MODULE 4

CHAPTER 4

Windows and Unix Forensics Investigation

Syllabus

Investigating Windows Systems - File Recovery, Windows Recycle Bin Forensics, Data Carving, Windows Registry Analysis, USB Device Forensics, File Format Identification, Windows Features Forensics Analysis, Windows 10 Forensics, Cortana Forensics

Investigating Unix Systems - Reviewing Pertinent Logs, Performing Keyword Searches, Reviewing Relevant Files, Identifying Unauthorized User Accounts or Groups, Identifying Rogue Processes, Checking for Unauthorized Access Points, Analyzing Trust Relationships

This chapter discusses the importance of operating systems forensics investigation. It focuses on the various features and services of the Windows Operating Systems to apply forensics investigation. Second section discusses about Unix Operating System Investigation based on Reviewing Pertinent Logs, Performing Keyword Searches, Reviewing Relevant Files, Identifying Unauthorized User Accounts or Groups, Identifying Rogue Processes, Checking for Unauthorized Access Points and Analysing Trust Relationships. .

4.1 INVESTIGATING WINDOWS SYSTEMS

GQ. Why Windows Forensic Investigation is required?

The Operating System (OS) of a computer is the collection of applications that interfaces with the hardware and manages the operation of its various parts, including the hard drive, CPU, memory, and numerous others. Because an OS controls file management, memory management, logging, user administration, and many other crucial functions, forensic investigation on an OS is accessible.

Windows Forensic Investigation focuses on 2 things :

(i) Extensive research on the Windows Operating System.

Microsoft created the widely used OS known as Windows. Windows makes use of the file systems FAT, exFAT, NTFS, and ReFS. Analyzing the crucial Windows places can help investigators find evidence.

(ii) Looking into Windows System Artefacts.

Items known as Windows artefacts include data on the actions taken by Windows users. Different artefacts have different types of information and places depending on the operating system. Sensitive data is acquired and reviewed in Windows artefacts during forensic examination.

Following are some artefacts of the windows operating system :

(1) **Recycle Bin :** This contains files that the user has deleted. A copy of the files that are deleted by the user is kept in the recycle bin. This method is known as "Soft Deletion." Recovering data from the recycle bin can provide solid proof.

(2) **Registry :** A database of values and keys called the Windows Registry contains information that forensic investigators can utilize.

(3) **Browsers :** Cookies, navigation history, website caching, and download history are just a few of the data that may be found in online browsers. During a forensic inquiry, all of this information could prove to be a great source.

(4) **Thumbs.db Files :** These save thumbnails of images that contain useful data.

(5) **Windows Error Reporting Forensics :** We may receive artefacts from this feature that show proof of programme execution. If a dangerous application fails to execute properly.

(6) **Remote Desktop Protocol Cache :** Attackers occasionally utilise RDP to move around a network when using the Windows "**mstsc**" client when connecting over RDP.

Whenever we connect to a system, cache files are automatically created and contain rarely modified sections of the system's screen. As a means of improving performance, these cached files provide us with valuable forensic evidence.

4.1.1 File Recovery

- The analysis of deleted files is a vital step in any type of digital forensic investigation. To be a successful digital forensic investigator, you must comprehend how Windows deletes files, where you can still locate them after they have been erased, and how to analyze these files (for example, by retrieving deleted files' metadata to support a criminal investigation). In order to help with the settlement of the current case, this section will provide a list of several tools and techniques for retrieving critically important files and document fragments.
- Due to the fact that file records are not truly erased and continue to exist on disc until they are replaced by new file records, it may be possible to harvest MFT file records from deleted files. Since **NTFS overwrites lost file records** before allocating new space for the MFT, the likelihood of successfully recovering deleted (marked for deletion) file records declines over time. The file name and the usual metadata, such as MAC timings, are contained in the file records. Such details might be quite helpful in an investigation.
- The data-runs for the file's **non-resident data** will likewise be known and easily recoverable if the file record is recovered. If the file is not fragmented, a physical disc search without the file record could still find and recover the lost file.
- Even if only a portion of a fragmented file is retrieved, it may still contain crucial information. Fragmented files are particularly difficult to recover completely through a physical search.

- Two alternatives are available for the analysis and file recovery of files stored in windows compatible file systems like FAT32 and NTFS: first, personally doing the investigation and second, utilizing programmes like "**Atopsy**."
- Since Autopsy is an automated forensic tool, it will automatically extract the data that is most frequently used in digital forensic analysis from any forensic images that are added to it. For the purpose of studying a given data source (such as a forensic image), Autopsy provides default ingest modules; it is up to you to select or deselect any module during the case creation wizard.

4.1.2 Windows Recycle Bin Forensics

- The Windows recycle bin holds files that have been removed by users but are still present on the system. It was initially introduced in **Windows 95.** For instance, Windows transfers the subject file to the recycle bin without really deleting it when a user deletes a file (using the conventional delete key on the keyboard after selecting the target file OR **picking a file, right-clicking it, and selecting "Delete" from the pop-up menu**).
- This is how Windows operates by default, but a user can change the settings for the recycle bin to permanently remove files without putting them there. In addition, some users press and hold the Shift key when deleting a file in order to perform the same thing.
- The default action of Windows when a user deletes a file is to place it in the recycle bin. The locations and file names of the recycle bin vary between Windows versions. Deleted files for Windows XP are kept in the "Recycler" folder in the root directory where Windows is installed often saved the "C:" disc but it can be different in higher versions.
- One or more folders can be found inside the "Recycler" folder; these folders are named using the unique security identifier (SID) for each user (e.g., S- 1-5-21- 2602240047-739648611- 3566628919-501); if a system has multiple users, each one will have a separate folder that contains the deleted files associated with that user account.
- Each user's recycle bin folder also contains another crucial file called "INFO2," which provides an index of all the files that the user has previously deleted. Additionally, it includes metadata about each deleted file, such as the file's original path, size, and deletion time.

- The Windows recycle bin has a finite amount of storage space. The default setting for the recycle bin in Windows is 10% of the available hard drive space;
- Let's test out deleting a file and utilizing Windows 10 and a free programme called $I Parse to analyze it.
- Open a command-line terminal and change the working directory to the **$Recycle.Bin** folder on the C: disc using the CD command. Use the DIR command with the /a switch to display the contents of the folder (to display hidden system files). Fig. 4.1.1 shows how these commands are displayed.

```
c:\$Recycle.Bin>dir /a
 Volume in drive C has no label.
 Volume Serial Number is 724F-8982

 Directory of c:\$Recycle.Bin

11/13/2017  10:28 PM    <DIR>          .
11/13/2017  10:28 PM    <DIR>          ..
01/28/2017  02:24 AM    <DIR>          S-1-5-18
01/27/2017  09:49 PM    <DIR>          S-1-5-21-2602240047-739648611-3566628919-1000
09/25/2018  03:26 PM    <DIR>          S-1-5-21-2602240047-739648611-3566628919-1001
05/16/2017  02:32 PM    <DIR>          S-1-5-21-2602240047-739648611-3566628919-500
11/13/2017  10:28 PM    <DIR>          S-1-5-80-3477044410-376262199-2110164357-2030828471
               0 File(s)              0 bytes
               7 Dir(s)  89,426,808,832 bytes free

c:\$Recycle.Bin>
```

(1D1)**Fig. 4.1.1 : $Recycle.Bin contents under Windows using DOS prompt**

- Fig. 4.1.1 shows that the $Recycle.Bin has four subfolders, each of which is a SID subfolder and represents the SID of the person who deleted the file. When a user sends a file to the recycle bin for the first time, each subfolder is generated.
- Now, we need to execute the following command to find out the name of the user account that is the owner of a particular SID subfolder:

  wmic useraccount get name,sid

- We can now determine which SID subfolder is located in the Recycle by having this display all user accounts on the target system. Bin is the target user's asset. (Refer Fig. 4.1.2)

```
Directory of c:\$Recycle.Bin

11/13/2017  10:28 PM    <DIR>          .
11/13/2017  10:28 PM    <DIR>          ..
01/28/2017  02:24 AM    <DIR>          S-1-5-18
01/27/2017  09:49 PM    <DIR>          S-1-5-21-2602240047-739648611-3566628919-1000
09/25/2018  03:26 PM    <DIR>          S-1-5-21-2602240047-739648611-3566628919-1001
05/16/2017  02:32 PM    <DIR>          S-1-5-21-2602240047-739648611-3566628919-500
11/13/2017  10:28 PM    <DIR>          S-1-5-80-3477044410-376262199-2110164357-2030828471-4165405235
               0 File(s)              0 bytes
               7 Dir(s)  89,426,808,832 bytes free

c:\$Recycle.Bin>wmic useraccount get name,sid
Name                SID
Administrator       S-1-5-21-2602240047-739648611-3566628919-500
DefaultAccount      S-1-5-21-2602240047-739648611-3566628919-503
Guest               S-1-5-21-2602240047-739648611-3566628919-501
MSSQLSERVER01       S-1-5-21-2602240047-739648611-3566628919-1009
MSSQLSERVER02       S-1-5-21-2602240047-739648611-3566628919-1010
MSSQLSERVER03       S-1-5-21-2602240047-739648611-3566628919-1011
MSSQLSERVER04       S-1-5-21-2602240047-739648611-3566628919-1012
MSSQLSERVER05       S-1-5-21-2602240047-739648611-3566628919-1013
MSSQLSERVER06       S-1-5-21-2602240047-739648611-3566628919-1014
MSSQLSERVER07       S-1-5-21-2602240047-739648611-3566628919-1015
MSSQLSERVER08       S-1-5-21-2602240047-739648611-3566628919-1016
MSSQLSERVER09       S-1-5-21-2602240047-739648611-3566628919-1017
MSSQLSERVER10       S-1-5-21-2602240047-739648611-3566628919-1018
MSSQLSERVER11       S-1-5-21-2602240047-739648611-3566628919-1019
MSSQLSERVER12       S-1-5-21-2602240047-739648611-3566628919-1020
MSSQLSERVER13       S-1-5-21-2602240047-739648611-3566628919-1021
MSSQLSERVER14       S-1-5-21-2602240047-739648611-3566628919-1022
MSSQLSERVER15       S-1-5-21-2602240047-739648611-3566628919-1023
MSSQLSERVER16       S-1-5-21-2602240047-739648611-3566628919-1024
MSSQLSERVER17       S-1-5-21-2602240047-739648611-3566628919-1025
MSSQLSERVER18       S-1-5-21-2602240047-739648611-3566628919-1026
MSSQLSERVER19       S-1-5-21-2602240047-739648611-3566628919-1027
MSSQLSERVER20       S-1-5-21-2602240047-739648611-3566628919-1028
Nihad               S-1-5-21-2602240047-739648611-3566628919-1001
WDAGUtilityAccount  S-1-5-21-2602240047-739648611-3566628919-504
```

(1D2)**Fig. 4.1.2: Getting specific SID owner's subfolder within $Recycle.in**

- We can use the CD command to access the target account's recycle bin once we know which one it belongs to. To view its contents, use the DIR command with the /a switch (see Figure Refer Figure 4.1.3).

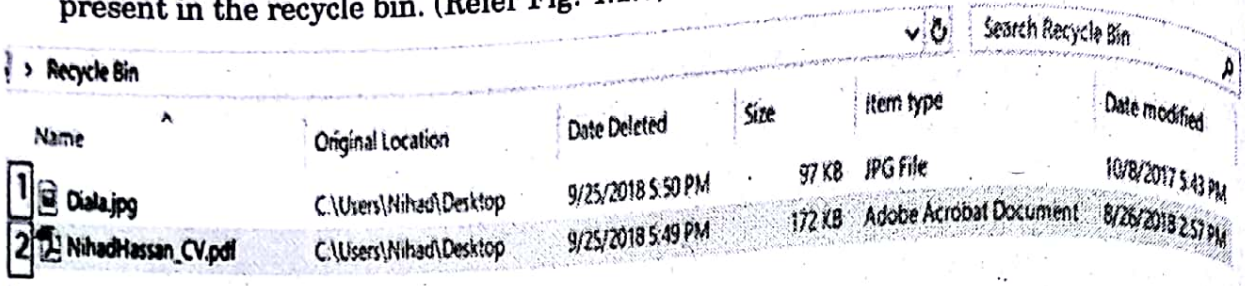
```
c:\$Recycle.Bin>cd S-1-5-21-2602240047-739648611-3566628919-1001

c:\$Recycle.Bin\S-1-5-21-2602240047-739648611-3566628919-1001>dir /a
 Volume in drive C has no label.
 Volume Serial Number is 724F-8902

 Directory of c:\$Recycle.Bin\S-1-5-21-2602240047-739648611-3566628919-1001

09/25/2018  05:50 PM    <DIR>          .
09/25/2018  05:50 PM    <DIR>          ..
09/25/2018  05:50 PM                94 $IOH0BYK.jpg
09/25/2018  05:49 PM               112 $IY2KKQM.pdf
10/08/2017  05:43 PM            98,662 $ROH0BYK.jpg
08/26/2018  02:57 PM           176,107 $RY2KKQM.pdf
01/27/2017  09:51 PM               129 desktop.ini
               5 File(s)        275,104 bytes
               2 Dir(s)  89,707,253,760 bytes free

c:\$Recycle.Bin\S-1-5-21-2602240047-739648611-3566628919-1001>_
```

Metadata Files - Begin with $I

Actual deleted files data - Begin with $R

(1D3)**Fig. 4.1.3 : viewing the contents of the target recycle bin reveals two deleted files**

- The metadata file and the actual data (recoverable data) of each deleted file are both present in the recycle bin. (Refer Fig. 4.1.4)



(1D4)**Fig. 4.1.4 : The recycle bin contains two deleted files.**

- Let's now look into the deleted file's metadata, often referred to as index files (begin with $I), in Windows recycle bin and subsequently using a tool called $I Parse. Follow these instructions to use this tool:

1. Visit "https://df-stream.com/recycle-bin-i-parser/"

   Download the application, and then extract the files.

