

Memory Forensics for Analyzing Malicious Activities

Rafid Asrar Prottoy

Student ID: 012 201 067

A Project in
The Department of
Computer Science and Engineering



Presented in Partial Fulfillment of the Requirements
For the Degree of Master of Science in Computer Science and Engineering
United International University
Dhaka, Bangladesh

January 2023

© Rafid Asrar Prottoy, 2023

Approval Certificate

This project titled "**Memory forensics for analyzing malicious activities**" submitted by **Rafid Asrar Prottoy, Student ID: 012 201 067**, has been accepted as Satisfactory in fulfillment of the requirement for the degree of Master of Science in Computer Science and Engineering.

Board of Examiners

1.

Supervisor

Mohammad Mamun Elahi

Assistant Professor

Department of Computer Science and Engineering (CSE)

United International University (UIU)

Dhaka-1212, Bangladesh

2.

Examiner

Prof. Dr. Salekul Islam

Professor & Head of the Department

Department of Computer Science and Engineering (CSE)

United International University (UIU)

Dhaka-1212, Bangladesh

3.

Ex-Officio

Prof. Dr. Md. Motaharul Islam

Professor and Director, MSCSE

Department of Computer Science and Engineering (CSE)

United International University (UIU)

Dhaka-1212, Bangladesh

Declaration

This is to certify that the work entitled “**Memory forensics for analyzing malicious activities**” is the outcome of the research carried out by me under the supervision of **Assistant Professor Mohammad Mamun Elahi**.



Rafid Asrar Prottoy

Student ID: 012 201 067

MSCSE Program

Department of Computer Science and Engineering (CSE)

United International University (UIU)

Dhaka-1212, Bangladesh

In my capacity as supervisor of the candidate's project, I certify that the above statements are true to the best of my knowledge.

Mohammad Mamun Elahi

Assistant Professor

Department of Computer Science and Engineering (CSE)

United International University (UIU)

Dhaka-1212, Bangladesh

Abstract

With the change of era, the growing dependency on the computer and Internet is needless to say in a word. Memory is a very important part of a computer that holds the necessary data that the processor uses. As the CPU's running process data is stored in the memory, capturing and preserving the memory information are very important to detect malicious activities. If the memory is volatile like in RAM, data can be easily lost by overwriting or power failure. So, creating the memory dump from the volatile and secondary memory is invaluable for memory forensics and identifying different malicious activities for forensic investigation. Memory dump information can be used forensically to detect malicious activities within the suspected device. Nowadays, Internet usage is increasing tremendously, so people face many attacks like malware originated from the Internet. The attacker uses the victim's machine to execute their plan anonymously. During the investigation, there will be voluminous amount of information to investigate. As malicious processes are smart enough to hide, finding the malicious processes are not that trivial. Investigators must relate the incident data from the memory dump information to identify the malicious activities. There are many challenges in creating the memory dump from the heterogeneous types of devices and investigating the collected memory dump if investigators do not use the right methods and tools, which will enable to create the memory dump mellifluously and to identify different malicious activities in a short time. Using the right tools and frameworks at the right time, the effectiveness of the investigation can be much better and faster. In traditional processes, there is no structural way to find malicious activity. So, in this project, we have proposed a method for investigating malicious activities in a more structured and efficient way from the captured memory dump and identifying malicious activities from a suspected machine.

Acknowledgment

By the grace of almighty ALLAH, first and foremost, I am thankful to ALLAH for giving me blessings and strength to complete this project satisfactorily. Secondly, I want to thank and also admire the help and guidance of Mohammad Mamun Elahi, Assistant Professor, Department of CSE, for being my mentor and the support and guidance towards the completion of this project. With his supportive guidance, this work was possible. I would like to express my gratitude to Prof.Dr.Md.Motaharul Islam, Professor and Director of MSCSE Program, Department of CSE and Prof. Dr.Salekul Islam, Professor & Director of IQAC, Department of CSE for giving me the opportunity to complete this project. I want to thank all my respected department teachers for their support and warm wishes. Lastly, I also express my gratitude to my family, who always kept me cheering up to complete this project in due time.

Table of Contents

LIST OF TABLES.....	viii
LIST OF FIGURES	ix
1. Introduction.....	1
1.1 Research Motivation and Objectives	3
1.2 Project Contributions	6
1.3 Organization of the Report	7
2. Background.....	8
2.1 Memory.....	8
2.1.1 Cache Memory.....	8
2.1.2 Primary Memory (Main Memory).....	9
2.1.3 Secondary Memory.....	10
2.2 Memory dump	11
2.3 Memory forensics	11
2.4 Malware	12
2.5 Importance of memory in digital forensic	13
2.6 Challenges.....	13
3. Related works	15
3.1 File system mapping in the memory dump and Problem Statement	15
3.2 Memory forensic Tools and Problem statements	15
4. Tools for Analysis and capturing the memory	17
4.1 Software and frameworks	17
4.2 Hardware Requirements of the software	19

4.2.1 AccessData FTK Imager Software Requirements:.....	19
4.2.2 Autopsy Software requirements	20
4.2.3 Volatility framework requirements.....	20
5. Memory Dump creation using different tools	22
5.1 Dumpit.....	22
5.2 Magnet Forensics.....	23
5.3 AccessData FTK imager.....	25
5.4 Belkasoft Live RAM Capturer.....	28
6. Malicious Process Identification from the primary memory dump.....	30
6.1 Accumulation of Materials	30
6.2 Proposed Method	31
6.2.1 Find Volatility.....	32
6.2.2 Profile Name Identification	34
6.2.3 Ghost process identification	36
6.2.4 Parent process identification.....	37
6.2.5 Process camouflage activity identification	38
6.2.6 Execution method identification.....	39
6.2.7 Backtrack the remote connection.....	40
6.2.8 Malicious activity identification	43
7. Malicious activity identification from secondary memory dump.....	45
7.1 Accumulation of Materials	45
7.2 Proposed Method.....	46
7.2.1 Case creation.....	48
7.2.2 Web and E-mail History investigation	52
7.2.3 Storage investigation	53

7.2.4 File type investigation.....	55
7.2.5 Installed Program investigation	56
7.2.6 Reverse engineering.....	57
8. Future works	60
9. Conclusion	61
10. References.....	62

LIST OF TABLES

Table 7.1: Options that Autopsy offers and directions table	46
---	----

LIST OF FIGURES

Figure 1.1: Process of Capturing RAM using Imager Software.....	2
Figure 1.2: Process of Capturing Disk image using Imager Software	3
Figure 1.3: Process of detecting and identifying malicious activities using the framework	5
Figure 4.1: Required configurations of the FTK imager	20
Figure 5.1: Dumpit software dashboard	22
Figure 5.2: Creating memory dump using Dumpit software	23
Figure 5.3: Creation of memory dump using Dumpit software.....	23
Figure 5.4: Magnet forensic software dashboard	24
Figure 5.5: Directory selection of memory dump using Magnet forensic software	24
Figure 5.6: Creating memory dump using Magnet forensic software	25
Figure 5.7: Creation of memory dump using Magnet forensic software.....	25
Figure 5.8: FTK imager software dashboard.....	26
Figure 5.9: Selection of the primary memory on the FTK imager	26
Figure 5.10: Directory selection of memory dump using FTK imager software	27
Figure 5.11: Creating memory dump using FTK imager software	27
Figure 5.12: Creation of memory dump using FTK imager software	28
Figure 5.13: Directory selection of memory dump using Belkasoft software.....	28
Figure 5.14: Creating memory dump using Belkasoft software.....	29
Figure 5.15: Creation of memory dump using Belkasoft software	29
Figure 6.1: Proposed process diagram for identifying the malicious process	31
Figure 6.2: Directory of the Volatility tool.....	32

Figure 6.3: Volatility tool's help command.....	33
Figure 6.4: “imageinfo” command on the Volatility tool.....	34
Figure 6.5: “pslist” command on the Volatility tool.....	36
Figure 6.6: Suspicious processes viewing on the Volatility tool.....	36
Figure 6.7: “pstree” command on Volatility	37
Figure 6.8: Processes parent-child view on Volatility.....	38
Figure 6.9: “psxview” command on Volatility.....	38
Figure 6.10: “cmdline” command on Volatility	39
Figure 6.11: Finding process ID on Volatility.....	39
Figure 6.12: Creating process dump using process ID on Volatility.....	40
Figure 6.13: “connscan” command on Volatility	40
Figure 6.14: Opening process dump on Windows	41
Figure 6.15: Opening process dump in Notepad	41
Figure 6.16: Searching IP address on the dumped process	42
Figure 6.17: Identifying IP address on the dumped process.....	42
Figure 6.18: Creating executable process using process ID on Volatility.....	43
Figure 6.19: Virustotal websites main web page.....	44
Figure 6.20: Virustotal websites result of the process.....	44
Figure 7.1: Proposed process diagram for identifying the malicious activities.....	47
Figure 7.2: Opening a new case in Autopsy Digital Forensic Software.....	48
Figure 7.3: Information fill-up for a new case in Autopsy Digital Forensic Software.....	49
Figure 7.4: Optional information fill up for new case in Autopsy Digital Forensic Software	49

Figure 7.5: Selection of secondary memory dump in Autopsy Digital Forensic Software	50
Figure 7.6: Data source selection in Autopsy Digital Forensic Software	50
Figure 7.7: Dashboard of Autopsy Digital Forensic Software	51
Figure 7.8: Partition viewing in Autopsy Digital Forensic Software	51
Figure 7.9: Web history viewing in Autopsy Digital Forensic Software	52
Figure 7.10: Logged-in email address viewing in Autopsy Digital Forensic Software	52
Figure 7.11: Sending email using the victim's machine	53
Figure 7.12: Visited websites and locations using the victim machine	53
Figure 7.13: Images downloaded into the victim's machine	54
Figure 7.14: More images downloaded in incident occurring time	55
Figure 7.15: Mismatched Extension images	56
Figure 7.16: Software installed into the suspected machine	57
Figure 7.17: Save files into the computer disk which is used in the investigation	58
Figure 7.18: File opening using Steganography software	58
Figure 7.19: Extracting hidden information using Steganography software	59

