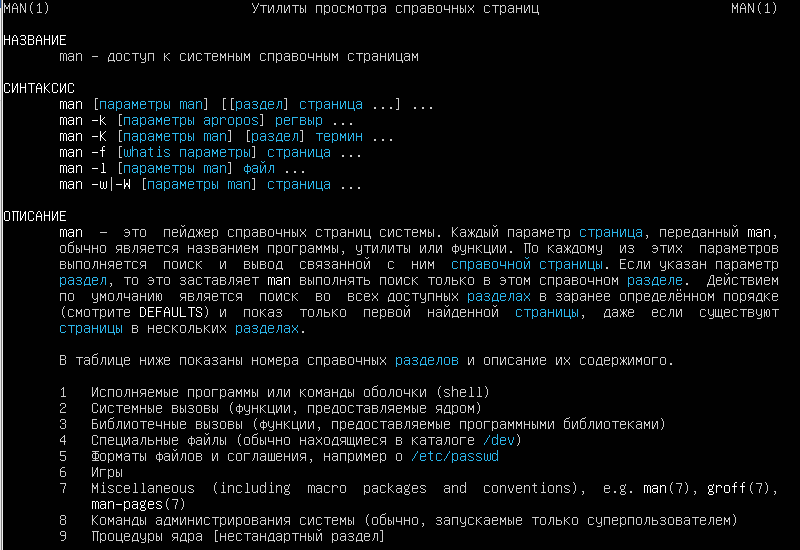
* exit – завершить сеанс работы с оболочкой
* shutdown – выключить компьютер или спланировать выключение. Если просто выключить питание, изменения не будут сохранены, что в некоторых случаях может привести к невозможной загрузке системы. Для правильного завершения работы системы перед выключением компьютера можно воспользоваться командой shutdown. Эта команда может быть выполнена только пользователем root
* reboot – перезагрузка компьютера и разрешает/запрещает использование клавиш Ctrl-Alt-Del для перезагрузки
* halt – немедленно выключить компьютер. Данная команда используется, когда требуется завершить работу и выключить компьютер.
* Ctrl+Alt+Del – выход из системы (вызов команды logout)
* Alt+Ctrl – вызов нового окна терминала
* Alt+Tab – переключение между окнами
* Alt+Fn (где n – это номер консоли) – комбинации клавиш для работы в консоли:
  + Alt+F1 – выбор активного окна
  + Alt+F2 – аналог «Выполнить» в Windows
  + Alt+F4 – закрывает окно терминала
  + Alt+F7 – захват окна для изменения его расположения
  + Alt+F10 – свернуть окно

Вывод: в ходе выполнения данного пункта была проведена регистрация системе, изучены правила входа в систему и завершения работы, также изучены некоторые команды, отвечающие за вход и завершение работы.

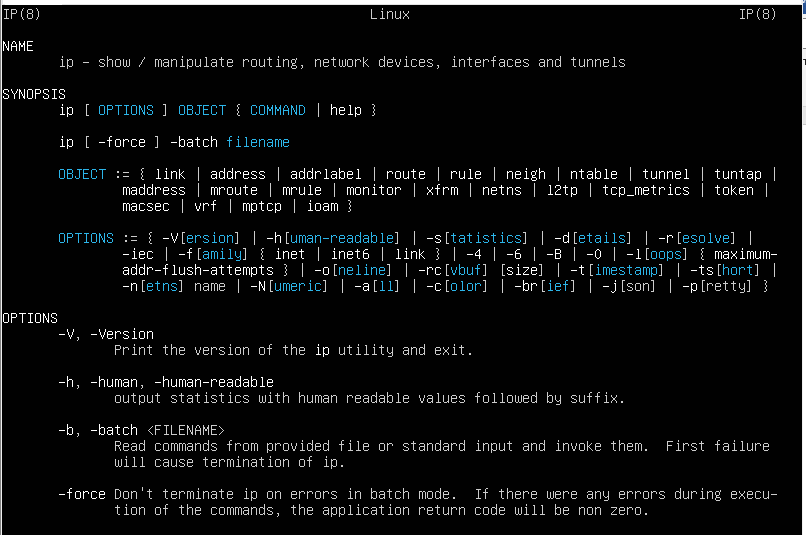
**2. Основные информационные команды. Научиться пользоваться диалоговым руководством man, apropos, ознакомиться со справочной системой в формате файлов HOW-TO.**

**man** – руководство пользователя Manual – позволяет получить краткую справку об установленном ПО и системе, а также по некоторым командам терминала. Справочная информация структурирована по разделам: команды пользовательского уровня, системные вызовы, библиотечные функции, драйвера устройств, стандарты форматов файлов, игры и демо-программы, файлы и документы, команды системного администрирования, внутренние интерфейсы и спецификации. Чтобы запустить руководство по какой-либо команде надо набрать команду "man" и через пробел указать название команды, по которой нужно руководство. После этого система запустить руководство по команде, если найдет.

Информация по самой утилите man

****

Информация об IP

****

Информация об tcp

****

Структура руководства:

В начале страницы можно увидеть название, краткое описание команды и синтаксиса и описание для чего предназначена программа. Есть определённые правила оформления страницы руководства. Есть руководства по командам, программам, функциям и т.д. Не во всех руководствах есть эти заголовки, так как некоторые из них применимы только к конкретным командам.

Список заголовков:

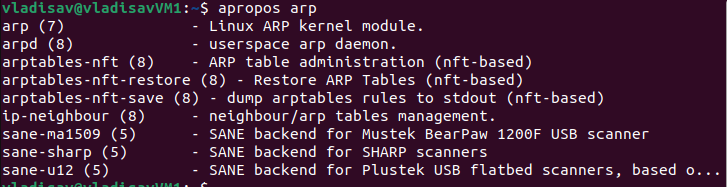
* Название (Name)- название команды, по которой просматривается руководство
* Синопсис (Synopsis)- краткое описание команды и синтаксиса
* Конфигурация (Configuration)- детали настройки для устройства
* Описание (Description)- описание основного назначения программы
* Опции (Options)- опции которые принимает команда
* Выходной статус (Exit Status)- возможные значения, возвращаемые командой при завершении работы
* Возвращаемое значение (Return Value)- если руководство запущено по какой-то библиотеке, то это указывает на значение, которое вернет библиотека функции, вызвавшая ее
* Ошибки (Errors)- список всех значение, которые может принимать errno в случае ошибки выполнения команды
* Окружение (Environment)- список переменных окружения, которые относятся к команде или программе
* Файлы (Files)- список файлов, которые использует команда или программа
* Атрибуты (Attributes)- список различных атрибутов команды
* Версии (Versions)- список изменений в ядре Linux или библиотеке, которую использует команда
* Соответствие (Conforming to)- описание любых стандартов, которым может соответствовать команда
* Заметки (Notes)- дополнительные заметки
* Ошибки (Bugs)- известные ошибки
* Примеры (Examples)- один или несколько примеров использования команды
* Авторы (Authors)- авторы, которые разработали и поддерживают команду
* Просмотрите также (See Also)- рекомендуемые материалы по команде

Цифры в скобках указывают на раздел руководства. Различают 9 разделов:

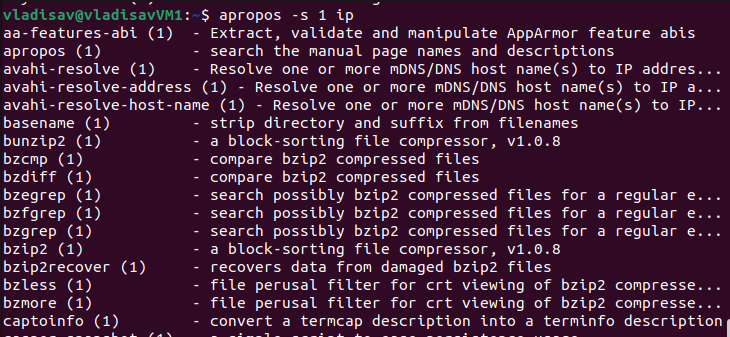
1. Основные команды (General commands)- команды, которые используются в командной строке
2. Системные вызовы (System calls)- функции ядра, которые может вызвать программа
3. Функции библиотек (Library functions)- общий набор функций и возможностей, используемых программами
4. Форматы файлов и соглашения (File formats and conventions)- форматы файлов как passwd, cron table, tar архивы
5. Специальные файлы (Special files)- обычно устройства, например, найденные в /dev, и их драйверы
6. Игры (Games)- описание команд, например, fortuna, которая при запуске показывает цитаты из БД
7. Дополнительно (Miscellaneous)- описание таких вещей как inodes, параметры загрузку
8. Администрирование системы (System administration)- команды, зарезервированные для использования root-ом.
9. Распорядок ядра (Kernel Routines)- информация, касающаяся внутренних операций ядра. Сюда входят функциональные интерфейсы и переменные, которые могут быть использованы программистами, которые разрабатывает драйвера устройств.