2. We must first remove the recycled file metadata before using this utility file. Enter the following into the command prompt to accomplish this

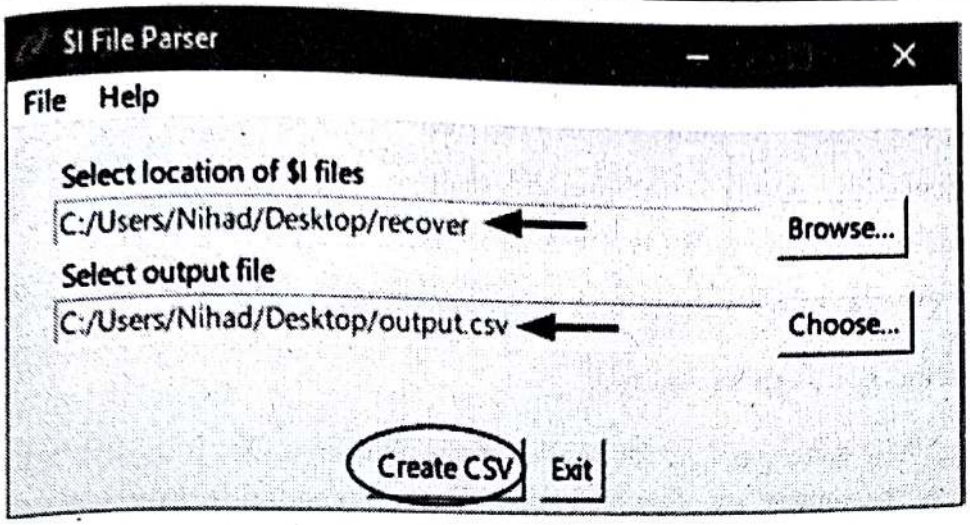
   "copy $I* \users\nihad\desktop\recover" (Refer Fig. 4.1.5)

```
c:\$Recycle.Bin\S-1-5-21-2602240047-739648611-3566628919-1001>copy $I* \users\nihad\desktop\recover
$IOH0BYK.jpg
$IY2KKQM.pdf
        2 file(s) copied.

c:\$Recycle.Bin\S-1-5-21-2602240047-739648611-3566628919-1001>
```

(1D5)**Fig. 4.1.5 : Copy the metadata files from recycled files and place them in a different folder for analysis**

3. Run the $I Parse application, go to "File menu ➤ Browse..., and select the folder that contains metadata files".

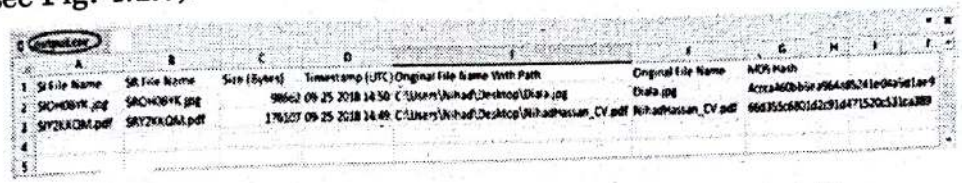
4. Click the Choose... button on the main application menu to choose where to save the output file, which will be a CSV file with the parsing results inside of it (see Fig. 4.1.6).



(1D6)**Fig. 4.1.6: Parse every file of metadata in the target directory**

5. After clicking on "Create CSV" is returns the CSV file of all parsed files.

   Open created CSV to view a list of all recycled files names in the target recycle bin along with file metadata information (i.e. original path, deletion date/time, and file size) (see Fig. 4.1.7).



(1D7)**Fig. 4.1.7 : CSV shows that information of recycled files.**

4.1.3 Data Carving

- Data carving is a sophisticated form of data recovery that is typically employed in digital forensic investigations to **extract a specific file from unallocated space** (raw data) utilizing the file's header and footer information without the aid of any file system structure (e.g., MFT).
- When the file system that originally organized these files on the hard drive is absent or defective, data carving may be the only way to recover crucial evidentiary files and file fragments in a criminal investigation.
- When removing a file(s) from a stream of network data that was captured, carving is also required.
- We should be aware that skilled forensic investigators can use data carving techniques to extract (recover) structured data and thus a file like a document or photo from non-structured or raw data

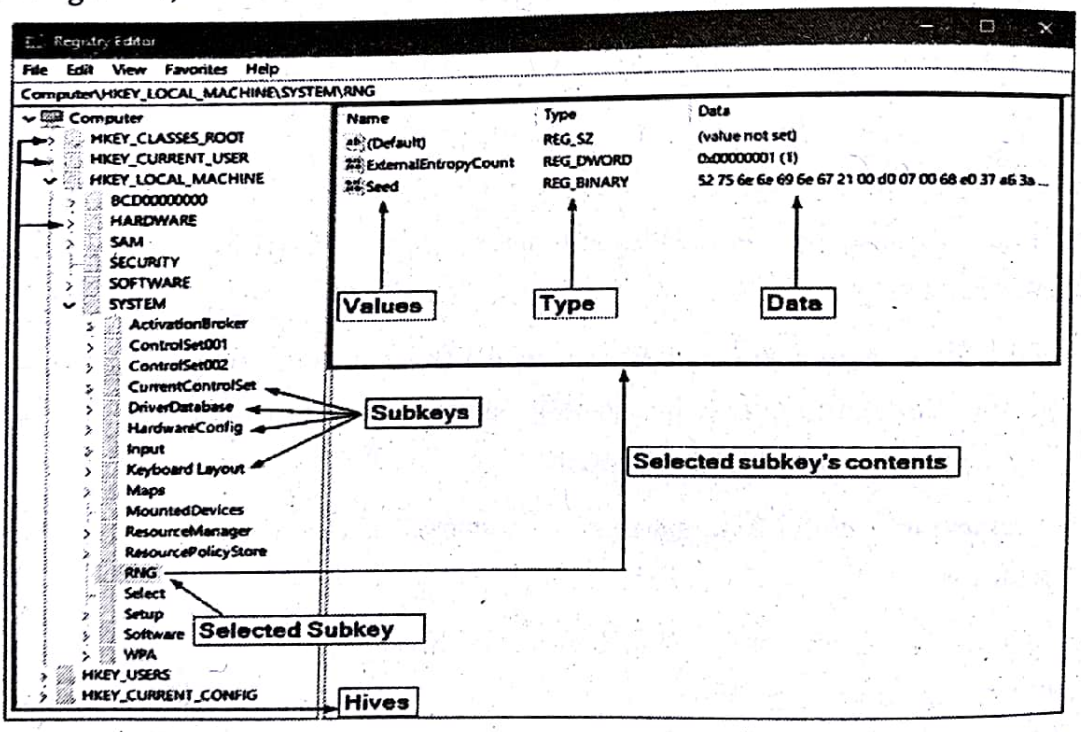
- Only a Hex editor can be used for file carving, but there are other tools that can help examiners. The following resources are available for free file carving:
  1. Foremost (http://foremost.sourceforge.net)
  2. Scalpel (https://github.com/sleuthkit/scalpel)
  3. Jpegcarver (www.seedstech.net/jpegcarver)
  4. List of data recovery (including some file carving)

4.1.4 Windows Registry Analysis

- The registry is considered as the brains of the Windows OS; it contains the vital data required for the operation of both the operating system and installed programmes.
- The Windows registry is a rich source of information that may be very helpful for any digital forensic investigation because almost every action taken by a Windows user is saved in it in some way.

4.1.4(A) Architecture of Registry Analysis

- The registry is a hierarchical database that contains user preferences, computer and application usage history, and Windows system configuration settings for hardware, software, and the operating system.
- Each node in the tree that makes up the structure of registry data is referred to as a key. In addition to data values, a key may also include other keys (subkeys) (see Fig. 4.1.8).



(1D8)Fig. 4.1.8 : Windows registry structure

Windows Registry Root Folders (Hives) provides following information

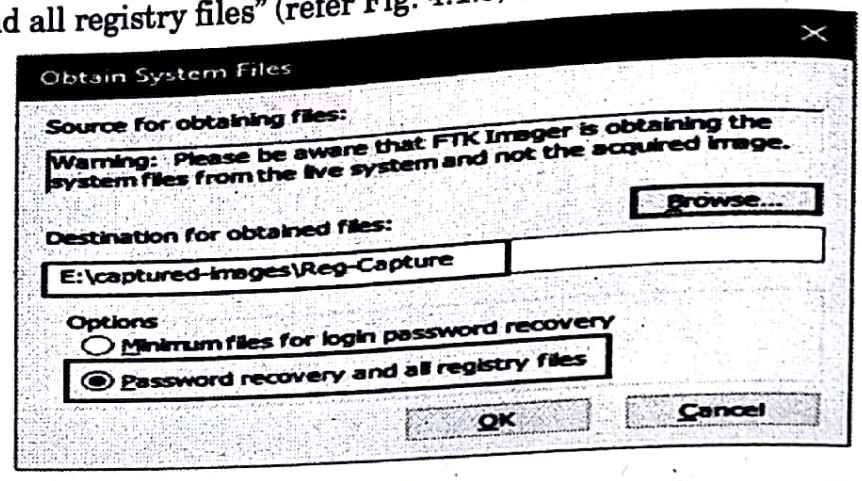
- **HKEY_CLASSES_ROOT** : Contains file association information (configuration information that tells Windows which program to use to open files).
- **HKEY_CURRENT_USER** : Stores configuration information (related to the installed software and operating system) to the currently logged-in user.
- **HKEY_LOCAL_MACHINE** : Contains the majority of the configuration information for currently installed programs and the Windows OS itself.
- **HKEY_USERS** : Contains configuration information (user profiles) for all active users on the system.
- **HKEY_CURRENT_CONFIG** : Does not store information itself; instead, acts as a pointer to another registry key (HKEY_LOCAL_MACHINE\System\CurrentControlSet\Hardware Profiles\Current). This hive keeps information about the hardware profile of the local computer system.
- There are two ways that digital forensic investigators can check the Windows registry :
  1. The forensic picture contains the register. As you would while using Windows File Explorer to browse files and directories, the computer forensic tool will be utilized to study registry files in this manner.
  2. Live analysis (for instance, when starting from the forensic image of the suspect). Using the Windows built-in registry editor, you can access the registry as you would on any other computer when using this approach.

4.1.4(B) Architecture of Registry Analysis

- When acquiring the target computer's system drive, computer forensics applications will also acquire Windows registry files. Additionally, we can take a live system and only take the registry files out, storing them separately for subsequent study (this is known as a "**Registry Image**").
- The example below demonstrates how to use AccessData FTK Imager to accomplish this. Use FTK Imager to obtain the target Windows machine registry by performing the following steps:
  1. Download AccessData FTK Imager and transfer it into your USB thumb drive.
  2. Attach the USB drive that contains FTK Imager to the suspect

Machine, open FTK Image, and go to File menu ➤ Obtain Protected Files...

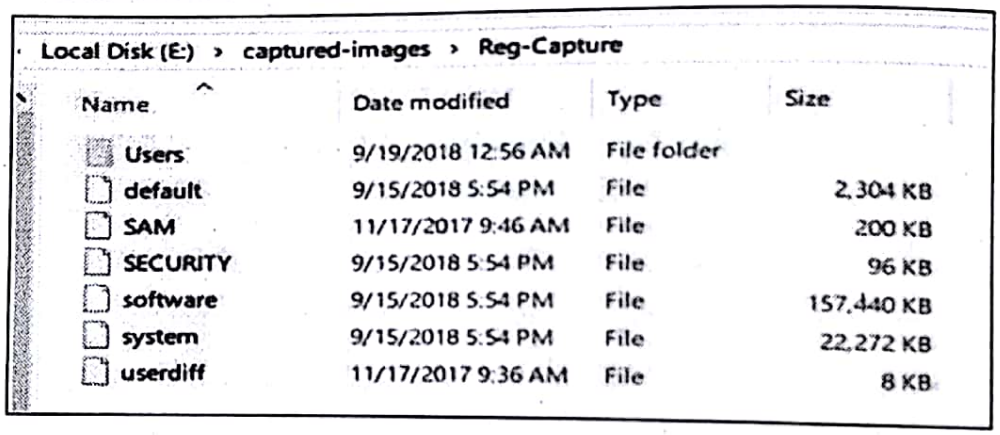
3. A new dialog appears; select where you want to store obtained files, "Password recovery and all registry files" (refer Fig. 4.1.9). Finally, click the "OK" button.



