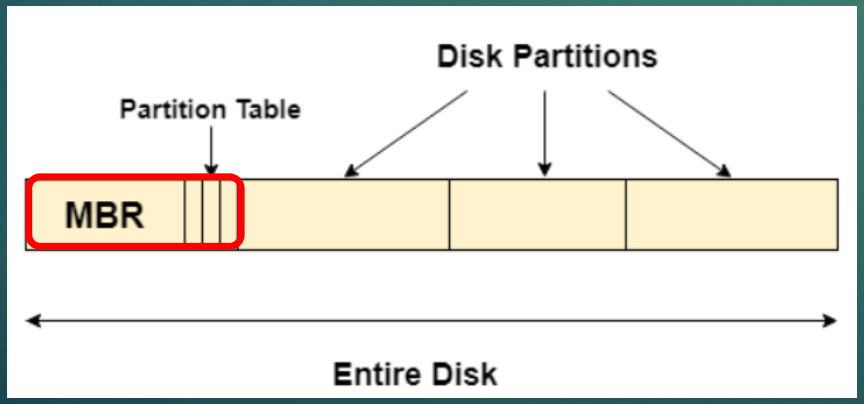
Revision on important Concepts this week-week5

EVIDENCE PROCESSING - OS & FILE SYSTEMS

DISK PARTITION – BEFORE A **DRIVE** (A,B,C,D) CAN BE USED BY ANY OS, A **PARTITION TABLE** NEEDS TO BE CREATED ON THE DRIVE. **PARTITION TABLE** IS STORED IN **MASTER BOOT RECORD (MBR),** SECTOR 0.

- The **MBR** is the information in the **first sector** of any hard disk that identifies how and where an operating system is located so that it can be **boot** (loaded) into the computer's main storage or random access memory. As such, the **MBR** holds the information on how the **logical partitions**, containing **file systems**, are organized on that medium.



Revision on important Concepts this week…

EVIDENCE PROCESSING - OS & FILE SYSTEMS

RAM SLACK & FILE SLACK

- What is **RAM Slack** & **File Slack**?
- Given required information such as **drive size**, **data/file size** and etc, how do we determine **cluster size**, calculate **RAM slack** and **File slack**.
- By default, **one sector** is always **512 bytes.** i.e 9 sectors = 9 x 512 = 4608 bytes.
- Depending on the file system (FAT32, FAT16 and etc), OS will allocate **cluster** to file so that data can be stored in file. Cluster is make up of sectors

File size is **5000** bytes

example

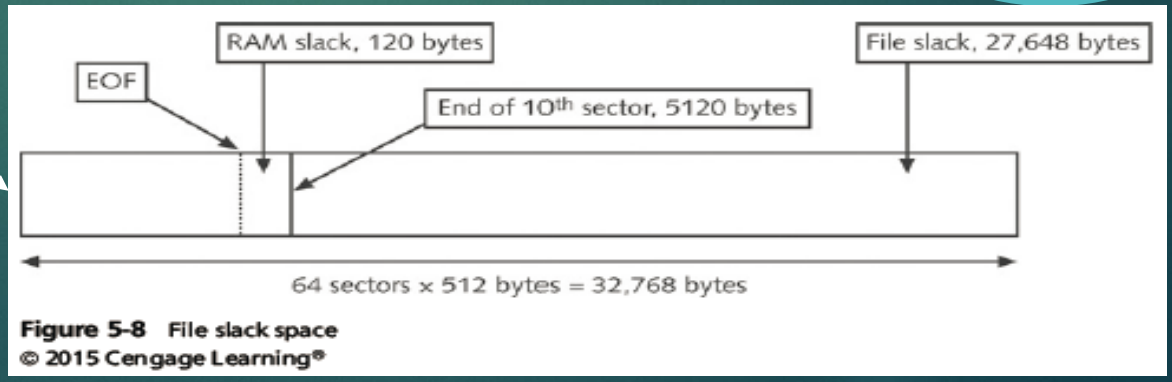


Figure 5-8 File slack space
© 2015 Cengage Learning®

Revision on important Concepts this week… (Cont)