Chapter 1

Introduction

Computers and other personal devices connected to the Internet already became a part and parcel of our daily lives along with the social media. Along with the ever increasing usage of cyber space, breaches on personal and organizational data as well as malware attacks on targeted systems are also increasing in an alarming rate day by day [1][2]. Memory is an essential part of a computer that holds the necessary data that the processor uses to run user applications. The primary purpose of memory is to store data, information, and instructions. Computer memory is responsible for storing data used later by the CPU and the instructions needed by the CPU to run the processes. As the CPU's running process data is stored in the memory, capturing and preserving the memory information will be required to detect and prevent the threat [3]. If the memory is volatile, then data can be easily lost by overwriting or power failure. So, the need to investigate and find malicious activities capturing the memory data or information and creating the memory dump is critical [4]. Memory dump information can be forensically used to detect malicious activities within the suspected device. It also can be used to find the motive of the incident by the professionals. Memory forensics plays an important role in digital forensic investigation as many malicious activities leave traces in the memory [5]. We can use memory forensics to investigate, identify and detect malware attacks and other malicious activities. Day by day, Internet usage is increasing, and people face many Internet-related attacks like malware. There are many challenges to capturing the memory from the suspected computer/device and investigating that memory dump information for finding the malicious activities [6]. In the case of volatile memory, the most critical challenge is timeliness [7]. If we fail to capture the memory in time, all the information in the memory can be overwritten or lost forever. So, capturing the memory in time is crucial to find forensic information. Although some information remains saved in the windows memory log, many sensitive data will be lost if we can't make a memory dump in time. We can collect the drive information as a memory dump to further investigate the incident. There are many tools available both commercial and open source for investigation into the compromised computers or devices by installing

the tools into the compromised computers or devices [9][10]. There will be a significant possibility of modification or loss of information or, in other words altering the data. In the investigation into the victim or compromised device, we don't want to do that in a special investigation case. Also, there will be a chance of overwriting the memory of the compromised machine as we are running different tools which create new processes. So, it's always preferable to make the memory dump when we can access the compromised or victim device and then analyze that memory dump to identify the malicious process or activity. When the investigation process starts (using different types of tools available for memory forensic investigation), there will be much information that the investigator has to go through. Malicious processes are designed to hide from the system logs, browsing data, the network information can be the source, and many more. An investigator needs go through all the information and try to relate those to the crime or incident. We can find the malicious activities in the memory dump captured from the compromised computer/device. So, it is essential to create the memory dumps correctly and investigate using the correct tools and processes.

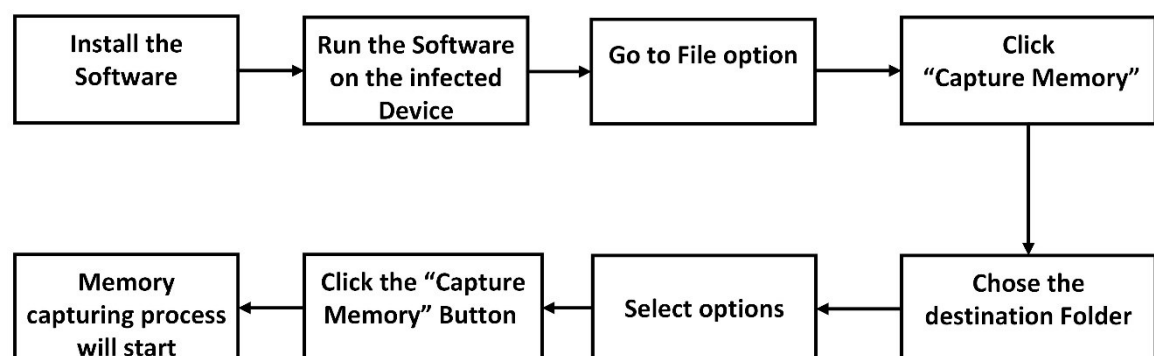


Figure 1.1: Process of Capturing RAM using Imager Software

Considering the importance and challenges of capturing the memory dump, and the relation of the memory dump and malware analysis, our objective in this project is to successfully capture the infected device memory and using a structured approach to analyze the memory dump to find helpful information to detect the presence of the malware. For that, we can take the help of some useful software or tools available on the Internet (all tools used in this project will be open source). First, we can identify an infected device and capture the primary memory as early as possible. After successfully capturing the memory, we can start our investigation. The process of capturing memory (both RAM and hard disk) is shown in figure 1.1 and figure 1.2.

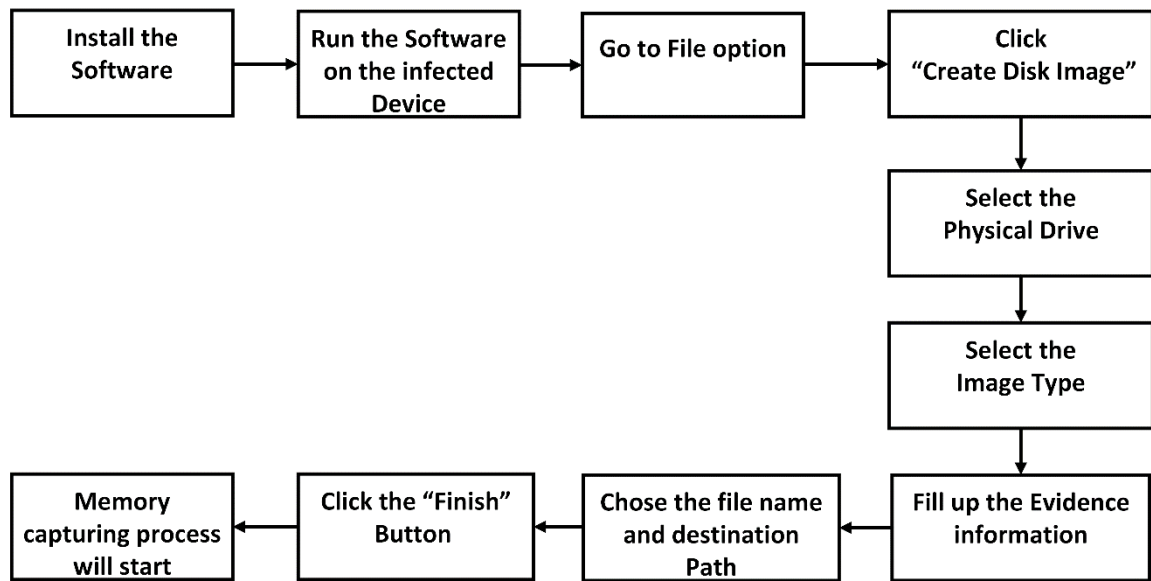


Figure 1.2: Process of Capturing Disk image using Imager Software

1.1 Motivation and Objectives

Although many open-source tools are available, there are some difficulties capturing the memory dump in an efficient way and in a timely manner. First of all, system must be in the live state. Inadvertently shutting down the device will delete all the contents in the RAM. Secondly, as RAM is a constantly changing volatile component of a device, action must be taken quickly, altering, as little as possible as the more time is passed, as well as the more activity that has taken place in the system, the less options exist to be able to recover useful information from the RAM memory. After successfully collecting the memory dump, we need to investigate the acquired memory dump to find the presence of any malware. As we know, malware always tries to hide from the system log, so when it successfully enters the victim machine, it will close the process it created. So, searching for processes with malicious activity along with the commands entered by the threat actor needs some techniques to track them.

Identifying a malicious process is also an important task. We can find which processes are created and which just stopped. Every process has some responsibility, or we can say some work to do. But if a process just ended within a second it started, then it can be a malicious

process [1]. We can find all the time and date when the process started as a running process with the running process information and the process ID that will help us to identify and investigate more in-depth. To determine the process running on the memory dump, which is collected from the compromised computer or device, we need to also know the profile information of the computer or device from which we created the memory dump. Without that profile information, we cannot run the command, which is vital to identify all the running processes on the device. In this work, we will investigate how to do that in descriptive form. We can also find the parent and child process information that will help us to identify which process is the parent process and which process is the child process, and also, in a parent process, how many child processes are running. Conversely, we can find which child process is running from which parent process.

Another big challenge lays in that many unethical professionals are trying to run the software so that no one can find the process that runs unauthorized in the drive memory. To do that, they use the command line to execute the run of the software or a process. If any remote host tries to run something to a device, they use the command line to do the action. For example, software is not directly run by the executable file but from the command line. The information of the command can be found in the command line. But if we close the command line, that information can't be accessed from the command line directly. We can see the command line information from the memory dump using different system defined commands. We can find all the commands written in the command line previously until the memory was captured from the suspected or compromised computer or device. Sometimes in the investigation process, if an investigator finds a process that looks suspicious or malicious and wants to investigate that process in details, then the investigator needs to make the specific process memory dump using the process ID. After that, they can thoroughly analyze that particular process to identify and detect the malicious activities related to the attacks. The malicious process always tries to run when we start the operating system without human interaction. For this reason, when we switch on a device and the operating system reboots itself, the malicious process is also run without human interaction. That is why we can see that process began to run into the device/computer. Investigators must find the process showing this behavior to identify the malicious process

or attacks. We can also determine that kind of process using some frameworks and commands.