(1D9)**Fig. 4.1.9 : Acquiring target Windows registry database using FTK Imager**

4. A status window will appear showing registry files' export progress.

- Now, open the directory where we have saved our registry files to see the resultant files; we should see the five files and one folder (refer Fig. 4.1.10)



(1D10)**Fig. 4.1.10 : A registry forensic image captured using AccessData FTK Imager**

We have successfully exported registry of target machine, Now it is possible to use different forensics tools to analyze it.

4.1.4(C) Registry Examination

- We'll presume that in order to do various forensic tests on the system, we booted it up using a suspicious forensic image.

☞ Automatic Startup Locations

- Windows includes a feature that enables applications to start up immediately as the operating system boots; this functionality is essential for some applications, such as

antivirus software, Which must run first to prevent any harmful malware before Windows is fully loaded.

- **Keyloggers and botnets** are examples of malicious software that can add entries to the Windows registry so that they run every time Windows boots. A record of every application that Windows has booted is kept in the Windows registry. The registry keys given below include a list of the applications that are auto-booted.

(1) HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services

(2) HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Explorer \ShellServiceObjects

(3) HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion \RunServicesOnce

(4) HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion \RunOnce

(5) HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion \Policies\Explorer\Run

(6) HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run

(7) HKEY_LOCAL_MACHINE\Software\Microsoft\WindowsNT\CurrentVersion \Windows

(8) HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion \ShellServiceObjectDelayLoad

(9) HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion Explorer\SharedTaskScheduler

(10) HKEY_LOCAL_MACHINE\Software\Microsoft\Active Setup\Installed Components

(11) HKEY_LOCAL_MACHINE\Wow6432Node\Microsoft\ActiveSetup\Installed Components

(12) HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion \Explorer\SharedTaskScheduler

(13) HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Drivers32

(14) HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunOnce

(15) HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run

(16) HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\Run

(17) HKEY_CURRENT_USER\Software\Microsoft\WindowsNT\CurrentVersion\Windows\load

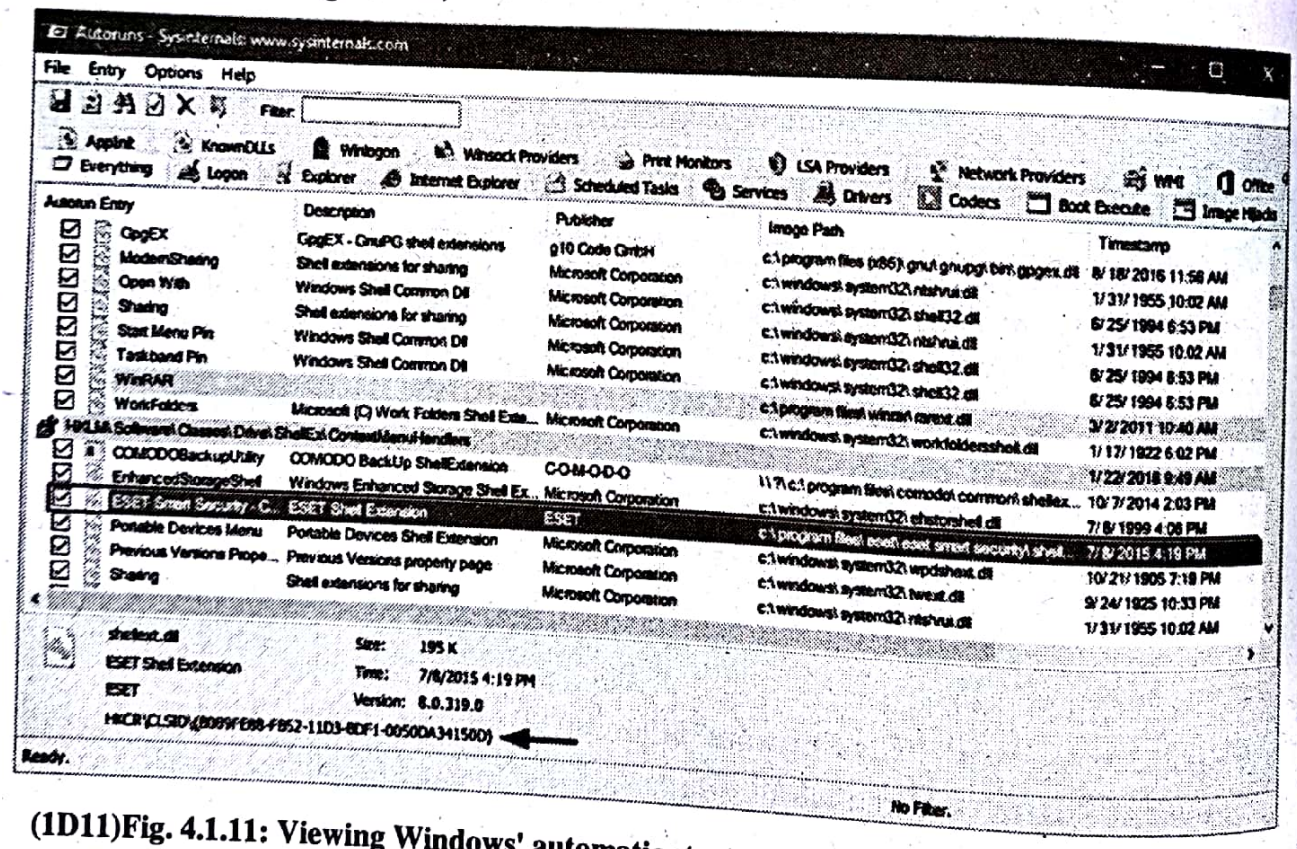
(18) HKEY_CURRENT_USER\SOFTWARE\Wow6432Node\Microsoft\Windows\CurrentVersion\Run (64 bit systems only)

(19) HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunServices

(20) HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunServicesOnce

(21) HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunOnceEx

Microsoft offers a portable tool called Autoruns that can look at every Autorun application. (Refer Fig. 4.1.11)



(1D11)Fig. 4.1.11: Viewing Windows' automatic startup applications and related registry key using Autoruns from Sysinternals

- Investigating startup applications can be very useful for forensics in various situations; for instance, malware can take control of a suspicious machine and launch **DDoS assaults** utilizing it secretly.
- Even though their PC was used to conduct a crime, a suspect may open up to investigators when something like this is being looked into. Installed Program Keys in the Windows Registry
- Forensic investigators can benefit greatly from knowing what applications are now or have previously been installed on the suspect computer. For instance, the presence of steganography and encryption tools, or remains from such tools, will indicate that the suspect machine may have hidden data or was only used to run such tools.
- Windows maintains details of all installed applications in the registry in the following locations:

(1) HKEY_LOCAL_MACHI
NE\SOFTWARE\MICROSOFT\WINDOWS\CURRENTVERSION\UNINSTALL

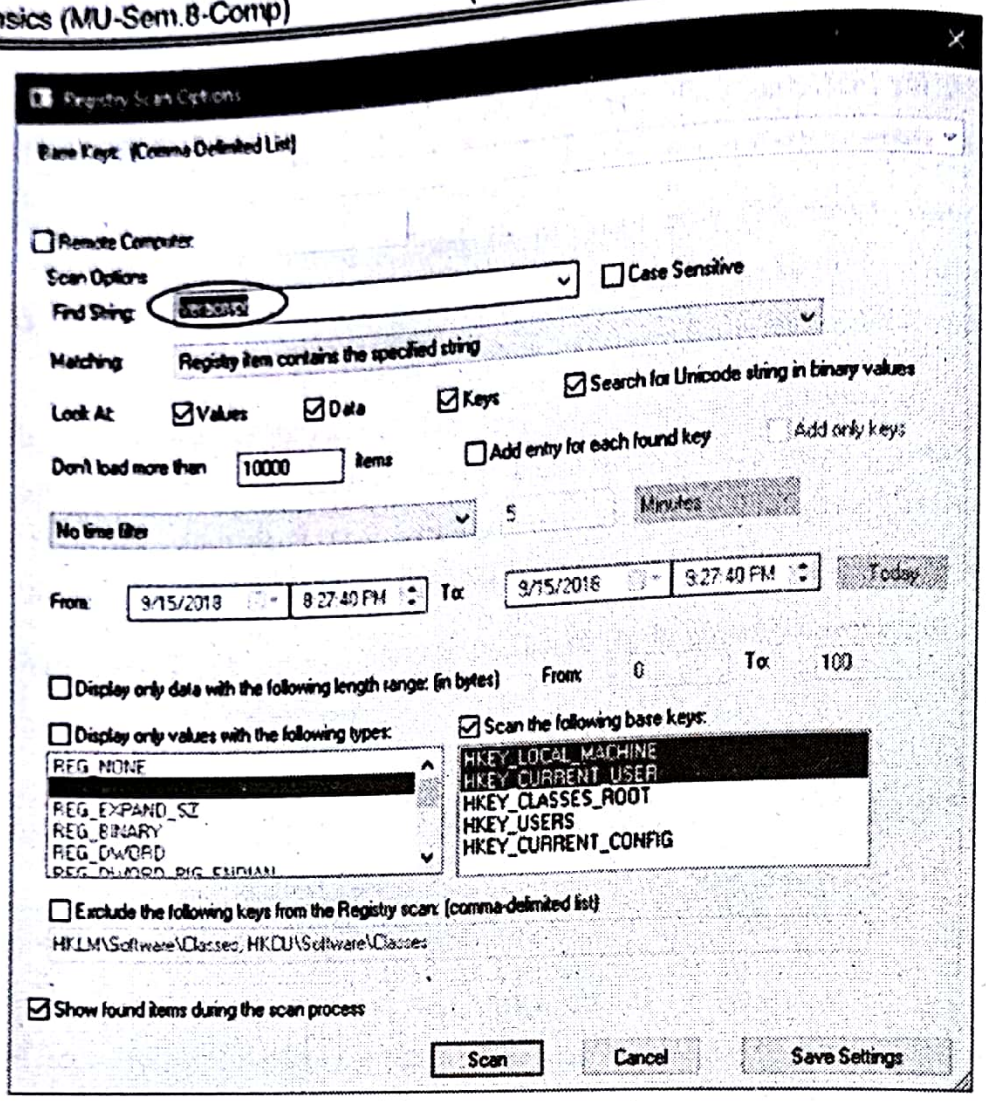
(2) HKEY_CURRENT _ U S E R \ S O F
TWARE\MICROSOFT\WINDOWS\CURRENTVERSION\UNINSTALL *

(3) HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432NODE\MICROSOFT
\WINDOWS\CURRENTVERSION\UNINSTALL **

(4) HKEY_CLASSES_ROOT\INSTALLER\PRODUCTS\<PRODUCT
CODE>\SOURCELIST\NET

(5) HKEY_CURRENT_USER\SOFTWARE\MICROSOFT\INSTALLER
PRODUCTS\<PRODUCTCODE>\SOURCELIST\NET

- To find an application that has been installed on the Windows registry or lost data, such as fragments of installed applications, abandoned applications, or any data objects that might be hidden in the Windows registry, we can use automated tools.
- RegScanner, a simple utility provided for free by Nirsoft. (**www.nirsoft.net/utils/regscanner.html**), searches the Windows registry using the user's specified search parameters. The user can click any item in the list of returned results to access the related value in RegEdit.
- The discovered registry values can also be exported into a.reg file. Once this utility has been run, a search option box will open where you can input your search criteria and modify some search options (Refer Fig. 4.1.12).



