A Novel Methodology for Offline Forensics Triage in Windows Systems

Dija S[1], Sreeja S C[2]

[1]Scientist F/ Associate Director, Cyber Forensics Section, CDAC, Thiruvananthapuram
[2]Principal Engineer, Cyber Forensics Section, CDAC, Thiruvananthapuram

Abstract - *The pervasive expansion of digital data leads to a marked increase in the volume of data collected from crime scenes. Consequently, the analysis of evidence gathered in any reported cybercrime is prolonged due to this. Windows Computers have thousands of artefacts holding forensically sound information. The analysis of the bitstream image of storage media collected from the Suspect's system re-covers lakhs of files and folders. Many cyber forensic professionals lack the necessary expertise to pinpoint the optimal starting point for analyzing these files and folders. This paper presents a novel methodology for Offline Cyber Forensics Triage, to retrieve crucial information by analyzing selected files. This crucial evidence may guide the analyst to proceed further in the right direction based on the observed results during the triage. Criminals may adopt various anti-forensics techniques to delete evidence. This paper also discusses the results of experiments conducted based on the described methodology to detect the traces of recently happened activities even after it is deleted from the system. The paper also includes steps to unveil various details of recently accessed files and programs including suspicious processes.*

Key Words: Triage, Offline Forensics, Cyber Forensics, Disk Forensics.

1. INTRODUCTION

Cyber forensics is a branch of forensic science that involves various techniques to recover, analyze, and present digital evidence in a legally admissible manner in cases involving cyber crimes. In traditional offline forensics, Cyber forensic analysis starts with the recovery of files and folders from the forensic image of the storage media collected from the Suspect's computer. Cyber forensic data recovery tools retrieve large amounts of digital data. Analyzing all these artifacts one by one is a tedious task. A usual practice is to look into the questionnaire submitted by the investigative agencies and to answer it one by one. But, this results in the loss of a lot of crucial evidence and that may mislead the analysis in the wrong direction. Cyber forensic analysts commonly use keyword searching and timeline analysis on the recovered files to minimize the artifacts to be analyzed in detail. However, this may not retrieve hidden, erased, tampered, or other malware-related information. Also, the timestomping techniques applied by the criminals may negatively affect the results shown by the tools. So detailed research is needed in this area to identify the artifacts which cannot be accessed or modified by malware or anti-forensics tools.

2. BACKGROUND INFORMATION

It is the need of the hour that digital forensics experts must continually stay one step ahead of data hiding, destruction and obfuscation techniques and any other anti-forensic measures currently in vogue. Numerous freely available and easy-to-use tampering tools make it difficult for forensic scientists to collect legally valid evidence and reconstruct a credible timeline [1]. Cyber forensic triage is the process of prioritizing the artefacts to be analyzed to obtain fast and useful results. Here, the artefacts are ranked in terms of importance or priority based on the evidential value in them. Prioritizing the evidence is paramount to the success of an analyst. The ability to distinguish between vital details for the reported case and irrelevant ones is of ultimate importance in cybercrime analysis. A successful forensic analysis depends on knowing where to find metadata associated with deleted files and how to interpret them. In this paper, we explore a fast and effective forensic analysis of Windows computers which is capable of selecting the highly valuable artefacts as the first step of analysis and guiding the analysts to decide how to proceed in the deep-dive analysis of storage media content.