Another challenging task during the investigation is that the investigator must know if any connection was established with the suspected/compromised computer or device with other networks. Sometimes the attackers try to access a remote computer with the telnet or SSH and launch an attack using that computer. Using remote access, attackers can run various malicious software or contain malware. So investigators need to know the network information to identify the source where the attack was launched or any malicious code was run into the targeted computer or device using the Internet network. Network-established connection searching is also possible for investigation using some frameworks and plugins. The complete process of detecting and identifying the malicious activities is shown in the figure 1.3.

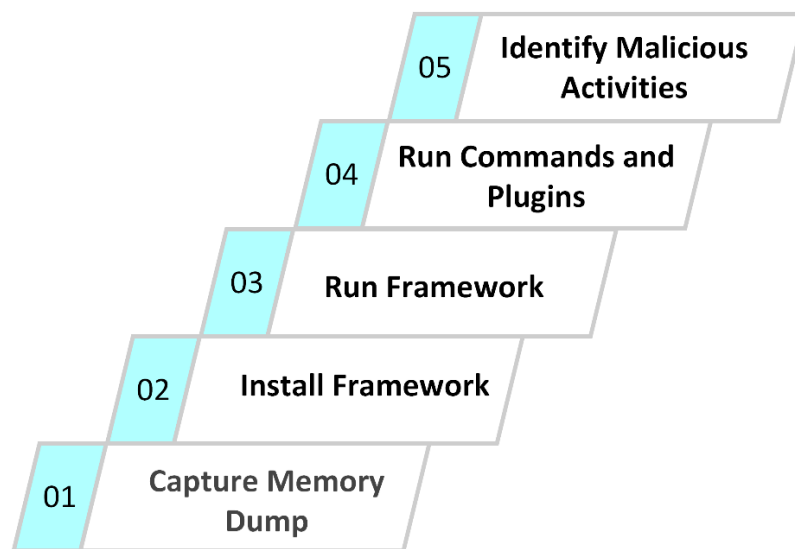


Figure 1.3: Process of detecting and identifying malicious activities using the framework

The steps of the process are described as follows. Many important pieces of information can be investigated in the disk or secondary memory to identify and detect malicious activities. So, disk images or disk memory dumps can be created using different tools. After getting the disk memory dump, the investigator can go through the investigation process into that memory dump and identify or detect malicious activities. After the memory dump of the disk is created, we can use different open-source tools to investigate the memory dump and try to identify the malicious activities. Using the tools, we can extract that information and investigate for identification. In the memory dump, we can find the

programs the attacker can use to do malicious activities using that device. There will be information on the detected files from the computer, which can be helpful. That file could be executable, and that executable file ran the malicious process. Also, in the memory dump, we have browsing history. We can search that history to find malicious activities [2], and the same goes for email information. If the suspected computer/device sent malicious mail, we could identify that using the tools. There can be hidden data in images, videos, audio, and many more. If any image was encrypted with some hidden information that could also be identified by investigation using some available tools.

Our objective of the project can be summarized using the following two-fold goals:

- To study various aspects of volatile memory capture from a live system and issues and challenges related to the efficient searching and analysis of the captured data.
- To select and employ an open-source tool to address the above issues and use a repeatable and structured way to identify the malicious activities done by the attacker or suspected person and the evidence of the malicious activities utilizing the memory dump.

1.2 Project Contributions

Our main contributions of this project work are as follows:

- Using forensically sound methods for capturing the main memory/ primary memory without altering the main artefacts of the suspected/ compromised computer or device using different types of free or open-source software/tools efficiently and in a timely manner. Maintain the evidence integrity of the captured data by comparing the hash value of the primary and captured memory to show that no artefacts were changed or modified during the memory dump creation process and preserved the hash value as proof or evidence of the work.
- Analyze the captured memory dump information using available tools and frameworks to identify the attacks and malicious processes or activities.
- Gathered information and made documentation so the data and findings could be understandable and the entire process is repeatable and structured.

1.3 Organization of the Report

This project report is organized as follows. Chapter 2 presents background concepts required to understand our project related information and challenges (different types of memory, memory dump, and many more). Chapter 3 abbreviates previous work on memory investigation and malicious activities find. Chapter 4 narrates the Software and tool used in our project to identify the activities and their hardware requirements. Chapter 5 describes different types of tools that can create a memory dump. Chapter 6 represents malicious activity identification using a volatility toolkit from the primary memory dump made from a suspected machine. And also proposed a structured method to find malicious processes efficiently. Chapter 7 explains malicious activity identification using Autopsy software from the secondary memory dump created from the alleged device. And also proposed a method to find malicious activity efficiently. Chapter 8 delineates the future works that can be done to investigate malicious activities in the future. Finally, chapter 9 concludes the paper.

Chapter 2

Background

2.1 Memory

Memory can be compared with the human brain. The primary purpose of memory is to store data, information, and instructions. When we say computer memory, it is the storage space in the computer where data is to be processed, and the instructions required for processing the data are stored. We can conclude that computer memory is a temporary storage area that holds data and instructions that the CPU needs to run the process. Computer memory has many types. Some of them are volatile, and others are non-volatile. But the responsibility of the memory is more or less the same.

Primarily, there are three types of memory in the computer system, and they are

- Cache Memory
- Primary Memory/Main Memory
- Secondary Memory

There are many differences between these types of memory. The uses of the memory depend on those features provided by the memory. Those features mainly depend on cost, productivity, build materials, capacity, and more. All the memory types are needed for almost every kind of computer system.

2.1.1 Cache Memory

Cache memory is a semiconductor memory, and the memory speed is very high, which helps to speed up the CPU process. This cache memory is used between the CPU and the main memory (which is called RAM) as a buffer. From the random data, the most continually used part of data and program by the CPU is used to hold in the cache memory. The most commonly used part of the data and program are transferred from the RAM to cache memory. The operating system handles this memory transfer between the RAM and

cache memory, and from the cache memory CPU, then access those data and program. There are some key advantages and disadvantages of the computer's cache memory.

Advantages:

- Cache memory is comparatively faster than the main memory (RAM) system.
- Cache memory comparatively devours less access time than the main memory.
- Cache memory only stores those programs executed in a small amount of time.
- This memory only stores those data temporally.

Disadvantages:

- The capacity of the cache memory is very limited.
- Cache memory is very costly memory.

2.1.2 Primary Memory (Main Memory)

The computer's most recent and currently working data and instructions are mainly held by the primary memory type, also called main memory. The capacity of the primary memory is limited, and when we restart or switch off the computer system, all the data held by the primary memory will be lost. The semiconductor device usually makes this primary memory. This type of memory is typically slower than the registers. Primary memory holds or stores the data and instructions required to be processed. An example of primary memory is RAM. There are some key advantages and disadvantages of the primary memory:

- This memory is mainly semiconductor memory.
- This memory is mainly the working memory of the PC.
- A computer system cannot run without the primary memory.
- Primary memory usually is much faster than secondary memory.

Disadvantages:

- Primary memory is volatile.
- Price of the primary memory is higher than the secondary memory.
- All the data stored in the primary memory will be lost if we restart or switch off the computer.
- Data will be lost if a power failure occurs.

2.1.3 Secondary Memory

Secondary memory is also known as the non-volatile memory or external memory. Secondary memory usually is slower compared to cache or primary memory. The CPU of the computer does not directly access this type of memory. This type of memory is generally accessed by the input and output routines. The information or data that the CPU will use first loaded into the main memory. After that, the CPU will access the main memory to use that data. This type of memory is mainly used to store the data/information permanently. An example of the secondary memory is HDD or CD-ROM.

Advantages:

- This type of the memory is mainly optical and magnetic memory.
- This memory is non-volatile.
- Data of the memory will not lose if any power failure happens.
- Data will be permanently stored in the secondary memory until we delete them.

Disadvantages:

- Secondary memory is much slower compared to primary memory.
- A computer system can run without the presence of the secondary memory.
- This type of memory is only used for data storage and can't access by the CPU.

A process data/information that the processor will use, or CPU typically stored in the volatile memory. As process data is stored in the memory. Since memory is volatile, capturing and preserving that memory information is vital to detect and prevent the threat that can be done individually or in the organization [1]. From the previous discussion, we can easily understand that if the computer is switched off or faces any power failure, we will lose all the data stored in the primary memory. So, capturing the volatile memory is very important, and also it helps us to make an analysis that captured memory. That can be done by memory dumping.

2.2 Memory dump

Memory dump is simply a snapshot capture of computer memory data, just like we take a screenshot of a screen to capture that screen information. A memory dump can contain much information that will become later forensic for investigation. In any incident like a security breach or system failure, a previous memory dump can be used to analyze the problem or detect the attack or malware [1]. A memory also dumps retention information of that particular computer or device, which will help us to mark why the unexpected happens. Suppose we can capture the memory dump from the device or computer in perfect time. In that case, we will have many forensic data which will be valuable for the investigator to examine the incident. That is done by memory forensics.

2.3 Memory forensics

Memory forensics, also known as memory analysis, alludes to the analysis of the valuable data or information located in the captured memory dump, which memory is volatile. Information security professionals conduct memory forensics to investigate if any attacks or malicious actions were launched by that suspected device whose device memory was dumped or captured. We can analyze the runtime system activity using memory forensics, including the open network connections and commands recently executed before memory capturing happened. In most cases, critical data regarding threats and attacks will exist in the system memory, like chat messages, injected code, network connection, encryption key, browsing history, and many more. As every process, data, or information first loads into the memory after the CPU uses that data, most of the information we need from the computers is in that memory. Most of them are non-cacheable. Without the memory dump, that information will be lost after a restart or overwrite. As we know, main memory has limited space and is also volatile. Regular and malicious programs must be loaded in the memory for execution. So, if we capture that memory and investigate that memory dump, we can track the attack and detect malicious programs. As the usage of the Internet in or day to day life is vastly increasing, we are facing many Internet related attacks like malware.