(1D12)**Fig. 4.1.12 : Options for registry scanning that the RegScanner application uses to search the Windows registry**

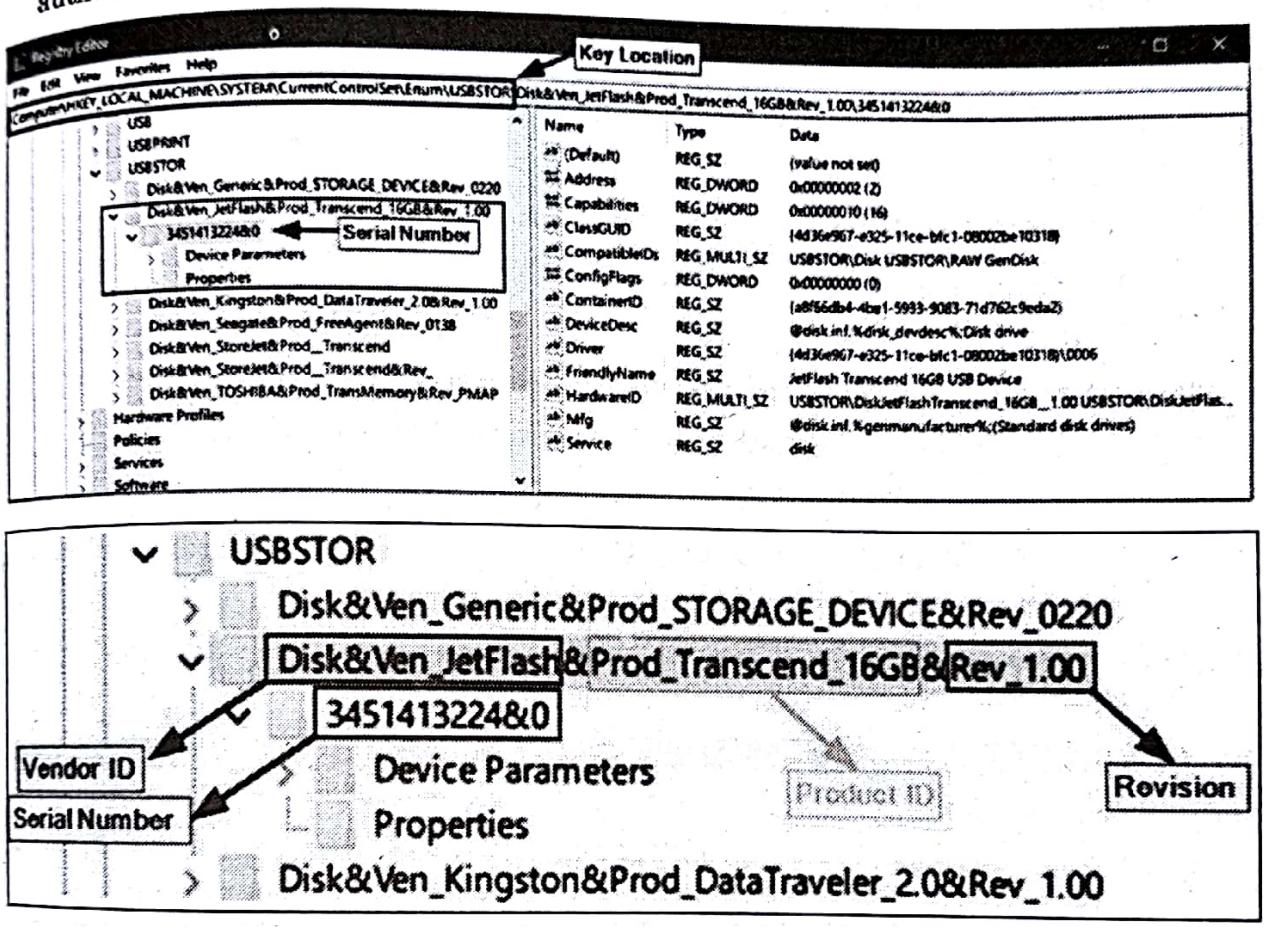
- Keep in mind that not all programmes require the installation of a registry key before being used; for example, portable applications can function on Windows without being installed (e.g., applications launched from a USB stick).

4.1.4(D) USB Device Forensics

- Windows keeps a history log of all previously connected USB devices, including the times of those connections and the user account that installed them. The **vendor ID, product ID, revision, and serial number** for each connected USB device are also stored in the Windows registry along with other crucial technical data.
- Windows uses five registry keys to store data pertaining to USB history, with each entry providing a unique piece of knowledge regarding the connected device. Investigators will be able to determine how an offender used removable devices, like a USB, to carry out or support their actions by integrating this information.

(1) HKEY LOCAL MACHINE\SYSTEM\CurrentControlSet\ \sEnum\ USBSTOR

All USB devices inserted into the operating system since its installation are listed here. Note that if the second character of the device serial number is "&," the connected device does not have a serial number, and the device ID has been generated by the system. It also displays the USB vendor ID (manufacturer name), product ID, and device serial number. For a list of previously connected USB devices to the author's computer, see Fig. 4.1.13.



(1D13)**Fig. 4.1.13 : USB connected devices history**

(2) **HKEY_LOCAL_MACHINE\SYSTEM\MountedDevices :** The MountedDevices subkey stores the drive letter allocations; it matches the serial number of a USB device to a given drive letter or volume that was mounted when the USB device was inserted.

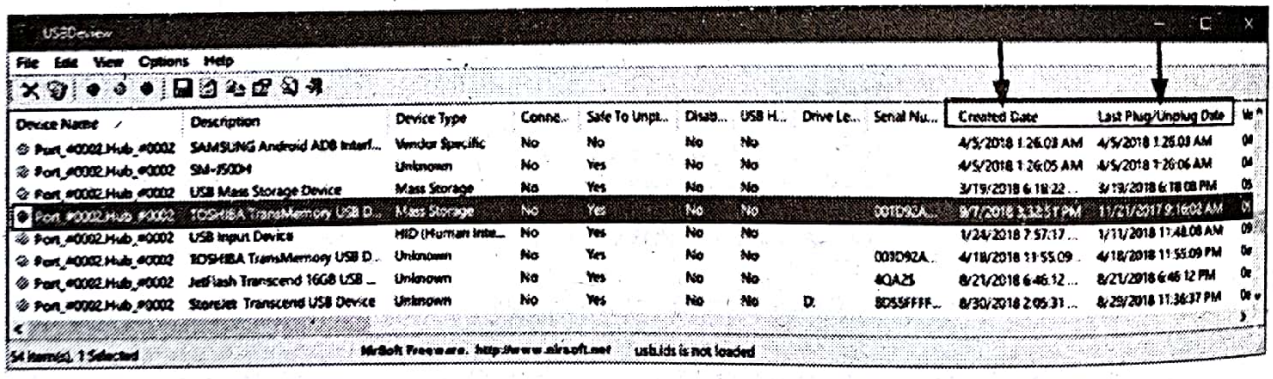
(3) **HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion \Explorer\MountPoints2 :** This key records which user was logged into Windows when a specific USB device was connected. The key also includes the "Last Write Time" for each device that was connected to the system.

(4) **HKEY_LOCAL_MACHINE\SYSTEM\Currentcontrolset\Enum\Usb :** This key holds technical information about each connected

USB device in addition to the last time the subject USB was connected to the investigated computer.

(5) **Identify the first time device was connected :** Check this file at **\Windows\inf\setupapi.dev.log** for Windows Vista, 7, and 8, and at **\Windows\inf\setupapi.upgrade.log** for Windows 10. On Windows XP, this file will be located at **\Windows\setupapi. log.** Search in this file for a particular USB device's serial number to learn when it was first connected to the subject system (in local time).

- The Nirsoft tool USBDeview (**www.nirsoft.net/utils/ usb devices view.html**) can be downloaded for free and performs all the activities we just completed manually for finding information about the current and previously USB connected devices.
- Extensive information (such as device name/description, device type, serial number, and much more) about each attached USB device will be displayed after this utility has been run on the target system.
- The Last Plug/Unplug Date in Fig. 4.1.14 indicates when a device was initially connected to the system. The reinsertion of the same device does not alter this date. The last time the same device was connected to the system is shown by the "Created Date" field.



(1D14)**Fig. 4.1.14: Viewing various artefacts about previously connected USB devices using USBDeview**

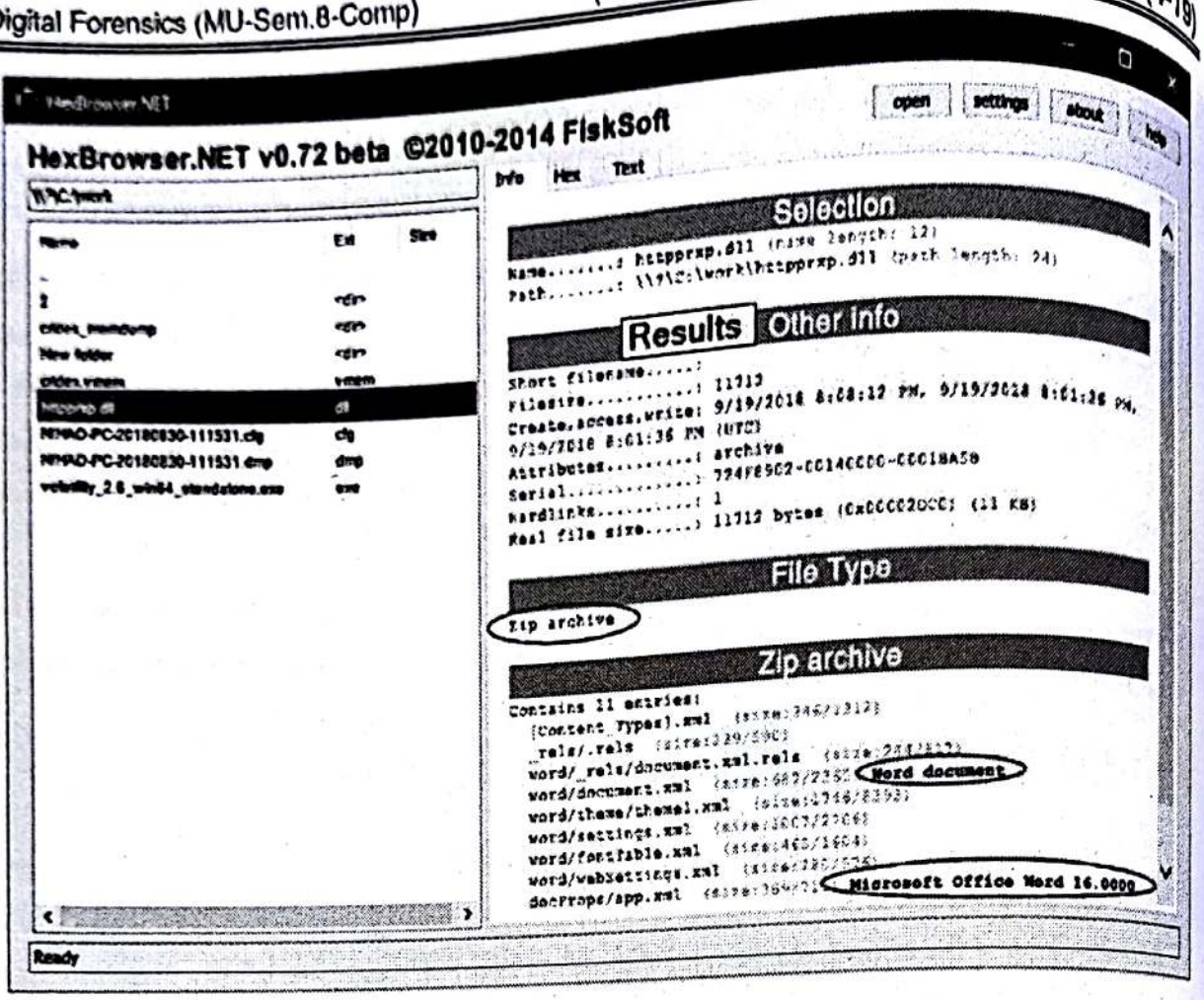
- Unfortunately, not all USB device types will leave traces in the Windows registry as we have described, for example, USB devices that connect to computers using the media transfer protocol (MTP). The MTP protocol is used by devices running newer Android OS versions, as well as Windows phones and Blackberry devices.

- When a USB device is connected to a Windows computer, the MTP protocol does not leave any traces in the Windows registry. In order to handle the research of such artefacts, a specialist tool is required.
- The website USB Detective (https://usbdetective.com) facilitates the detection of USB devices that connect to Windows using the MTP protocol. Although you must subscribe to the premium professional edition to use all functions, it also includes a wealth of options for thoroughly examining linked USB devices, including the creation of timelines of each device's individual connection/disconnection and deletion timestamps.
- In order to obtain the traces from a Windows machine from a USB device connected over an MTP connection, specific processing is required. To find out if this capability is available, check the manual for your computer forensic software.