- **Cluster, space required for a file is made up of "number of sectors"**

- **Number of Cluster Required to Store a File**
  - (FileSize) / (ClusterSize)
    = Round Up (ClusterRequired)
  - While ClusterSize is determined by no. of sectors

Table 5-2 Sectors and bytes per cluster

Drive size	Sectors per cluster	FAT16
8–32 MB	1	512 bytes
32–64 MB	2	1 KB (1024)
64–128 MB	4	2 KB
128–256 MB	8	4 KB
256–512 MB	16	8 KB
512–1024 MB	32	16 KB
1024–2048 MB	64	32 KB
2048–4096 MB	128	64 KB

- **RAM Slack**
  - Per Sector Size = **512** Bytes (in general)
  - (SectorsRequired) = (FileSize) / (SectorSize) = Round up (SectorsRequired)
  - Round Up (SectorRequired) * Sector Size = Total Sector Size Required
  - Total Sector Size – File Size = **RAM Slack**

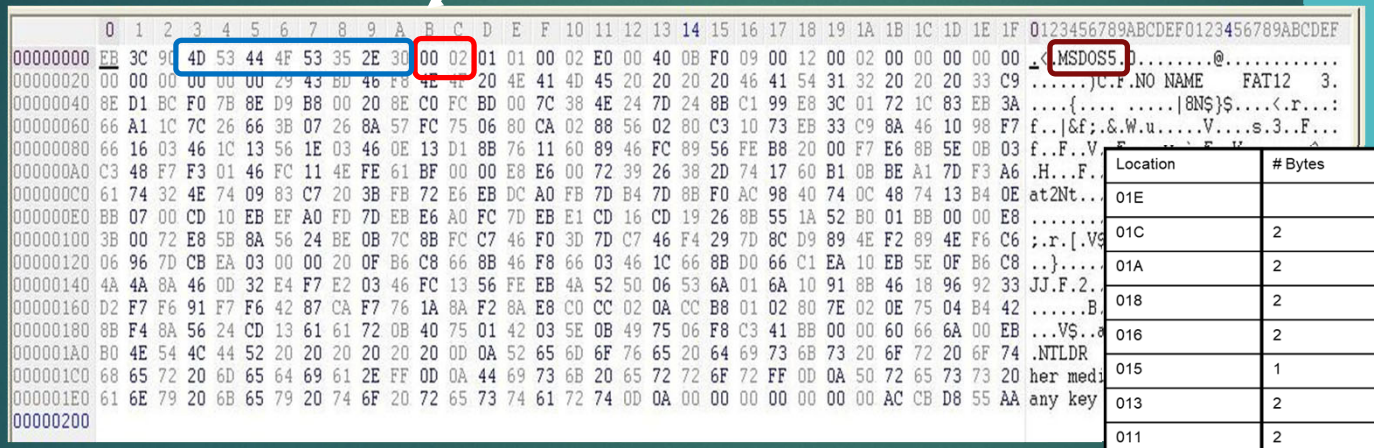
- **File Slack**
  - File Slack = (SizeOfClusterRequired) – (FileSize) – (RAMSlack)

Revision on important Concepts this week… (Cont)

- **File Allocation Table (FAT) –** A File structure database that Microsoft originally designed for floppy disks. Three major variants: FAT12, FAT16 and FAT32
  - Must understand concept of FAT. i.e **Cluster is make up of sectors and one sector is 512 Bytes.**
  - **Cluster number** is the logical address in OS
  - Need to be able to interpret **specifications of FAT** – to understand FAT file structure

00B

Address:-
00000000



Sample disk view of a **FAT file structure**

Location	# Bytes	Meaning	Value
01E		OS Boot Loader	
01C	2	# Hidden Sectors	0
01A	2	# Heads	2
018	2	# Sectors / Track	18 (12x)
016	2	# sectors/ FAT	9
015	1	Media Bytes	F0: (floppy)
013	2	# logical sectors	2880(0B40x)
011	2	# Root Dir entries	224 (00E0x)
010	1	# FATS	2
00E	2	# Boot Sectors	1
00D	1	# Sectors/Cluster	1
00B	2	# Bytes/Sector	512 (0200x)
003	8	OEM Name ID	MSDOS5.0
000	3	Jmp to loader	EB 3C 90