apropos- поиск в именах справочных страниц и кратких описаниях. apropos выполняет поиск ключевого слова в этом описании. Ключевое слово может представлять собой регулярное выражение (-r), шаблон (-w) или точное искомое ключевое слово (-e). При использовании этих параметров может потребоваться указывать ключевое слово в кавычках или экранировать (\) специальные символы для того, чтобы избежать их интерпретации оболочкой командной строки. Стандартные правила поиска позволяют искать по названию страницы и границам слова в описании. Команда apropos является эквивалентом команды man -k. Данная команда полезна, когда пользователь не знает или не помнит точного названия инструмента (или команды).

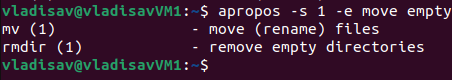
Простой поиск, используя команду "apropos arp". Например, когда хотим найти команды, которые включают в свое описание или название слово arp.



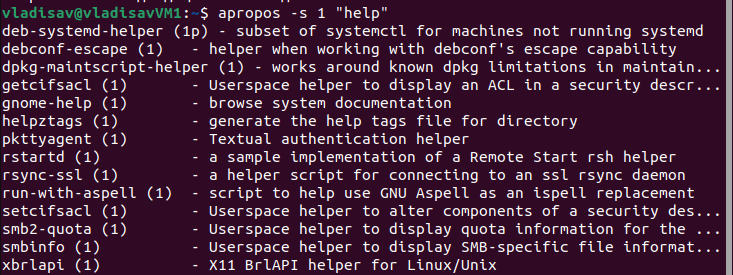
Используя флаг -s и номер мы можем выполнить вывод информации только из man-страниц определенного раздела



По ключевому слову "move" находятся и строки со словами "remove". Чтобы этого избежать, необходимо использовать ключ "-е".



Поиск по фразе, используя команду "apropos -s 1 *"<фраза> "*



Файлы справки HOW-TO имеют вид текстовых файлов с описанием алгоритма для выполнения определенной задачи. Могут храниться как централизовано, так и в папке с программой, по которой требуется обучающая информация. Но такой способ справки уже устарел, т.к. многие программы имеют встроенную систему документации, не доступную, ни через man, ни через info. Такие файлы можно найти на сайте " https://www.opennet.ru/docs/5.shtml". Здесь содержатся подробные инструкции по установке и использовании файлов. Например, "Linux Security HOWTO"



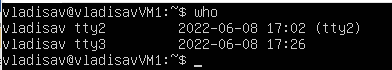
Файлы HOW-TO рассчитаны на предоставление информации по разрешению возникших проблем или настройке тех или иных компонент. Такие файлы содержат аннотацию, структуру, описание и алгоритм действий.

**Вывод**: в ходе выполнения данного пункта были получены навыки пользования рукводоством man и apropos, также познакомились со справочной системой в формате файлов HOW-TO.

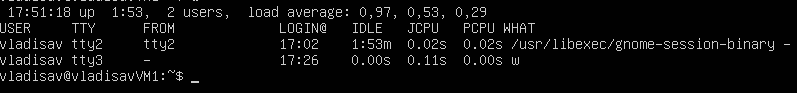
**3. Управление учетными записями пользователей. Получить информацию о пользователях с помощью команд who, w, id, logname, finger. Создать новую учетную запись командой adduser. Ознакомиться с форматом и использованием файла паролей /etc/passwd, /etc/shadow. Привести примеры назначения учетных записей псевдопользователей. Определить состав и свойства домашнего каталога пользователя, профиля пользователя, профиля по умолчанию /etc/skel.**

Получим информацию о пользователях с помощью следующих команд:

* **who** - получение информации об активных пользователях, которые подключены к системе, в том числе и о терминальных сессиях, через которые происходит подключение

****

* **w** – получить более детальную информацию о пользователях и используемых ими терминалах. Также выводит на экран текущий процесс, запущенный в терминале, время работы пользователя в терминале

****

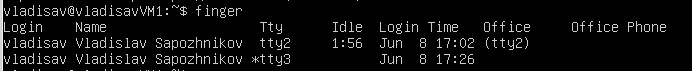
* **id** – вывести информацию о пользователе: идентификаторы пользователя, его группы, некоторые права и настройки пользователя

****

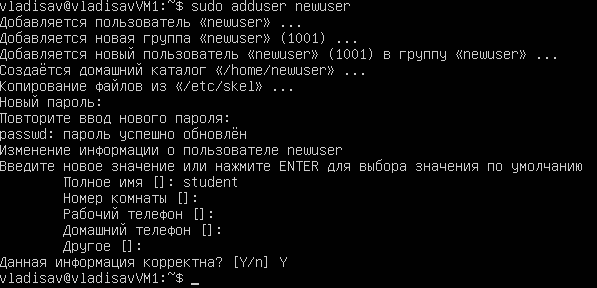
* **logname** – выводит имя пользователя, который зарегистрирован в системе на данный момент

****

* **finger** – вывести информацию о пользователе, вошедшим в систему

****

Cоздадим нового пользователя с помощью команды "adduser"

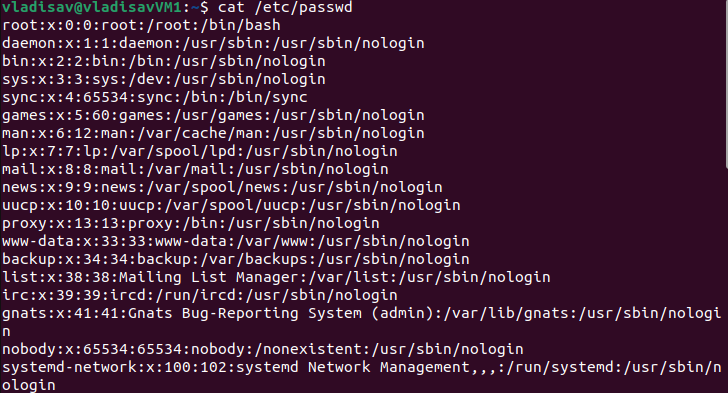
****

После запуска данной команды необходимо ввести пароль для нового пользователя. Затем будет предложено ввести дополнительную информацию о пользователе: имя, номер комнаты, телефоны и комментарий. Вводить эту информацию необязательно, можно пропустить нажатием клавиши "Enter". В результате выполнения этой команды будут выполнены следующие действия:

* Создается новый пользователь с указанным именем
* Создается группа с тем же именем
* Создается домашний каталог пользователя в директории /home/имяпользователя
* В домашний каталог копируются файлы из директории /etc/skel. В данной директории хранятся файлы, которые копируются в домашний каталог всех новых пользователей.

/etc/passwd- файл, содержащий в текстовом формате список всех пользовательских учётных записей, имеющихся в системе. Является первым и основным источником информации о правах пользователя операционной системы.

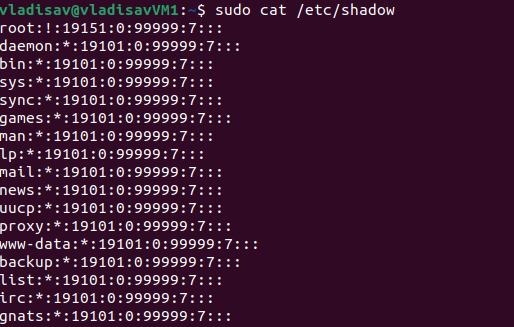
Посмотрим содержание файла при помощи команды cat.