4.1.4(E) File Format Identification

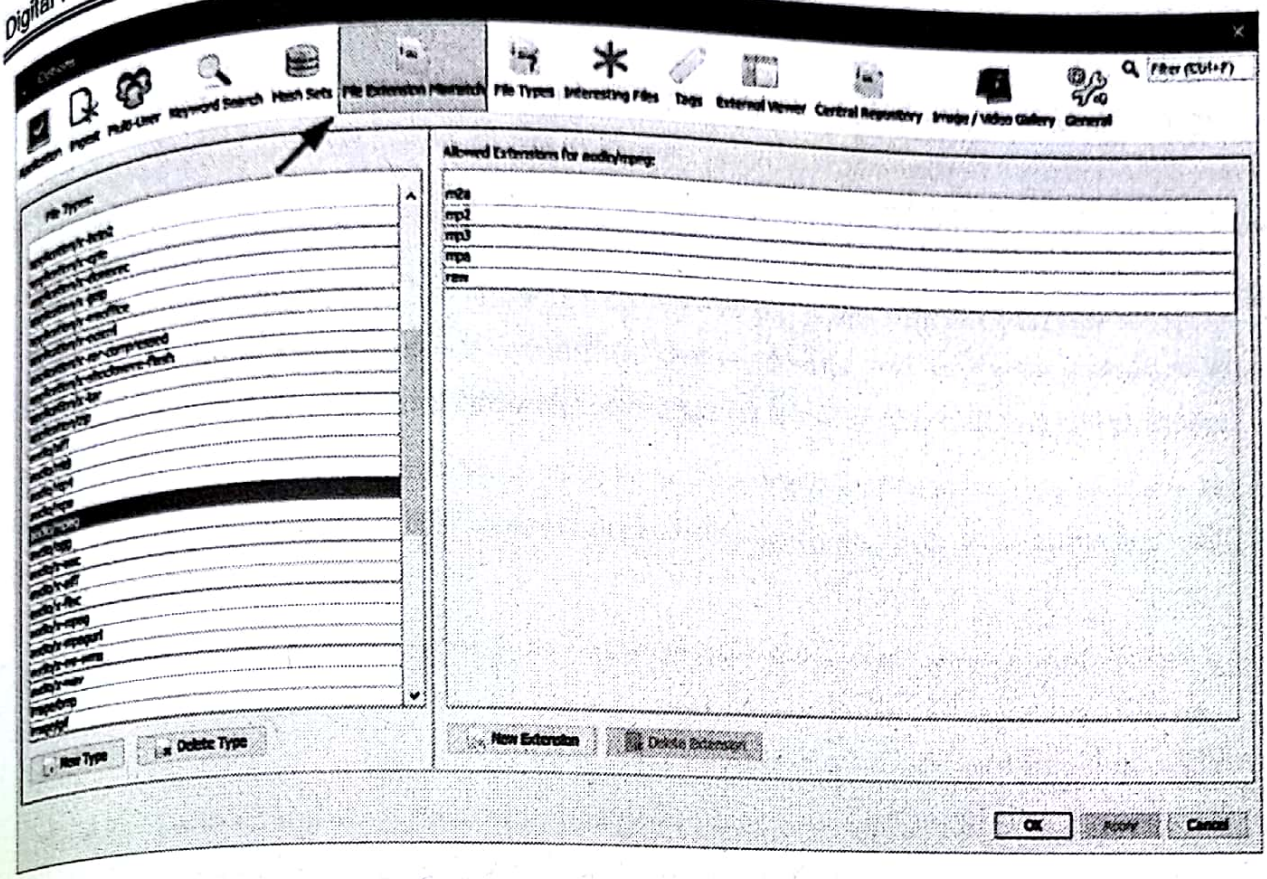
- In order to determine whether an attempt has been made to mask the original file type (by changing the file extension to conceal it from the investigators' sight), file headers and extensions are compared with a known database of file headers and extensions.
- As we are all aware, every file in Windows has a distinct signature that is typically kept in the first 20 bytes of the file. Any file can have its original file signature verified by looking at it in Notepad or a Hex editor.
- A Windows application called HexBrowser can identify more than 1,000 different file formats and provides comprehensive information on each one. Follow these easy steps to use this tool:

1. Download HexBrowser from www.hexbrowser.com.
2. Select the suspicious file by clicking the "Open" option in the main application menu, and then you're done!
3. View the outcomes in the application window's right pane (see Fig. 4.1.15).

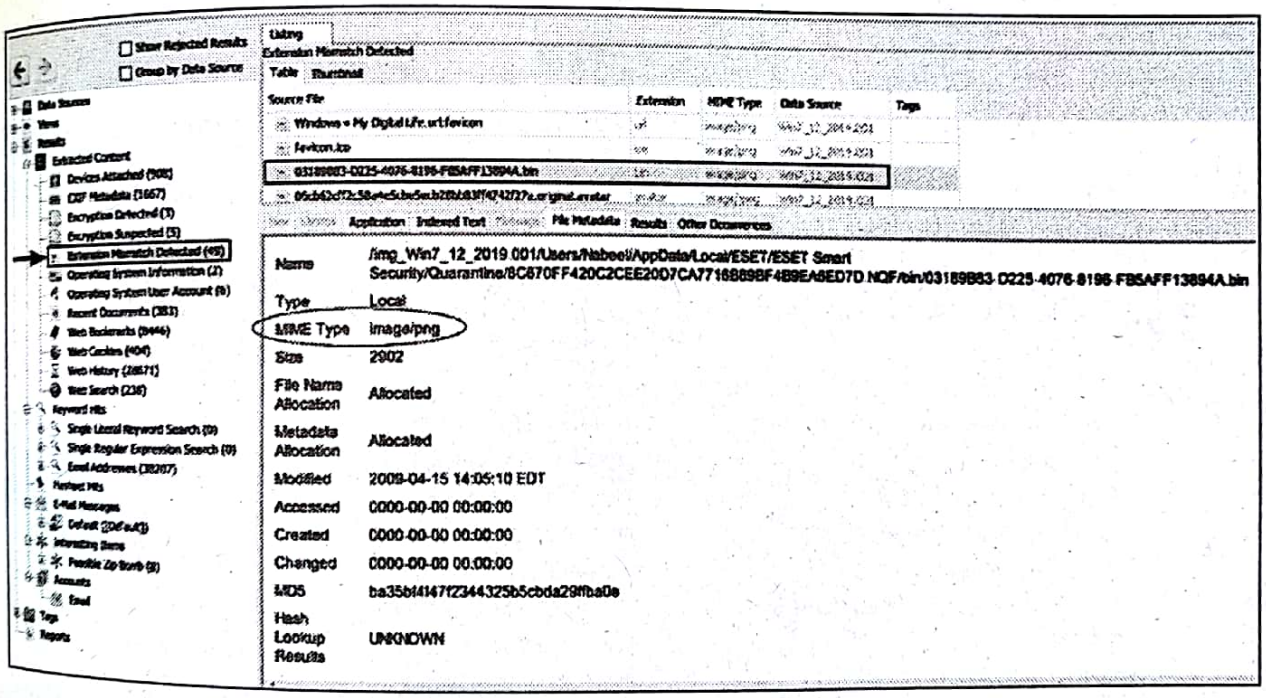


(1D15)Fig. 4.1.15 : To discover specified original file format. In this example, a file with a DLL extension was investigated, and HexBrowser discovered that the original file type is MS Word 2016.

- The "Extension Mismatch Detector" module must be enabled in order to leverage Autopsy's ability to find file extension mismatches. By selecting
- Tools menu > Options > File Extension
- Mismatch, we can further customize file mismatch search options. According to your case needs, you can add or remove extensions from this point (see Fig. 4.1.16) and the outcomes are displayed in the Results tree under "Extension Mismatch Detected" (see Fig. 4.1.17).



(1D16)**Fig. 4.1.16 : Configuration of File Extension Mismatch in Autopsy**



(1D17)**Fig. 4.1.17: Discovering file mismatch results using Autopsy**

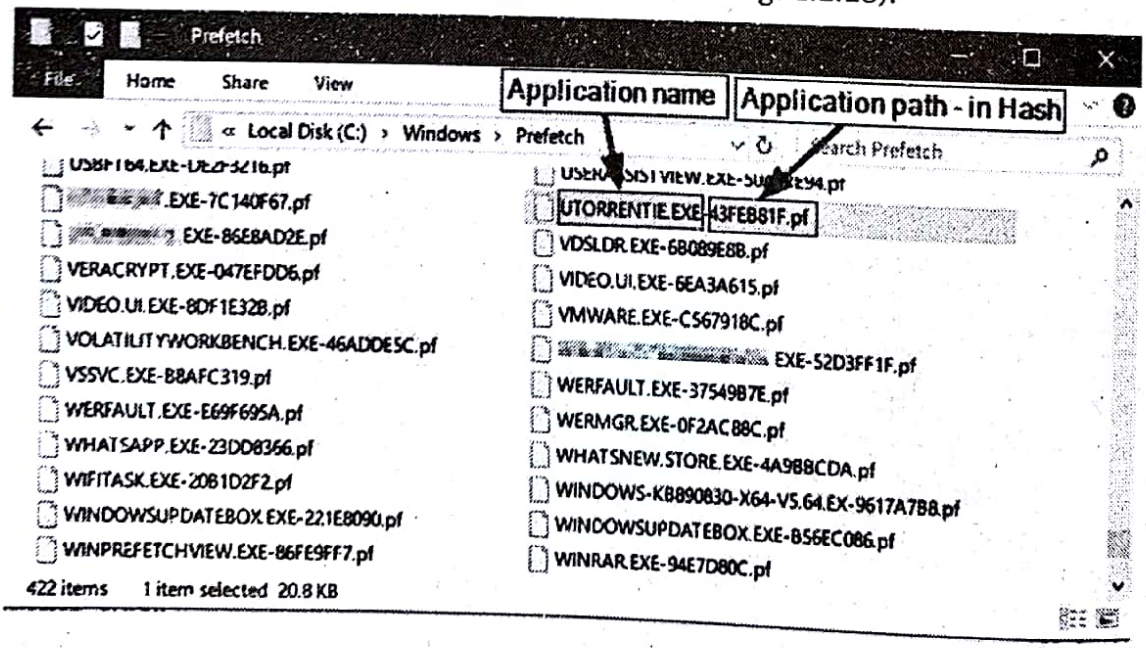
4.1.4(F) Windows Features Forensics Analysis

The Windows OS provides a variety of features that can be used to enhance or modify some of its features to make them more user-friendly. It is crucial to look at these qualities because they may include digital evidence.

Windows Prefetch Analysis

- Windows uses a function called prefetch to hasten the loading of applications. When a user starts an application for the first time, Windows produces a Prefetch file.
- It then records which files were loaded as part of this application execution as well as the last time this application was launched so that Windows can load it more quickly the next time a user launches it.
- Even if the subject application was uninstalled after execution, the Prefetch feature can still reveal what applications were run on the target machine because they are still visible in the Windows Prefetch folder.
- The subsequent Windows registry key contains the configuration for the Prefetcher:

  HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\MemoryManagement\PrefetchP
- Prefetch files are kept by Windows at C:WindowsPrefetch. Prefetch files are all named based on standard naming conventions. Prior to the eight-character hash of the location where the application was executed and the.PF extension, the name of the currently running application appears first (see Fig. 4.1.18).



(1D18)**Fig. 4.1.18 : Contents of Windows Prefetch folder**

- Following are the simple portable tools for reading windows prefetch files.
  (1) WinPrefetchView from Nirsoft (www.nirsoft.net/utils/win_prefetch_view.html)
  (2) Prefetch Parser (https://ericzimmerman.github.io/#!index.md)

Windows Thumbnail Forensics

- When a user selects to view files as thumbnails, Windows caches thumbnails of graphical files (JPEG, BMP, GIF, PNG, TIFF), some document types (DOCX, PPTX, PDF), and video files in the thumbs.db thumbnail cache file for later easy viewing.
- Investigators can learn about prior files (such as images) that were there on a system even after the user erased them by looking into this feature because image thumbnails might still be present at thumbs.db.
- Thumbnail previews are kept in one central location in the system by more recent versions of Windows. Database has a thumbnail search file in addition to an index file.
- Thumbs Viewer is a portable utility for extracting thumbnail images from the database files included in all Windows OS versions, including Thumbs.db, ehthumbs.db, ehthumbs vista.db, Image.db, Video.db, TVThumb.db, and musicThumbs.db. It is available for download at "https://thumbsviewer.github.io."

Jump Lists Forensics

- Microsoft introduced a new function for Windows users with Windows 7. Users can view previously browsed or accessed files for each installed application using the Jump Lists functionality.
- In criminal situations when the user's online actions are the center of the investigation, investigating this feature has great forensic value since it provides in-depth information into the user's computer habits and recently viewed files.
- The location of Jump List files for each user on Windows is "\Users\<username>\AppData\Roaming\Microsoft\Windows\Recent\" We can distinguish Jump Lists in to automatic and custom type.

Automaticdestinations-Ms

- These files are created in
  \Users\<username>\AppData\Roaming\Microsoft\Windows\
- Recent\AutomaticDestinations-ms"

- created automatically by Windows when a user opens an application or accesses a file. Jump Lists are contained within OLE containers and are named according to the application that has opened the relevant file.

☞ **Customdestinations-Ms**

- These files are created in \Users\<username>\AppData\Roaming\Microsoft\Windows\
- Recent\CustomDestinations-ms when a user pins a file to the Start menu or task bar.
- The application name (AppID), which is made up of 16 hexadecimal digits, is used to identify the file for both AutomaticDestinations-ms and CustomDestinations-Ms, followed by the ".customDestinations-ms" or "automaticDestinations-ms" extension.
- During application running, the system or the application names these AppIDs.
- The investigator can determine the identity of the application used to access or view the required file in the Jump List by learning the AppID name. Jump List IDs are listed on a variety of websites, including Forensics

  Wiki (www.forensicswiki.org/wiki/List_of_Jump_List_IDs)

  and GitHub (https://github.com/4n6k/Jump_List_AppIDs/blob/master /4n6k_AppID_Master_List.md).
- To automatically extract the data from Windows Jump List files, Nirsoft provides a utility. JumpListsView (www.nirsoft.net/utils/jump_lists_view.html) is a tool that shows information from the Jump Lists, such as the name of the file the user opened and the time and date that the open event occurred. To find out which application was used to open a subject file, users must still manually check the AppID.

☞ **LNK File Forensics**

- Windows shortcut files (LNK extension) are a type of metadata file that points to an application or file on the Windows operating system. These files are frequently discovered on a user's desktop, but we can also find them scattered throughout other places. When a user opens a local or remote file, Windows may automatically produce LNK files or the user may create them manually.
- In addition to the computer where it is presently located, LNK files also contain a variety of essential information about the machine where the file was initially produced.

