**Week 7 NTFS Analysis**

**Everything a file in NTFS:**

* **Most common file system in use by all recent versions of Windows**
* **Important data allocated to files**
* **No specific layout like other file systems, entire system considered a data area, any sector can be allocated to a file**
* **Only consistent layout = first sectors of volume contain the boot sector and boot code**

**PBR:**

* **At first sector of NTFS partition**
* **MFT Logical cluster at offset 48-55**

**MFT:**

* **Primary source of metadata in NTFS**
* **All file and directory have an entry in MFT**
* **Info such as sizeinbytes, attributes, content**
* **Each NTFS volume has own MFT at any position in vol**
* **To determine position, look at byte offset 48-55 in PBR**

**Endianness**

* **Attribute of system indicating if int are from left or right o right to left**
* **Big endian = left justified**
* **Little endian = right justified**

**In encase data is presented in big endian (hence CT should all be big endian?). Must convert to little endian.**

**MFT Entry**

* **All same structure. 1024 Bytes in size**
* **At offset 0x38 is start of string of attributes**
* **Header describes properties of record; attribute describes aspects of file (e.g. name to data)**
* **Header has info like Record Type (file/directory), Record #, Active/inactive flag.**
* **Header and attributes are essential elements of MFT entry**
* **1st 4 bytes combines to form ‘FILE’ even if folder**
* **‘BAAD’ = unusable entry**
* **MFT records are reused never deleted. Once deleted, file flagged, made available for re-use in future file or directory allocations**

**NTFS System Files**

* **1st 12 entries in MFT used by system files**
* **All start with $ and are hidden**
* **Perform raw disk access using forensic tools (Encase) to access these files**

**Change Logs**

* **NTFS maintains logs for files and directories**
* **Logs used to reverse file system operations if failure**
* **For FI, evidence for file system activity, good if attacker has attempted to delete file**
* **$Logfile is that file**
* **Tracks transactions that change the structure of volume**
* **Entires in file Contain same attributes in MFT**
* **Circular and can roll over frequently**

**NTFS Attributes**

* **Each attribute = varying-length stream id by 4-byte attribute type at offset 0x0**
* **Terminated with 0xFFFFFFFF. First attribute normall = 0x00000010 (Little Endian) $Standard\_Information**

**Typical File Record**

* **$Standard\_Information and one $File\_Name mandatory for all files and directories**
* **File name has special characters / >8 characters. 2nd $File\_Name attribute for DOS compatible name**
* **For directories, content is an index that lists all their files**
* **$Index\_Root provides header for index, all entries fit into $Index\_root if few enough**
* **If too long, $Index\_Allocation used to track which cluster index will reside at**

**Analysing Timestamps**

* **Most important metadata stored in MFT**
* **1 MFT entry at least 2 sets of attributes containing MACE (Entry Modified) attributes**
* **1 within $Standard\_Information other $File\_Name**
* **File properties shows $Standard\_Information timestamps (does not include entry modified)**

**Resident and Non-Resident attributes**

* **Used when content of attributes (such as $Data) is too large**
* **Extension MFT records used to hold some attributes**
* **Base MFT record to reference the file, $Attribute\_List references all extension records of file.**
* **Attributes content held entirely within attribute (base or extension MFT record) = Resident**
* **Not held entirely within attribute = Non-resident (attribute hold external cluster information)**
* **At offset 0x08 non-resident flag:**
  + **Resident = 0**
  + **Non-resident = 1**
* **$Data most likely non-resident**
* **Files too long will have data held in external clusters, $Data will hold external cluster information thus, non-resident**

**$Bitmap**

* **Map of logical clusters (use and not used).**
* **Purpose = Find free space for file allocation**
* **Also, a MFT record attribute to map MFT record, 1 = in-use, 0 = out of use record**

**$Boot file (VBR)**

* **16 sectors at logical sector 0 in active partition with VBR and bootstrap code = $Boot**

**Week 6**

**Primary Partitioning:**

* **Subdividing of a hard disk**
* **1 hard disk up to 4 primary partitions**
* **If multiple OS, each one assigned to its own primary partition**

**Volume vs Partition:**

* **Partition = Collection of (physically) consecutive sectors**
* **Volume = Collection of (logically) addressable sectors**
* **Partition created on single hard disk, auto becomes a volume and assigned a drive letter**

**Extended Partitioning:**

* **1 of the 4 primary partitions can be subdivided into multiple partitions (volumes)**
* **Allowing 2 or more volumes to exist within one partition**
* **Referred to as extended partition**