Specifications of FAT

Revision on important Concepts this week… (Cont)

- **NT File System – NTFS :** To improve on FAT file system. In **NTFS**, everything written to disk is a file.
  - Understand how **NTFS** files are stored in **NTFS** system
    - Namely "**Resident**" and "**Non-Resident**" - 2 types

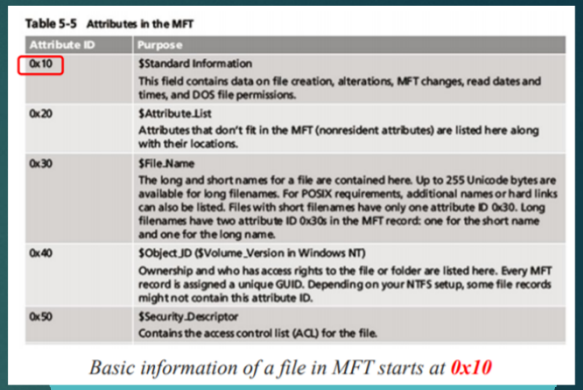


Table 5-5 Attributes in the MFT

Attribute ID	Purpose
0x10	$Standard Information This field contains data on file creation, alterations, MFT changes, read dates and times, and DOS file permissions.
0x20	$Attribute_List Attributes that don't fit in the MFT (nonresident attributes) are listed here along with their locations.
0x30	$File_Name The long and short names for a file are contained here. Up to 255 Unicode bytes are available for long filenames. For POSIX requirements, additional names or hard links can also be listed. Files with short filenames have only one attribute ID 0x30. Long filenames have two attribute ID 0x30s in the MFT record: one for the short name and one for the long name.
0x40	$Object_ID ($Volume_Version in Windows NT) Ownership and who has access rights to the file or folder are listed here. Every MFT record is assigned a unique GUID. Depending on your NTFS setup, some file records might not contain this attribute ID.
0x50	$Security_Descriptor Contains the access control list (ACL) for the file.

*Basic information of a file in MFT starts at **0x10***

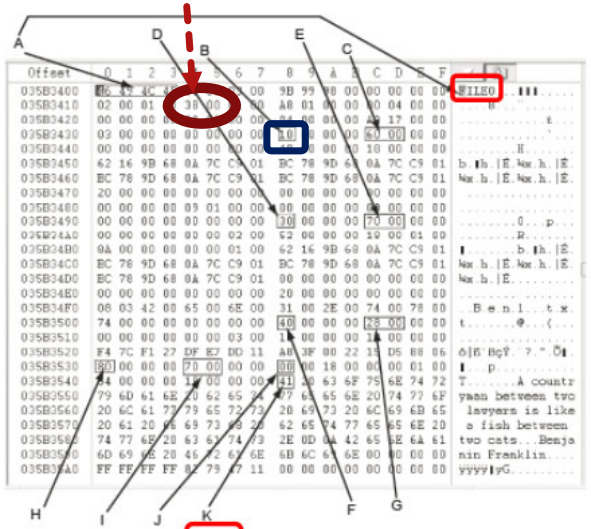
- First data set of **NTFS** disk
  - Is the **Partition Boot Sector**
  - Next is **Master File Table (MFT)**
    - **Each file** on an NTFS volume is represented by **a record** in master file table (**MFT**)
- **MFT** contains information about **all files** on the disk
- In the **MFT**, the **first 15 records are reserved for system files**
- Need to understand **MFT Structures** as well as **Attributes in the MFT**

The **Master File Table (MFT)** allocates space for each file record. The **attributes** of a file are written to the allocated space in the MFT. Small files and directories (typically **512 bytes** or smaller), can entirely be contained within the master file table's record. Such file is also known as "Resident" file.