- LNK files are useful for forensic analysis since they show the following :
  (1) Creation, modification, and access times for the linked file and the LNK file itself.
  (2) Past computer usage by the user; for instance, if a suspect transfers a file to a USB drive or completely deletes it from his or her computer, the accompanying LNK file will still be present, providing important details about previous operations performed on the target machine.
  (3) Size of the linked file.
  (4) The referenced file's original location.
  (5) The volume's name and serial number, which included the linked file.
  (6) The computer's original network path and network adapter MAC address
- When users right-click on the LNK shortcut, users can see details like the path and MAC time of the LNK file.
- However, by using third-party software made specifically for processing LNK files, a lot of data can be extracted. The following are two well-liked applications for looking at LNK files.

1. **Windows File Analyzer (WFA)** tool, information obtained from distinct Windows files such as shortcut files, Prefetch files, Index.dat, thumbnail databases, and others is decoded and displayed. Follow these steps to use this utility to extract LNK file information from a certain directory:
   (i) Download the utility (it's portable, free software) from www.mitec.cz/wfa.html.
   (ii) Open the tool, select the target folder under File > Analyze Shortcuts > Browse. You will be prompted by the application to choose the target OS. Finally, press the "OK" button to complete the action.
2. **Link Parser :** Link Parser (www.4discovery.com/our-tools), a free portable tool for extracting data from LNK files, was created by 4Discovery. It may analyze a single object, several items, an entire forensic image, or a folder and report all LNK files discovered along with their details (about 30 attributes). The acquired data may be exported as a CSV file.

Event Log Analysis

- Windows maintains a record of important events (including software and hardware events) that have affected the operating system, applications, or other services.

- In addition to assisting system administrators or users in identifying the exact cause of a specific event (for example, replacing a hard drive before it completely fails), recording events like low memory, excessive access to the hard drive, failed login, and others can help them predict future events.
- Windows event logs assist examiners in discovering what a user has done on a computer at a specific moment from a computer forensic perspective.
- The Windows event log can log five distinct types of events:
  (1) **Error :** Indicates a serious issue, such as when a service doesn't load properly during launch.
  (2) **Caution :** Although not a large occurrence, it could cause major issues in the future.
  (3) **Information :** Shows that a service, application, or driver is operating successfully.
  (4) **Success Audit :** Indicates that a security event was successful (for example, a successful login is noted as a "Success Audit event").
  (5) **Failure Audit :** Reverses a Success Audit (for instance, a "Failure Audit event" is reported when a user is unable to enter into Windows).
- The following are the essential components of each event in a log entry:
  (1) **User :** The account logged onto the computer at the time the event happens.
  (2) **Event ID :** A number produced by Windows that designates the type of event.
  (3) **Source :** The thing that brought about the incident.
  (4) **Computer :** The name of the device on which the incident took place.
  (5) **Date and time :** The time and date that the incident took place.
  (6) **Description :** Describe the events that led up to the occurrence.
- Windows has a user-friendly graphical user interface (GUI) for viewing recorded events; to access it, go to Control Panel > Administrative Tools > Computer Management.
- FullEventLogView is a portable application created by Nirsoft that displays all Windows event logs in a single table (www.nirsoft.net/utils/full event log view.html). With the help of this application, users may look at events that have been saved locally, remotely, or in an exported Windows log file with the .evtx extension. A TXT or HTML file can be created from an event list, making it simple to manipulate the results in software.

- Other tools for examining Windows event logs are available, both for free and for purchase. The principal ones are as follows:
- Log parser (available for download at www.microsoft.com/en-us/download/details.aspx?id=24659). SQL query language is used to query the Windows event log.
- The Log Parser Lizard GUI is available at www.lizard-labs.com/log parser lizard. aspx. It uses SQL to do queries on the registry, file system, Windows event log, IIS log, active directory services, and more. There is a free trial available for this commercial application.

Hidden Hard Drive Partition Analysis

- The majority of hard discs (HDD and SSD) typically have partitions. A partition can be divided for a variety of purposes; for instance, a user may choose one disc to contain the OS files and a second to contain user-private data.
- Hard drives can be partitioned by more people than just end users; for instance, the majority of computer manufacturers construct a hidden partition to hold a backup copy of an installed operating system. Hidden partitions, however, can also include files of relevance and evidence-gathering data.
- The DiskPart command-line application, a tool of the Microsoft Windows family, can be used to determine whether a specific hard disc or USB stick contains hidden partitions (Windows 10, 8, 8.1, 7, Vista, XP, and Server 2003).

Windows Minidump File Forensics

- Depending on the version of Windows, the Windowsminidump or Winntminidump will contain a copy of the computer memory at the time of the crash when a Windows PC experiences a Blue Screen of Death. Windows is capable of producing several memory dumps; for example, Windows 10 offers five varieties:
  (1) One-line memory dump (256 KB)
  (2) Kernel memory dump, second
  (3) Full dump of the memory
  (4) Auto-memory dump (default option)
  (5) Dynamic memory dump

- We can configure the memory dump feature under Windows by going to **Control Panel ➤ System ➤ Advanced system settings ➤ Advanced tab ➤ Startup** and Recovery pane – Settings button.

☞ **Pagefile.sys, Hiberfil.sys, and Swapfile.sys**

- Three crucial system files: Swapfile.sys, Hiberfil.sys, and Pagefile.sys are necessary for the Windows OS to operate correctly.
- The three files come hidden, so you must see hidden files including protected system files.

☞ **Pagefile.sys**

Virtual memory has a forensic value since it can store crucial information that has been moved from RAM. Forensic examiners should not ignore this worth. For instance, pieces of encrypted data could still be present there, and encryption keys or passwords (or pieces of them) might potentially be discovered there.

☞ **hiberfil.sys.**

- This file, which roughly occupies 3/4 of our RAM, is utilized by Windows to provide the hibernation feature. Since it only saves the kernel session and device drivers in more recent versions of Windows (such as 8 and 10), the hibernation file is noticeably smaller than it was in prior versions of Windows (such as 7 and Vista).
- A precious of data about the machine that is now running can be stored in hiberfil.sys.

☞ **Swapfile.sys**

- Every time a user attempts to access an idle process again, its information is transferred back to the RAM memory, where it is used to store the idle and other non-active items removed from the RAM memory.
- We can see that **Pagefile and Swapfile** coexist on a system drive in more recent versions of Windows (such 8 and 10), and we can assume that these two files combined make up what is now referred to as virtual memory in the Windows OS. In current Windows versions (8, 10), the Swapfile has a fixed size of 256 MB.
- For each partition where it is activated, the Volume Shadow Copy Service (VSS), a service that is accessible in all Windows versions starting with Windows XP, coordinates the construction of a consistent snapshot of the data at a particular point in time.

- When some of Windows' files become corrupt, VSS helps in recovery; in such a circumstance, Windows will restore the clean version of the files from earlier backups (restore points). These restore points are created at predetermined intervals that Windows has set. The system restore feature can be used to access the VSS capability, which is only available on NTFS-formatted discs.
- Finding deleted files and folders that are still accessible in restore points after the user deletes the original copies is the biggest benefit from investigating this feature. Investigating Windows restore points snapshots can provide a variety of forensically helpful information.
- Windows restore point snapshots also contain the registry hive files, making it possible to investigate earlier Windows registry snapshots. Use the free portable utility RegistryChangesView from Nirsoft (www.nirsoft.net/utils/ registry changes view.html) to extract Windows registry files from a specific Windows restores point.

4.1.4(G) Windows 10 Forensics

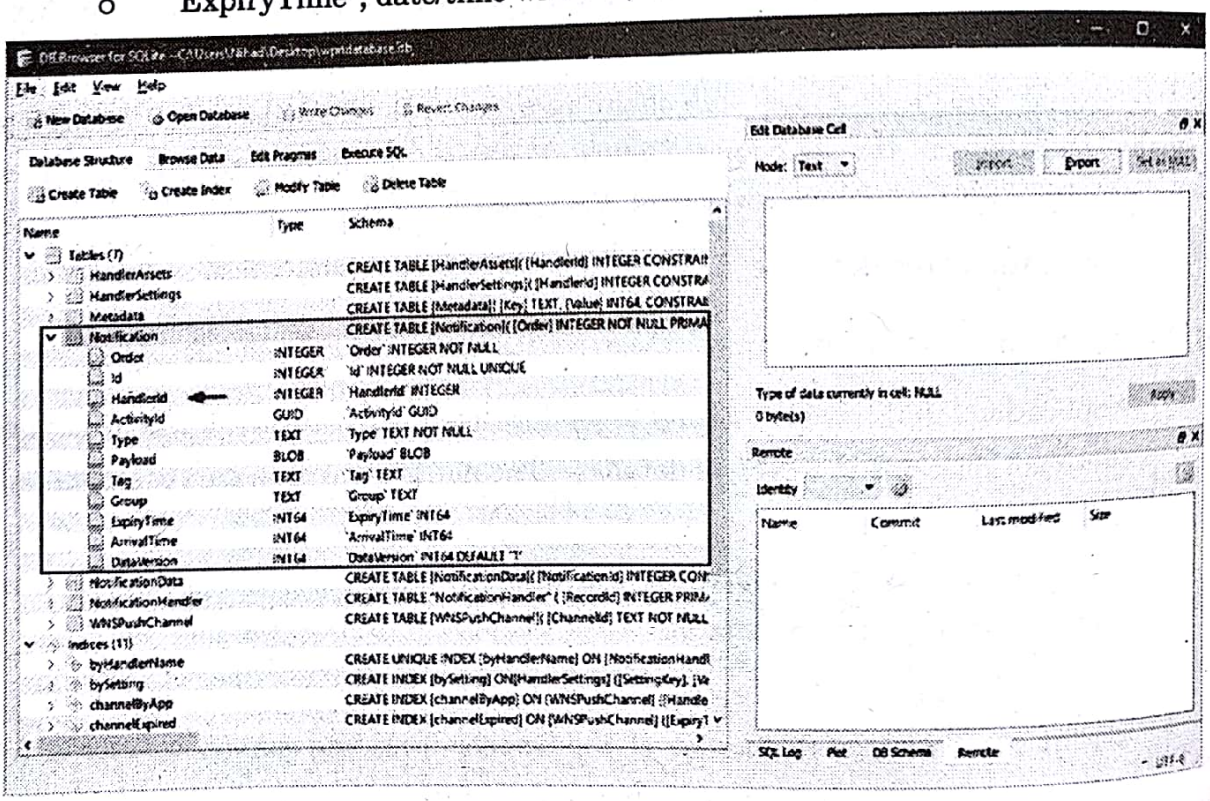
- Microsoft's voice-activated digital assistant Cortana, the Edge browser, Windows 10 apps, and many other new features and applications are made available to Windows 10 users.
- The most significant was the launch of the Universal Program Platform (UAP), which makes it possible for a single application to run across several device types, including laptops, desktops, Internet of Things (IoT) devices, tablets, smartphones, and more.
- Windows 10 comes equipped with many new features. We have gone through the many features in the Windows feature Forensics Analysis, now here we will investigate most two unique services of Windows 10:

  (1) Notification area database (2) Cortana forensics

(1) Notification area database

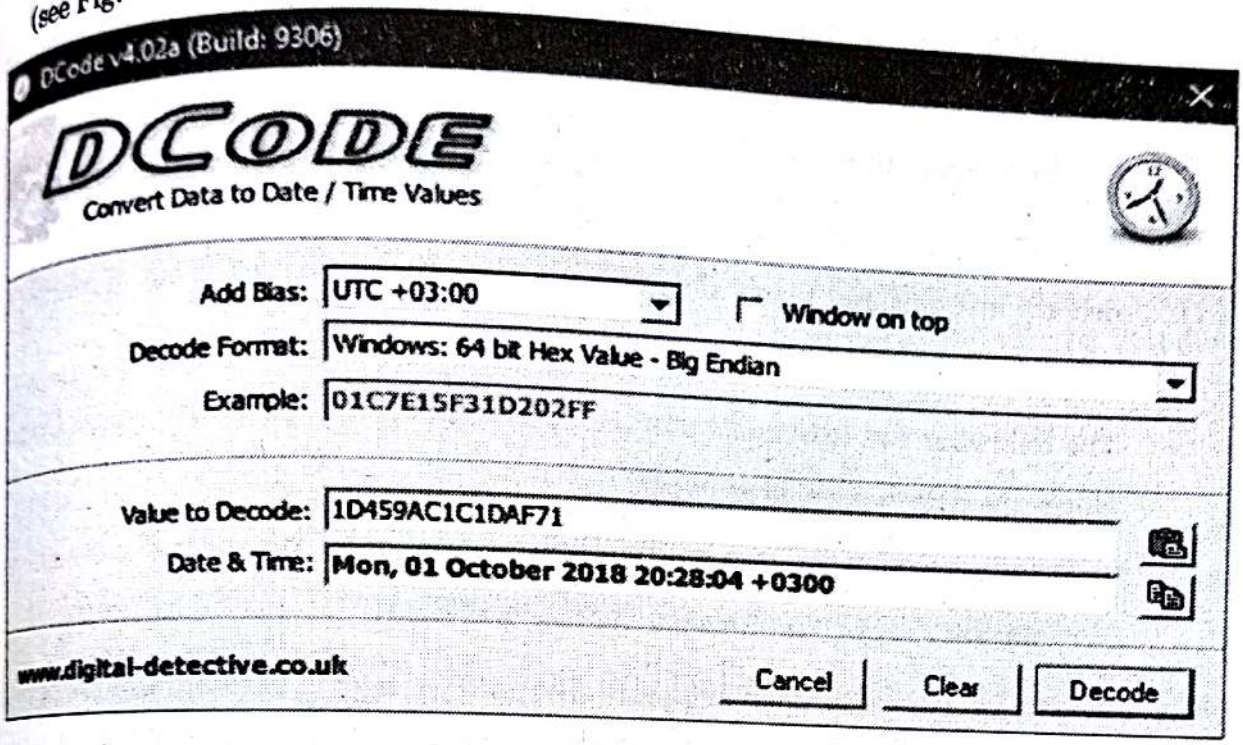
- Applications that have the ability to produce systray notifications will later save these notifications in a central database. The notification area database can be found at **\Users\<UserName>\AppData\Local\Microsoft\Windows\Notifications** under the name wpndatabase.db
- The notification database stores different notification types that Windows users see in the bottom right corner of the screen, including pop-up messages from different OS components (such as backup and restore), email alerts, and messages related to particular apps, like Torrent downloads, among others.

