A picture containing logo

Description automatically generated

Khoa Công nghệ Thông tin

**Assignment 03.01:**

**MBR VÀ GPT**

**TRUY XUẤT SECTOR**

Bộ môn An toàn và Phục hồi Dữ liệu

A picture containing outdoor, sign

Description automatically generatedGiảng viên

ThS. Thái Hùng Văn

Sinh viên

19120659 – Phạm Văn Thành

20120382 – Hoàng Thu Thủy

MỤC LỤC

[1 Kiến trúc MBR và GPT 3](#_Toc149869482)

[1.1 MBR 3](#_Toc149869483)

[1.2 GPT 6](#_Toc149869484)

[2 Chương trình truy xuất sector logic 11](#_Toc149869485)

[2.1 Chuẩn bị dữ liệu 11](#_Toc149869486)

[2.2 Đọc thông tin bootsector 13](#_Toc149869487)

[2.3 Tạo Image file cho volume 15](#_Toc149869488)

[3 Danh mục tham khảo 17](#_Toc149869489)

# Kiến trúc MBR và GPT

– Ổ cứng máy tính có thể được chia thành nhiều phân vùng. Để quản lí các phân vùng này, người ta lưu thông tin chúng thành bảng phân vùng (bảng Partition).

– MBR (Master Boot Record) và GPT (GUID Partition Table) là hai chuẩn để lưu trữ các thông tin trên.

– Kiến trúc MBR là một cách tổ chức ổ đĩa cứng truyền thống, được sử dụng rộng rãi trong các hệ điều hành Windows. MBR là viết tắt của Master Boot Record, là một khu vực đặc biệt nằm ở phần đầu tiên của ổ đĩa, chứa thông tin về các phân vùng và mã khởi động. MBR cho phép tối đa bốn phân vùng chính trên một ổ đĩa, mỗi phân vùng có thể chứa một hệ điều hành khác nhau. MBR cũng có thể tạo ra các phân vùng mở rộng để chứa nhiều phân vùng logic hơn. Tuy nhiên, MBR có một số hạn chế, như không thể quản lý các ổ đĩa lớn hơn 2 TB và không hỗ trợ tính năng bảo mật UEFI.

– Kiến trúc GPT (GUID Partition Table) là một tiêu chuẩn mới cho việc phân vùng ổ đĩa cứng, thay thế cho kiến trúc MBR (Master Boot Record) cũ. Kiến trúc GPT có nhiều ưu điểm so với MBR, như hỗ trợ dung lượng ổ đĩa lớn hơn 2TB, cho phép tạo nhiều phân vùng hơn 4 phân vùng chính, và sử dụng mã nhận dạng duy nhất (GUID) để xác định mỗi phân vùng. Kiến trúc GPT cũng có khả năng phục hồi dữ liệu khi bị lỗi nhờ có bản sao lưu của bảng phân vùng ở cuối ổ đĩa.

## MBR

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Address** | | **Description** | | **Size (bytes)** |
| **Hex** | **Dec** |
| 0000 | 0 | Bootstrap code area | | 446 |
| 01BE | 446 | Partition entry 1 | Partition table (for primary partitions) | 16 |
| 01CE | 462 | Partition entry 2 | 16 |
| 01DE | 478 | Partition entry 3 | 16 |
| 01EE | 494 | Partition entry 4 | 16 |
| 01FE | 510 | 55h | MBR signature | 2 |
| 01FF | 511 | AAh |
| MBR, total size: 446 + 4x16 + 2 = | | | | 512 |