Revision on important Concepts this week… (Cont)

MFT and File Attributes (Cont)

0x14 - > 2nd row, 5th position = 38 00

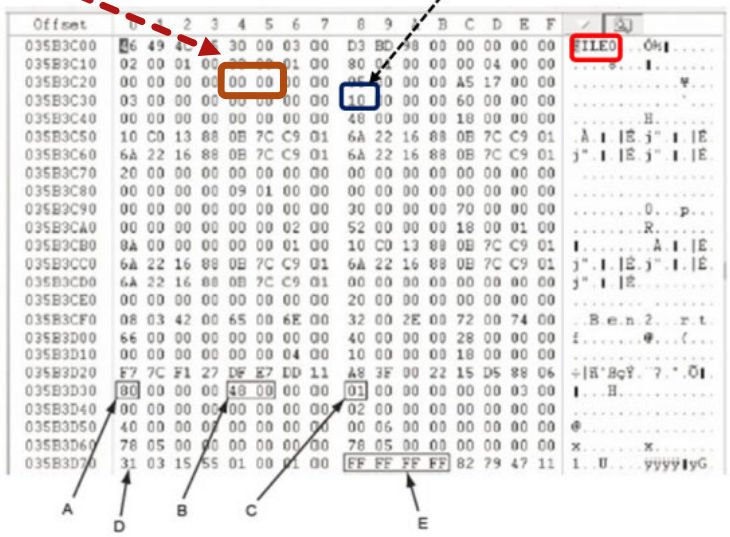


A: All MFT records start with FILE0
B: Start of attribute 0x10
C: Length of attribute 0x10 (value 60) **60 00 → 00 60**
D: Start of attribute 0x30
E: Length of attribute 0x30 (value 70) **70 00 → 00 70**
F: Start of attribute 0x40
G: Length of attribute 0x40 (value 28)
H: Start of attribute 0x80
I : Length of attribute 0x80 (value 70)
J : Attribute 0x80 resident flag
K: Starting position of resident data

Figure 5-10 Resident file in an MFT record
Courtesy of X-Ways AG, *www.x-ways.net*

Resident file attributes

Start of attribute 0x10



A: Start of nonresident attribute 0x80
B: Length of nonresident attribute 0x80
C: Attribute 0x80 nonresident flag
D: Starting point of data run
E: End-of-record marker (FF FF FF FF) for the MFT record

Figure 5-12 Nonresident file in an MFT record
Courtesy of X-Ways AG, *www.x-ways.net*

Non-resident file attributes

Table 5-5 Attributes in the MFT

Attribute ID	Purpose
0x10	$Standard Information This field contains data on file creation, alterations, MFT changes, read dates and times, and DOS file permissions.
0x20	$Attribute_List Attributes that don't fit in the MFT (nonresident attributes) are listed here along with their locations.
0x30	$File_Name The long and short names for a file are contained here. Up to 255 Unicode bytes are available for long filenames. For POSIX requirements, additional names or hard links can also be listed. Files with short filenames have only one attribute ID 0x30. Long filenames have two attribute ID 0x30s in the MFT record: one for the short name and one for the long name.
0x40	$Object_ID ($Volume_Version in Windows NT) Ownership and who has access rights to the file or folder are listed here. Every MFT record is assigned a unique GUID. Depending on your NTFS setup, some file records might not contain this attribute ID.
0x50	$Security_Descriptor Contains the access control list (ACL) for the file.