2.4 Malware

Malware is a short form of malicious software that can be a virus, trojan, worm, or other harmful computer program used by a hacker to gain access to sensitive information or destroy other processes. The term malware is referred to any software which is intentionally designed to cause damage to a computer, its network, or its server. The software identified as malware is altogether its intended use, not a technique or technology used to build that software. We always overlap the terms virus and malware. A virus is a type of malware, so all viruses are malware but not every malware is a virus. There are some significant types of how malware spreads [3].

A worm is a malicious software that has the power to reproduce itself and that can spread from one computer/device to another computer/device.

A virus is a segment of computer code that is used into a program's code and nipping it to do some malicious action, and it also can spread itself by tweaking other programs.

A trojan can masquerade itself and trick the user into activating it to do the damage it created for. Typically, a trojan cannot reproduce itself, but it depends on the software's creator and purpose.

Adware distributes unwanted or malicious advertisements, which can annoy anyone during work. These malwares are not directly harmful to the users who browse Internet content. Though it is not too malicious, but it helps to make others annoying because these types of ads can lead users to download harmful applications. That downloaded application can be directly toxic to the users.

Traditional malware infects devices using executable files though fileless malware doesn't affect files directly. For a notable example – Operation Cobalt Kitty was fileless malware attack. The Oceanlotus Group penetrated various corporations in the last six months before being discovered.

A bot is a software program that performs an automated task without human interaction. Creating a botnet, a bot with infection spreads to other devices. This type of bot-compromised machine can be controlled, and hackers use it to create massive attacks by it.

By using spyware, the cybercriminals can monitor user activities. They use keystrokes to log in, and it helps cyber criminals access all user information using antivirus software. Users can prevent spyware. They can also use anti-tracking browser extensions to remove spyware from the user's site to site.

Ransomware is one kind of malicious activities which encrypts and holds data for not paying fees to hackers timely. If payments are not being paid, victims get threatened. It has been shown, however, that victims may still lose data even if they pay the required costs.

2.5 Importance of memory in digital forensic

In digital forensics, memory forensics is referred to as memory analysis. Exploration of volatile data remains in the computer's memory dump. Generally, it's not easier to detect tracks on hard drive data. Information security professionals can easily see malicious behaviors or invasions using memory forensics. Volatile data remains a temporary memory when a computer is in a live form. When a computer gets switched off, volatile data is lost instantly. It dwells as a volatile storage in a computer. It also includes data such as clipboard information, browsing data, and message information. For example - without saving data or information, if anyone works on a word document or other page, that will be lost by switching off the computer. A memory dump is a snapshot capture of computer memory data. It is also named a core dump or system dump. Memory dump has also come from specific instants. It can take on valuable forensics data about the system's state before an incident can happen. Memory dump comprises RAM data that can be used to identify an occurrence's detailing. It also can be used for the identification of malicious activities.

2.6 Challenges

Although we can capture the memory of when the incident happened, and information about the incident appeared, there will be some challenges in successfully capturing the memory dump from suspected computers or devices. Some of the challenges of capturing the memory dump were discussed in the previous part of this paper. Not only will the devices/computers' memory dump capturing be challenging, but there will also be some challenges with the evidence maintenance and investigation processes. As we collect the memory dump from the suspected computer or device in live form, there will be a high chance of altering the computer or device information. Also, in the investigation process, we have to consider the challenge of changing the artifact's information hashed value. All

of the challenges that might or will be challenging during the entire process of analyzing and identifying the malicious activities are as follows:

- Capturing from the suspected computers/devices memory dump within the perfect time can be difficult.
- Suspected device/ computer can be switched off intentionally or unintentionally. So, capturing the memory dump from the live computer or device can be difficult in that case.
- Using the right tools for the memory dump-capturing process can be difficult as the configuration of the computer or device may vary.
- During the memory dump capturing process from the suspected computer or device, all the information will be lost if the investigator faces power failure.
- If the investigation process is done into the suspected computer, then there will be a chance of altering the artifacts. Captured and investigated memory dump's hash value must be the same.
- Memory size of the suspected computers or devices may vary, so the size of the memory dump can't be defined. The memory dump's size needs more time to investigate all the information in the memory dump.
- Many different tools can be used to collect memory dumps, but extraction and efficient searching for useful information can be challenging.
- Many different tools can be used in the investigation process for malicious activity identifications, but efficient searching for malicious activities, including malicious processes, can be challenging.
- Many investigation tools do not support all the operating systems and versions of the memory dump. So, choosing a tool can be challenging.
- Different types of memory dump extensions of the suspected memory dump are possible, and many tools may not support that specific extension.
- Identifying any hidden process can be challenging as they do not need human interaction.
- Identifying the processes remotely installed into the system can be challenging.
- There will be a large amount of data when we extract the memory dump using different tools. However, identifying malicious activity from that vast data can be difficult without any specific process.
- Identifying different types of multimedia files that may contain hidden malicious activities can be challenging.

Chapter 3

Related works

This chapter of the project book discusses related work for memory dump creations and analysis from the created memory dump. In the section 3.1, memory dump-related research and problem statements discoursed. In the section 3.2, digital memory forensic tools and frameworks related to research and problem statements were deliberated.

3.1 File system mapping in the memory dump and Problem Statement

In this section, we will discuss the memory mapping into the memory dump of the machine. In the [4] paper, the authors discussed the different file system mapping from the suspected memory. The authors mentioned a file carving method from the memory image. But also identify the method's effectiveness, which is far less in the present time as modern operating systems try to avoid the fragmentation on the disk. Paging is used to load the memory from the hard drive at current systems. From the DLL files (Windows shared libraries), file types can be identified as DLL holds the code that is used by multiple programs. But using the DLL files, searching the current running program or any other extension can be very difficult as it doesn't show the programs in a structured way. Also, extra information like start time and date can't be found easily either. The authors [5] compared different imagers performance, and from those comparisons, the FTK imager did a good performance of capturing the memory dump from the suspected memory machine. So, if anyone uses that toolkit, then the possibility of searching will increase. If we use a memory dump analysis software as we have used, then all the data will show in a structured way.

3.2 Memory forensic Tools and Problem statements

In this section, we will discuss the tools used for memory forensics and the findings that researchers did. In [6], the authors discussed different tools and frameworks used in the investigation process of memory forensics. The authors compared those tools with what they couldn't and what they couldn't. But at present, we have some more tools and frameworks that can perform the task better than the tools discoursed by the author, and most important, these tools and frameworks are open-source, so we can develop our

customized plugins to use them when it is needed. So, if a toolkit can perform all the tasks an investigation needs, it will be more convenient and efficient for searching the malicious activities from the memory dump. Also, the tools which the authors discussed have a platform-oriented problem. Most of the tools are platform-oriented, so most of the standard operating systems like Windows, Linux, and macOS, this tool will not support all operating systems. Also, most tools help with a limited number of operating system versions and file format support. Tools and frameworks need to be supported by almost all operating system types, and the number of file support formats has to a sufficient number of amounts. The features offered by the tools are pretty enough. Still, the investigator needs as many features and plugins to investigate the image efficiently. Investigators can investigate in more structured ways, and the evidence will be more robust. In [7] author discussed the web browser forensic for detecting malicious activities. From the browser cookies, user system inside information, and DNS cache history can be the perfect place to search malicious activities. But if we search that way, there will be a chance of altering the artifacts of the machine. The hash value can be changed if any minor artifacts are altered. So, in the digital forensic investigation process, this will be less acceptable. Also, if we search only from browser activities, sometimes it won't be enough to identify the malicious activities. In [8] author discussed the Autopsy digital forensic toolkit searching method using keyword search. Utilizing the keyword search autopsy, specifically search from the disk image and show all the results of that specific keyword. But how the keyword will conduct in the actual investigation scenario and other autopsy features was not discussed. Autopsy offers many valuable features which can be very useful in the investigation process of the disk image. The plugins of the autopsy digital forensic tool can analyze the disk image and show helpful evidence that can be later investigated for malicious activities finding more accurately.

Chapter 4

Tools for Analysis and capturing the memory

4.1 Software and frameworks

For capturing the memory dump and analyzing that memory dump, we need something that offers some related features [9]. For that, we can take the help of some useful software or some frameworks available on the Internet (all tools used in this project will be free or open source). Some tools are free of cost or open source and offer capturing memory features. One of them are AccessData FTK Imager. This is a forensic toolkit, and we can capture current primary memory using it. Using FTK imager software, we can create memory dump and paging files for 32-bit and 64-bit windows systems [9]. AccessData FTK Imager also supports the Linux and Mac operating systems as well. For Linux operating system, investigators can use a graphical user-friendly all-inclusive AccessData FTK Imager paid GUI. That will work as we are doing on the windows operating system version. The features will also be the same as the windows operating version. As we are using the free version, our features will be less than the paid version of the AccessData FTK Imager. But the free version of AccessData FTK Imager has all the features we need to conduct our project work, like capturing all the information from the suspected computer or device as a memory dump. A Mac version of AccessData FTK Imager also supports the Mac operating system. We can download the AccessData FTK Imager software for the Mac version and install it like other software installation processes for the Mac operating system. Some unique features make AccessData FTK Imager different from other imager software. The most magnificent of them is the performance of the imager. As far as we know, it is the only forensic tool that uses parallel actions utilizing the multiple cores of the CPU. This helps the tool boost the imaging process. From the available documentation of the AccessData FTK Imager, they claim that investigators can use this imager more efficiently (more than 150% than other tools) four times better than other imager software in the investigation process. It will take some time, depending on the RAM size and how much data are stored in the RAM—capturing memory using AccessData FTK Imager is not that difficult.

