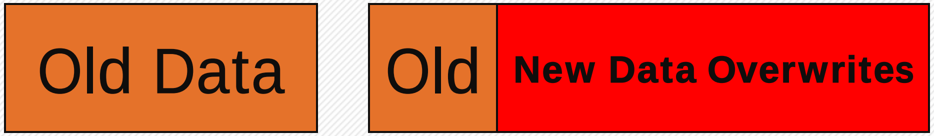
CYBER502x
Computer Forensics

Unit 5: Windows File Systems

Basic concepts in Windows

- Clusters
  - The basic storage unit of a disk
  - The piece of storage that an operating system can actually place data into
  - Different disk formats have different cluster sizes
- Slack space
  - If they are not filled up-which, the last one almost never is –this excess capacity in the last cluster



What does a file system do?

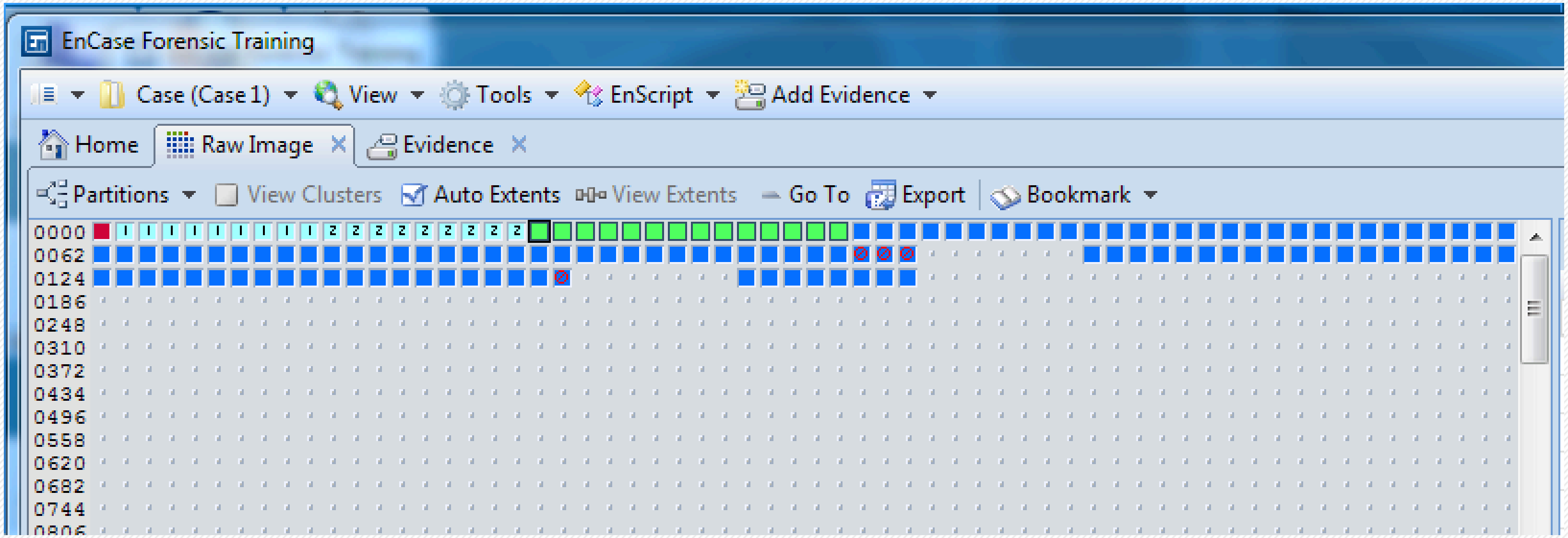
- Make a structure for an operating system to stores files
- For you to access them by name, location, date, or other characteristic.
- File System Format
  - The process of turning a partition into a recognizable file system

Windows File Systems

- File Allocation Table (FAT)
  - FAT 12
  - FAT 16
  - FAT 32
  - exFAT
- NTFS, a file system for Windows NT/2K
  - NTFS4
  - NTFS5
- ReFS, a file system for Windows Server 2012

FAT File System Structure

- The boot record
- The File Allocation Tables
- The root directory
- The data area