3. RELATED WORK

There are various models for Cyber Triage for Incidence Response which have been attempted to deal with the entire process related to the analysis of the digital evidence. Due to the need for information to be obtained in a relatively short time frame, the model usually involves an on-site/field analysis of the computer system in question. F. Marturana and S. Tacconi [2] have presented a triage methodology for automating the categorization of digital media using machine learning. The method-ology explained in the paper was applied in two use cases: copyright infringement and child pornography exchange to prove its viability. Marcus K. Rogers, James Goldman, Rick Mislan, Timothy Wedge, Steve Debrota[3] explain another model an onsite approach for the identification, analysis, and interpretation of digital evidence in a short time frame. This approach is employed without the need to transport the system(s) or media back to the lab for an extensive investigation or to procure a full forensic image(s). Muhammad Shamraiz Bashir, Muhammad Naeem Ahmed Khan [4] explain a model for performing digital forensics by describing step-by-step procedures to perform forensic analysis on the compromised machines and store all the necessary logs and system files in a database for later use. This assists the analyst to understand the nature of the attack and can compare the data with the blacklist database to detect novel attack patterns. But the whole process is still very time con-suming. Kyung-Soo Lim, Antonio Savoldi Changhoon Lee and Sangjin Lee[5] demonstrate a methodology to automatically gather evidence according to general categories, such as live data, Windows Registry, file system metadata, instant messaging services clients, web browser artifacts, memory dump and page file. The system emphasises the need for triage in Live forensics. X. Du and M. Scanlon[6] pro-pose a methodology for the automated metadata-based classification of suspicious file artifacts using supervised machine learning to solve the challenge of detecting quickly pertinent file artifacts to a digital investigation. All the above-mentioned pa-pers explain the various methods for triage in both offline and online forensics. They also explain different procedures to perform forensic analysis. However, the pro-posed model focuses mainly on corelating various forensically relevant artefacts extracted from the Windows System and generating comprehensive reports that help the investigator to obtain fast and efficient results.

4. OFFLINE FORENSICS TRIAGE

Various models have been attempted to deal with the entire process related to the analysis of digital evidence. This process is time-consuming for a huge volume of digital evidence and may fail when considering time-critical situations such as child luring, kidnapping, and terrorist threats. It was determined in these situations the need for quick information and investigative leads outweighs the need for an in-depth analysis of all the potential digital evidence. The proposed method explores the location and underlying formats of various encoded and compressed artefacts which may hold forensically sound evidence and ensure fast and effective cyber forensics analysis in traditional offline forensics. This may reveal the details of malware-initiated crimes and crimes in which anti-forensics tools have been used to delete or erase crucial evidence.

4.1 Important Artefacts

In this paper, we have conducted intensive research to identify the crucial artefacts that need to be examined during the investigation of Windows machines. The arti-facts that need to be focused on the initial analysis vary based on the nature and domain of reported crime. In case of a malware-initiated crime, the starting point would be to detect the details of recently executed programs from various potential artifacts. Also, there is a need to identify file-less malware, if any, hiding inside the registry hives or other locations. However, in the case of cybercrime where the criminals extensively use various anti-forensics tools to hide and erase the evidence, some other methodologies are to be adopted to retrieve the evidence. The paper explains the method to locate, process and extract various artefacts needed to find suspicious activities and the metadata associated with the deleted files. The main artifacts that are focused on in this methodology are Prefetch files, Registry files, $USNJrnl file, Browser files, Shimache, AmCache, Recycle bin, and Event logs. These artifacts are in different formats and are to be analyzed in detail to reconstruct the evidence stored in them.

4.2 Analysis of Artefacts

This section explains the result of forensics triage to conduct a comprehensive analysis on Windows Computers which may provide crucial evidence during the traditional offline forensic analysis. Fig. 1 depicts the cyber forensic triage model which extracts various artefacts from Windows machines and corelate them to identify the following critical findings

1. List of deleted/modified/accessed files
2. List of suspicious Programs/Applications
3. Network information
4. List of Timestomped files
5. Web Browser activities
6. Recent User/ Account activities
7. List of Signature mismatched and Encrypted files
8. List of Wiping tools used.

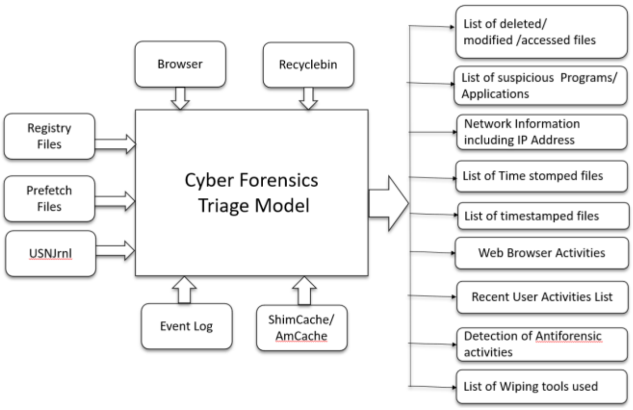

Browser
Recyclebin
Registry Files
Prefetch Files
USNJrnl
Cyber Forensics Triage Model
Event Log
ShimCache/ AmCache
List of deleted/ modified /accessed files
List of suspicious Programs/ Applications
Network Information including IP Address
List of Time stomped files
List of timestamped files
Web Browser Activities
Recent User Activities List
Detection of Antiforensic activities
List of Wiping tools used