****

Каждая строка файла описывает одного пользователя и содержит семь полей, разделённых двоеточиями:

* Имя пользователя (логин)
* Хеш пароля
* Идентификатор пользователя (uid)
* Идентификатор группы (gid)
* Информационное поле GECOS (полное имя пользователя, номера телефонов и т.д.)
* Начальный (домашний) каталог
* Регистрационная оболочка (shell)

/etc/shadow — это текстовый файл, содержащий информацию о паролях пользователей системы.

****

Как и в файле passwd, каждое поле в файле shadow отделяется двоеточием:

* Имя пользователя. Совпадает с именем пользователя в файле /etc/passwd
* Пароль (зашифрованный). Пустая запись означает, что для входа пароль не нужен, запись "\*'' - вход заблокирован
* Количество дней (с 1 января 1970), когда пароль был сменен в последний раз.
* Число дней до смены пароля (0 показывает, что он может быть сменен всегда)
* Число дней, после которых пароль должен быть сменен (99999 показывает, что пользователь может не менять пароль фактически никогда)
* Число дней, в течение которых пользователь получает предупреждения о необходимости пароль сменить (7 для полной недели)
* Число дней после окончания действия пароля, когда еще можно работать. Если пароль не сменить, после данного срока аккаунт будет заблокирован
* Число дней, начиная с 1 января 1970, после которых пароль будет заблокирован.
* Зарезервировано для возможного будущего использования

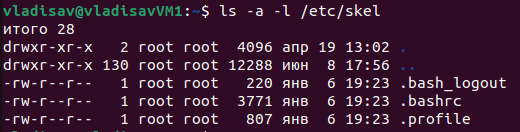
**Псевдопользователи:**

Список выше показывает намного больше пользователей, чем ожидается, потому что в нем перечислены все пользователи системы. Если мы хотим отличить обычных пользователей от системных пользователей, можно обратиться к номеру идентификатора пользователя (UID). Обычный пользователь имеет UID, больший или равный 1000. У специальных пользователей UID от 1 до 500. Для пользователя root зарезервирован идентификатор под номером 0. Рекомендуется присваивать реальным пользователям идентификаторы, следующие после 500, поскольку в системе может быть много «неперсонифицированных» пользователей.

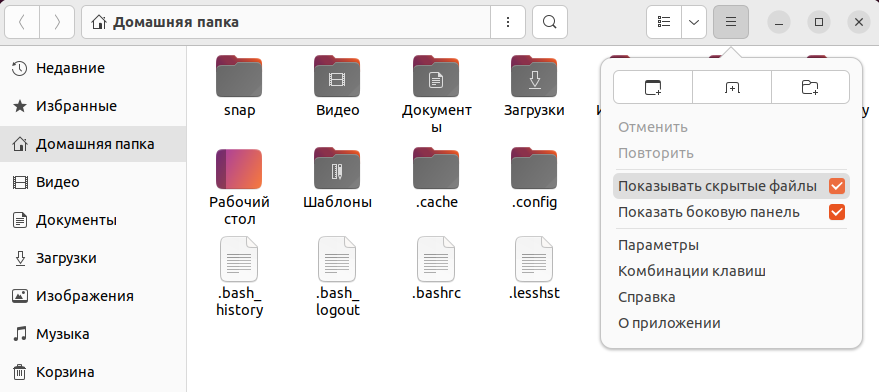
Некоторые разновидности псевдопользователей:

* nobody- предназначен для запуска программ, служб и т.п., которым не требуются какие-либо специальные разрешения
* daemon- используется серверными процессами системы
* sys- владеет системными файлами
* adm- владеет файлами бюджета
* bin- владеет исполняемыми файлами пользователя
* uucp- используется UUCP
* lp- используется подсистемами 1р или lpd
* Usenet, audit, cron, mail, new- обслуживают связанные с ними процессы и файлы

/etс/skel - файл, который содержит в себе файлы и настройки поведения командной оболочки Bash. Посмотреть содержимое домашнего каталога с скрытыми файлами можно с помощью команды "ls -a -l /etс/skel"

****

Или через графический интерфейс, включив показ скрытых файлов.

****

По умолчанию включает в себя 3 файла конфигурации:

* .bash-logout
* .bashrc — настройка поведения консоли
* profile — размещение пользовательских исполняемых файлов по умолчанию

Все файлы этой директории копируются в домашнюю папку нового пользователя (/home/%username%) при создании командой adduser или useradd. Точка перед именем файла – признак скрытого файла. Домашний каталог служит в качестве хранилища для личных файлов, каталогов и программ пользователя, включая личные файлы конфигурации. /etc/skel позволяет создать домашний каталог по умолчанию для всех новых пользователей на компьютере или в сети и, таким образом, убедиться, что все пользователи начинают с одних и тех же настроек или среды.

**Вывод**: в ходе выполнения данного пункта были изучены команды, связанные с учетными записями пользователей. Также было произведено создание учетной записи с помощью команды "adduser". Было разобрано содержание файлов etc/passwd и /etc/ shadow.

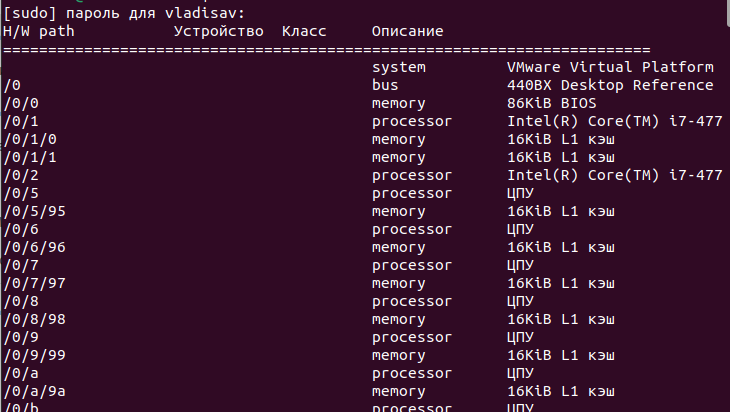
**4. Работа с устройствами и файловой системой. Изучить принцип именования устройств, назначение и применение специальных файлов устройств в классическом формате Unix и в современном формате udev. Найти в каталоге /dev специальные файлы основных устройств компьютера и составить отчет об аппаратной конфигурации. Сравнить распределение аппаратных ресурсов (I/O Ports, IRQ, Mem) в linux с аналогичными данными в Windows (из 2 работы). Изучить свойства и параметры операции монтирования файловых систем.**

Различают несколько видов файлов: обычные файлы, каталоги и специальные файлы. К обычным файлам относят файлы с важными данными. Каталоги — это файлы, в качестве данных которых выступают списки других файлов и каталогов. В данных каталога осуществляется связь имени файла с его индексным дескриптором. Поэтому один и тот же файл может существовать под разными именами и/или в разных каталогах: все имена будут связаны с одним и тем же индексным дескриптором (жесткие ссылки). Файлы всегда содержатся в каталогах. Специальные файлы — это механизм использования ввода/вывода. Большинство специальных файлов находятся в каталоге /dev. Они используются для обозначения физических устройств компьютера. Размера у специального файла не существует. Так как этот файл обозначает физическое устройство. Другими словами, это ссылка на соответствующий драйвер.

Специальные файлы делятся на два типа:

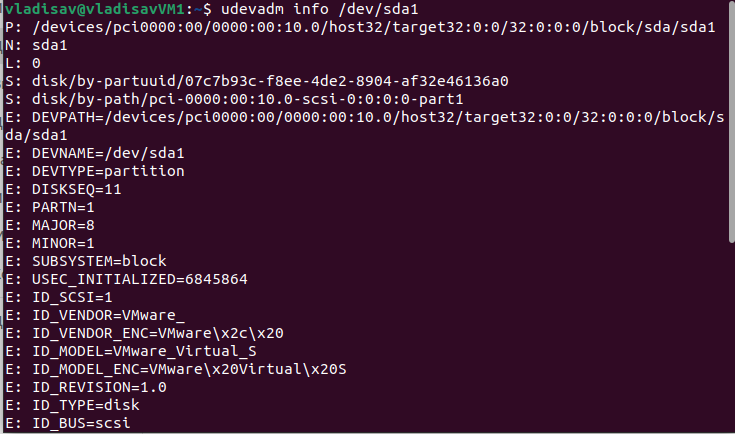
* Блочные - запись и чтение допускаются только блоками
* Символьные - можно читать и писать отдельными байтами.