Boot record

- The first sector of a FAT12 or FAT16 volume
- The first 3 sectors of a FAT 32 volume
- Defines the volume, the offset of the other three areas
- Contains boot program if it is bootable

FAT (File Allocation Table)

- A lookup table to see which cluster comes next
- File Allocation Table for FAT 16
  - One entry is 16 bits representing one cluster
  - Each entry can be
    - The cluster contains defective sectors (FFF7)
    - the address of the next cluster in the same file (A8F7)
    - a special value for "not allocated" (0000)
    - a special value for "this is the last cluster in the chain“ (FFFF)

Directory entry structure

- Starting from the root directory.
- Each directory entry is 32 bytes long
- It contains information of
  - file name followed by extension
  - Type (file or subdirectory)
  - The address of the first data cluster (Byte 26-27)
  - The length of the file (byte 28-29)
  - Time and date

How to locate a file

- A directory entry that contains the file
- Find the first cluster in the directory (root or subdirectory)
- Find the chain of clusters that contain the data

File Deletion and Recovery under FAT

- Does not entirely remove the contents of that file from the disk
- The system replaces the first character of the file name with the hex byte code "E5h".
- Unallocate the clusters in FAT table

Recover Folders in FAT partition

- Recover Folders
    - Searches through the unallocated clusters that had “.” and “..”
    - Their directory entries were overwritten in the parent directory

System Format

- Two types high-level formatting in Windows
    - A quick format
        - It zeros out the root directory entries
        - Zeros out the file allocation table entries
        - The data area is not touched
        - EnCase – recover folder will help find many information
    - A full format
        - It writes the hex character F6h or zeros to the whole disk

Things are different in NTFS…

- Journaling FS
  - Changes were first recorded to a log file, then written to the disk
- Enhanced security
  - Permissions for each file, dir.
- Robust
- Maintains much more information about system and user actions
- MFT

