* [**Лабораторная работа №3. Файловые подсистемы.**](https://xn--80aqobguv5e.xn--p1ai/%D0%BE%D0%B8%D0%B1/lr3.html)
  + [**Основные теоретические сведения**](https://xn--80aqobguv5e.xn--p1ai/%D0%BE%D0%B8%D0%B1/lr3.html#id1)
  + [**Задания к лабораторной работе**](https://xn--80aqobguv5e.xn--p1ai/%D0%BE%D0%B8%D0%B1/lr3.html#id2)
  + [**Вопросы к лабораторной работе**](https://xn--80aqobguv5e.xn--p1ai/%D0%BE%D0%B8%D0%B1/lr3.html#id3)

**Лабораторная работа №3. Файловые подсистемы.**

**Основные теоретические сведения**

**Цель:** Получение теоретических и практических навыков работы с таблицами разделов(MBR и GPT), создания разделов и файловых систем.

**Консольные команды:**

* fdisk <параметры> - Консольная программа для управления дисками (Работает только с MBR).
* parted <параметры> - Консольная программа для управления дисками (Работает как с MBR, так и с GPT).
* dd <параметры> - Консольная программа копирования данных.
* mkfs.<тип файловой системы> <раздел диска> - Класс консольных команд создания файловых систем на разделах.
* mount -t <тип файловой системы> <раздел диска> <точка монтирования> - Консольная программа монтирования разделов жесткого диска.

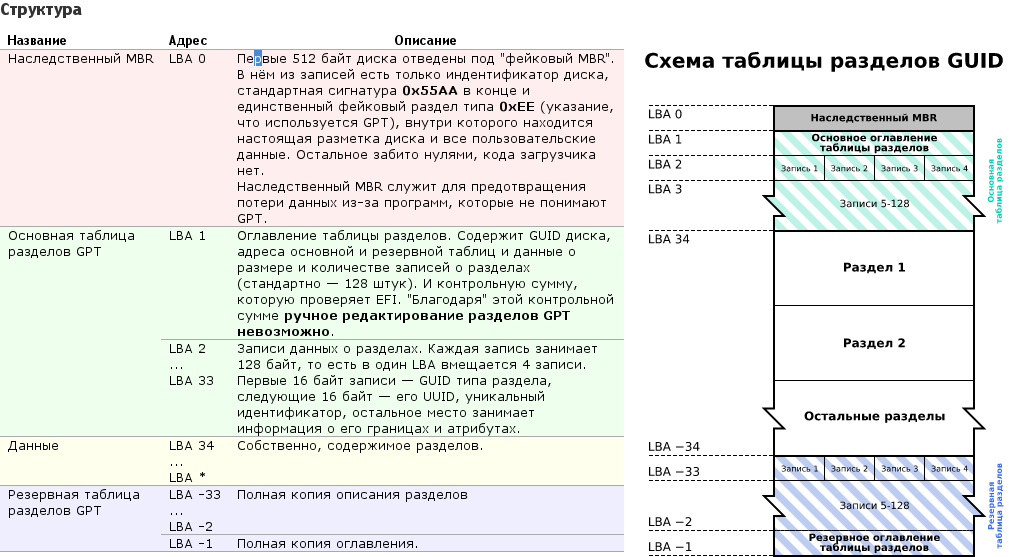
Диск делится на разделы. Как именно диск делится на разделы, определяется таблицей разделов. Таблицы разделов бывают двух типов : MBR и GPT.

**Структура MBR**

Первые 512 байт (первый сектор диска) главного устройства хранения данных занимает MBR (Master Boot Record). В состав MBR входит 446 байт кода загрузчика, четыре записи по 16 байт - это таблица разделов, 2 байта сигнатуры. Таблица разделов может состоять из первичных разделов (до 4) и логических разделов(до 128).

**Структура GPT**

GUID Partition Table, аббр. GPT — стандарт формата размещения таблиц разделов на физическом жестком диске. Он является частью расширяемого микропрограммного интерфейса (англ. Extensible Firmware Interface, EFI) — стандарта, предложенного Intel на смену BIOS. EFI использует GPT там, где BIOS использует главную загрузочную запись (англ. Master Boot Record, MBR). В GPT нет собственной программы-загрузчика, вместо этого он работает в паре с EFI. Внутри GPT используется адресация логических блоков LBA, которая абстрагирована от физики устройства (в отличие от CHS — «Цилиндр-Головка-Сектор»). Каждый логический блок занимает 512 байт. LBA 0 — первые 512 байт диска, LBA 1 — следующие, и так далее. Отрицательные значения LBA означают смещение в блоках с конца диска. Последний блок имеет смещение «-1» (LBA -1).



**Примечание**

На данный момент наиболее распространенной схемой разбиения дисков является MBR. Но с развитием средств хранения данных и их объемов, возможностей MBR становится недостаточно. Это связанно с невозможностью обеспечивать доступ к разделу диска емкостью более чем 2.2 TB. На сегодняшний день уже доступны диски емкостью более 6 TB, а так же, применяются различные технологии по объединению дисков в массивы, такие как RAID и LVM. Таким образом, применение схемы разбиения дисков на основе GPT становится все более актуальным.