Fig -1: Cyber Forensics Triage model for correlating the Windows artefacts

1. **Identification of Recently deleted/ modified/accessed file list.**

The recently deleted files can be identified by processing the recycle bin files and USNJrnl files. When users delete files or folders from their system, these items are often sent to the Recycle Bin instead of being permanently removed from the storage media. The Recycle Bin contains deleted files and folders, which may retain valuable metadata, such as file names, creation dates, modification dates, and file paths. This information can be essential for understanding the events' timeline or tracking the files' origin. This can include the types of files they frequently delete, patterns of behavior, or potential attempts to conceal data. In Windows 10 the Recycle Bin is typically located in the root directory of each partition, and its folder name is $Recycle.Bin.

The files which are Modified/Accessed can be identified by processing the Windows Registry files and USNJrnl files. Windows Registry can be considered a treasure box of evidence related to cybercrime. It holds the details of all recent activities performed by the user on the computer. The following registry keys hold the information about recently accessed files.

1. HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs.

2. HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\ ComDlg32\OpenSavePidlMRU

This is a strong indicator that a suspect had knowledge of all files that were viewed. MRU, or 'most recently used' lists contain entries generated as a result of specific actions performed by the user [7].

$USNJrnl File is an NTFS system file that records each and every change made to files or folders in an NTFS volume. It is stored in the hidden system file $USNJrnl. Successful decoding of the details in these streams may provide very useful information to a forensic investigator to track details of suspicious executables, including every change happening when accessing the program. Even when files are deleted or renamed, the $USNJrnl file may contain records of these actions. The file contains two Alternate Data Streams (ADS), $Max and $J[8]. The $Max contains metadata of the Journal such as the maximum size and $J contains the content of the journal such as the date and time of change in a file, the reason for the change, MFT Reference Number etc.

2. Identification of Suspicious Programs/ Applications

The investigator can identify suspicious events by processing the Event logs. Windows 10 holds event log files in .evtx format. Security Event Log records events based on auditing criteria provided by local or global group policies and many other evidential artefacts. System Event Log holds the details of events related to Windows services, system components, drivers, resources etc. So, the analyst can identify the Services stopped, systems rebooted, crashed services etc. from this log. Application Log records events logged by applications. In addition to these 3 main logs, Custom application logs are available for storing the details of the Task scheduler, Terminal Services, PowerShell, WMI, firewall, DNS related activities. Each of these custom logs also provides various crucial evidence related to the reported cybercrime. Event logs can be found in the location C:\WINDOWS\system32\winevt\Logs. Suspicious events can be detected by processing the Event ID extracted from the event logs. If a particular event is found suspicious then the source application for causing that event can be identified from the logs.

The recently executed process and applications can be identified by processing Registry and prefetch files. Software used by attackers will create a footprint within the Registry, leaving the investigator clues about the incident. The in-depth analysis of registry files will provide valuable insight into the activity that occurred on the system. The following registry key holds the information related to the last executed applications.

HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\LastVisitedPidlMRU

ShimCache or Application Compatibility Cache is a critical forensically relevant artefact in the Windows Registry. When a program is executed, an entry will be created in the ShimCache. This helps in malware analysis to identify the number of times the suspicious version of the program was executed. The Registry Key related to ShimCache can be found and located as

HKLM\SYSTEM\CurrentControlSet\Control\Session Manager\AppCompatCache\AppCompatCache.

Amcache hive in the registry also holds details of executed programs on a Windows Computer. It is present in the location C:\Windows\appcompat\Programs. Information like the executable name, file path, version, and hash value can be extracted by parsing the Amcache hive.

Windows Prefetch Files are a treasure to forensic investigators from which they can get information about the programs that are executed on a Windows Computer. A prefetch file with a .pf extension will be created when a program is executed for the very first time. By analyzing a prefetch file, we can get the name, last execution time, frequency of execution and other details of the recently executed programs. The Prefetch files can be found in the location C:\Windows\Prefetch folder. When an application is not running from its normal location, we can detect it through prefetch file analysis, and it may be categorized as a potentially suspicious application.