С помощью команды "sudo lshw -short" получим список классов устройств используемого компьютера



udev – это механизм, используемый для создания и именования узлов устройств в каталоге /dev, соответствующих устройствам, присутствующим в системе.

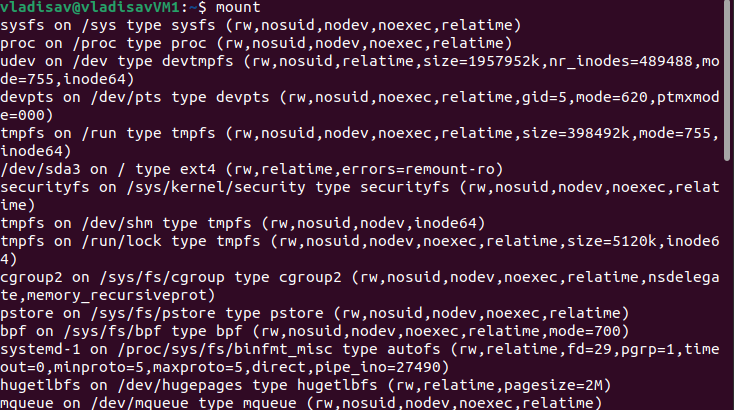
При помощи команды "udevadm info /dev/sda1" узнаем атрибуты устройства /dev/sda1 из БД udev:



**Ознакомиться с основными командами работы с дисками: mount, umount, df, mkfs, fsck, fdisk:**

**mount**- монтирует файловую систему. Все файлы, доступные в Unix системах, составляют иерархическую файловую структуру, которая подобна растущему дереву имеет ветки (каталоги) и листья (файлы в каталогах). Корень этого большого дерева обозначается как /. Физически файлы могут располагаться на различных устройствах. Команда mount служит для подключения файловых систем разных устройств к этому большому дереву.

Если ввести команду "mount", то она выдает большой список всех смонтированных файловых систем в текущий момент и параметров монтирования

****

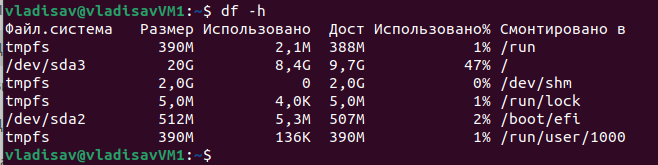
Рассмотрим некоторые из них:

* sysfs - с помощью этой файловой системы вы можете задавать различные настройки ядра во время выполнения
* tmpfs позволяет размещать любые пользовательские файлы в оперативной памяти компьютера. Достаточно создать блочное устройство нужного размера, затем подключить его к папке, и вы можете писать файлы в оперативную память
* procfs - по умолчанию смонтирована в папку proc и содержит всю информацию о запущенных в системе процессах, а также самом ядре
* securityfs- эта файловая система предназначена для использования модулями безопасности
* ext2, ext4 — это стандартные файловые системы Linux

**umount**- применяется для размонтирования файловых систем, предварительно смонтированных посредством вызова утилиты mount.

**df**- выводит имена файлов устройств разделов, полный объем их пространства, объем занятого и свободного пространства, процентный показатель использования разделов, а также точки их монтирования

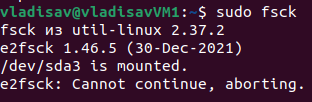
Выведем все дисковое пространство файловой системы в удобном виде с помощью команды "df -h"



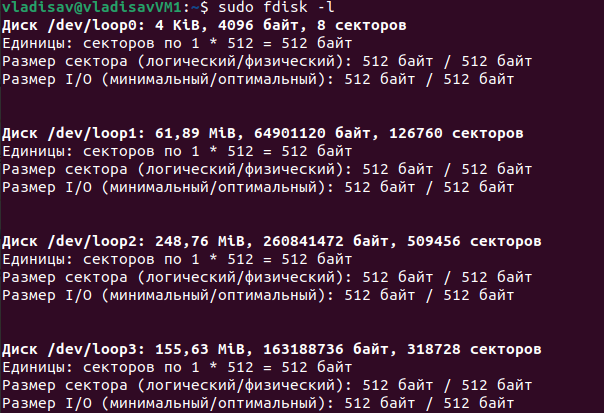
Здесь записи представлены в виде таблицы, каждый столбец имеет свое название: тип файловой системы, ее размер, сколько использовано и доступно, также сколько использовано в процентном соотношении и где смонтировано.

**mkfs**- позволяет создавать и форматировать файловые системы

**fsck** - используется для проверки и исправления ошибок одной или нескольких файловых систем. Также fsck параллельно проверяет данные на разных физических дисках, чтобы сократить общее время, необходимое для полной проверки всех дисков. Если файловых ввести команду fsck без параметров, то она по порядку проверит файловые системы, указанные в /etc/fstab.



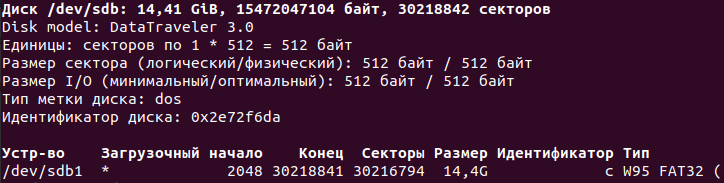
**fdisk** - используется для управления разделами жёсткого диска. При помощи флага -l просмотрим детальную информация о разделах жесткого диска



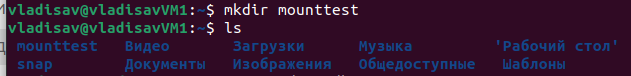
Здесь можно посмотреть более детальную информацию об устройстве: имя файла, список разделов, формат таблицы разделов, список разделов, размер раздела и его файловую систему

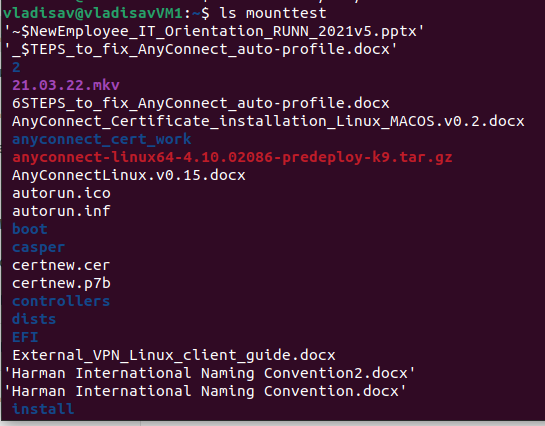
**Выполнить монтирование файловой системы на сменном носителе. Рассмотреть описание автоматически монтируемых файловых систем в файле /etc/fstab. Объяснить использованный в fstab формат ссылки на устройство корневой файловой системы.**

Подключим к виртуальной машине флешку для её монтирования



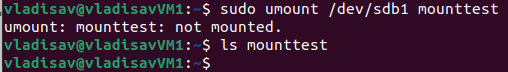
В корневом каталоге создадим папку, смонтируем в нее файловую системы с флешки при помощи mount и посмотрим её содержимое:





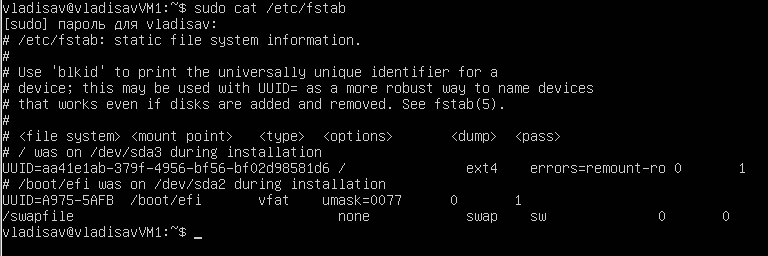
Видим, что в данном каталоге отображается содержимое флешки

Теперь выполним размонтирование флэшки и также посмотрим содержимое папки после размонтирования



**fstab** — это файл, содержащий список устройств хранения информации и параметры монтирования. Различные накопители, которые необходимо автоматически присоединить во время загрузки операционной системы, по порядку перечисляются в файле fstab.