Bảng 1. Bảng mô tả Master Boot Record[[1]](#footnote-1)

|  |  |  |  |
| --- | --- | --- | --- |
| **Offset (bytes)** | | **Field length (bytes)** | **Description** |
| 0x00 |  | 1 | Status (0x80 = bootable, 0x00 = non-bootable, other = invalid) |
| 0x01 |  | 3 | CHS address of first block in partition. The format is described in the next 3 bytes. |
|  | 0x01 | 1 | Head |
|  | 0x02 | 1 | Sector is in bits 5 – 0, bits 9 – 8 of cylinder are in bits 7 – 6 |
|  | 0x03 | 1 | Bits 7 – 0 of cylinder |
| 0x04 |  | 1 | Partition type:   * 0x07: Phân vùng chứa “Windows”: NTFS (LBA) * 0x83: Phân vùng chứa “Linux” * 0x00: Phân vùng không sử dụng * 0x0C: Phân vùng chứa WIN95: FAT32 (LBA) * 0x0F: Phân vùng chứa Extended (EBR) * 0x06,0B: Phân vùng chứa FAT(16,32) |
| 0x05 |  | 3 | CHS address of last block in partition. The format is described in the next 3 bytes. |
|  | 0x05 | 1 | Head |
|  | 0x06 | 1 | Sector is in bits 5 – 0, bits 9 – 8 of cylinder are in bits 7 – 6 |
|  | 0x07 | 1 | Bits 7 – 0 of cylinder |
| 0x08 |  | 4 | LBA (Logical block addressing) of first sector in the partition |
| 0x0C |  | 4 | Number of blocks in partition, in little-endian format |

Bảng 2. Bảng mô tả Partition entry[[2]](#footnote-2)

* **Minh họa với Partition Table cụ thể**

– Master boot record có Partition table như sau:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, hàng

Mô tả được tạo tự động

– Đĩa cứng trên được chia thành 2 phân vùng:

|  |  |  |  |
| --- | --- | --- | --- |
| **Phân vùng** | | **1** | **2** |
| **Trạng thái** | | 0x80 : Khởi động | 0x00 : Không khởi động |
| **Bắt đầu (địa chỉ phần cứng)** | **Head** | 1 | 0 |
| **Track**  **(Cylinder)** | Byte thứ 3 (bits 7-0 của cylinder):  0x00 = 0000 0000  -> T = 00 0000 0000 = 0 | Byte thứ 3: 0xFF = 1111 1111  -> T = 11 1111 1111 = 1023 |
| **Sector** | Byte thứ 2: 0x01 = 0000 0001 -> S = 000 001 = 1  (00 là bits 9-8 của cylinder) | Byte thứ 2: 0xC1 = 1100 0001 -> S = 000 001 = 1 |
| **Loại phân vùng** | | 0x0C: Phân vùng chứa Win95 - FAT32(LBA) | 0x0F: Phân vùng Extended |
| **Kết thúc (địa chỉ phần cứng)** | **Head** | 0xFE = 254 | 0xFE = 254 |
| **Track**  **(Cylinder)** | Byte thứ 7: 0xFF = 1111 1111 -> T = 11 1111 1111 = 1023 | 1023 (tương tự phân vùng 1) |
| **Sector** | Byte thứ 6: 0xFF = 1111 1111 -> S = 111111 = 63 | 63 (tương tự phân vùng 1) |
| **Sector logic đầu tiên** | | 0x3F = 63 | 0x04A85300 = 78140160 |
| **Số lượng sector** | | 0x04A852C1 = 78140097 sectors | 0x04A891C1 = 78156225 sectors |