3. Network Information

The network-related details can be extracted by processing the Windows registry files. Network configuration parameters of connected networks and details relating to the Network Interface Cards on the system are all stored within the Windows Registry. The following Registry keys hold the network details including the IP Address of the system.

1. HKLM\ SOFTWARE\Microsoft\Windows NT\ CurrentVersion\ Network-Cards
2. HKLM\SYSTEM\CurrentControlSet\Services\ Tcpip\Parameters\Interfaces

The registry also stores information related to the wireless network connected to the system. The registry key location where wireless network information is stored is as follows.

HKLM\SOFTWARE\Microsoft\WindowsNT\CurrentVersion\NetworkList\Signatures\Unmanaged

4. Timestomped Files List

Timestomping is a technique used to backdate a file to a time/date chosen by the adversary. They usually use this technique to make the file's creation and modification time similar to those surrounding it to blend in. This anti-forensic technique can be detected by analyzing the ShimCache and USNJrnl. ShimCache tracks the executable file's last modified date, file path, and if it was executed. If the current file's modified time is not equal to ShimCache modified time, the file can be identified as a timestomped file. Timestomped files can also be detected by processing NTFS $MFT attributes. Among the $MFT attributes, there are $STANDARD INFORMATION ($SI) and $FILENAME ($FN) of interest that contain useful meta-information about the files such as the filename, the extension, the timestamps, etc. [9]. When a suspect changes the time of a file the time in $SI changes and $FN remains the same. If the time stamps extracted for both attributes are different, then we can confirm the file is timestomped. Also, USNJrnl records the modifications done in the metadata of a file. Upon parsing the USNJrnl, BASIC_INFO_CHANGE records the last time the test file had its meta-data altered [10],

5. Web Browser Activities

Browser files contain important information related to Suspects' Internet activities and hence its analysis is indispensable in both offline and live forensic analysis [11]. Monitoring Web browser activities is a crucial part of determining the internet browsing behavior of the suspect. Visited Sites, Cookies, Download History, In-Private Browsing, Content, Searched Keywords, the items bookmarked by the user, installed browser plugins, saved credentials, etc. contained in the browser files can be used for reconstructing the suspect's online browsing behavior. The details of browsers such as Google Chrome, Firefox, and Opera are stored in SQLite format. The browsing information of Internet Explorer and Microsoft Edge are stored as EDB files. But in the latest version of Windows 10, the details of the Edge browser are stored in SQLite format. The browsing details of the safari browser are stored in plist file format. The location of browser files and forensically relevant information which can be reconstructed from browser files is shown in Table 1.

Table -1: Browser file Locations and Artefacts

SI. No	Browser Name	Location	Artefacts
1	Google Chrome	C:/Users/{user}/AppData/Local/Google/Chrome/User Data/Default	Visited URLs, Download history, Bookmarked sites, Search terms, Cookies, Autofill, Login details
2	Internet Explorer	C:/Users/{user}//AppData/Local/Microsoft/Windows/WebCache	
3	Mozilla Firefox	C:/Users/{user}/AppData/Roaming/Mozilla/Firefox/Profiles	
4	Microsoft Edge	C:/Users/{user}/AppData/Local/Microsoft/Edge/User Data/Default	
5	Apple Safari	C:/Users/{user}/AppData/Roaming/Apple Computer/Safari	
6	Opera	C:/Users/{user}/AppData/Roaming/OperaSoftware/Opera Stable	

6. Recent User/ Account Activities

During case analysis, identifying the user activities is very critical. This includes the user search details, run commands, recently visited folders (Shellbag details), and plugged USB information. The Registry key that holds the above-mentioned artefacts is given in Table 2.