Рассмотрим содержимое файла /etc/fstab



В каждой строчке содержится следующая информация:

* Устройство — это раздел диска, который нужно примонтировать
* Файловая система- указывает в какой файловой системе нужно монтировать это устройство
* Точка монтирования - куда нужно примонтировать это устройство
* Опции - параметры монтирования файловой системы
* Проверка - указывает в какой очереди нужно проверять устройство на ошибки: 1 - в первую очередь, 2 - во вторую, 0 - не проверять
* Резерв - указывает нужно ли делать резервную копию раздела, может принимать значения только 0-нет и 1-да

**Вывод**: в ходе выполнения данного пункта была проведена работа с устройствами и файловой системой с помощью различных команд, рассмотрены описание автоматически монтируемых файловых систем в файле fstab.

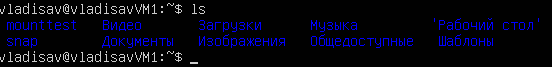
**5. Управление файлами.**

**Изучить и опробовать: получение информации о файлах и каталогах: форматы команды ls, команды файловых операций: cd,pwd,cp,mv,rm,mkdir,rmdir, операции над группами файлов, создание прямой и символической ссылки ln, команды вывода cat,tee,more,file, поиска find, фильтра grep, перенаправление потоков ввода-вывода (, >>) и каналы (|)**

**Получение информации о файлах и каталогах**

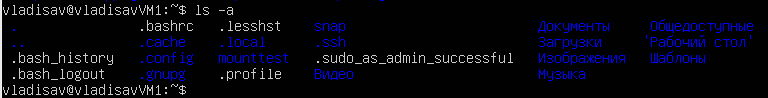
**ls**- используется для вывода содержимого каталогов и информации о файлах. В качестве аргументов ls принимает адреса каталогов или файлов. Синтаксис команды: ls опции/ путь к папке

Если ввести команду "ls" без аргументов, то выведется содержимое каталога

****

Основные параметры команды ls:

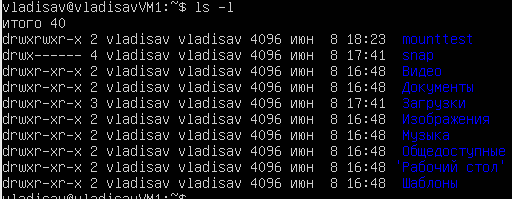
* -a - отображать все файлы, включая скрытые

****

* -g - показывать подробную информацию

****

* -l - выводить подробный список, в котором будет отображаться владелец, группа, дата создания, размер и другие параметры

****

**Команды файловых операций**

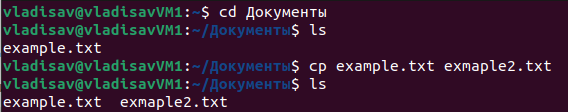
**cd** - изменить рабочую директорию. Эта команда позволяет изменить текущую папку на другую. По умолчанию, текущей считается домашняя папка

****

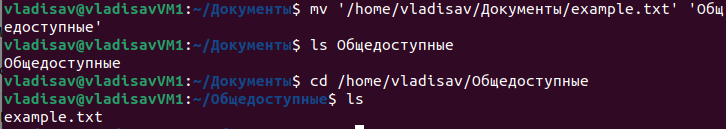
**pwd**- показать полный путь к текущему файлу



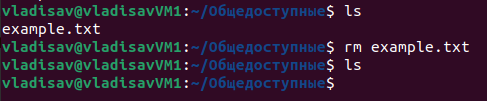
**cp**- копирование файлов в каталоги



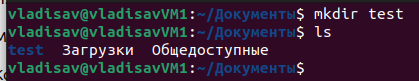
**mv**- перемещение или переименование файлов и каталогов. Переместим файл "example.txt" в каталог "Общедоступные"



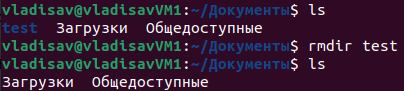
**rm**- удаление файлов. Удалим из "Общедоступные" файл "Smena.txt"



**mkdir**- создание каталога



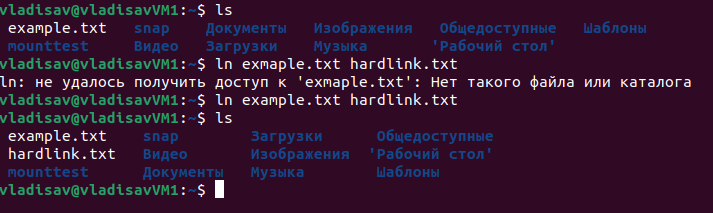
**rmdir**- удаление каталога

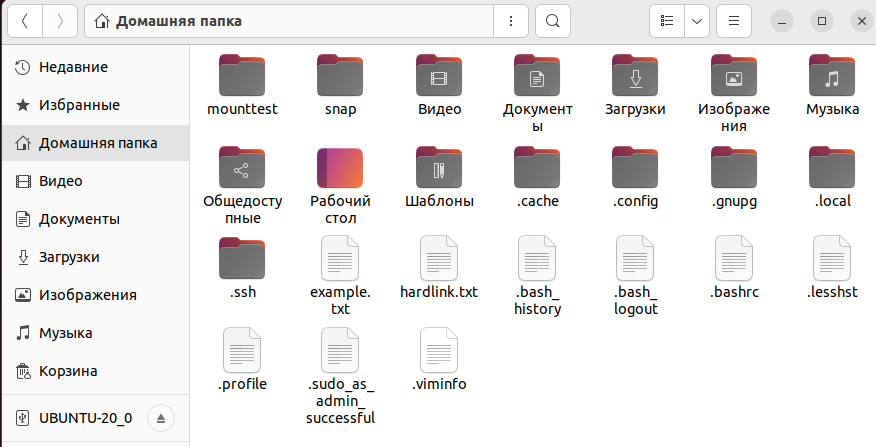


Создание прямой и символической ссылки ln:

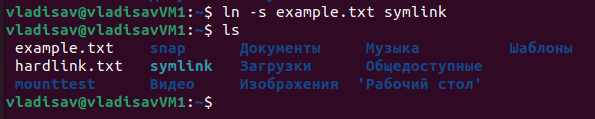
**ln** - команда, устанавливающая связь между файлом и именем файла

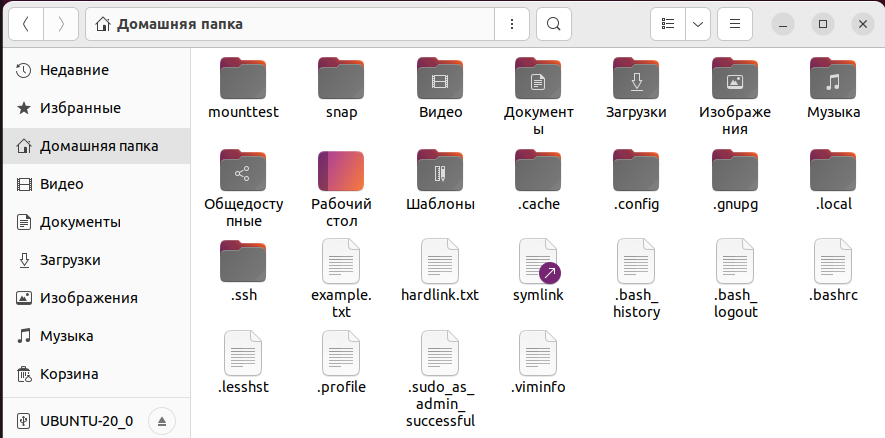
* ln файл имя ссылки- создаётся (hard link) жёсткая ссылка





* ln -s файл имя\_ссылки- создаётся (symbolic link) символьная ссылка





Команды вывода:

**cat**- считывает данные из файлов и выводит их содержимое



**tee**- считывает стандартный ввод, затем записывает его содержимое в стандартный вывод.

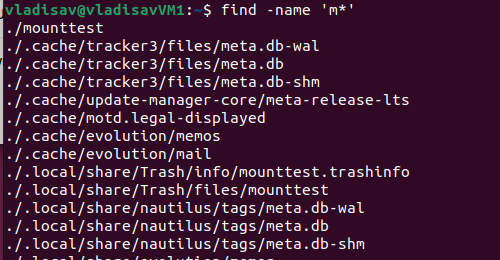


**more**- постраничный просмотра файлов в терминале

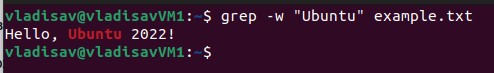
**file**- отображает тип файла



**find**- поиск файлов по имени и другим свойствам. Может производить поиск в одном или нескольких каталогах с использованием критериев, заданных пользователем.