For analyzing the memory dump, very few tools are available on the Internet that offer some related features. One of them is Autopsy. An Autopsy is one of the best open-source platforms that is the easiest, quick, most efficient, and most useable in almost every computerized device investigation. Using this open-source software called Autopsy, we can analyze various device memory dumps like HDD's, SSD's, flash drives, smart mobile phones, and memory card. This software was initially created for only the windows operating systems, but now we have support for the Linux and Mac operating systems. The first version of the Autopsy software was released in back 2001. Autopsy provides some organized information that can help in the analysis part. This Autopsy software has a graphical interface that displays the results, which can be very helpful during the investigation process. This software is automated and supports intuitive workflow [10]. Autopsy processes the memory dump and provides organized data. From that, we can visualize process information and some other information. There are some selected options that autopsy process for us to investigate the memory dump. Using the Autopsy software, we can extract the browsing information of the suspected computer or devices from the memory dump we created. We can also extract the mailing information from the memory dump of the suspected computer or device. We can see the index search of the computer used in the crime and from which we created the memory dump. We can also investigate the deleted file of the suspected computer, which can be found using the software. There are also different types of Images, Video, Audio, and Documentation sections from which we can investigate and find malicious activities. Although we use the volatility framework to investigate the network connections, we can also do that using the Autopsy software. But because of the convenience of other artefacts, we are using the volatility framework to investigate from the main memory dump.

For analyzing the main memory (RAM), there are fewer options available in the current marketplace, which is free and open source. One of them is volatility. Volatility is an entirely open-source framework used wildly in the investigation field. Volatility contains a vast collection of tools for analyzing the memory dump information for malicious activities and also for malware identifications [11]. This volatility framework was implemented in python language under the general public license. Volatility uses the volatile memory information (RAM) to monitor the running process and its state information. Because of this, it can investigate the network, ports, connections establishment, process list, process parent-child relations, DLL, and registry files sections.

Using volatility, we can extract the command line writing information and clipboard information from the memory dump created by the suspected computer or device [12]. As the framework is open source and written in Python, we can also right our contribution if we need any extra features. This volatility framework supports all versions of operating systems like Windows, macOS, and Linux. This framework has efficient and comparatively faster algorithms that make this tool analyze from the created memory dump (from suspected computer/device RAM) without inappropriate memory use and unwanted overhead. Volatility foundation has a massive community with related professionals, so it has the most recent features associated with the investigator and worker needs. Lastly, Volatility was constructed by the digital forensic, malware, and incident response professionals to do the most effective analyzing tasks.

4.2 Hardware Requirements of the software

4.2.1 AccessData FTK Imager Software Requirements:

AccessData FTK imager software requires a minimum of 4-core processors. But they recommended a minimum of 8-core processors. The minimum physical memory of the computer system is 8 Gigabytes, but for fluent use, they recommended 16 Gigabytes. If we use 16 gigabytes of RAM, then the process of memory capturing will be more efficient than 8 Gigabytes of RAM as the RAM size is double. The software will run if we use HDD as a storage drive, but in the recommendation of the software specification, they preferred the SSD storage drive. As we are using the Windows operating system, the software specification recommends a Windows 7 64-bit system or later [9]. If we use a flash drive (pen drive) to store the memory dump file from the suspected computer or device using AccessData FTK Imager Software, then the flash drive has to be a minimum of 3.1 generation 1. Still, as the RAM size can be a maximum of 64 GB, we have to ensure that the flash drive size is greater than 64GB, as the memory dump size will be larger than the RAM size. Using 128 GB of the flash drive will be better because there will be no tension of flash drive memory overflow.

Laptop Install		
Component	Basic	Recommended
Processor	4 cores	8 cores
Memory	8GB RAM	16GB RAM (2GB/core)
Storage	7200 RPM	SSD
OS	Windows 7 64-bit	Windows 7 x64 / Server 2008 R2
Network	1Gbit NIC minimum	1Gbit NIC minimum
Other	USB interface for license dongle unless using soft dongle	

Figure 4.1: Required configurations of the FTK imager

4.2.2 Autopsy Software requirements

Autopsy is a powerful software for executing the investigation process on the created memory dump from the suspected computer or device. We need the minimum configuration of the device for running and using this powerful software. The 64-bit version of Autopsy requires a minimum of 8GB RAM (16 GB recommended). As we are investigating from the memory dump of the disk memory (secondary memory) or flash memory (secondary memory), the memory size will be huge, so we need extra RAM to support the running process. That's why they recommend a total of 16 GB of RAM to utilize the efficiency of the software. The minimum number of CPU cores required for the efficient running of the software is 4. When the 64-bit version of Autopsy is installed on Windows, it will be limited to a maximum heap size of 4GB, leaving the remaining memory for the operating system, the internal Solr text indexing service, and other applications [9]. If you wish to change the maximum heap size after installation, you can change the Maximum JVM Memory value in the Runtime section.

4.2.3 Volatility framework requirements

We are using the volatility framework to investigate the captured memory dump of the Random Access Memory (RAM) from the suspected computer or device. We are using version 2.6 of the volatility in our project. There is also a new version of the volatility, version 3, which is in the active development phase. More plugins are added with this version daily to improve the efficiency of the investigation of the memory forensics field. This software can run with the minimal hardware configuration of the computer. Still, if we use a less powerful computer, it will do the work, but the process of workflow and investigation will be more time-consuming. So, the more we use a higher configuration,

the less time it will take to process and extract the information from the memory dump. A pre-requisite for installing the volatility tool on the computer is Python 2.7. As we know from the earlier discussion, volatility was written in Python, so Python 2.7 is required to run the tool on a Windows computer. We also need the ujson (parsing library of JASON) and pillow (imaging library of python) libraries for running the tools on the Windows device [13].

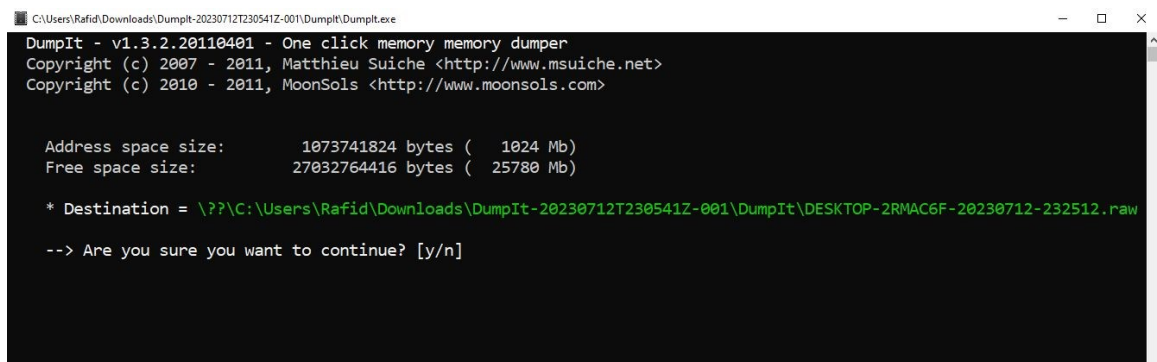
Chapter 5

Memory Dump creation using different tools

Our main goal for this project is to find malicious activity from the memory dump captured from the suspected device or computers. To achieve our goal, we have to investigate the memory dump and try to find the presence of malicious activities done from the alleged computer or done into the suspected computer or device. To analyze the memory dump captured from the suspect computer/device's RAM, we first have to capture the memory. Some free and paid available tools can be used for capturing the RAM. This kind of software can be managed from the Internet. Some of them have paid versions, so if we need to use that kind of software, we must first purchase it. Some free tools are discussed here, and the detailed process of creating the memory dump is also shown below.

5.1 Dumpit

MoonSols Dumpit is a memory-capturing tool that can be executable in Windows 32-bit and 64 bits. The Dumpit is a free and straightforward tool that Matthieu Suiche wrote. We can download this software for free and create a memory dump from the suspect device. This Dumpit is a very compact tool that can capture the contents of the target device's system RAM. Dumpit software is a console utility, but we won't need to open the command line. We have to double-click the .exe software. After that, the software will execute. The copy of the RAM contents that we call memory dump will save to the current directory of the device.



```
C:\Users\Rafid\Downloads\DumpIt-20230712T230541Z-001\DumpIt\DumpIt.exe
DumpIt - v1.3.2.20110401 - One click memory memory dumper
Copyright (c) 2007 - 2011, Matthieu Suiche <http://www.msuiche.net>
Copyright (c) 2010 - 2011, MoonSols <http://www.moonsols.com>

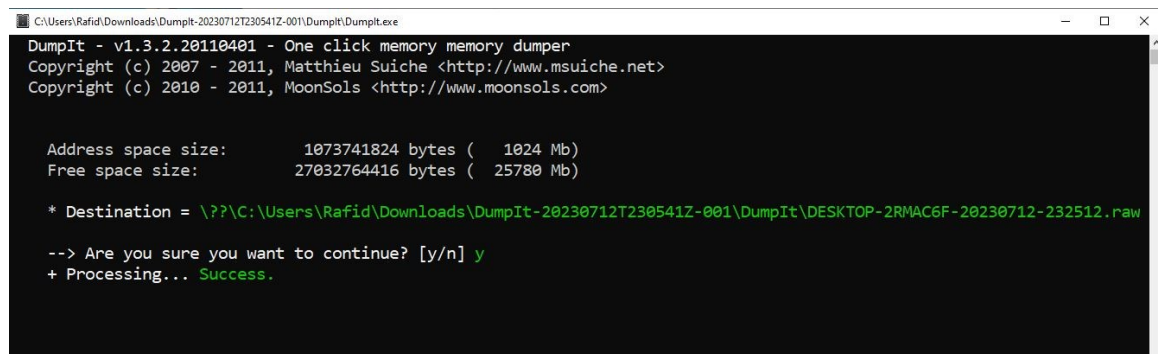
Address space size:      1073741824 bytes ( 1024 Mb)
Free space size:        27032764416 bytes ( 25780 Mb)

* Destination = \??\C:\Users\Rafid\Downloads\DumpIt-20230712T230541Z-001\DumpIt\DESKTOP-2RMAC6F-20230712-232512.raw

--> Are you sure you want to continue? [y/n]
```