Table -2: Registry Keys for storing various User Activities

SI. No	Artifacts	Registry Key
1	Plugged USB Details	○ HKLM\SYSTEM\Current ControlSet \Enum\USBSTOR\ ○ HKLM\SYSTEM\Current ControlSet \Enum\SWD\WPDBUSENUM\
2	Search Terms	HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\WordWheelQuery\
3	Recently Visited Folders	HKEY_CURRENT_USER \Software\Microsoft\Windows\Shell\BagMRU\
4	Run Commands	Software\Microsoft\Windows\CurrentVersion\Explorer\RunMRU\

7. Signature Mismatch and Encryption Detection

Anti-forensics activities are done by cyber criminals to destroy evidence to make it more difficult for investigators to uncover their activities. Detection of these activities is crucial in preserving and strengthening digital evidence. Anti-forensics activities like changing the extension of the file in order to hide it from the preliminary analysis can be identified by the Signature mismatch detection method. Signature-based detection is a technique used to identify the original type of file by checking the signature of a file. For that, a signature database for known file types should be maintained. The signature of the file is compared with the signature of the file type stored in the data-base and check whether the signature is the same. If the signature is different, the file should be identified as a signature mismatch file. Password protection is another anti-forensics technique used by the suspect to hide critical information. Here also the encrypted document files can be detected by checking their signatures.

8. Detection of wiping tools

File wiping is a technique used by adversary teams to delete and overwrite files in a system. The existence of a file-wiping utility can be used as proof of anti-forensics techniques used by the attackers. The execution of a wiping tool like SDelete can be identified by processing the USNJrnl and prefetch file. SDelete tool overwrites the contents as well as metadata of the file it deletes and renames the file several times. By analyzing the prefetch files, number of execution and execution time of the ap-plication can be identified. Compare the execution times of the wiping tool with the files which has been deleted/ overwritten/ renamed at that time. This data can be retrieved by analyzing $USNJrnl file. The amazing aspect of USNJrnl is that it can provide a rolling history of changes of every file. From the USN reason code the operations like 'DataOverwrite' and 'RenameOldName' performed in a file can be identified and thus confirm the usage of the wiping tool.

5. EXPERIMENTATION RESULTS AND DISCUSSION

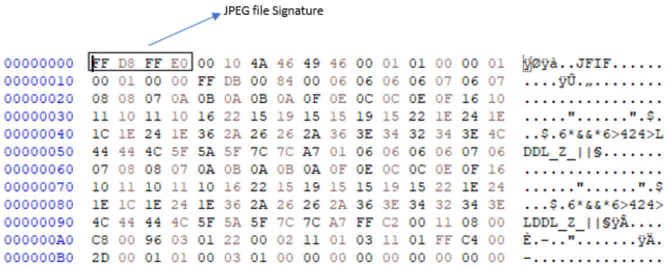
The remnants of recently occurred activities continue to exist in the same location for a period of time even after the original file or programs have been deleted and overwritten from the storage media. Successful reconstruction of the data from these files may provide crucial initial evidence by recovering the details of suspicious crime-related programs and files even after their deletion. We have installed five applications in four Windows 10 machines and taken the bitstream copy of each system. We have performed an analysis of various artefacts extracted from Windows systems and found the following results given in Table 3. The results obtained by correlating various Windows remnants are shown in Table 4.

Table -3: The traces of suspicious application in various artefacts before and after deletion

SI. No	Source	Before Deletion	After Deletion
1	ShimCache	✓	✓
2	Registry	✓	✓
3	USNJrnl	✓	✓
4	AmCache	✓	✓
5	Prefetch	✓	✓

Table -4: Artefacts obtained by correlating various Windows files

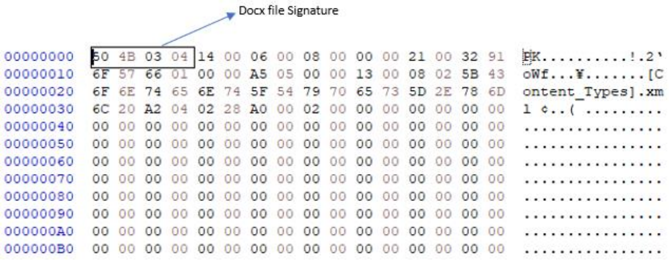
SI. No	Artifacts	Windows files
1	Deleted Files	Recycle bin, USNJrnl
2	Modified/ Accessed Files	NTUSER.DAT (Registry file), USNJrnl
3	Timestomped files	ShimCache, USNJrnl
4	Suspicious Programs	Event logs, Prefetch file, ShimCache, AmCache
5	User Activities, Network Information	NTUSER.DAT, SYSTEM (Registry files)