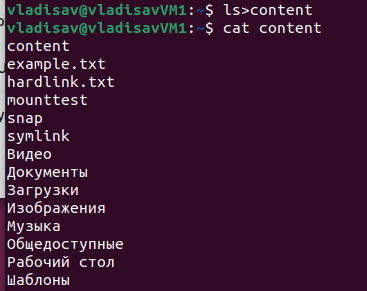


**grep**- выполняет поиск образца в текстовых файлах и выдает все строки, содержащие этот образец.

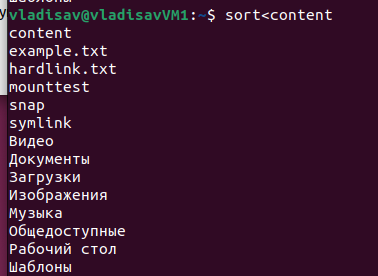


Перенаправление потоков ввода-вывода (, >>) и каналы (|):

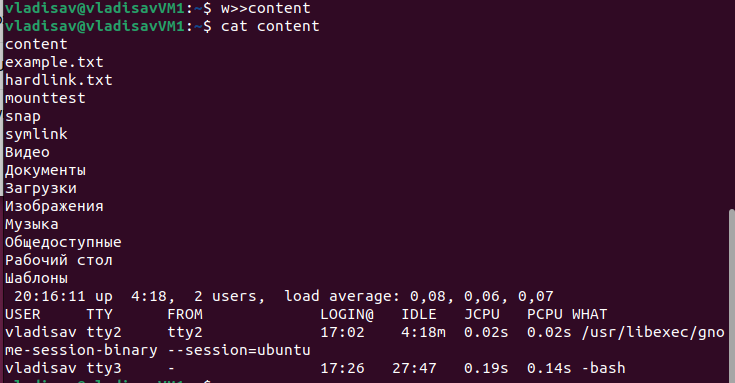
* >file – перенаправляет стандартный вывод в file. Если в файле есть информация, она будет перезаписана.



* <file – использовать файл как источник данных для стандартного потока ввода.

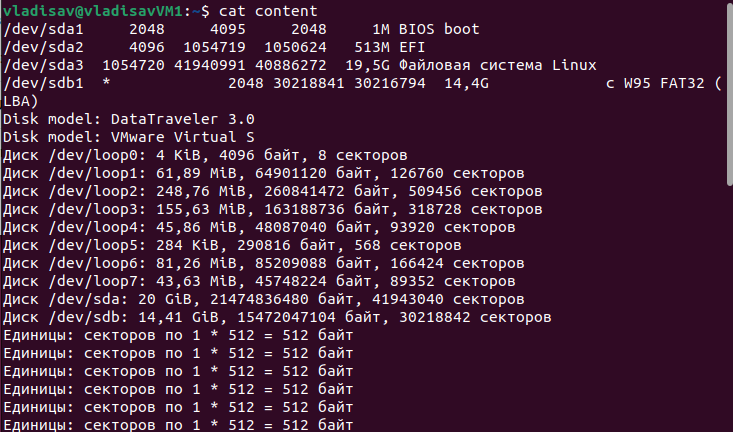


* >>file- добавляет вывод в конец file. В отличие от ">file" не перезаписывает файл, а добавляет в конец.



* |- перенаправляет вывод одной команды в качестве ввода другой в определенное место, например в файл. "sudo fdisk -l | sort > Proverka "- вывод в файл информации о разделах жесткого диска с последующей сортировкой





**Вывод**: в ходе выполнения данного пункта были изучены команды для работы с файлами и каталогами, создание прямых и символических ссылок.

**6. Управление доступом к файловой системе.**

**Состав индексного дескриптора файла. Определение владельца-пользователя и владельца группы, команды chown, chgrp. Смысл режимов доступа r w x для файла и каталога. Дополнительные флаги SUID, SGID, StikyBit для файла и каталога. Выполнить управление правами доступа: chmod. Опробовать различные варианты и составить таблицу, сопоставляющую различные права доступа на каталог и файлы в нем и доступные или недоступные действия пользователя над файлами (в том числе варианты -w-, --x, показать действие права x на каталог). Найти, как назначаются права на symlink, сравнить с NTFS.**

Состав индексного дескриптора файла:

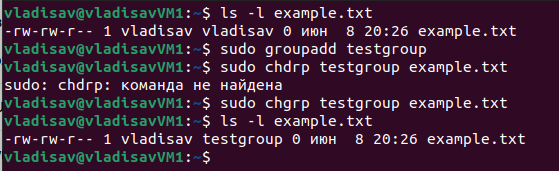
* di\_gid- идентификатор группы владельцев (GID)
* di\_mtime- время последней модификации файла
* di\_ctime- время создания файла
* di\_atime- время последнего обращения к файлу
* di\_size- размер файла в байтах
* di\_uid- идентификатор владельца файла (UID)
* di\_mode- тип и права доступа к данному файлу

Как и в любой ОС некоторые пользователи или группы пользователей имеют свои разрешения на файлы и каталоги. Например, обычным пользователям закрыт доступ к некоторым важным системы файлам и каталогам. С помощью команд "chown" и "chgrp" можно изменять владельца и владельца -групп.

**chown**- команда, позволяющая изменить владельца-пользователя.

****

**chgrp** - команда, позволяющая изменить владельца-группу для файла.

****

Смысл режимов доступа r, w, x для файла и каталога заключается в безопасности. Как говорилось выше, для каждого пользователя, группы и остальным владелец файла или каталога хочет ограничить доступ к ним, выставляя соответствующие параметры. Различают 3 параметра доступа к файлам и каталогам:

* r (read)- доступ на чтение
* w (write)- доступ на запись
* x (execute)- доступ на исполнение

Дополнительные флаги SUID, SGID, StikyBit для файла и каталога существуют с целью упрощения выполнения различных программ обычным пользователям от имени суперпользователя.

* SUID- если этот флаг установлен, то при выполнении программы, id пользователя, от которого она запущена, заменяется на id владельца файла. Это позволяет обычным пользователям запускать программы от имени суперпользователя
* SGID- этот флаг работает аналогично, но различие заключается в том, что пользователь считается членом группы, с которой связан файл, а не групп, к которым он действительно принадлежит. Если SGID флаг установлен на каталог, все файлы, созданные в нем, будут связаны с группой каталога, а не пользователя
* StikyBit- если этот флаг установлен, то пользователи могут только создавать, читать и выполнять файлы, но не могут удалять файлы, принадлежащие другим пользователям

**chmod**- предназначена для изменения прав доступа к файлам и каталогам

Различают следующие виды прав:

* r – чтение
* w – запись
* x – выполнение
* s - выполнение от имени суперпользователя

Также для следующих категорий пользователей можно установить эти права:

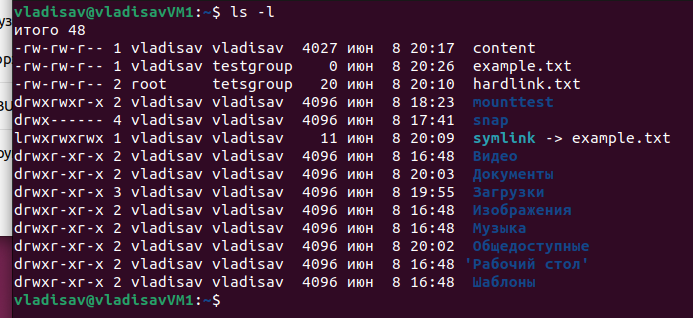
* u - владелец файла
* g - группа файла
* - все остальные пользователи

Все эти права можно комбинировать, указав знак "+".