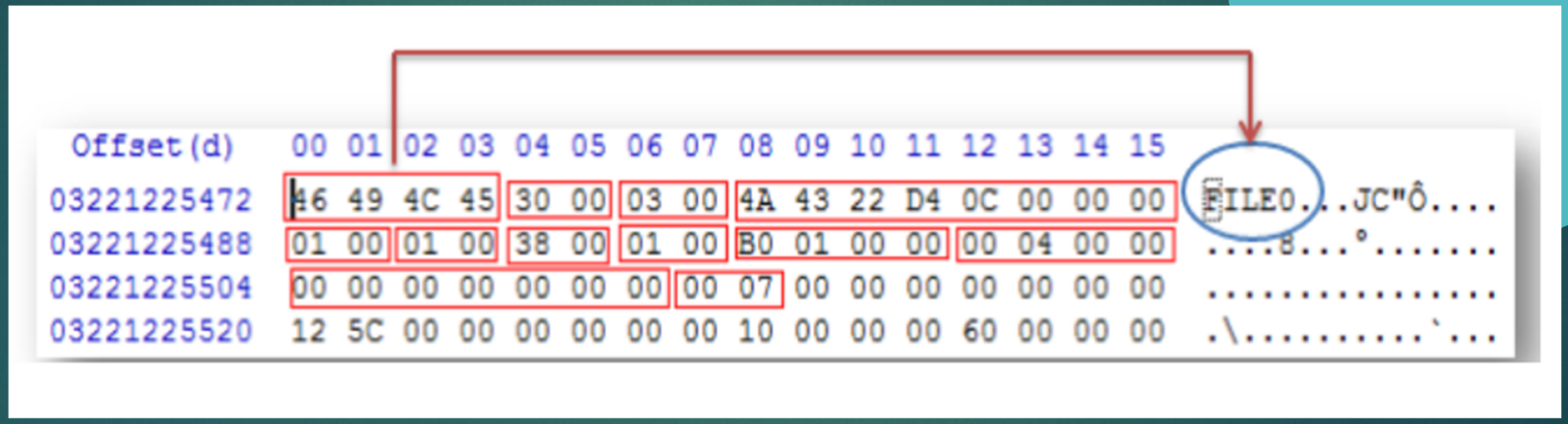
*Basic information of a file in MFT starts at **0x10***

*See **slide 29 on chapter 5** for more information on attribute ID*

- *At offset 0x14* - length of the header (indicates where the next attribute starts) *38 00 → 00 38 = 56 bytes!!*

Taking a closer look on NTFS Master File Table Records

- The first field in each MFT entry is the signature also known as MFT Record Identifier (or magic number), and a standard entry will have the ASCII string "FILE.“ or the value 0x46494c45.



Revision on important Concepts this week…

- Is there a difference between **Steganography** and **water marking**?

  Steganography and **watermarking** bring a variety of very important techniques on how to hide important information in an undetectable and/or irremovable way in audio and video data.

  Steganography: hide the very existence of the data. Adversary doesn't know of a secret communication.

  Watermarking: either **visible** or **invisible** and used to identify **ownership** and **copyright**.

Some quizzes…

1

1. What is the purpose of BIOS (Basic Input Output System)?

- ○ To improve on computer memory
- ○ To ensure image is well displayed on monitor
- ○ It contains programs that perform input and output at hardware level
- ○ It transfers files out of hard disk when hard disk is full

2

1. Cluster size vary according to disk drive size and file system. For FAT file system, what is the size of cluster for a 256MB disk drive?

- ○ 2KB
- ○ 4KB
- ○ 8KB
- ○ 16KB

3

1. Basic information of a file or folder in NTFS MFT environment, starts at attribute ID:-

- ○ 0x10
- ○ 0x20
- ○ 0x30
- ○ 0x40

4

1. What is the purpose of Registry in windows environment?

- ○ To register all files and folders
- ○ To store hardware and software configuration information
- ○ To enhance speed of memory in the computer
- ○ To act as a network device like router or switch when required

5

1. What is the issue with virtual machine when perform digital forensic?

- ○ Virtual machine does not have wifi
- ○ Virtual machine always do not have enough virtual disk space
- ○ Virtual machine does not have file slack and unallocated space
- ○ Virtual machine is always not stable and can crash easily

This Week Lab - **Steganography**

- Download "**Prac 5v2.zip**" from BrightSpace and unzip the file in your Magnet VM windows environment.
- We are going to work on **Steganography** this week
- Follow instructions in "**Pract 5 Labv5.pdf**" document to work on your lab this week. Remember to **unzip** each exercise before you start your work.
  - Note : Exercise 9 (zip file 8) is about **EXIF file type**, **EXIF image viewer** website and **SPAM MIMIC** site.
- This will be your **last practical** before term ends…

Since this is the last lab exercise….

No submission is required!!!

Do spend time on your assignment 1 when you are done with this week lab. Presentation for assignment 1 is on week7!!!