JPEG file Signature

```
00000000  FF D8 FF E0 00 10 4A 46 49 46 00 01 01 00 00 01  ÿØÿà..JFIF......
00000010  00 01 00 00 FF DB 00 84 00 06 06 06 06 07 06 07  ....ÿÛ.„........
00000020  08 08 07 0A 0B 0A 0B 0A 0F 0E 0C 0C 0E 0F 16 10  ................
00000030  11 10 11 10 16 22 15 19 15 15 19 15 22 1E 24 1E  ....."......".$.
00000040  1C 1E 24 1E 36 2A 26 26 2A 36 3E 34 32 34 3E 4C  ..$.6*&&*6>424>L
00000050  44 44 4C 5F 5A 5F 7C 7C A7 01 06 06 06 06 07 06  DDL_Z_||§.......
00000060  07 08 08 07 0A 0B 0A 0B 0A 0F 0E 0C 0C 0E 0F 16  ................
00000070  10 11 10 11 10 16 22 15 19 15 15 19 15 22 1E 24  ......"......".$
00000080  1E 1C 1E 24 1E 36 2A 26 26 2A 36 3E 34 32 34 3E  ...$.6*&&*6>424>
00000090  4C 44 44 4C 5F 5A 5F 7C 7C A7 FF C2 00 11 08 00  LDDL_Z_||§ÿÂ....
000000A0  C8 00 96 03 01 22 00 02 11 01 03 11 01 FF C4 00  È.–.."......ÿÄ.
000000B0  2D 00 01 01 00 03 01 00 00 00 00 00 00 00 00 00  -...............
```

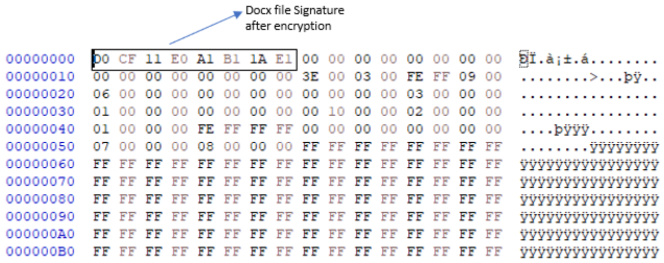
Fig -2: Structure of a jpeg file

For the detection of password-protected files, we have encrypted ten office document files which include MS Word, Excel and PowerPoint files. Fig. 3 shows the hex view of a sample .docx file before encryption. The signature of the .docx file is 504B0304.


Docx file Signature

```
00000000  50 4B 03 04 14 00 06 00 08 00 00 00 21 00 32 91  PK..........!.2’
00000010  6F 57 66 01 00 00 A5 05 00 00 13 00 08 02 5B 43  oWf...¥.......[C
00000020  6F 6E 74 65 6E 74 5F 54 79 70 65 73 5D 2E 78 6D  ontent_Types].xm
00000030  6C 20 A2 04 02 28 A0 00 02 00 00 00 00 00 00 00  l ¢..( .........
00000040  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000050  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000060  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000070  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000080  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000090  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000000A0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000000B0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
```

Fig -3 Structure of a docx file before encryption


Docx file Signature after encryption

```
00000000  D0 CF 11 E0 A1 B1 1A E1 00 00 00 00 00 00 00 00  ÐÏ.à¡±.á........
00000010  00 00 00 00 00 00 00 00 3E 00 03 00 FE FF 09 00  ........>...þÿ..
00000020  06 00 00 00 00 00 00 00 00 00 00 00 03 00 00 00  ................
00000030  01 00 00 00 00 00 00 00 00 10 00 00 02 00 00 00  ................
00000040  01 00 00 00 FE FF FF FF 00 00 00 00 00 00 00 00  ....þÿÿÿ........
00000050  07 00 00 00 08 00 00 00 FF FF FF FF FF FF FF FF  ........ÿÿÿÿÿÿÿÿ
00000060  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
00000070  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
00000080  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
00000090  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
000000A0  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
000000B0  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF  ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
```

Fig -4 Structure of a docx file after encryption

As part of the signature mismatch analysis, we have conducted three experiments. In our first experiment the extension of a picture file with jpeg extension is modified to pdf. The hex view of the jpeg file is shown in the Fig. 2. As shown in the figure, the signature of the file after modification of the extension remains the same. So according to our methodology, if the signature is not matching the file's extension that file will be identified as a signature mismatched file. In our second and third experiment the extension of a picture file with gif extension has been modified to zip and the extension of a document file with pdf extension has been changed to ppt accordingly. As discussed in the previous case the signature of gif file and pdf file remains same even though the extension has been changed. Both these files were detected as signature mismatched files with the help of the signature database which contains the signature of all major type of files. After encryption, the signature changes as shown in Fig. 4. A signature database has been maintained which contains the signature of all known file types. By checking the signature of the encrypted file with the corresponding signature in the signature database, the file can be identified as encrypted.