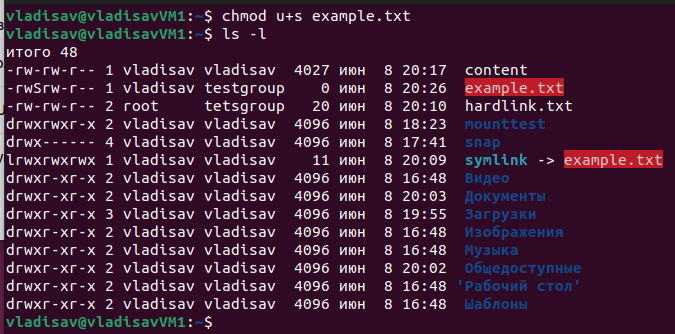
Можно не только словами обозначать права, но и цифрами:

* 0 - никаких прав
* 1 - только выполнение
* 2 - только запись
* 3 - выполнение и запись
* 4 - только чтение
* 5 - чтение и выполнение
* 6 - чтение и запись
* 7 - чтение запись и выполнение

Попробуем переназначить права для файлов. Но сначала посмотрим список каталогов и файлов с исходными разрешениями с помощью команды "ls -l"

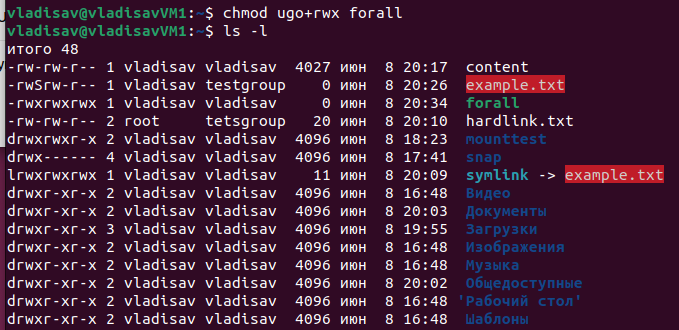
****

Теперь назначим для файла "example.txt" разрешение только для владельца файла и выполнение от имени суперпользователя

****

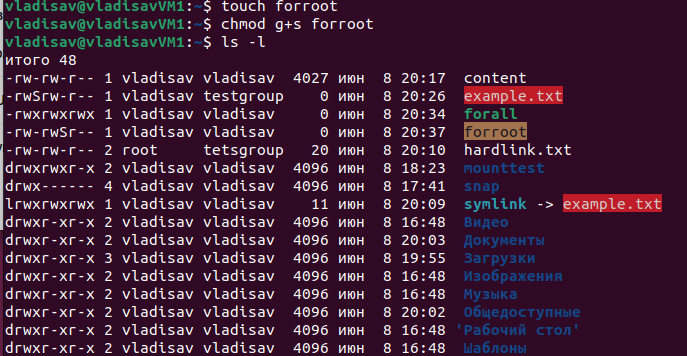
Теперь этот файл может выполняться только имени суперпользователя и является собственностью владельца. Также видим, что файл теперь выделяется красным цветом.

Назначим для файла "forall" разрешение для всех и все действия

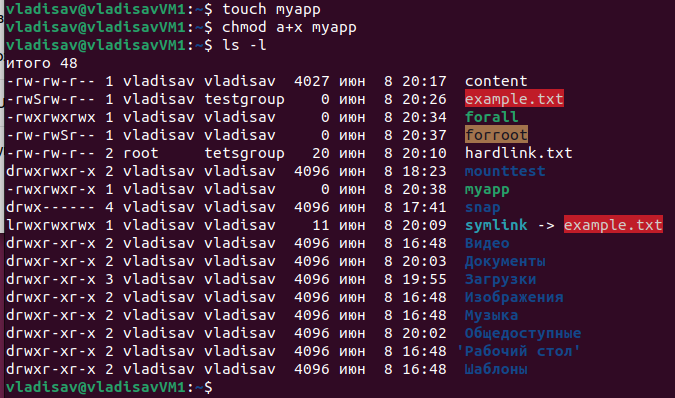
****

Теперь этот файл доступен всем и с ним можно производить любые действия. Также видим, что файл теперь выделяется зеленым цветом.

Назначим для файла "forroot" разрешение для группы и выполнение от имени суперпользователя

****

Теперь изменим разрешение файла "myapp" только на выполнение файла, как программы, но не на изменение и чтение

****

При таких правах мы не можем открыть и изменить этот файл, но можем выполнять с ним другие действия, например, перемещение, как на рисунке выше.

Составим таблицу основных разрешений прав на каталог и файл

|  |  |  |  |
| --- | --- | --- | --- |
| **Флаг** | **Маска** | **Права на каталог** | **Права на файл** |
| **000** | --- | Отсутствие прав | Отсутствие прав |
| **001** | --x | Доступ к файлам и их атрибутам | Права на выполнение |
| **010** | -w- | Отсутствие прав | Права на запись |
| **011** | -w x | Все, кроме доступа к именам файлов | Права на запись и выполнение |
| **100** | r-- | Только чтение имен файлов | Права на чтение |
| **101** | r-x | Чтение имен файлов и доступ файлам и их атрибутам | Права на чтение и выполнение |
| **110** | r w - | Только чтение имен файлов | Права на чтение и запись |
| **111** | r w x | Все права | Полные права |

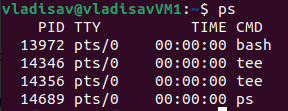
Права доступа на symlink никак не используются и не могут быть изменены. При попытке изменить права на symlink, права изменяются у самого объекта, на который она ссылается.

**Вывод**: в ходе выполнения данного пункта изменялись владелец-пользователь и владельца групп, разобран состав индексного дескриптора файла, рассмотрены дополнительные флаги для файла и каталога, совершались манипуляции с правами доступа и опробованы различные варианты права доступа на каталоги и файлы.

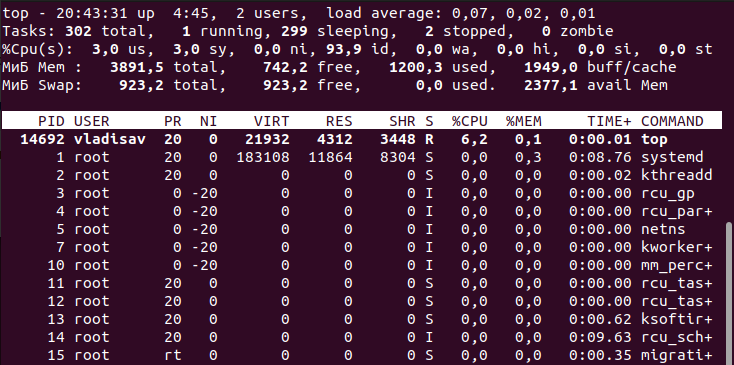
**7. Управление заданиями. Научиться запускать и управлять заданиями в оперативном и фоновом режимах. Команды jobs,ps,top,fg,bg,kill,stop,at,Ctrl+Z, nice. Интерпретировать информацию о состоянии и свойствах процессов.**

Задания могут выполняться прямо сейчас или в фоновом режиме. В фоновом режиме задания выполняются и занимают какой-то объем памяти, но при необходимости мы можем их задействовать на передний план без потерь данных и ожиданий. На переднем плане может быть только одно задание. Поэтому задание на переднем плане считается заданием, с которым пользователь взаимодействует прямо сейчас.

* **jobs** - команда, которая выводит список фоновых процессов, запущенных в данном терминале
* **ps**- команда, которая выводит список всех запущенных процессов

****

* **top** - команда, выводящая информацию о всех процессах, запущенных в системе: идентификатор процесса, пользователь, запустивший процесс, использование процессом системных ресурсов, время работы процесса, команда, запустившая процесс

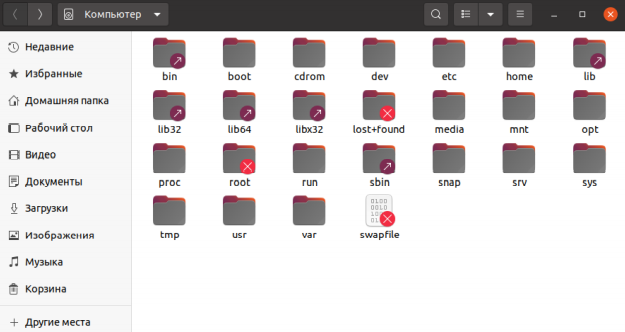
****

* **fg,bg** - команды, позволяющие осуществлять переключение между фоновым и оперативным режимами выполнения программы прямо во время выполнения
* **kill**- команда, позволяющая принудительно завершить процесс
* **at** - команда, позволяющая запустить определенную задачу в указанное время
* **Ctrl+Z**- сочетание клавиш, позволяющее остановить выполнение процесса. Данное сочетание клавиш часто необходимо в случае запуска длительных по времени выполнения программ в оперативном режиме, так как в отличие от команды stop, это сочетание клавиш может прервать выполнение программы, не дожидаясь освобождения терминала
* **nice**- команда, позволяющая запустить процесс с заданным приоритетом. Приоритет задается в диапазоне от -20 (высокий) до 19 (низкий).