**Процесс загрузки**

Процесс загрузки компьютера является многоступенчатым процессом, и начинается он с инициализации системных устройств набором микропрограмм, называемых BIOS (Basic Input/Output System), которые выполняются при старте системы. После того, как BIOS успешно проверит системные устройства, идет процесс поиска загрузчика в MBR устройств хранения (CD/DVD диски, USB диск, HDD, SSD и др.) или на первом разделе устройства. После того, как загрузчик получил управление, он получает таблицу разделов и готовит к загрузке операционную систему. В семействе загрузчиков GNU/Linux яркими представителями являются GRUB и LILO. В них MBR состоит из небольшой части ассемблерного кода. Стандартный загрузчик Windows/DOS в состоянии проверить только активный раздел, считать несколько секторов с этого раздела и затем передать управление операционной системе. Он не в состоянии загрузить Linux, так как не наделен необходимым функционалом. GRand Unified Bootloader (GRUB) - это стандартный загрузчик для операционных систем семейства GNU/Linux, и всем пользователям рекомендуется по умолчанию установить его в MBR, для того чтобы иметь возможность загружать операционную систему с любого раздела, первичного или логического.

**Пример работы с MBR**

Существует специальный набор команд для работы с MBR. Так как он расположен на диске, то может быть сохранен и, в случае необходимости, восстановлен.

* dd if=/dev/sda of=/path/mbr-backup bs=512 count=1 - Для создания резервной копии MBR
* dd if=/path/mbr-backup of=/dev/sda bs=512 count=1 - Для восстановления MBR
* dd if=/dev/sda of=/path/mbr-boot-code bs=446 count=1 – Для сохранения только загрузочного кода
* dd if=/dev/sda of=/path/mbr-part-table bs=1 count=66 skip=446 - Для сохранения только таблицы разделов
* dd if=/path/mbr-backup of=/dev/sda bs=446 count=1 – Для восстановление загрузочного кода из файла mbr-backup
* dd if=/path/mbr-backup of=/dev/sda bs=1 skip=446 seek=466 count=66 - Для восстановления только таблицы разделов
* dd if=/dev/zero of=/dev/sda bs=446 count=1 - Для очистки MBR, но при этом оставить таблицу разделов

**Задания к лабораторной работе**

* Добавьте в виртуальную машину с операционной системой Linux виртуальный жесткий диск (делается это в настройках виртуальной машины).
* Запустите виртуальную машину с операционной системой Linux.
* Ознакомьтесь с командой fdisk и ее возможностями из справочной документации.
* Создайте таблицу разделов (3 первичных и 1 логический) с помощью команды fdisk на **добавленном** виртуальном диске (обычно это диск /dev/sdb).
* Запишите изменения на диск
* Проверьте факт создания разделов используя команду fdisk. (Так же, создание разделов можно проверить используя команду ls /dev/sd\*)
* Отформатируйте созданные разделы в файловую систему ext4.
* Ознакомьтесь с командами mount и umount и их возможностями из справочной документации.
* Смонтируйте созданные разделы и создайте там произвольные файлы.
* Сделайте резервную копию MBR с помощью утилиты DD.
* Сотрите таблицу разделов MBR с помощью утилиты DD.
* Восстановите MBR с помощью утилиты DD.
* Смонтируйте разделы и проверьте целостность данных.
* Отмонтируйте разделы.
* Установите gdisk <sudo apt-get install gdisk>
* Создайте таблицу разделов GPT (5 первичных разделов) с помощью gdisk.
* Отформатируйте созданные разделы в файловую систему ext3.
* Смонтируйте созданные разделы и создайте там произвольные файлы.
* Сделайте резервную копию GPT с помощью утилиты DD, предварительно определив необходимое количество байт для резервной копии.
* Сотрите GPT с помощью утилиты DD.
* Восстановите GPT с помощью утилиты DD.
* Смонтируйте разделы и проверьте целостность данных.
* Отмонтируйте разделы.
* Определите достоинства и недостатки таблиц разделов MBR и GPT.

**Вопросы к лабораторной работе**

1. Что записано в первом секторе главной загрузочной записи MBR ?
2. Функциональное назначение MBR и GPT ?
3. Структура GPT.
4. Какое максимальное количество первичных разделов можно создать при использовании таблицы разделов MBR ?
5. Какое максимальное количество первичных разделов можно создать при использовании таблицы разделов GPT ?
6. Как сохранить информацию о структуре MBR ?
7. Как создать 10 разделов c файловой системой ext3 на диске в таблице разделов MBR ?
8. Как стереть код загрузчика в MBR ?
9. Как можно смонтировать раздел диска с файловой системой в режиме только для чтения?
10. Как можно осуществить восстановление GPT разделов в случае сбоев?

**Составьте отчет о выполнении лабораторной работы.**

**Включите в него копии экрана и ответы на вопросы лабораторной работы.**