6. FUTURE WORK

An enormous volume of storage media content is present on modern computers. Therefore, the development of a selective imaging tool in offline forensics is a need of the hour. This necessitates the retrieval of high-priority artifacts and their examination to decide whether a deep dive analysis is needed for further investigation. A tool is being planned for such acquisition and the analysis methodology explained in this paper can be used there. The technology and solution developed as part of the pro-posed project shall be used to support the investigator to initially identify crucial evidence and then proceed accordingly in the further investigation. The various analysis features will help the cyber forensics investigators to effectively point out the suspicious activities that have happened in the suspect machine. In addition to this, con-tenuous research and development are required in these areas since the format and methodologies used in various artefact files may change from time to time with the release of each minor version of Operating Systems.

7. CONCLUSION

One of the major challenges confronted by Cyber Crime Analysts is the vast quantity of digital data to be analysed in a reported cybercrime. The proposed methodology helps the analyst to quickly identify the initial crucial evidence that helps to decide on where to concentrate and how to proceed in the deep dive analysis in Offline Forensics. This paper discusses the details of various artefacts present on Windows systems that hold information related to recently accessed files and programs even after they have been deleted or overwritten from the storage media. These initial key indications identified will lead the investigator to focus more on the specific artefacts in further investigation. This may ensure effective cybercrime analysis and minimize the total time taken to retrieve the evidence. This methodology also reveals suspicious programs and the anti-forensics attempts done by the criminals.

REFERENCES

[1] Michael Galhuber, Robert Luh, "Time for Truth: Forensic Analysis of NTFS Timestamps", The 16th International Conference on Availability, Reliability and Security, 2021.

[2] F. Marturana, S. Tacconi, A machine learning-based triage methodology for automated categorization of digital media, Digital Investigation 10 (2) (2013) 193–204.

[3] Marcus K. Rogers, James Goldman, Rick Mislan, Timothy Wedge, Steve Debrota, " Computer Forensics Field Triage Process Model", Journal of Digital Forensics, Security and Law, Vol. 1(2)

[4] Muhammad Shamraiz Bashir, Muhammad Naeem Ahmed Khan, "A triage framework for digital forensics", ISSN 1361-3723 March 2015.

[5] Kyung-Soo Lim, Antonio Savoldi Changhoon Lee, Sangjin Lee, "On-the-spot digital investigation by means of LDFS: Live Data Forensic System", Mathematical and Computer Modelling, 2012

[6] X. Du, M. Scanlon, Methodology for the automated metadata-based classification of incriminating digital forensic artefacts, arXiv preprint arXiv:1907.01421.

[7] Farmer, D. J. (n.d.). A forensic analysis of the Windows Registry. Retrieved March 13, 2011, from http://www.forensicfocus.com/downloads/windows-registry-quick-reference.pdf

[8] Dija S, Ajana J, Indu V, Sabarinath M, "Cyber Forensics: Discovering Traces of Malware on Windows Systems" IEEE Recent Advances in Intelligent Computational Systems (RAICS), December 03-05, 2020, Trivandrum

[9] ALJI Mohamed, CHOUGDALI Khalid, "Detection of Tiestamps Tampering in NTFS using Machine Learning", The International Workshop on Emerging Networks and Communications, 2019.

[10] David Palmbach, Frank Breitnger, "Artifacts for Detecting Timestamp Manipulation in NTFS on Windows and Their Reliability Forensic Science International: Digital Investigation, 2020"

[11] Dija S, Indu S, Sajeena A, Vidhya J A, "A Framework for Browser Forensics in Live Windows systems", IEEE International Conference on Computational Intelligence and Computing Research(ICCIC).