## GPT

|  |  |  |
| --- | --- | --- |
| **Offset** | **Length (bytes)** | **Contents** |
| 0 (0x00) | 8 | Signature ("EFI PART", 45h 46h 49h 20h 50h 41h 52h 54h or 0x5452415020494645ULL on little-endian machines) |
| 8 (0x08) | 4 | Revision 1.0 (00h 00h 01h 00h) for UEFI 2.8 |
| 12 (0x0C) | 4 | Header size in little endian |
| 16 (0x10) | 4 | CRC32 of header |
| 20 (0x14) | 4 | Reserved; must be zero |
| 24 (0x18) | 8 | Current LBA (location of this header copy) |
| 32 (0x20) | 8 | Backup LBA (location of the other header copy) |
| 40 (0x28) | 8 | First usable LBA for partitions (primary partition table last LBA + 1) |
| 48 (0x30) | 8 | Last usable LBA (secondary partition table first LBA − 1) |
| 56 (0x38) | 16 | Disk GUID in mixed endian |
| 72 (0x48) | 8 | Starting LBA of array of partition entries (usually 2 for compatibility) |
| 80 (0x50) | 4 | Number of partition entries in array |
| 84 (0x54) | 4 | Size of a single partition entry (usually 80h or 128) |
| 88 (0x58) | 4 | CRC32 of partition entries array in little endian |
| 92 (0x5C) | \* | Reserved; must be zeroes for the rest of the block (420 bytes for a sector size of 512 bytes; but can be more with larger sector sizes) |

Bảng 3. Bảng mô tả Primary GPT Header (nằm ở sector 1)[[3]](#footnote-3)

|  |  |  |
| --- | --- | --- |
| **Offset** | **Length (bytes)** | **Contents** |
| 0 (0x00) | 16 | Partition type GUID (mixed endian) |
| 16 (0x10) | 16 | Unique partition GUID (mixed endian) |
| 32 (0x20) | 8 | First LBA (little endian) |
| 40 (0x28) | 8 | Last LBA (inclusive, usually odd) |
| 48 (0x30) | 8 | Attribute flags (e.g. bit 60 denotes read-only) |
| 56 (0x38) | 72 | Partition name (36 UTF-16LE code units) |

Bảng 4. Bảng mô tả GUID Partition entry (sector 2 – 33)[[4]](#footnote-4)

* **Minh họa với Partition Table cụ thể**

– Primary GPT Header như sau:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

* Sector khả dụng đầu tiên cho phân vùng(address 022|8): 0x22 = 34 -> sector cuối cùng của primary partition table = 33.
* Sector khả dụng cuối cùng cho phân vùng(address 023|0): 0x78BFDE = 7913438 -> sector đầu tiên của secondary partition table = 7913439.
* Sector bắt đầu của partition entries: 0x02 = 2
* Số lượng partition entries: 0x80 = 128
* Kích thước của một partition entry: 0x80 = 128

– Vì sector bắt đầu của partition entries = 2, sector 2 như sau:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự động

Đĩa cứng trên được chia thành 2 phân vùng:

|  |  |  |
| --- | --- | --- |
| **Phân vùng** | **1** | **2** |
| **Loại phân vùng** | EBD0A0A2-B9E5-4433-87C0-68B6B72699C7  (Phân vùng Windows Basic Data Partition) | EBD0A0A2-B9E5-4433-87C0-68B6B72699C7  (Phân vùng Windows Basic Data Partition) |
| **Sector logic đầu tiên** | 0x80 = 128 | 0x38B080 = 3715200 |
| **Sector logic cuối cùng** | 0x38B07F = 3715199 | 0x78B07F = 7909503 |
| **Tên phân vùng** | Basic data partition | Basic data partition |

**1.3 So sánh**

So sánh giữa MBR và GPT có thể được thực hiện dựa trên các tiêu chí sau:

* ***Số lượng phân vùng:*** MBR chỉ hỗ trợ tối đa 4 phân vùng chính hoặc 3 phân vùng chính và 1 phân vùng mở rộng. GPT hỗ trợ tối đa 128 phân vùng chính trên hệ điều hành Windows và không giới hạn số lượng phân vùng trên hệ điều hành Linux.
* ***Kích thước ổ đĩa:*** MBR chỉ hỗ trợ ổ đĩa có dung lượng tối đa 2 TB. GPT hỗ trợ ổ đĩa có dung lượng lên đến 9.4 ZB (Zettabyte), tương đương với khoảng 9.4 tỷ TB.
* ***Tính tương thích:*** MBR tương thích với hầu hết các hệ điều hành cũ và mới, cũng như các thiết bị khởi động như BIOS và UEFI. GPT chỉ tương thích với các hệ điều hành mới hơn, như Windows Vista trở lên, Linux kernel 2.6 trở lên và macOS 10.4 trở lên. GPT cũng chỉ khởi động được trên các thiết bị sử dụng UEFI, không khởi động được trên BIOS.
* ***Tính bảo mật:*** MBR không có tính năng bảo mật nào đặc biệt, dễ bị lỗi và mất dữ liệu do viết sai hoặc virus. GPT có tính năng bảo mật cao hơn, bằng cách sử dụng mã nhận dạng duy nhất (GUID) cho mỗi phân vùng và sao lưu bảng phân vùng ở cuối ổ đĩa để phục hồi khi cần.

# Chương trình truy xuất sector logic

## Chuẩn bị dữ liệu

-Để tạo một file bootsector, chúng em chuẩn bị một USB được format theo FAT16, bên trong USB có chứa một vài file .txt và một subdirectory:

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Biểu tượng máy tính

Mô tả được tạo tự động

Sau khi sử dụng HxD để mở USB, lúc này thông tin hiển thị là toàn bộ dữ liệu của USB nhưng trong đó, thực tế thông tin partion data, file system, bootsector,… chứa ngay ở đầu volume nên chúng em quyết định lấy 1MB đầu tiên ( offset 000000 -> FFFFFF) copy-paste ra một vị trí khác và đặt tên file là test.bin (File này tụi em có gửi kèm source code).

Ảnh có chứa văn bản, ảnh chụp màn hình, màn hình, số

Mô tả được tạo tự động

Thông tin lúc này của Test.bin sẽ bao gồm:

+ MBR và bảng phân vùng (1 sector – 512 byte)

+ Bootsector của phân vùng FAT16 (1 sector – 512 byte)

+ 1-2 bản copy của File Alloction Table (FAT)

+ Thư mục gốc (Root Directory)

+ Các thư mục và dữ liệu khác

## Đọc thông tin bootsector

* Chương trình nhận đường link và kiểm tra xem đây có phải FAT16, sau đó hiện thông báo vị trí sector logic đầu tiên:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ

Mô tả được tạo tự động

* Chương trình hiện nội dung của Bootsector:

Ảnh có chứa văn bản, đồ điện tử, ảnh chụp màn hình, màn hình

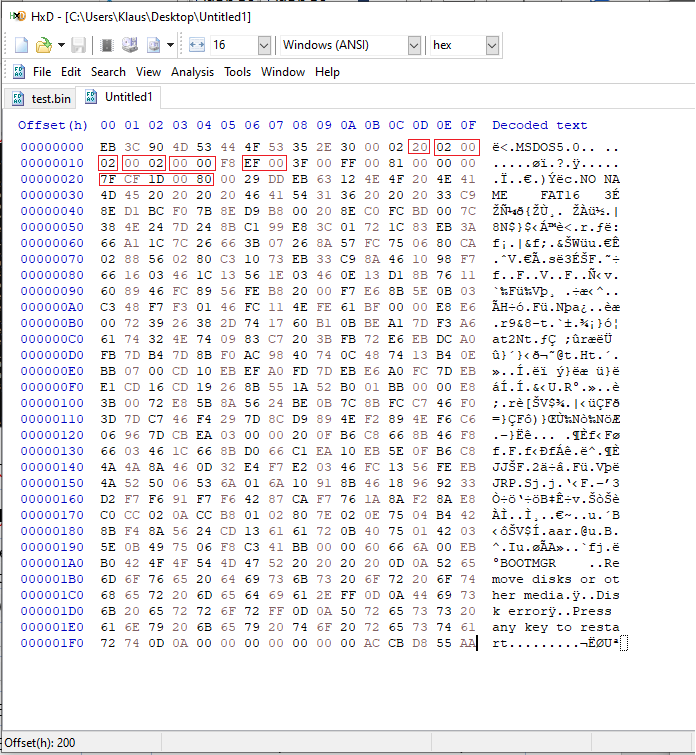
Mô tả được tạo tự động

* Chương trình hiện các thông tin lấy được từ BootSector:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ

Mô tả được tạo tự động

\*) Kiểm tra tính đúng đắn của chương trình:



|  |  |  |
| --- | --- | --- |
| **Offset(hexa)** | **Số byte** | **Kết quả** |
| 0x0D: Số sector của cluster (sectors\_per\_cluster) | 1 | 0x20 (hex) = 32 (dec) |
| 0x0E: Số sector trước bảng FAT (reserved\_sector) | 2 | 0x02 = 2 |
| 0x10: Số lượng bảng FAT (number\_of\_FATs) | 1 | 0x02 = 2 |
| 0x11: Số Entry của RDET (root\_dir\_entries) | 2 | 0x0200 = 512 (entry) |
| 0x13: Số sector của volume (total\_sectors\_short) | 2 | 0x00 = 0 |
| 0x16: Số sector của FAT (Fat\_size\_sector) | 2 | 0xEF = 239 |
| 0x20: Kích thước volume (nếu 2 byte tại offset 13 là 0) (total\_sectors\_long) | 4 | 0x1DCF7F = 1953663 |
| … | … | …. |

Bảng 4 – Các thông số Bootsector

* Chương trình hiển thị chính xác thông tin của bootsector như khi giải tay (bảng 4).

## Tạo Image file cho volume

* Chương trình sẽ tự động tạo image file khi người dung truyền vào đường dẫn tới volume và tên file muốn tạo

Eg: Tạo ở volume usb (D:\) với tên file là test.img:

Ảnh có chứa văn bản, đồ điện tử, ảnh chụp màn hình, phần mềm

Mô tả được tạo tự động

Ảnh có chứa văn bản, phần mềm, Biểu tượng máy tính, số

Mô tả được tạo tự động

* Chương trình kiểm tra tự chạy tiếp phần việc còn dở dang khi có sự cố gây gián đoạn trong lần chạy trước:

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Phông chữ

Mô tả được tạo tự động

# Danh mục tham khảo

* Master boot record, Wikipedia, [en.wikipedia.org/wiki/Master\_boot\_record](https://en.wikipedia.org/wiki/Master_boot_record)
* GUID Partition Table, Wikipedia, [en.wikipedia.org/wiki/GUID\_Partition\_Table](https://en.wikipedia.org/wiki/GUID_Partition_Table)
* Tìm hiểu và so sánh MBR với GPT và Legacy (Bios) với UEFI/EFI, [Tìm hiểu và so sánh MBR với GPT và Legacy (Bios) với UEFI/EFI | Anhdv Blog (anh-dv.com)](https://anh-dv.com/thu-thuat-hay/so-sanh-mbr-voi-gpt-va-legacy-voi-uefi)
* [Tìm hiểu và so sánh giữa MBR với GPT và BIOS với UEFI (blogtinhoc.com)](https://blogtinhoc.com/tim-hieu-va-sanh-giua-mbr-voi-gpt-va-bios-voi-uefi.html)
* Thông tin về bootsector, <https://www.compuphase.com/mbr_fat.htm>
* Cách đọc bootsector, <https://www.youtube.com/watch?v=x4frzwTvxCg>

1. Theo Wikipedia, *Master boot record* [↑](#footnote-ref-1)
2. Theo Wikipedia, *Master boot record* [↑](#footnote-ref-2)
3. Theo Wikipedia, *GUID Partition Table* [↑](#footnote-ref-3)
4. Theo Wikipedia, *GUID Partition Table* [↑](#footnote-ref-4)