

Терминал.  
Файлы.  
Директории.



# Содержание

- Файловая подсистема Linux — NTFS, BTRFS, ZFS, EXT4
- Терминал — bash, потоки и man
- Файлы и каталоги — типы и команды

# Файловая подсистема Linux

«Все есть файл»

**Файловая система** – способ организации хранения файлов на носителей

**Файл** – упорядоченная совокупность информации на цифровом носителе

**Директория** – это файл, содержащий информацию о группе файлов



# NTFS

## Преимущества:

- Поддерживает большие файлы(256 ТБ)
- Поддержка сжатия файлов
- Поддержка шифрования
- Контроль и восстановление целостности файлов
- Поддержка функции изменения размера разделов

## Недостатки:

- Несовместим со старыми ОС
- Ограничения на имена файлов
- Отсутствие поддержки прав файлов POSIX

# BTRFS

## Преимущества:

- Поддержка COW (copy-on-write)
- Самовосстановление
- Поддержка сжатия файлов
- Снимки
- Дедупликация (только одна копия дублированных данных)

## Недостатки:

- Нестабильный RAID
- Высокая фрагментация в случае HDD

# ZFS

## Преимущества:

- Масштабируемость (до 1 миллиарда ТБ)
- Поддержка RAID
- Поддержка сжатия данных
- Моментальные снимки
- Контроль целостности и восстановление файлов

## Недостатки:

- Требует много оперативной памяти и процессорного времени
- Не подходит для обычных накопителей

# XFS

## Преимущества:

- Высокая скорость и надёжность
- Масштабируемость
- Дефрагментация
- Высокая параллелизация операций ввода/вывода

## Недостатки:

- Отсутствие защиты от тихих сбоев накопителя
- Высокая чувствительность к большому количеству мелких данных

# EXT4

## Преимущества:

- Поддержка большого объема файлов
- Аккуратен с флэш памятью
- Надежен
- Быстрая проверка файловой системы
- Неограниченное количество подкаталогов

## Недостатки:

- Функционально устарел
- Плохая масштабируемость



# Содержание

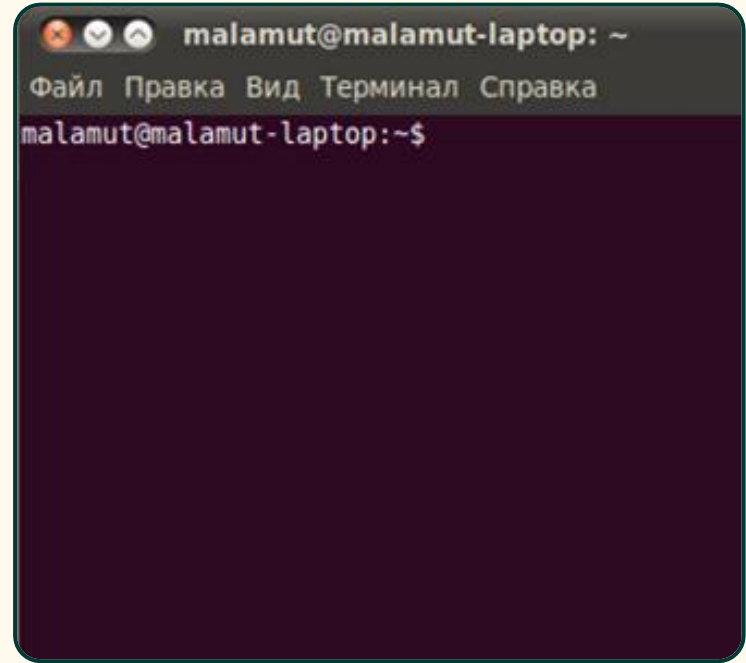
- Файловая подсистема Linux — NTFS, BTRFS, ZFS, EXT4
- Терминал — bash, потоки и man
- Файлы и каталоги — типы и команды

# Текстовая оболочка

Интерфейс пользователя, в котором вся информация представлена исключительно в виде текста

Стандартные потоки привязаны к файловым дескрипторам с номерами 0, 1 и 2:

- Стандартный поток ввода (**stdin**) — 0;
- Стандартный поток вывода (**stdout**) — 1;
- Стандартный поток ошибок (**stderr**) — 2.



# Управление потоками

- **< файл** : направить данные из файла в поток ввода
- **> файл** : направить поток вывода в файл
- **2> файл**: направить поток ошибок в файл
- **>> файл**: направить поток вывода в файл(запись в конец файла)
- **2>> файл**: направить поток ошибок в файл(запись в конец файла)
- **&>файл (>файл 2>&1)**: направить поток ошибок и вывода в файл
- **>&-** : закрыть поток вывода перед вызовом команды
- **2>&-** : закрыть поток ошибок перед вызовом команды

# Manual.Структура.

**man** – интерфейс, используемый для просмотра справочных руководств системы.

## ЗАГОЛОВКИ:

- **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 (Ключи):** 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, которая вызвала 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

# Manual.Структура.

## ЗАГОЛОВКИ:

- **Файлы (Files):** Список файлов, которые использует команда или программа, например, конфигурационный файл
- **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, например, POSIX.
- **Заметки (Notes):** Дополнительные заметки.
- **Баги (Bugs):** Известные ошибки.
- **Примеры (Examples):** Один или несколько примеров использования команды.
- **Авторы (Authors):** Люди, которые разработали и поддерживают команду.
- **Посмотрите также (See Also):** Рекомендуемые материалы по команде.