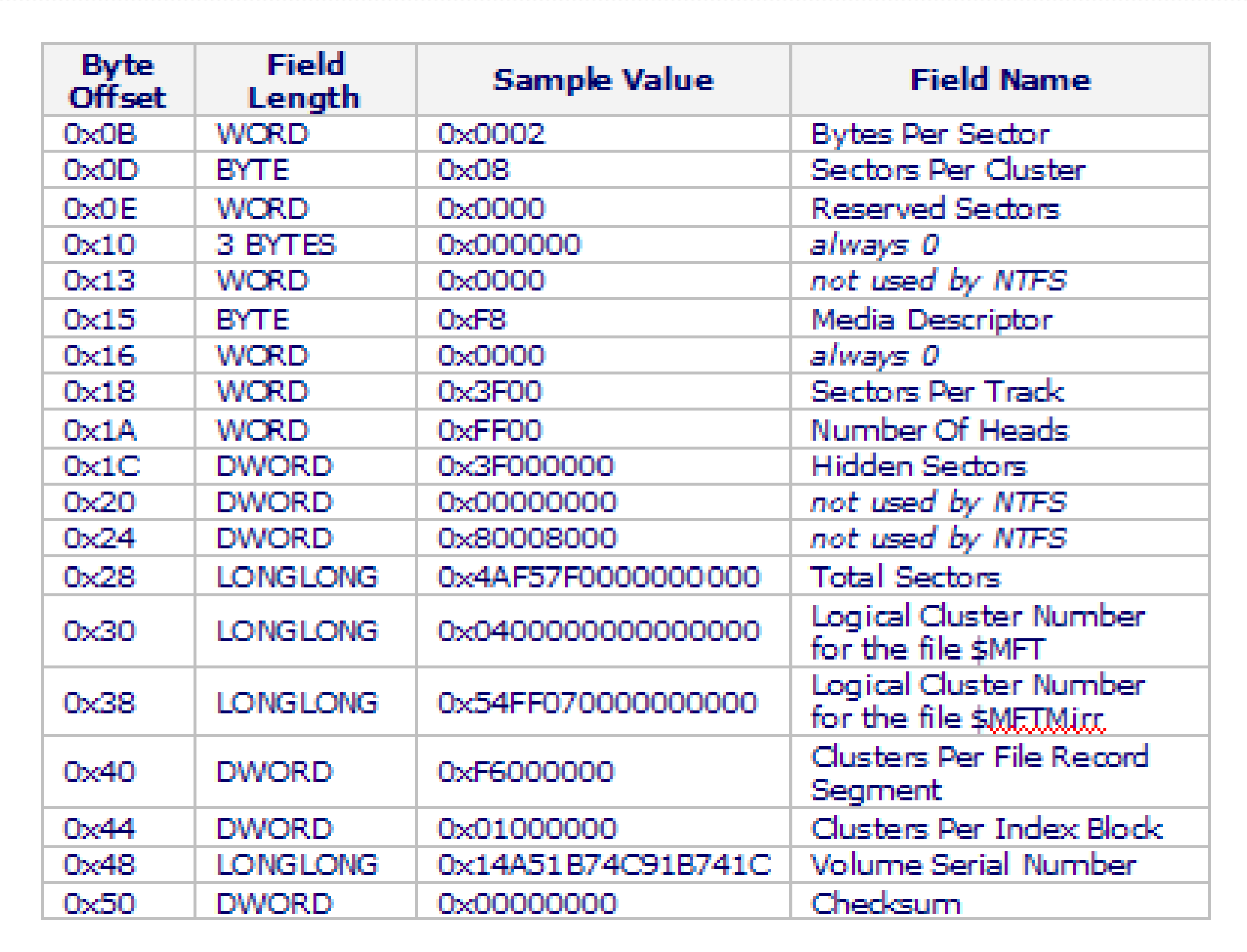
NTFS

- Used by WinNT, WinXP, …, Windows 7, …
- Supports all sizes of clusters from 512 bytes up to 64 Kbytes.
- Represents character strings in 16-bit Unicode.
- Use 64 bits for addressing the clusters.
- Master File Table

NTFS Volume Boot Sector

- Begins in the first sector of the partition, can use up to 16 sectors
- Contains
  - Information of volume label and size, the location of the key metadata files
  - Program code to load the OS (It will generally load NTLDR)

Partition boot sector



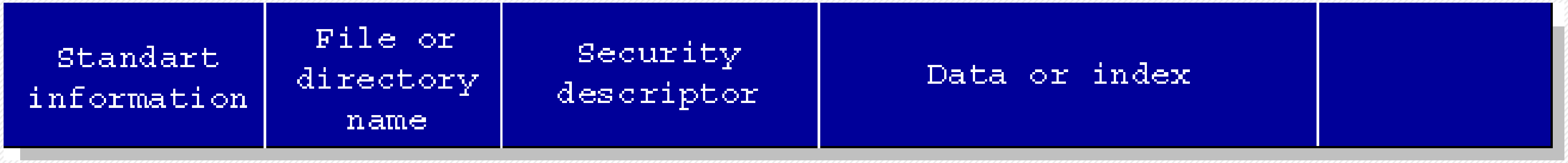
Byte Offset	Field Length	Sample Value	Field Name
0x0B	WORD	0x0002	Bytes Per Sector
0x0D	BYTE	0x08	Sectors Per Cluster
0x0E	WORD	0x0000	Reserved Sectors
0x10	3 BYTES	0x000000	*always 0*
0x13	WORD	0x0000	*not used by NTFS*
0x15	BYTE	0xF8	Media Descriptor
0x16	WORD	0x0000	*always 0*
0x18	WORD	0x3F00	Sectors Per Track
0x1A	WORD	0xFF00	Number Of Heads
0x1C	DWORD	0x3F000000	Hidden Sectors
0x20	DWORD	0x00000000	*not used by NTFS*
0x24	DWORD	0x80008000	*not used by NTFS*
0x28	LONGLONG	0x4AF57F0000000000	Total Sectors
0x30	LONGLONG	0x0400000000000000	Logical Cluster Number for the file $MFT
0x38	LONGLONG	0x54FF070000000000	Logical Cluster Number for the file $MFTMirr
0x40	DWORD	0xF6000000	Clusters Per File Record Segment
0x44	DWORD	0x01000000	Clusters Per Index Block
0x48	LONGLONG	0x14A51B74C91B741C	Volume Serial Number
0x50	DWORD	0x00000000	Checksum

Master File Table

- A system file created during the formatting of an NTFS volume.
- Record every files and directory on the volume, including an entry for itself.
- Record 16 system files.