Figure 5.1: Dumpit software dashboard

After opening the software, we have just to hit the “y” button, and that’s it! The software will start the work, and after finishing, the memory dump will be visible in the current directory in a file with .dmp extensions.



```
C:\Users\Rafid\Downloads\DumpIt-20230712T230541Z-001\DumpIt\DumpIt.exe
DumpIt - v1.3.2.20110401 - One click memory memory dumper
Copyright (c) 2007 - 2011, Matthieu Suiche <http://www.msuiche.net>
Copyright (c) 2010 - 2011, MoonSols <http://www.moonsols.com>

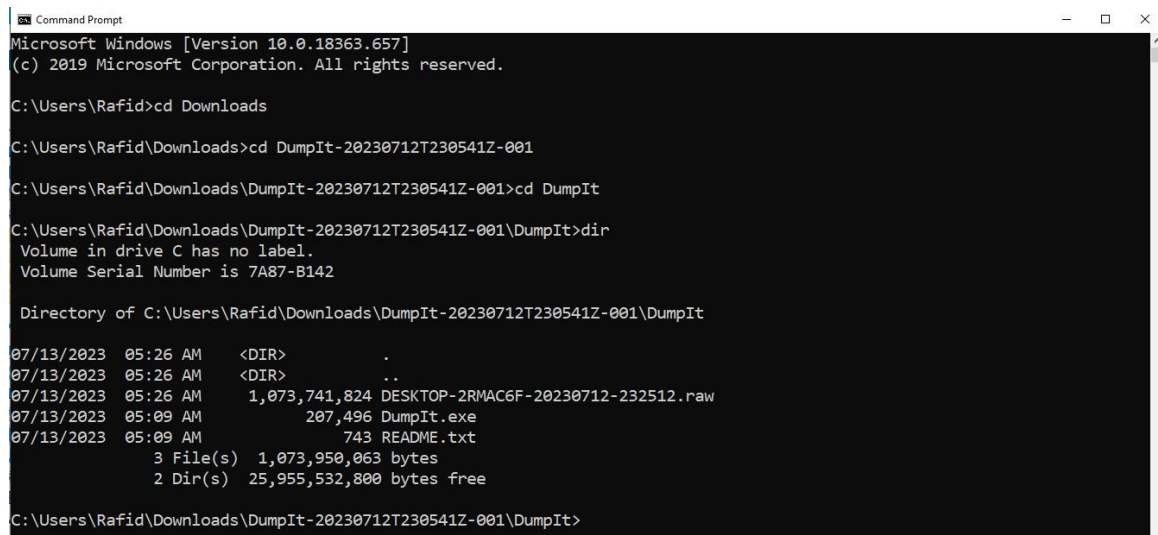
Address space size:      1073741824 bytes ( 1024 Mb)
Free space size:         27032764416 bytes ( 25780 Mb)

* Destination = \\?\C:\Users\Rafid\Downloads\DumpIt-20230712T230541Z-001\DumpIt\DESKTOP-2RMAC6F-20230712-232512.raw

--> Are you sure you want to continue? [y/n] y
+ Processing... Success.
```

Figure 5.2: Creating memory dump using Dumpit software

The memory dump, also known as a memory image, is successfully created using those simple steps.



```
Microsoft Windows [Version 10.0.18363.657]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Rafid>cd Downloads
C:\Users\Rafid\Downloads>cd DumpIt-20230712T230541Z-001
C:\Users\Rafid\Downloads\DumpIt-20230712T230541Z-001>cd DumpIt
C:\Users\Rafid\Downloads\DumpIt-20230712T230541Z-001\DumpIt>dir
Volume in drive C has no label.
Volume Serial Number is 7A87-B142

Directory of C:\Users\Rafid\Downloads\DumpIt-20230712T230541Z-001\DumpIt

07/13/2023  05:26 AM    <DIR>          .
07/13/2023  05:26 AM    <DIR>          ..
07/13/2023  05:26 AM             1,073,741,824  DESKTOP-2RMAC6F-20230712-232512.raw
07/13/2023  05:09 AM             207,496      DumpIt.exe
07/13/2023  05:09 AM              743      README.txt
               3 File(s)  1,073,950,063 bytes
               2 Dir(s)  25,955,532,800 bytes free

C:\Users\Rafid\Downloads\DumpIt-20230712T230541Z-001\DumpIt>
```

Figure 5.3: Creation of memory dump using Dumpit software

5.2 Magnet Forensics

The Magnet Forensics is a RAM-capturing tool that is entirely free and can capture the memory dump of the suspect’s system/device. It allows investigators to analyze and help recover valuable evidence that can only be found in the system memory. We can capture the memory contents in RAW format using this software.

First, we have to double-click on the executable magnet.exe to start the capturing process of the RAM. There is a section that has a button called “Browse.” From here, we must select the path to save the captured memory dump and the file’s name.

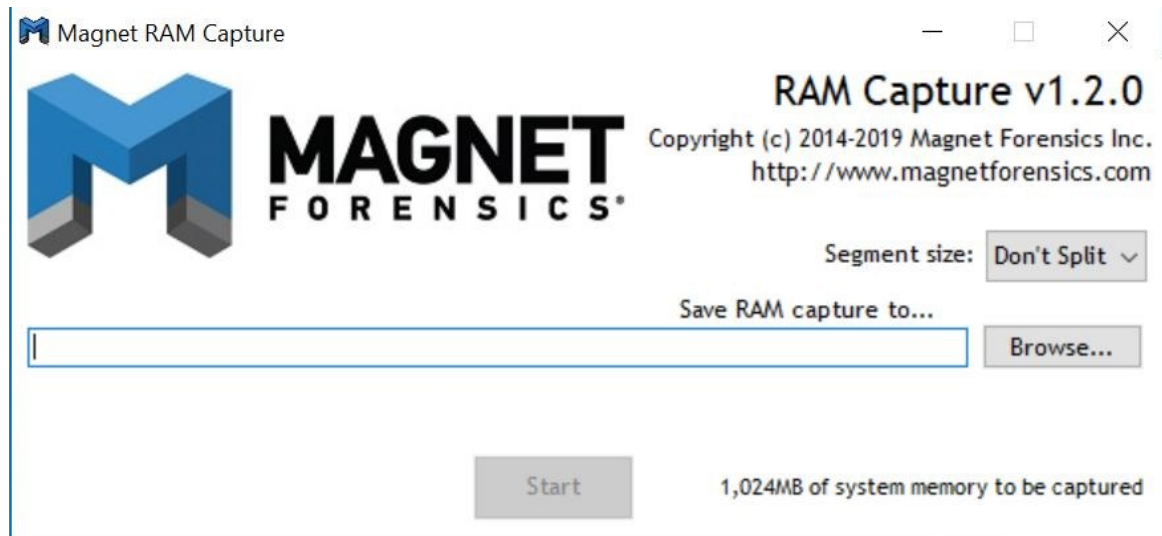


Figure 5.4: Magnet forensic software dashboard

After selecting the path and creating the name, we must hit the “Start” button.

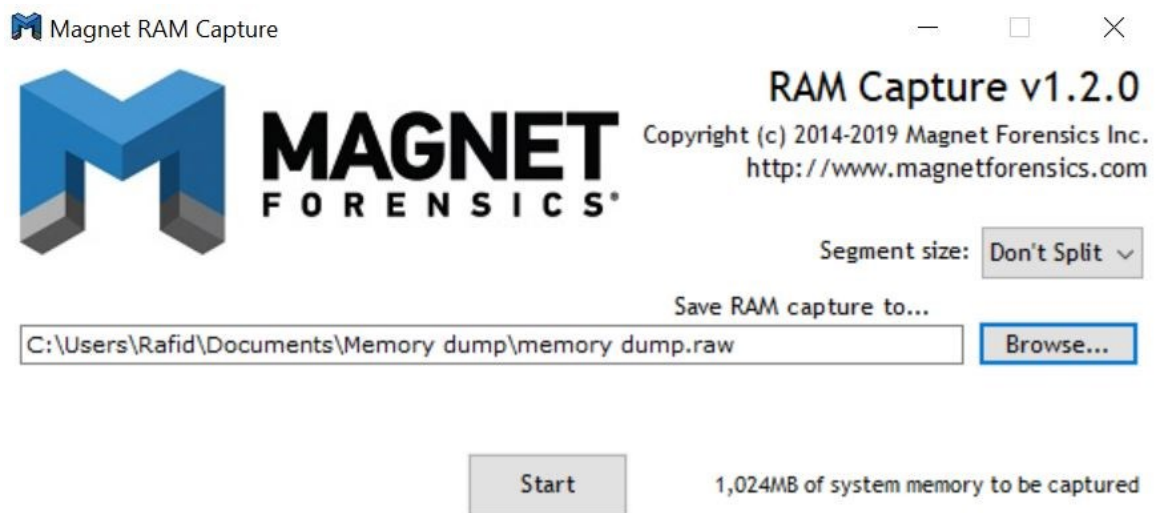


Figure 5.5: Directory selection of memory dump using Magnet forensic software

After clicking the Start button, the process of RAM capturing will start. Process compilation is dependent on the size of the RAM. If the memory is small, it will take less time to complete the creation of the memory dump, but if the memory size is big, it will take more time to create the suspected computer or device.