# Manual.Разделы.

- **Основные команды (General commands):** Команды, которые используются в командной строке
- **Системные вызовы (System calls):** Функции ядра, которые может вызвать программа
- **Функции библиотек (Library functions):** общий набор функций и возможностей, используемых программами
- **Форматы файлов и соглашения (File formats and conventions):** Форматы файлов как passwd, cron, tar архивы

- **Специальные файлы (Special files):** обычно устройства, например, найденные в /dev, и их драйверы
- **Дополнительно (Miscellaneous):** Описание таких вещей как inodes, параметры загрузки
- **Администрирование системы (System administration):** Команды и демоны, зарезервированные для использования root-ом
- **Распорядок ядра (Kernel Routines):** Информация, касающаяся внутренних операций ядра

# Содержание

- Файловая подсистема Linux — NTFS, BTRFS, ZFS, EXT4
- Терминал — bash, потоки и man
- Файлы и каталоги — типы и команды

# Строение файла

Состоит из:

- Жесткая ссылка
- Inode (индексный дескриптор)
- Данные

**Inode** – метаданные :

- длина в байтах
- идентификатор устройства
- идентификатор владельца
- идентификатор группы
- Режим файла
- дата последнего изменения Inode, дата последней модификации файла и дата последнего доступа
- счетчик жестких ссылок
- указатели на блоки диска, хранящие содержимое файла



# Строение файла

Типы файлов		Назначение
Обычные файлы	-	Хранение символьных и двоичных данных
Каталоги	d	Организация доступа к файлам
Символьные ссылки	l	Предоставление доступа к файлам расположенных в любом месте
Блочные устройства	b	Интерфейс для взаимодействия с аппаратным обеспечением компьютера
Символьные устройства	c	
Каналы	p	Организация взаимодействия процессов в системе
Сокеты	s	

# Команда ls

Стандартные параметры:

- **-a**: вывести всё
- **-l**: вывести атрибуты
- **-h**: читаемый вид размера файла
- **-R**: просмотр вложенных директорий
- **-c**: сортировка по имени/дате

1	2	3	4	5	6	7	8
-	rwxr--r--	3	root	root	101560	апр 28 2022	file

1. Тип файла

2. Режим прав файла

3. Количество ссылок

4. Имя владельца

5. Имя группы

6. Размер файла в байт

7. Дата последнего изменения

8. Имя файла

# Каталоги

- **/home** – каталог пользователей
- **/root** – каталог рута
- **/media** – каталог внешних устройств
- **/mnt** – каталог для временного монтирования
- **/boot** – каталог ядра и загрузчика
- **/etc** – каталог конфигурационные файлы системных программ
- **/dev** – каталог различных устройств
- **/bin** – каталог системных утилит-программ
- **/sbin** – каталог системных утилит для администрирования
- **/lib** – каталог библиотек
- **/usr** – каталог пользовательских программ
- **/opt** – для программ которые устанавливаются вручную
- **/tmp** – каталог временных файлов
- **/var** - содержит файлы с изменяемыми данными(логи, кеш, временные файлы)

# Команды - каталоги

→ **mkdir** (создание):

- ◆ -m: режим прав каталога
- ◆ -p: создает недостающие родительские каталоги
- ◆ --verbose: вывод сообщений для каждого каталога

→ **rmdir**: удаляет пустые каталоги

→ **pwd**: путь текущей директории

→ **cd**: смена директории

→ **rm -rf**: удаление каталога с содержимым

→ **mv**:  
переименование/перемещение директории

→ **cp -r**: копирование директории и содержимого

# Команды - файлы

- **file**: выводит тип файла
- **stat**: выводит содержимое inode
- **touch**: изменяет временные штампы файла
  - ◆ **-a**: изменить время последнего доступа к файлу
  - ◆ **-c** : не создавать файл
  - ◆ **-m**: изменить время последней модификации файла
- **mknod**: создает файл блочного устройства
- **mkfifo**: создает файл символьного устройства
- **cp**: копирование файла
- **mv**: перемещение/переименование файла
- **rm**: удаление файла

# Команды - содержимое файлов

- **cat:** объединение и вывод содержимого файлов
- **split:** разбить файл на части
- **more:** постраничный вывод файла
- **less:** полный просмотр файла
- **od:** вывод дампа файла в восьмеричном формате
- **cut:** удаление секций из строк содержимого файла
- **grep:** поиск содержимого в файле
- **sort:** сортировка содержимого
- **wc:** вывод определенного числа строк, слов и байтов содержимого файлов
- **tr:** перевод или удаление символов
- **uniq:** вывод повторяющихся строк
- **head:** вывод начала содержимого файла
- **tail:** вывод конца содержимого файла
- **fold:** вывод содержимого файла с определенной длиной строки

# Оболочка ZSH

Современная, красивая  
командная оболочка

<https://ohmyz.sh/>

```
~ > mkdir zshhh
~ > cd zshhh
~/zshhh > git init
Initialized empty Git repository in /home/michiell/zshhh/.git/
~/zshhh > touch zshhh.txt
~/zshhh > gaa
~/zshhh > gcam "first commit"
[master (root-commit) 40ef851] first commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 zshhh.txt
~/zshhh > Zsh is great!
zsh: command not found: Zsh
X ~/zshhh > 
```

# Ссылки

- [Web Man](#)
- [Oh my zsh.sh](#)
- [ZSH Темы](#)
- [ZSH Плагины](#)
- [Шпаргалка команд](#)
- [Установка Linux рядом с Windows](#)
- [Кастомизация ZSH](#)
- [Искусство командной строки](#)