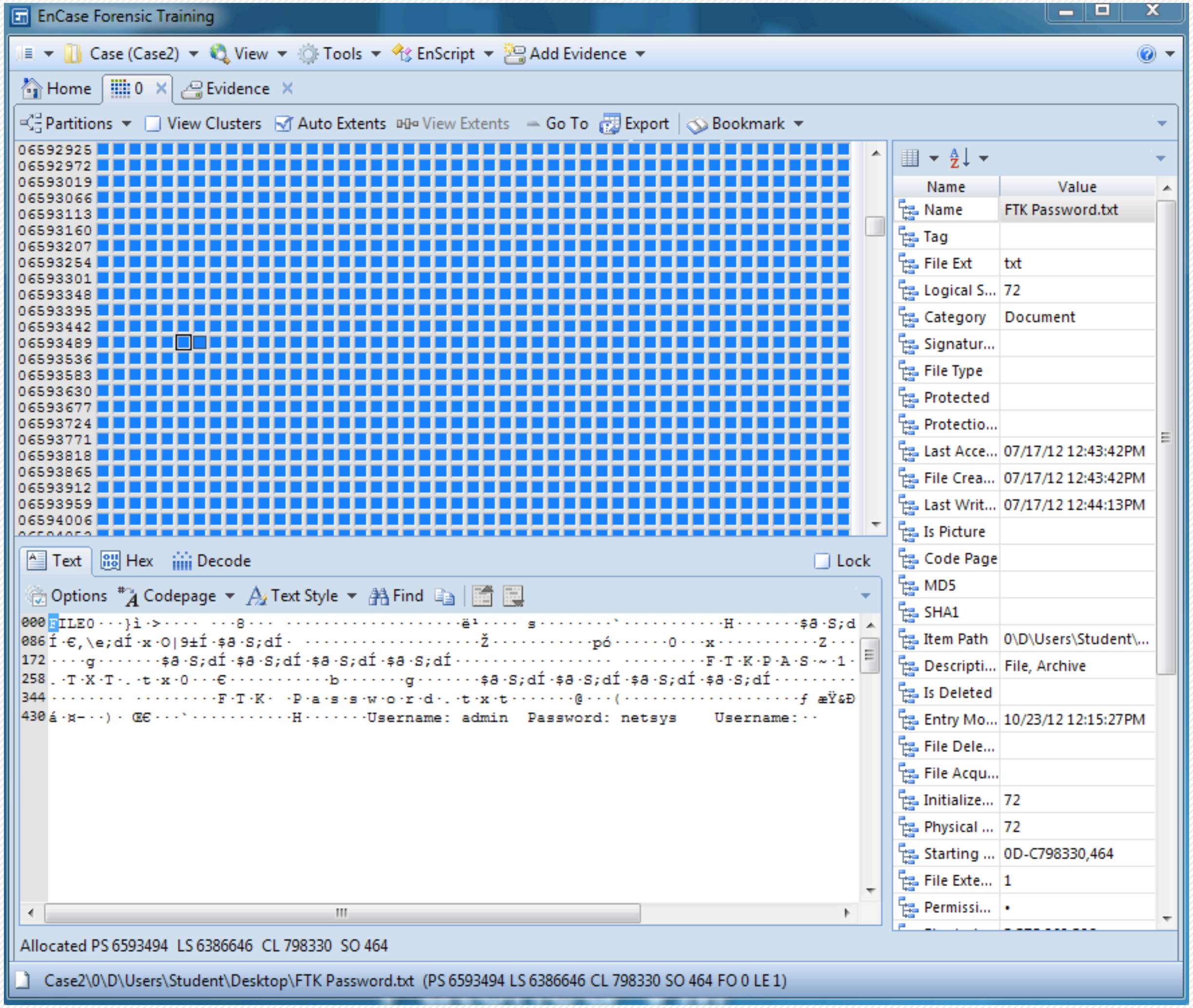
Master File Table (Cont'd)

- Each file record store attributes
  - $File Record Head (first 42 bytes)
    - MFT number, sequence #, Link count, file type, size, etc
  - $STANDARD_INFORMATION
    - MAC time, file characteristics (Hidden, System,…)
  - $FILENAME-Up to 255 characters
  - $DATA or associated cluster addresses