**What is File System:**

* **Users to store data in hierarchy of files and directories**
* **Organized in a way where computer knows where to find structural & user data**
* **NTFS, FAT, ReFS**

**Purpose of File System**

* **Track file/folder (object) name**
* **Track starting cluster of the object**
* **Track fragmentation of object (non-contiguous allocation clusters)**
* **Track cluster status on volume**

**Master Boot Record (MBR):**

* **512-bytes boot sector**
* **Holds boot instructions**
* **1st sector (absolute sector 0) of a partitioned storage device i.e. hard disk**
* **Tells computer how hard drive is partitioned and to load OS**
* **Holds disk’s primary partition table with 4 entries (1st sector of the drive)**
* **0x1BE offset of start of partition table**
* **Each entry list starting sector, # of sectors, the type of file system (16 bytes long)**
* **Performs:**
  + **Scans partition table for active partition**
  + **Finds starting sector**
  + **Loads copy of boot sector (VBR) from active partition to memory**
  + **Gives control to the code in the boot sector**
* **End of MBR = 2-byte structure called signature word/end of sector marker set too 0x55AA**
* **If not, hard drive will not boot**

**Partition Table:**

* **Information to id the type, size and location of partition and active or not**
* **4 primary or 3 primary and one extended partition. Each entry = 16-bytes long**

**Graphical user interface, text, application

Description automatically generated**

**Graphical user interface

Description automatically generated with medium confidence**

**Winxp and Windows:**

* **XP starting sector 63**
* **63 = one track of a physical disk image, first partition selected to start after track 0**
* **Win 7 and later, 1st partition not C: but rather SYSTEM\_DRV starting at sector 2048**

**Week 4 Windows Artifacts**

**Why look through windows systems files and folders?:**

* **Folder structure can provide evidence by itself**
* **Determine Windows version by looking at directory structure**
* **Locating OS artifacts is important**

**User Files and Folder:**

* **Series of folders and files are created**
* **Both local logins and network authentication**

**Root Folder:**

* **Named after user’s login name**
* **Folders containing user’s documents configuration and environment data are subfolders**
  + **Application Data**
  + **Cookies**
  + **Desktop**
  + **Favourites**

**NTUSER.DAT (in ROOT):**

* **Contains files and preferences settings unique to each user**
* **Each user has one of this file**
* **A Registry file, contains registry settings for their account**
* **Updated by OS when user logs out, Last written time = possible user last log out**
* **Registry viewer to view these files**

**Recycle Bin:**

* **When a file is deleted, it goes into the Recycle Bin**
* **In Win 2000/XP, deleted file name, path, DaT of deletion stored in hidden file INFO2.**
* **Win 7/8/10/Vista no more INFO2 file**
* **$R contains the actual file content**
* **$I contains original filename, file size, DaT file deleted (gone to recycle bin)**

**Low Folders:**

* **Presence of low folders in** 
  + **Cookies**
  + **Temporary Internet Files (TIF)**
  + **History**
* **Protect system against malware (thus only presence in Internet Explorer)**
* **Data in these folders run at lowest possible level privilege**

**Cookies Folder:**

* **A piece of text stored on a computer by the web browser**
* **Created by website, placed on user’s computer**
* **Purpose:**
  + **Authentication**
  + **Storing site preferences**
  + **Shopping cart contents**
  + **Identifier for a server-based session**
* **Cookie Names =** [**username@domainname.txt**](mailto:username@domainname.txt)
* **Each cookie is contained in 1 file**
* **Index.dat manages all cookies. Stores info about each cookie and pointers to the cookie file**
* **Index.dat file contains dates, cookie itself contains dates**
* **Internal dates of cookie = last modified by website and its expiration date**
* **CookieView to view cookies**

**TIF:**

* **Stores files that are downloaded/cached from the internet**
* **Used to track web browsing history**

**Email folder**

**Online Comms tool: Skype & MS Team**

**Recent Folders:**

* **Contains link files that links to recently accessed files, folders and applications**

**Documents Folder:**

* **Where all compliant applications store user-generated data**

**Sent To Folder:**

* **Options a user has after right clicking an object and choosing Send To**
* **Default = Documents, Desktop**
* **Can create entries into the list of options**

**Temp Folder:**

* **Temporary files by Windows as different processes and programmes are running**
* **Often exact copies of files stored elsewhere on the computer, other times they are duplicates which are waiting to be processed by the computer**