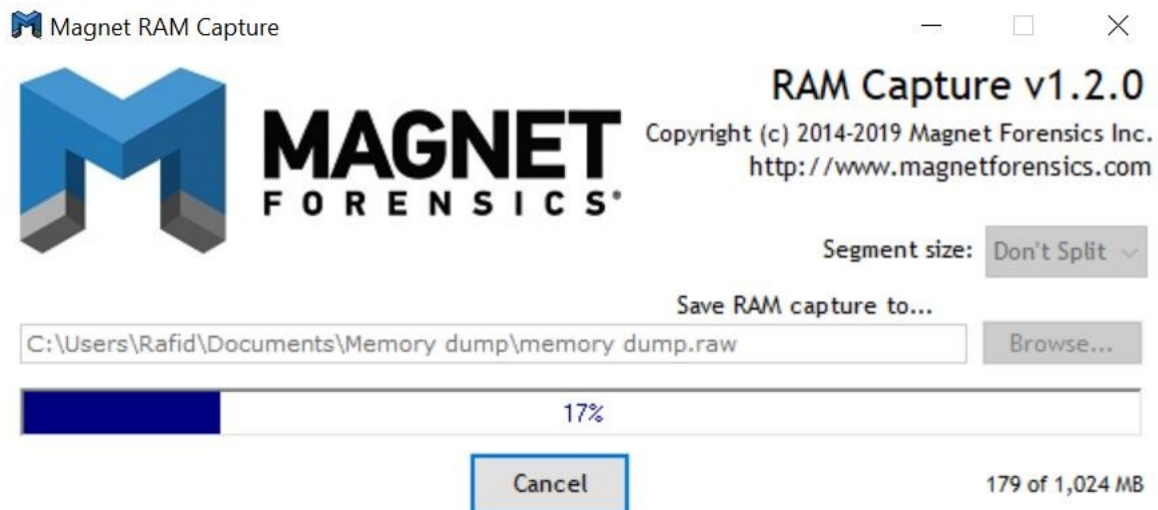


Figure 5.6: Creating memory dump using Magnet forensic software

When the memory image creation process is completed, there will be a message saying the RAM is successfully saved to the selected path.

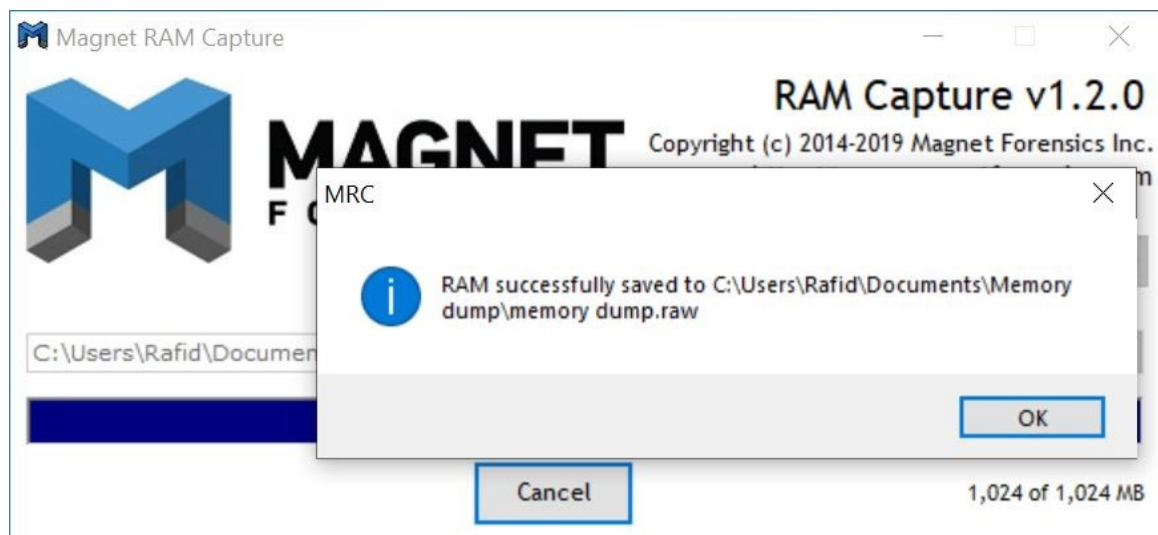


Figure 5.7: Creation of memory dump using Magnet forensic software

5.3 AccessData FTK imager

Using FTK imager software, we can create memory dump and paging files for both 32-bit and 64-bit windows systems. To use the software, first, we have to download the software from the Internet (this software is free) and install the software into the target system. FTK

imager is more efficient and faster than other software, no matter how much data we deal with or the size. After opening, the software's user interface will look like the image below.

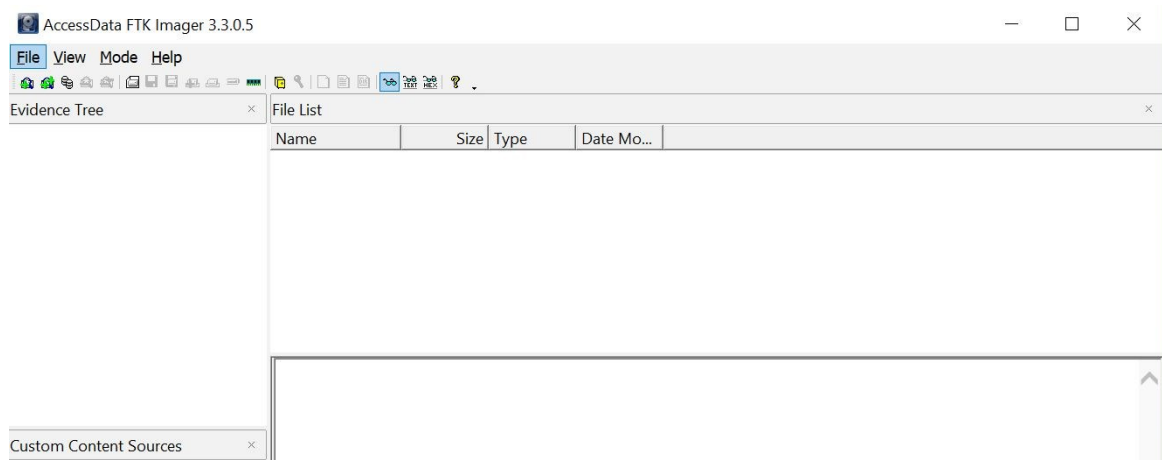


Figure 5.8: FTK imager software dashboard

After clicking the file option, the UI will look like the below image. To start the process, we have to click the file option on the left side of the UI. After that, we have to select the capture memory option from the file.

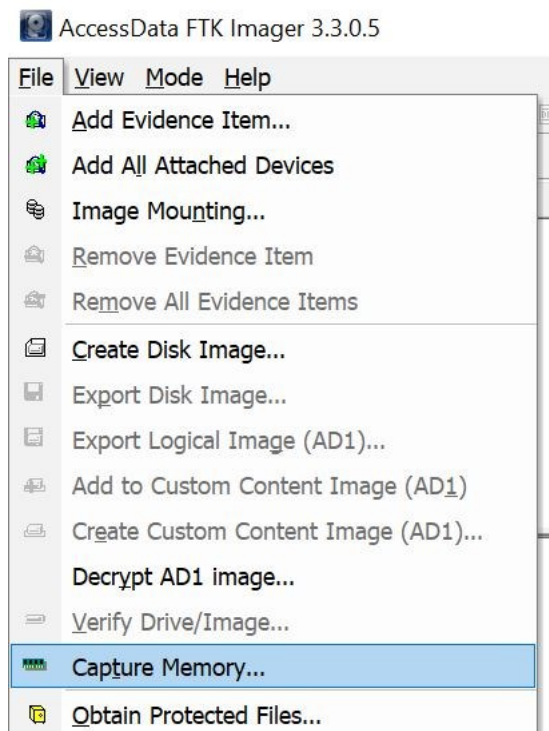


Figure 5.9: Selection of the primary memory on the FTK imager

After clicking the capture memory option, we now have to provide some information needed to create the memory dump. We have to give the file name and destination path. If

we need the pagefile, click the include pagefile and the AD1 file if required. After providing all the information, we have to click the “capture memory” button to start the memory imaging process.

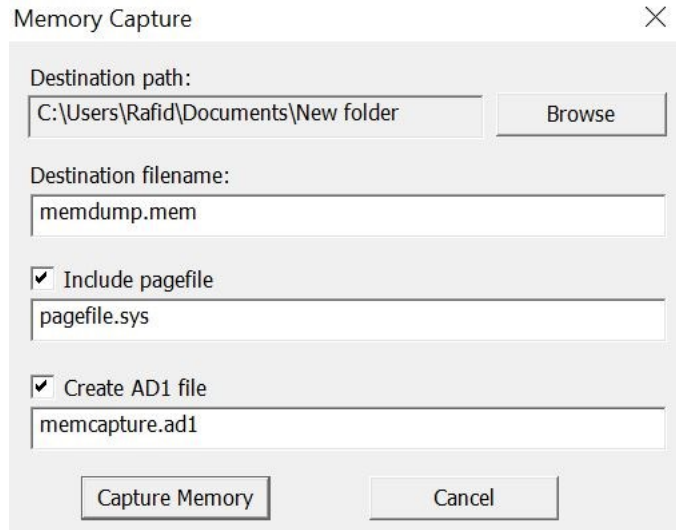


Figure 5.10: Directory selection of memory dump using FTK imager software

After clicking the “capture memory” button, it will show that the capture memory process is started, and the percentage of the process compilation will appear on the screen like below.