Standart information	File or directory name	Security descriptor	Data or index	

A small file that resides inside $MFT entry



```
000 FILE0···}ì·>····  ···8··· ·····················ë¹····  s·········`············H·······$ð·S;d
086 Í·€,\e;dÍ·x·O|9±Í·$ð·S;dÍ·  ···················Ž···········pó······0···x············Z···
172 ····g·······$ð·S;dÍ·$ð·S;dÍ·$ð·S;dÍ·$ð·S;dÍ··················  ·········F·T·K·P·A·S·~·1·
258 .·T·X·T·.·t·x·0···€···········b·······g·······$ð·S;dÍ·$ð·S;dÍ·$ð·S;dÍ·$ð·S;dÍ·········
344 ········  ··········F·T·K·  P·a·s·s·w·o·r·d·.·t·x·t·······@···(·····················ƒ æŸ&Ð
430 á·¤–··)·  Œ€···`············H·······Username: admin  Password: netsys    Username:··
```

Name	Value
Name	FTK Password.txt
Tag	
File Ext	txt
Logical S...	72
Category	Document
Signatur...	
File Type	
Protected	
Protectio...	
Last Acce...	07/17/12 12:43:42PM
File Crea...	07/17/12 12:43:42PM
Last Writ...	07/17/12 12:44:13PM
Is Picture	
Code Page	
MD5	
SHA1	
Item Path	0\D\Users\Student\...
Descripti...	File, Archive
Is Deleted	
Entry Mo...	10/23/12 12:15:27PM
File Dele...	
File Acqu...	
Initialize...	72
Physical ...	72
Starting ...	0D-C798330,464
File Exte...	1
Permissi...	•

Allocated PS 6593494 LS 6386646 CL 798330 SO 464

Case2\0\D\Users\Student\Desktop\FTK Password.txt (PS 6593494 LS 6386646 CL 798330 SO 464 FO 0 LE 1)

R·I·T

Master File Table (Cont'd)

- $Attribute list
  - If a file’s information is larger than one MFT record, it can point to other locations for additional MFT info.
- A flag for allocation status
  - flag is set to zero when the record is marked for deletion, or is unallocated

Master File Table (Cont'd)

- Each directory stores
  - Index entries for each file in the folder
    - File name, standard_information
- directory content
  - $INDEX_ROOT –contains the index entries
  - $INDEX_ALLOCATION (when cannot fit)
    - The addition data are stored in index buffers
    - $INDEX_ALLOCATION stores index buffers' locations.

MetaFiles

- $MFT
- $MFTMIRR
- $LOGFILE
- $VOLUME
- $ATTRDEF
- .
- $BITMAP
- $BOOT
- $BADCLUS
- $SECURE
- $UPCASE
- $EXTEND

http://resilientfilesystem.co.uk/refs-master-file-table

$BITMAP File

- Keeps track of cluster usage
- It uses one bit to record the status of each cluster on the volume
    - If a cluster is used, the corresponding bit is changed to one
    - Else, the bit is zero

What is happening…

- …when you create a file on an NTFS volume
  - The $BITMAP file will be modified
  - An MFT record will be created for the file
  - information in $File Record Head, $STANDARD_INFORMATION, $FILENAME and $DATA will be filled
  - An index entry will be inserted into its parent folder’s MFT record

What is happening…

- …when you delete a file on an NTFS volume
  - Its cluster references in the $BITMAP file are changed to zero
  - The MFT record for that file is marked for deletion
  - The index entry for the file is removed from its parent’s MFT record.
    - The entries below it are moved up, thereby overwriting the deleted entry.

ReFS-NTFS’s next generation

- Resilient File System (ReFS)
  - Introduced in Windows Server 2012 and Windows 8
  - Forensic Investigation of ReFS
    - https://redmondmag.com/articles/2016/02/01/stepping-up-refs.aspx
    - http://resilientfilesystem.co.uk/
- Address two major areas
  - The need for a larger size of storage
  - Providing continual reliability
    - Self-repairing, Handling hard drive failure
    - Copy-on-write (COW)
    - block cloning (ReFSv2) for Hyper-V