**Desktop Folder:**

* **Items are intentionally placed or app-generated during installation**

**Week 3 (Evidence Extraction and Analysis)**

**Slack Space:**

* **Unused space between end of file and end of cluster**
* **When a file does not occupy the entire cluster, the remaining space is slack space**
* **2 areas to consider**
  + **Space between the end of the file and the sector in which the file ends (AKA RAM SLACK). Padded with data as determined by the OS**
  + **Second = Remaining sectors in the cluster that contains no data**
    - **OS dependent but can either be untouched or wiped**
    - **If no wiped, remnants of the previous file will be there. Expose sensitive information**

**Unallocated Space:**

* **Area of hard drive where data can be written**
* **Requires specialized computer forensic software to analyse**
* **Contains deleted files, or partially deleted files. When file is deleted, pointers removed but data remains in unallocated space until OS stores another files, overwriting the data**

**Allocated or Active Space:**

* **Area on hard drive that contains the OS and user data (files) that are easily accessible to the user**
* **File slack (slack space) is allocated space, typically contains data that hold parts of the previous file or random contents of memory**

**Evidence Examination:**

* **Purpose to extract and analyse digital evidence**
  + **FI should generate hash and have the hashes agree**
  + **On image copy to avoid tampering the original evidence**
* **Extraction = recovery of data from its media**
* **Analysis = Interpretation of the recovered data, put in a logical and useful format**

**Types of Evidence Extraction**

1. **Manual Inspection**
   * **Good on small size of data**
   * **Used to validate results**
   * **Increased confidence in results**
2. **File Signature**
   * **Look at file extensions**
   * **Attackers rename files to disguise them as another file type. Thus, only file extension is risky (may miss relevant evidence)**
3. **Timestamp and metadata**
   * **File Names, Sizes, MAC times, MD5 hashes**
   * **Can be on last accessed or modified time**
   * **Known files by comparing MD5 hash with database**
4. **String and keyword searching**
   * **List of keywords used to search evidence files**
   * **Common = Based on file names/pattern in name OR KW in content**
   * **Conditions like encoding can render string search useless**
   * **Important that FI know how string they are searching for is represented in the data they are searching**
5. **File Carving**
   * **Search unique sequence of bytes that corresponds with the header or first few bytes of file**
   * **Common file format = standardized header (marking beginning) and sometimes footer (mark end of file).**

**Common Analysis methods**

* **Understand what suspect was trying to do with the computer**

1. **Timeframe analysis**
   * **Determine what events occurred on a computer**
   * **Used to associate usage of computer to an individual, at the occurrence of event**
   * **2 Methods: 1. Review event logs**
   * **2. Review timestamps in file system metadata E.g., last modified, last accessed, created.**
2. **Data hiding analysis**
   * **Used to detect and recover data concealed on computer. Indicates knowledge, ownership or intent**
   * **Methods: 1. Steganography**
   * **2. Id mismatch of file headers against file extensions. Mismatch = malicious intent to conceal data**
   * **Access all password-protected, encrypted and compressed files. User has tried to conceal data from unauthorized users**
3. **App and File analysis**
   * **Look at apps and files**
   * **Inference provides details such as proficiency of user**
   * **To identify details:**
     + **Review file names for relevance or patterns**
     + **Examine file content**
     + **Relate the files to the application installed**
     + **Look for similarities between files. E.g., Relate internet history to cache or email files to email attachments**

**Collecting Evidence From OS:**

**Chat Logs**

* **Criminal activities could be facilitated by chat room discussions**
* **Most chat software keeps temporary log of conversations**

**Web Browsers**

* **Indirect evidence of specific crimes. E.g., Searching of how to build a bomb**
* **Check browser history. E.g., Internet Explorer browser history**

**Event Logs**

* **Generated by system-wide auditing & monitoring mechanisms built into Windows OS**
* **Review logs to identify:**
  + **Successful and Failed logon attempts and their origin**
  + **Track creation, stop and start of services**
  + **Track usage of specific apps**
  + **Track changes to user permissions**
  + **Track alterations to audit policy**
  + **Monitor event generated by installed applications**
* **Application Log**
  + **Activities related to user programs and applications**
  + **Info audited = error or info that an app wants to report (Host-based security tools like Antivirus and Intrusion prevention systems often record events to this log)**
* **Security Log**
  + **Logs:**
    - **Users log on and off attempts**
    - **Account creation**
    - **Changes to user privileges/credentials**
  + **Local/Group policy can config which security events are logged**
* **System Log**
  + **Logs event by a variety of core OS components**
  + **Include Windows service events, changes to system time, network config issues**
* **Logs are stored within registry key**