Advanced Computer Forensics

EnCE EnCase Forensics: The Official EnCase Certified Examiner Study Guide

Chapter 2

File Systems

Disk Basics - 1

- Hard Disk
  - Physical Device
    - Referred to as a numeric value (0, 1, 2 etc)
- Logical Volume
    - Referred to by letters A: floppy and C etc. volumes on physical disk
- Disk Preparation
    - Partition – create partition table
      - MBR – Master Boot Record
        - 4 partition limit
        - Disk size to 2 TB
        - No backup copy of partition table
      - GPT – GUID Partition Table
        - 128 partitions
        - 8 ZB hard drive
        - Has a copy of the partition table

Disk Basics - 2

- Disk Preparation
  - Partition – create partition table
    - Right Click -> choose partition style
      - Creates a new blank partition table in the ***first sector of the hard drive***
  - Create a new Volume
    - Right Click -> “unallocated” NEW VOLUME
      - Specify volume size
      - VBR – will be created
  - Format the Volume
    - Choose file system – Structure of how information will be written and recalled
      - FAT, NTFS, ExFAT
  - Sectors and Clusters
    - Based on File Systems used
      - Allocation blocks or clusters will be established with groupings of sectors
      - Sectors are commonly 512 bytes
      - Clusters keep addressing manageable in very large drives
      - Windows default – 8 sectors per cluster – 4096 bytes

FAT Basics - 1

- Directory Entry
  - Directory Entries do NOT contain any data
  - Data is contained in Data allocation units or Clusters
    - Clusters
      - 1 or more sectors
      - Smallest unit in which a file or directory can be stored
      - If a file is bigger than 1 cluster than it is allocated more than one
      - Directory only keeps track of the starting cluster/extent

FAT Basics -2

- File Allocation Table
  - Tracks
    - Sequence of clusters for a file
    - Allocation of clusters (used or available)
    - Bad clusters
  - Three Versions of FAT
    - FAT 12 – 12 bit entry (to mark clusters) – 4,084 maximum addressable clusters
    - FAT 16 – 32 bit entry – 65,524 maximum addresses
    - FAT32 – (4 highest bits are reserved so really only 28bits to address with – 268,435,456
      - MBR – imposes a 2 TB limit (67,092,481)
    - ExFAT – 32 bits with none reserved – 4,294,967,285

FAT Basics - 3

- Directory Entry
  - Every file and directory is referenced and described in a separate directory entry. (32 bytes in length – 0 logical size)
    - Name
    - Logical Size
    - Dates / Times
    - Starting Cluster/Extent (beginning cluster)
    - Extension
    - File Attributes

FAT Basics - 4

- Physical Layout of FAT
  - Reserved (Volume boot sector)
  - FAT (File Allocation Table)
  - Data Storage area (directory entries and data)

FAT Basics -5

- Reserved Area
  - Volume Boot Sector (boot sector VBS - VBR)
    - Size is defined in the boot sector data, but for FAT12 and FAT 16 usually only one sector
    - FAT 32 – length is defined in boot sector – usually 0, 1, and 2 with backups at 6, 7, and 8
      - Total reserved area is usually 32 sectors
      - FSINFO – File System Information (sector offset – SO) 48-49 – usually located in sector 1 between the boot and sector 2 which is a continuation of the bootstrap (it’s backup is usually in sector 7)
      - Meant to indicate how many free clusters and where the next one is for the OS

FAT Basics - 6

- Reserved Area
  - Volume Boot Record
    - Located at sector 0 – there are 4 distinct sections
      - Jump instruction (first 3 bytes)
        - Where to find the beginning of the OS bootstrap
      - BIOS Parameter Block
      - Boot code and error messages
      - Signature bytes – 510-511 (hex 55 AA)

Clustering

- Clustering combines a set of contiguous sectors and treats them as a single unit
- Called a cluster or file allocation unit
  - Instead of numbering the sectors, clusters were numbered
  - Allowed partition sizes up to 2 GB
- DOS, Windows 3.1, and the first version of Windows 95 all use FAT16
  - Newer OSs also support FAT16

[Image]

How FAT Works

- Windows looks for the first cluster marked 0000 (good & available for use)
- If the file fits in that cluster, *FFFF* is put in the status column
- If the file is larger than the cluster, Windows finds the next open cluster
  - That open cluster's number is put in the first status field to know where to link
  - Process continues until the file is fully stored
  - Last cluster's status field is marked *FFFF* (end-of-file)

Fragmentation

- Fragmentation occurs when files are spread across drives (not contiguous)
  - Individual files are broken into pieces that fit into a sector or cluster
  - The pieces are stored on the hard drive but may not be stored in contiguous clusters
- Fragmentation slows down the system during hard drive reads and writes
- Programs such as Disk Defragmenter or Speed Disk can be used to defragment files, folders, or both

Fragmented Files

- Take longer for a system to piece together and can impact performance

[Instructor Selected Image]

Slack

[Instructor Selected Image]