- Windows notifications have forensic relevance since they can show what prior users have done on the target computer.
- The database for the notification area is SQLite (.db extension). Follow these steps to analyze the information in this database:
  - Go to http://sqlitebrowser.org and download DB Browser for
  - SQLite; select the version appropriate for your OS version.
  - Launch the program, go to File menu ➤ Open Database; browse to Users\<UserName>\AppData\Local\Microsoft\Windows\
  - Notifications; and select wpndatabase.db.
  - In Fig. 4.1.19, we can see the database schema of the Windows notification area (wpndatabase.db). In the Notification table, we can find the following attributes:
  - "HandlerId", which tells which program, has created the notification (retrieve program name from table "NotificationHandler").
  - "Payload" contains notification contents.
  - "ArrivalTime"; date/time when notification received.
  - "ExpiryTime"; date/time when notification will be deleted from the database.



(1D19)**Fig. 4.1.19 : Browse Windows notification area database using DB Browser**

- The "**ArrivalTime**" and "**ExpiryTime**" values are stored in decimal format; in order to convert them into a readable format, we must first convert the number into Hex.
- Then, using the DCode tool (previously used; accessible at www.digital-detective.net/dcode), we can convert the number into a readable date/time (see Fig. 4.1.20).



(1D20)**Fig. 4.1.20 : Convert date/time attributes from decimal values into a readable format using the DCode tool**

(2) Cortana forensics

- A voice-activated personal assistant called Cortana was created by Microsoft for the iOS platform, much like Apple Inc.'s Siri. When Windows 10 was released, Cortana, a relatively new feature, was also added to Windows desktop.
- It was first featured in Windows phone version 8.1. Its primary function is to give Windows 10 customers a personalized experience by making suggestions when they search, remembering events, sending emails on their behalf (when configured properly), searching the Web, checking the weather, and many other helpful things. Cortana functions by acquiring knowledge.
- Therefore, it will understand the user's own habits and attitudes better when the user communicates with it more (either by typing or through the PC microphone), which will result in more accurate results in subsequent interactions.

- In addition to web searches and geolocation data (the latitude/longitude of the triggered location-based reminders), Cortana can offer a wide range of details about a user's prior activity on the target machine from the perspective of digital forensics.
- Remember that despite the useful information that can be obtained from the Cortana feature, we cannot always assume that it will be turned on by default on Windows computers.
- This is because the feature has a reputation among Windows users for being a privacy invader, and many of them have already deactivated it out of concern for their privacy.
- The following two extensible storage engine (ESE) databases are where Cortana stores the data it uses for its work:
  - \Users\<UserName>\AppData\Local\Packages\Microsoft.Windows.Cortana_xxxx\AppData\Indexed D\IndexedDB.edb
  - \Users\<UserName>\AppData\Local\Packages\Microsoft.Windows.Cortana_xxxx\LocalState\ESEDatabase_CortanaCoreInstance\CortanaCoreDb.dat

- The "CortanaCoreDb.dat" contains forensically important information on user geolocation data, user-set reminders, and the locations and times at which these reminders have been activated. Please be aware that Cortana has access to a significant amount of personal data about its users, but it appears that Microsoft has moved many Cortana user interactions to its cloud servers.
- Another location where some Cortana-related artefacts can be found on the local machine is Users\<UserName>\AppData\Local\Packages\Microsoft.Windows.Cortana_xxxx\LocalState\LocalRecorder\Speech
- This folder stores voice command (WAV audio files) recordings issued by a user to Cortana to perform a task.
- The Cortana database cannot be decoded by all computer forensic suites; always read the documentation or the tool's features before purchasing. For instance, EnCase contains a script to decode user-specified IndexedDB.edb files' Cortana search words.

4.2 INVESTIGATING UNIX SYSTEMS

Comparison of UNIX and Windows

GQ. What is a difference between UNIX and Windows?

The UNIX and Windows operating systems differ significantly in a number of key ways. The following are some of the key differences between the UNIX and Windows operating systems:

- The Command Line Interface is a component of the UNIX operating system (CLI). Windows operating system, in contrast, has a Graphical User Interface (GUI).
- The UNIX operating system supports multiprocessing. In contrast, the Windows operating system does not support multiprocessing. UNIX is an OS with an open and free source. Windows, on the other hand, is a licenced OS.
- A command-based OS is UNIX. Windows, on the other hand, uses a menu-based OS. Files can be thought of as independent files, and Unix is completely case sensitive. Windows, in contrast, has the choice of case sensitivity.
- The Unix operating system is renowned for its exceptional stability. Despite recent improvements in Windows stability, most Unix systems continue to be significantly superior in this regard.
- There isn't much hardware support for Unix systems. It's possible that certain hardware doesn't have drivers. In contrast, practically all of the drivers for any piece of hardware are already included in the Windows operating system.
- It is possible to install the adaptable Unix operating system on a variety of devices, including mainframes, supercomputers, and microcomputers. New software design concepts are also encouraged by Unix, such as connecting smaller, simpler tools to solve issues rather than creating large, monolithic applications. Comparatively easy to use in terms of capabilities, but less powerful than Unix, is the Windows operating system.
- The ERR and STD.IO file systems are used by the UNIX operating system, and the UFS (Unix File System) treats each physical drive as if it were a single logical drive. Its file system is reliable and efficient. The file system is represented as a hierarchical tree with a single root.

- Windows, on the other hand, uses the File Allocation Table (FAT32) and New Technology File System (NTFS) systems to manage files and requires the owner of executables before executing them. Several hard discs, including C, D, and E, include folders where files are kept.
- Under the UNIX operating system, users may save two identical files. In contrast, the Windows operating system does not allow the user to save two files with the same name.

4.2.1 Reviewing Pertinent Logs

- Numerous log files found in Unix operating systems can provide crucial information during incident response. In addition to logging system activities like logons, startup, and shutdowns, Unix network service events are also recorded.
- The majority of log files can be found in the common directory /var/log. Other directories, like /usr/adm or /var/adm, are used by various Unix variants. Some logs are stored in unexpected places, including /etc.
- Consult operating system-specific documentation if you're unsure. Not all log files are present on the system in question, to boot. On a network server or security appliance like a firewall or an IDS, you might locate relevant logs.

Network Logging

- The syslog (system log) file is arguably Unix's most valuable logging feature. This log records activities from Unix systems and programs. The syslog configuration file, typically /etc/syslog.conf, regulates how syslog functions.
- The system's syslog daemon, syslogd, operates to log messages. Syslog also provides the option of remotely logging messages via a network.
- Overall, syslog's logging functionality is very strong and adaptable. Syslog logs to a variety of files in the default log directory on most varieties of Unix, although the most helpful logs are typically the messages, secure, and syslog files.
- Which kinds of messages are transmitted to which logs are determined by the syslog configuration file. Each contains three fields:

  (1) The subsystem that generated the log file is identified by the facility field. For instance, the mail facility's sendmail logs. There are eight facility types: auth (security), authpriv, cron, daemon, kern, lpr, mail, mark, news, syslog, user, uucp, and local0-7.

(2) The severity of the log is indicated by the priority column. Debug, info, notice, warning, err, crit, alert, and emerg are the eight priority levels.

(3) How the log will be recorded is specified in the action box. The action could be a log file's name or even a remote logging host's IP address.

- Four facility/priority entries are displayed in this configuration entry, and they are all logged to the /var/adm/messages file. Every facility having a priority level of err or higher is identified by the leading *.err. Any mail facility message with a critical priority or above is logged, according to the mail.crit entry. This example's action field instructs syslog to log all messages that match the facility/priority criteria to the /var/adm/messages file.
- This system snap demonstrates that a sendmail relay attempt (PID 5857) was made on the Pearl computer system, however the relay attempt was unsuccessful.

☞ Remote Syslog Server Logs

- The text log files created locally by the syslog daemon are typically world-readable but can only be modified by root.
- As a result, any attacker with administrator-level access might easily alter the syslog log files by deleting certain entries, changing certain entries, or introducing false messages. These alterations are essentially hard to find.
- Do not trust the logs if you believe an attacker has achieved root-level access to the system where they are stored. Repetitive logging to a safe, remote syslog server is the only method to know for sure if an attacker altered the log files.
- A pristine duplicate of the log file should be present on the remote syslog server in the event that a system is breached and the log files are altered, or if the attacker deletes the entire log file.
- Of course, the attacker could add fictional entries to the remote syslog server, but without first compromising the remote server, the attacker would be unable to change or delete entries.
- In light of this, the remote syslog server needs to be a protected (secure) host with restricted access, ideally just console or secure shell (ssh), which also makes use of system logging.
- To prevent access based on the compromising of passwords from other systems, the accounts and passwords for the server should be unique.

☞ TCP Wrapper Logging

- TCP Wrappers is yet another incredibly useful application that uses syslog in addition to all the other applications that benefit from the system logging feature.
- A host-based access control for TCP and UDP services is called TCP Wrappers. Syslog is used to log any connection attempts to "wrapped" services.
- Log entry contains a wealth of useful details, including the time and date of the attempted logon as well as the hostname (victim), service (sshd), user (root), and IP address of the system.
- This entry demonstrates a connection to the victim's TFTP server on April 26 from the host 10.10.10.10. One of the investigator's most effective tools is the correlation of connections and file-access timings.

☞ Host Logging

Numerous log files are made available by Unix to record host activity. Su command execution, users who are currently logged in, failed logon attempts, and cron job (scheduled programme) execution are some of the more useful logs.

☞ su Command Logs

- A user can change user IDs while still in a session by using the su command. This command is occasionally used by attackers to try to acquire root access to a system. Every attempt to run the su command on the system is recorded by Unix.
- The log displays the su attempt's time and date, whether it was successful, the terminal from which it was made, and the user ID both before and after the su attempt. Su attempts are logged in the messages or syslog file on some varieties of Unix, whereas on others, a distinct su log file is kept in one of the log directories.

☞ Logged-on User Logs

- Information on users who are logged on to the system is kept in the utmp or wtmp file. Depending on the type of Unix used, the log file has a distinct name and contains somewhat different data.
- The name of the user, the terminal used to log on, and the time of the logon are the three pieces of fundamental data that are stored. Instead of being saved as a text file, the file is kept in a binary data format.

- We cannot presume the integrity of these files even though the wtmp or utmp logs are saved in a binary format and cannot be easily altered with vi or other comparable editors. Numerous well-known hacker tools, like zap, can erase specific entries from these files.
- User must use the proper client software, such as w, who, finger, or last, to query the utmp or wtmp log file. You will require the retrieval utility's operating-system-specific version.

☞ Logon Attempt Logs

On the majority of Unix systems, unsuccessful and successful logon attempts are automatically recorded. Console logons are also kept in one of the log files, such as the messages file on Linux systems, along with the logon attempts for network services like FTP or ssh.

☞ Cron Logs

With the help of the Cron feature in Unix, users can schedule applications for execution in the future. Every cron job that is run is recorded in a file named cron, which is often located in the default logging directory or in /var/cron/log.

☞ User Activity Logging

In addition to logons, various user activities are also logged in Unix logs. The commands that users have executed are recorded in process accounting logs and shell history files.

☞ Process Accounting Logs

- Process accounting is a Unix feature that logs every command executed by every user. By default, this kind of logging is disabled. You won't be able to use this functionality if the acct or pacct log file does not exist on the system. We can examine the contents of either of these files if they are present by using the lastcomm or acctcom command.
- A binary file serves as the process accounting log file. No open-source attack tools that can alter this file are known to us. The attacker would have to destroy the log file in order to get rid of this proof.