**Вывод:** в ходе выполнения данного пункта были получены навыки в управлении процессами в фоновом и оперативном режимах с помощью консольных команд.

**8. Базовая файловая структура Linux. Структура и назначение основных каталогов файловой системы Linux. Получение информации о системе с помощью псевдофайловой системы /proc.**

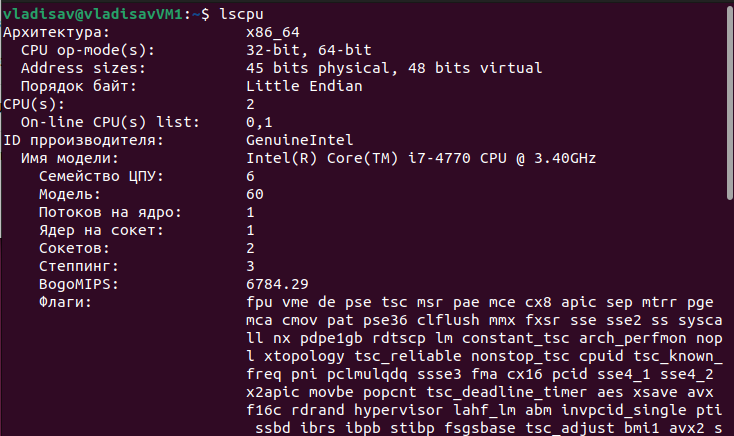
В версиях дистрибутивов Linux каталоги могут между собой отличаться. В ходе выполнении данной работы использовался дистрибутив Ubuntu.

****

* /bin - содержит стандартные утилиты Linux, основные исполняемые файлы, доступные всем пользователям, а также содержит символьные ссылки на исполняемые файлы.
* /boot - содержит конфигурационные файлы загрузчика в каталоге grub, образы ядра, файлы Initrd.
* /dev - содержит файлы устройств. Большинство устройств в Linux представляют из себя файлы в особой файловой системе. К ним обращается система для выполнения задач, связанных с вводом/выводом
* /etc - содержит конфигурационные файлы операционной системы и всех сетевых служб
* /home - домашний каталог пользователей
* /lib, lib32, lib64, libx32- здесь находятся различные библиотеки и модули ядра. В процессе установки различных программ устанавливаются зависимости для корректной работы программы. Эти зависимости- набор собранных особым образом файлов, которые подключаются во время установки к устанавливаемой программе
* /lost+found- содержатся восстановленные после некорректного размонтирования файловой системы файлы и каталоги
* /mnt и /media- в этих каталогах содержатся точки монтирования
* /opt - размещаются установленные программы, имеющие большой дисковый объем, или вспомогательные пакеты.
* /proc - это каталог псевдофайловой системы procfs, которая используется для предоставления информации о процессах. Он существует только во время работы системы в оперативной памяти компьютера. Многие из утилит, выводящие информацию о системе, берут свои исходные данные из этого каталога
* /root - каталог пользователя root.
* /run - это совершенно новый каталог для хранения данных, которые были запущены приложениями, требующимися в процессе работы
* /sbin - набор утилит для системного администрирования, содержит исполняемые файлы, необходимые для загрузки системы и ее восстановления
* /snap- каталог, содержащий установленные файлы и ПО пользователем
* /srv- содержит служебные каталоги различных сетевых служб
* /sys- каталог псевдофайловой системы sysfs, предоставляющей информацию о системе
* /tmp - каталог, в котором хранятся временные файлы. Система автоматически регулярно очищает этот каталог
* /usr- содержит пользовательские программы, документацию, исходные коды программ и ядра. Это один из самых больших каталогов файловой системы. В этот каталог устанавливаются практически все программы
* /var - содержит файлы, которые подвергаются наиболее частому изменению

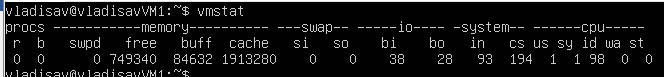
Так как разной информации много поэтому используем команду "/proc" с некоторыми из параметров для получения информации о системе:

**lscpu**- информация о процессоре

****

Здесь содержится информация о поставщике, названии модели, частоте, количестве ядер, количество физических процессоров и многая другая полезная информация

**vmstat**- информация о памяти, включая общее доступное количество и количество свободной памяти

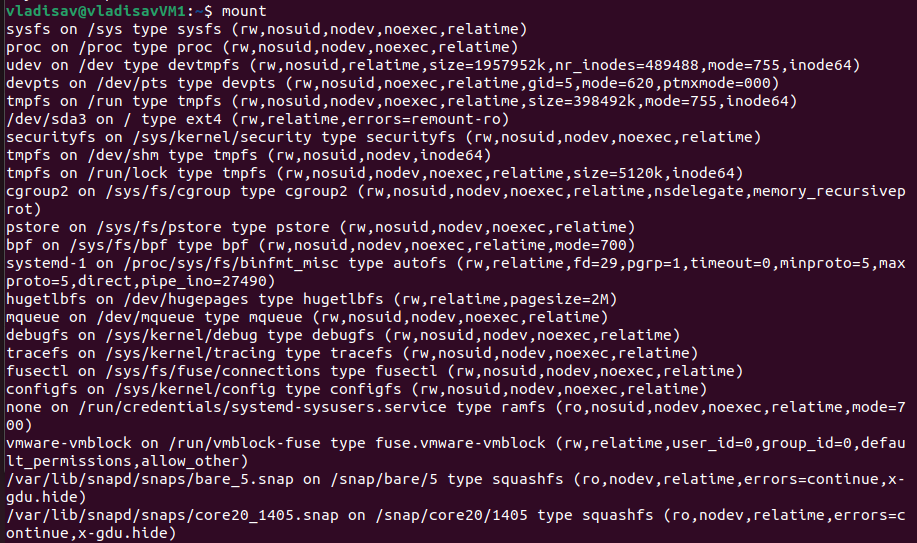
****

Здесь содержится информация о доступном объеме памяти, какой объем не используется, занятая память под разные режимы, общий объем области подкачки, зарезервированная память и многая другая информация.

**devices**- выводит список устройств. Выводится большой список все устройств. Все эти устройства разделены на 2 блока: символьные устройства и блочные устройства.

**filesystens**- список поддерживаемых файловых систем

**mount**- список смонтированных файловых систем



**partitions**- информация о доступных разделах

**version**- версия ядра

**swaps**- показать файлы подкачки

**net/dev**- показать сетевые интерфейсы

**interrupts**- показать прерывания

**buddyinfo**- содержится информация, которую можно использовать при диагностике проблем с фрагментацией памяти

**diskstats**- статистика дискового ввода-вывода по каждому устройству

**Вывод**: в ходе выполнения данного пункта были изучены структура и назначения каталогов файловой системы Linux в дистрибутиве Ubuntu, также получена основная информацию о системе с помощью псевдофайловой системы /proc.

**Вывод по лабораторной работе**:

В ходе выполнения данной лабораторной работы была произведена большая работа по изучению операционной системы Linux в целом. Была произведена регистрация в системе, использованы информационные команды, производилось управление учетными записями пользователей, произведена работа с устройствами и файловой системой, произведено управление файлами, произведено управление доступом к файловой системе, получены навыки запуска и управления заданиями в оперативном и фоновом режимах также с помощью консоли. Были изучены структура и назначение основных каталогов файловой системы Linux в дистрибутиве Ubuntu