Directory Entries EnCase - 1

- Function of FAT
  - How a file is stored
    - **Keeps track of file location (cluster addressing)**
    - **Clusters are allocated or unallocated**
      - **Reading a File**
        - OS Looks in the parent directory reads info. Regarding the file
        - Starting cluster and length
        - After reaching the length is stops nothing else within the sector or cluster is considered.
          - Logical size – actual number of bytes a file takes up
          - Physical size – actual number of clusters a file occupies
      - **Reading larger files**
        - Ones that take up more than one cluster
        - Need to determine how many clusters (remember there can be no partial clusters)
        - Each cluster will have the next cluster within it's entry until the 0xFFFF (end of file)

Directory Entries EnCase - 2

- Effects of Deleting and Restoring
  - **Deleting a File**
    - An hex E5 is placed in the first character of the filename
    - Because the OS knows the # of clusters it marks each with a 0 (available for use) up to and including the EOF cluster
    - No data is lost and all clusters are available
      - Logical size – actual number of bytes a file takes up
      - Physical size – actual number of clusters a file occupies
  - **Restoring a file**
    - Reverse the process
    - Replace the 0xE5 with an underscore or a known character
    - Go to the FAT entry for the starting cluster enter the next clusters address and then EOF once you get to the end of the file

Directory Entries EnCase - 3

- Slack Space
  - **RAM Slack/Sector Slack**
    - End of data until the end of the sector
    - 95B and later is filled with zeros
  - **File Slack**
    - End of the written sector to the end of the cluster
  - **Viewed in red within EnCase**
  - **Data in a logical file is black as a default**
    - Information within the File Slack could contain data from previous files

NTFS Basics - 1

- New Technology File Systems
  - **$MFT Tracks**
    - Filename
    - Starting cluster
    - Length of file and other metadata
    - Clusters used
    - Allocated and Unallocated space
  - **$MFT File**
    - Entries of all files and folders on a HD (partition table)
    - Database entry for every file and directory in the partition (even an entry for itself)
    - Fixed length of 1,024 bytes
    - Each entry has a header followed by attributes (if 480 or less the file could be contained within the entry record) Resident Data – cluster runs are stored here if the file is not a resident file
  - **Deleting a File**
    - Sets a flag indicating the file is not in use
    - Data runs are usually left intact, thus file retrial is more reliable than with FAT
    - $MFT can grow, but never shrinks – therefore deleted entries can quickly be overwritten
    - $BITMAP – tracks allocation of clusters

NTFS Basics - 2

- New Technology File Systems
  - **Volume Boot Record**
    - Created when a partition is formatted with NTFS
    - 16 sectors are reserved for its use – usually only 8 are used for data
    - Bytes 3-6 are NTFS
    - Backup of the VBR is located at the last sector of the partition
    - File system data is contained in files
      - $MFT – similar to the FAT directory
      - $Bitmap – similar to the FAT1 and FAT2 (allocated/unallocated space)
    - 1 bit for each cluster in the partition (0 unallocated 1 allocated)
  - **Deleted File**
      - $Bitmap must be updated to mark the cluster as unallocated
  - **Formatting**
    - Windows 7 full format will “wipe”
    - Windows 7 “quick” default – will not

CD File Systems - 1

- ISO 9660
    - **Restrictions**
        - Uppercase characters, number, and underscore for file names
        - 8.3 naming convention
        - Directory names 8 characters
        - Nesting sub-directories limited to 8
        - Files are contiguous
    - **Updates**
        - Up to 30 characters in a name
        - Files don't have to be contiguous
    - **Joliet**
        - Up to 64 characters for naming
        - Directories can have extensions
        - You will see both an ISO 9660 directory and Joliet directory
            - Two separate directory structures pointing to the same data

CD File Systems - 2

- **UDF**
  - Uses Packet writing to write information in increments for CD-R/RW
  - 255 characters for files
  - Need the drivers to read a UDF format (sometimes causes issues for examiners)
  - Can use 3rd party to convert to Joliet
- **HFS**
  - Mac
  - Unreadable on a PC
  - Hybrid directories – HFS and Joliet pointing to the same data
- **Rock-Ridge**
  - For Unix
  - Also has an ISO9660 directory that can be read
- **EnCase - CD Inspector**
  - Can even read "tough" CD formats – images them with a .zip extension to be opened within EnCase

exFAT

- **ExFAT**
  - Designed for flash media might be referred to as FAT64
  - File size limit of 16 EiB – (1 exbibyte=2^{60} bytes)
  - Great for examiners who need space
- **OS Support**
    - Vista SP 1
    - XP Server 2003 – SP 2 or higher and patch KB955704
    - Server 2008
    - Windows 7
    - OS X Snow Leopard (10.6.5)
    - OS X Lion
    - Linux (working on it)
- **4 Regions of exFAT**
  - Main boot region
  - Backup boot region
  - FAT region (normally only FAT1 is found unless TFAT is configured then FAT2 will be found
  - Data region

DOL Disclaimer and CCBY

This workforce product was funded by a grant awarded by the U.S. Department of Labor’s Employment and Training Administration. The product was created by the grantee and does not necessarily reflect the official position of the U.S. Department of Labor. The Department of Labor makes no guarantees, warranties, or assurances of any kind, express or implied, with respect to such information, including any information on linked sites, and including, but not limited to accuracy of the information or its completeness, timeliness, usefulness, adequacy, continued availability or ownership.