☞ Shell Histories

An associated command shell, such as the Bourne (sh), Korn (ksh), or Bourne-Again (bash) shell, is available to users who have interactive access to Unix computers. These shells offer the opportunity to record every command and its command-line arguments. The history file is often kept in the user's home directory as a hidden file.

What Can Happen

Your system has just been accessed by an intruder as root. The deletion of the.bash history file is one of the attacker's first actions. The file is afterwards linked to /dev/null, making it unable to log commands.

Where to Look for Evidence

- Inspection of shell history files is needed, whenever we examine a Unix system that might have been compromised.
- The history file was likely erased by the hacker if the history feature was activated and the history file was missing.
- It is even another clear sign that the system has been compromised if the history file has a connection to /dev/null. Also take notice of the file's creation date and time; the intrusive party could be left as trail of evidence for further inquiry.

4.2.2 Performing Keyword Searches

In practically every incident response investigation, from examples of email harassment to remote network compromise, keyword searches play a crucial role. A wide variety of ASCII strings, such as a username, MAC address, IP address, or the backdoor password of an attacker, can be used as keywords. We can perform keyword searches on the physical level, looking at a drive's whole contents, or on the logical file structure.

String Searches with grep

- The powerful, adaptable grep command is the primary tool for string searches. to perform a string search on a file.
- The functionality of grep varies between versions. When compared to many other, earlier Unix varieties, the GNU versions of grep that come with Linux are far more feature-rich. On a Solaris system, we must first use other tools, such strings, to extract the ASCII strings from the binary file in order to get the same outcomes.

File Searches with find

- Another useful command for string searches is find. You can use the find command to find any filename that matches a regular expression.

- For many searches, the find command is useful. It may look across a file system for files that fit a wide range of criteria, such as modification or access times, file owners, strings found inside files, strings found in file names, and more. Using the robust exec capability, we can also use find in conjunction with other commands like strings or grep.

4.2.3 Reviewing Relevant Files

- Numerous files will almost certainly include information on any specific incident. The likelihood of you finding every pertinent file is substantially lower though. We employ a few methods to assist in determining which files are most likely to be relevant to a specific incident.
- Using the information gathered during the initial response to Unix, as well as the **time/date stamps on the necessary files,** are a few of these ways. We also look through system and configuration files that are frequently accessed by attackers.

Incident Time and Time/Date Stamps

- We must first know the timing of the suspected incident in order to search for files and directories that were accessed, updated, or created around that time. A network IDS may have identified and logged the attack as it occurred, for example, thus the timeframe may be very precise.
- On the other hand, the timeline might be broad, as in the instance when a system administrator connected the machine to the Internet two weeks ago, and proof of breach was discovered today. The first step is to confirm that the system time on the IDS matches that of the victim system if you have a reliable record from an external source (like network IDS) of when the attack happened.
- Reviewing time and date stamps should be done with the intention of following up on the pertinent time frames you've already identified. It is likely that all of the files or directories that were viewed, changed, or created during this time will be relevant objects.
- For each file or directory, the Unix file system stores one of three timestamps:
  (1) The last time a file or directory was accessed is indicated by the atime, or access time. This covers even read-only access (such as cat filename).
  (2) The mtime, or modification time, logs the most recent modification to a file.

(3) Similar to the mtime, the ctime also keeps track of when the inode value was last modified. Events like altering ownership or permissions could affect its value.

- It might be good to save the time/date stamps from your initial response right away if you didn't already. Use the ls commands to get the atime, mtime, and ctime in order to save the time/date stamps for Unix. Save the output of these commands to the forensic workstation or magnetic media.

Special Files

- Certain file and directory types seem to appear in occurrences on a regular basis. SUID and SGID files, odd and hidden files and directories, configuration files, and the /tmp directory are some of these files and folders. Let's explore how these files might be important to research on Unix.
- SUID and SGID Files
- Unix contains features known as set userid (SUID) and set groupid (SGID), which are designed to allow programs to operate with higher privileges than those of the user running the program. For example, if user Bob executes a program, that program runs with the privileges of user Bob. However, if the program is SUID and Bob executes it, the program runs with the privileges of whichever user owns the executable, usually the root. SGID works the same way, except that the program runs with the privileges of the associated group.
- SUID and SGID root programs are the source of most privilege-escalation attacks on Unix systems, and they are also a favorite backdoor for attackers. A SUID root copy of /bin/ksh (the Korn shell) on most Unix systems will provide root privileges to any user who executes it. This is also known by attackers as a rootshell. To an investigator, a suspicious SUID root program is cause for alarm.
- Investigate further if you notice anything odd, such an SUID root programme in /tmp. We frequently observe a straightforward copy of /bin/ksh in the /tmp directory.
- Unusual and Hidden Files and Directories. Attackers frequently conceal files and directories from the untrained eye. Any file or directory that begins with a dot (.) under Unix is concealed from view and won't be listed in a ls command listing without the -a option.
- Additionally, hackers frequently give files and directories names that appear innocent, like /tmp/.X11-R5 for a directory or rpc.auditd for a sniffer. A name consisting only of three dots is very typical for directories.

- All of these names are similar to names of already-existing files and folders, so an administrator would not instantly be suspicious if they appeared in a directory listing or a process table listing. Knowing when to look deeper is the first step in spotting this kind of obfuscation, like in the example of directories with several dots.
- Logs from snoopers are obviously suspicious and are frequently hidden or renamed. They are beneficial to an attacker since they allow for the passive acquisition of data and network credentials. Sniffer records, fortunately, can also be extremely valuable to the investigator.

☞ Configuration Files

- Configuration files are a key piece of evidence in many incidents. With all the functionality built into the Unix operating system, a skilled attacker can easily modify an application to perform malicious tasks.
- Common targets are files that control access to the victim's system, such as the TCP wrapper configuration files /etc/hosts.allow and /etc/hosts.deny. An attacker can modify or delete these files to allow specific computers to freely connect to the victim's system.
- The Internet daemon configuration file inetd.conf (located in the /etc directory) controls many network services on Unix systems. Services such as Telnet, FTP, TFTP (and many others) are started through this file. An attacker can add entries to this file to force the victim's system to listen on many ports, or enable services like her TFTP that were previously disabled.

☞ Startup Files

- There are various places in the Unix operating system where services and applications can be launched. One of the most important files of this kind, the inetd.conf file, was just stated. Cron, rc startup files, and user startup files are further instances.
- The cron feature is used to schedule programmes for execution in the future, as was previously indicated. Cron jobs for different users are kept in the **/var/spool/cron or /usr/spool/cron directory.** This directory contains files with user account names, and any jobs saved there are run under that user's rights.
- For instance, tasks in the /var/spool/cron/root file are run as root. Cron jobs are a preferred hiding place for trojan applications as a result. Each file that is performed by cron tasks should be carefully examined because it could contain harmful code.

- The rc directory is another place where startup files can be found. This directory, which is typically called /etc/rc.d or something similar, includes a list of the programmes that run as soon as a Unix system boots up.
- These configuration files generally govern applications like sendmail and portmapper. To run trojan applications at bootup, attackers can simply add an entry to any of the startup scripts.
- Verify that the programmes being executed from the rc directory are real and have not been altered by an attacker. Check each of the startup scripts for fictitious entries.
- Furthermore, startup files are kept in each user's home directory. When users log in or when various applications are executed, the system automatically consults files like.login,.profile,.bashrc,.cshrc, and.exrc. These files can contain malicious code that attackers can insert. Check for erroneous entries in all configuration files of this kind.

Tmp Directory

On a Unix system, the only world-writable file system by default is the /tmp directory. This makes it a favorite hangout for attackers and a place where malicious tools are frequently stored. Additionally, a lot of publicly accessible exploits save temporary files in the /tmp directory during privilege-escalation attacks, and occasionally they leave trail evidence. In the event of an incident, thoroughly examine the /tmp directory to see whether any hidden directories or suspicious files are there.

4.2.4 Identifying Unauthorized User Account or Groups

- On victim systems, attackers frequently modify account and group information. This update may take the shape of new accounts or an increase in the privileges associated with existing accounts.
- Typically, the aim is to build a backdoor for future access. On suspected victim systems, you should audit user and group accounts to ensure that an attacker did not change this data. Information about **Unix system accounts** may be audited easily.

User Account Investigation

- User information is stored in the /etc/passwd file. This is a text file that can be easily examined by various mechanisms. All users on Unix systems have an entry in the /etc/passwd file.
- There are seven colon-delimited fields in the entry: the username (lester), the password (in this case, shadowed), **the user ID (512), the group ID (516), the GECOS field (for comments; in this case, Lester Pace), the home directory, and the default login shell.**

- It concerns if there are additional user accounts that were not made by the system administrator. Check to be sure that any daemon, sync, or shutdown accounts or any other accounts that should not be available for remote logon have not been tampered with. Moreover, carefully record each user ID and group ID.
- A user account with a user ID of 0 or 1 is suspicious. These user IDs signify, respectively, access at the root and bin levels. A backdoor for an attacker to get privileged access is presumably present if a typically privileged user account has been elevated in privilege level.

Group Account Investigation

- Group accounts use the group IDs shown in the **/etc/passwd and /etc/groups files.** The file contains a list of the groups and the users that belong to each group. It is significant to remember that a group can exist without having an entry in the group file. The group ID stored in the password file determines a group's membership.
- In the course of your system audit, keep an eye out for any users who are members of super-privileged groups. For instance, a user account that belongs to the bin group should be investigated further because this access gives the user account access to sensitive system files, which is often forbidden.

4.2.5 Identifying Rogue Processes

- Examining an active system makes it much simpler to spot rogue processes. All listening ports and active processes should have been noted during the initial inquiry.
- To ensure the legitimacy of the currently active processes, we need closely inspect them. Additionally, check all binaries connected to running processes and listening services to make sure they haven't been altered.
- We carefully log running processes and listening ports during our initial examination. Upon closer inspection. Although a distinct FTP daemon seems to be functioning, telnet and FTP should both be controlled by inetd.

  (1) By inserting the # as the first character, we can see from /etc/inetd.conf that the FTP service has been disabled.

  (2) The next step is to look in /usr/sbin for any file with the name "ftpd" in the file system:

We are now very close to figuring out the entire scope of this incident after getting the file's time/date stamps and studying the binary.

4.2.6 Checking for Unauthorized Access Points

- An extremely reliable and effective operating system is Unix. Network services are just one of many features that Unix has added throughout the course of its long history.
- The **Network File System (NFS), telnet, finger, rlogin, and a host of additional network services** are all included in the Unix operating system's default installation. As with a phone line linked to a modem, any networked function on a Unix system has the potential to grant illegitimate users some level of remote access.
- X Servers, FTP, telnet, TFTP, DNS, sendmail, finger, SNMP, IMAP, POP, HTTP, and HTTPS are some of the most popular access points that we have observed hackers exploit. Sadly, this is only a partial list.
- All network services should be looked at as potential access points when we investigate the Unix system. A successful intrusion could have already trojanized network services, making them susceptible and giving attackers access to your system.
- Did you discover anything odd during our examination of the configuration files, startup files, and listening sockets? What "regular" services were active on the system when the suspected event occurred? By providing answers to these queries, we can identify potential entry points for hackers into our system.
- Check each potential access point to make sure it is configured securely and has the newest software or patch updates. To ensure that the apps are free of trojans, we can compare checksums with versions of each application that are known to be trustworthy.

4.2.7 Analysing Trust Relationship

- Trust relationships within Unix systems were once a major attack mechanism. Trust between Unix systems can be established using various services.
- The most common services are rlogin, rsh, Network Information Service (NIS and NIS+), NFS, and ssh. Trust relationships save system administrators and users time.
- If machine A trusts machine B, the user of machine B can access machine A without additional credentials. For system administrators managing dozens of systems, this feature is very attractive to use.

- Trust relationships are typically configured through files such as **/etc/hosts.equiv or any .rhosts file** in the user's home directory.
- A trust relationship can be established over ssh via a shared key and her NFS share. Additionally, host-based access controls such as firewalls and TCP wrappers are often configured to allow specific source IP addresses to communicate with protected hosts. This is another form of trust. Examine all possible trust relationships to determine if they played a role in the incident.

Descriptive Questions

Q. 1 What is Windows Forensic Analysis?
Q. 2 What are Forensic Artifacts?
Q. 3 List at least five data acquisition methods for mobile phones.
Q. 4 Top Open-Source Tools for Windows Forensic Analysis
Q. 5 What are the 5 rules of evidence?
Q. 6 What is a file system?
Q. 7 How to perform keyword search in Unix? State advantage.
Q. 8 Explain user and group management in Unix OS.
Q. 9 Write a brief note on Reviewing Pertinent Logs in Unix.
Q. 10 Write a note on Windows and Unix file system.
Q. 11 Write a brief note on Windows Recycle Bin Forensics,
Q. 12 Explain Windows Registry Analysis.
Q. 13 State difference between investigation and Forensics.
Q. 14 Elaborate the Windows OS file recovery process.
Q. 15 Explain the concept of "Trust Relationships".

Chapter Ends...

❑❑❑