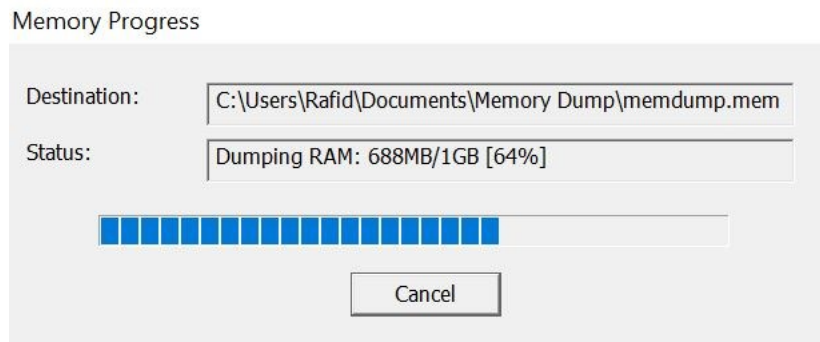


Figure 5.11: Creating memory dump using FTK imager software

When the memory dump process is completed, a message will be shown on the screen saying, “Memory capture finished successfully.” The path to the memory image will also appear on the screen.

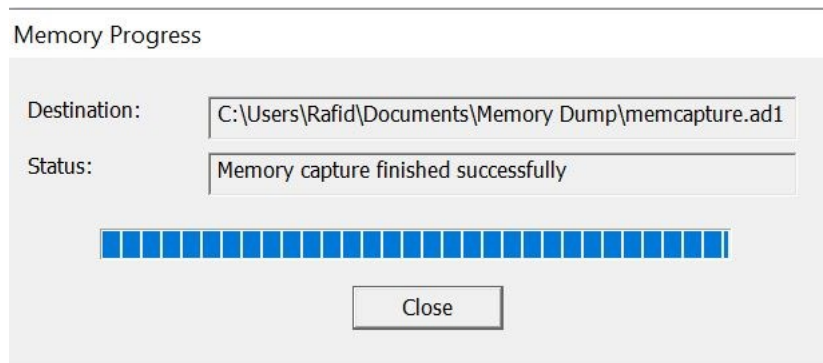


Figure 5.12: Creation of memory dump using FTK imager software

Using these simple steps, we can create a memory dump of the targeted device using this AccessData FTK imager software.

5.4 Belkasoft Live RAM Capturer

Belkasoft is a free forensic tool capable of capturing all the information of the device's volatile memory. Even if some active anti-debugging protects the system, we can capture the memory using this tool. This tool is supported for both 32-bit and 64-bit windows systems. We can use this captured memory in any RAM analysing tool to investigate suspicious activity tracking. First, we must download and install this tool into the system to use it.

After installing the tool, we have to open the tool and select the path where we want to save the memory image. After choosing the path, we must click the "Capture" button.

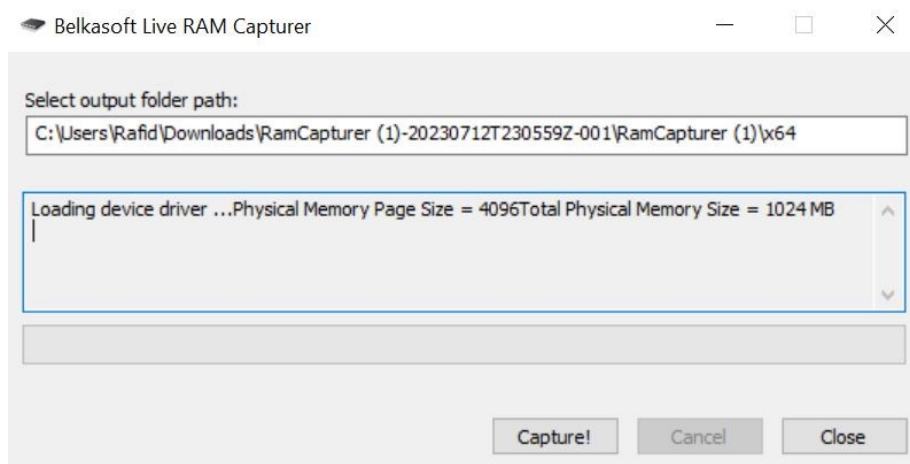


Figure 5.13: Directory selection of memory dump using Belkasoft software

After clicking the "Capture" button, the tool will start the process of imaging the RAM, and the process's progress will show on our screen.

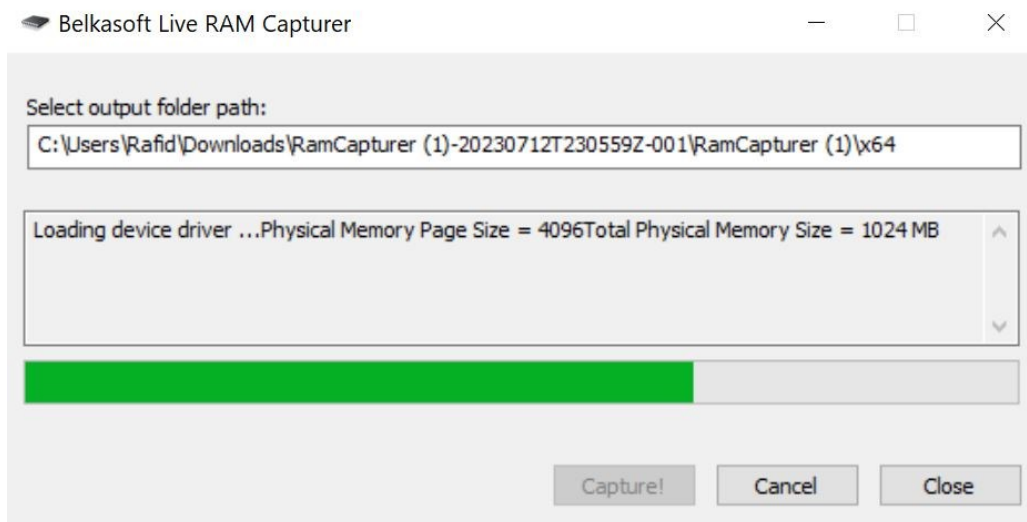


Figure 5.14: Creating memory dump using Belkasoft software

After the process end, the green line, which indicates the percentage of the process, will become full, and the memory image will be created successfully.

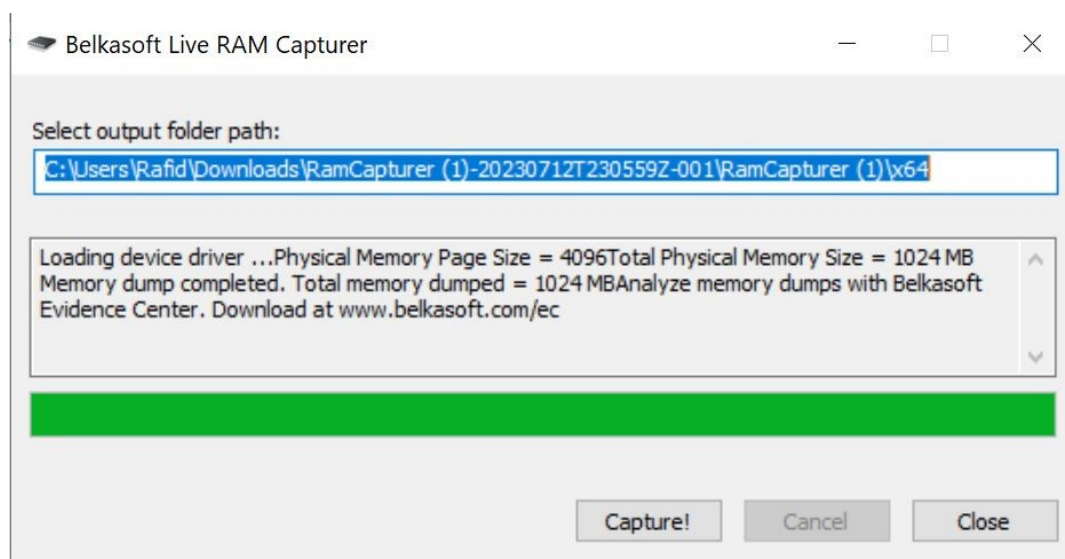


Figure 5.15: Creation of memory dump using Belkasoft software

Chapter 6

Malicious Process Identification from the primary memory

dump

Volatility can be a helpful memory forensic toolkit for examining memory dumps produced during the acquisition process. Using Volatility, investigators can efficiently search, predict, locate and detect malicious activities using the primary memory dump. Volatility can be used in all operating systems like Windows, Linux, and macOS. As an operating system, Windows is the platform upon which the Volatility is installed and operated. Since Volatility is a command-line tool, we must employ the Windows command prompt or Windows PowerShell to execute commands for memory forensics. Before we can put Volatility to work, we must get it from the Internet; to do so, we must visit the Volatility Foundation website. From there, we can download Volatility. We are now ready to use the Volatility to investigate malicious activities after downloading it.

6.1 Accumulation of Materials

We need a memory dump that contains the malicious process run or running during the memory dump creation event for malicious process identification. After that, the investigator will go through the memory dump and run Volatility plugins to identify the malicious process. There is also another way of collecting the memory dump. We can run some malicious processes on a device and then create the memory dump using Image creating tools discussed earlier in this paper. But if we run the malicious process directly on our machine, that will be very harmful to our device, and there is a very high possibility of losing sensitive data of the device. We can also run the process on the virtual machine to avoid data loss. But collecting such a malicious process is very difficult, and their resources are limited. Windows firewall will detect and delete the file if the process is already in the registry. Also, we can't 100% ensure that no casualty will happen. So, there safer way is to collect the memory dump that is already contained in the malicious process. But the resource limitation problem remains. No website provides such a memory dump. As we are investigating through the Windows operating system, that also limits our search area. By searching the Internet, we collected a minimal number of the memory dump. From

the details of the source, we ensured that the collected memory dump contained malicious activities and what types of the malicious processes were in the memory dump.

6.2 Proposed Method

To identify the malicious process from the primary memory dump, investigators run different commands or plugins that Volatility offers and try to relate and detect the malicious process. The Volatility provides many plugins, and all the plugins have different outputs. From that, investigators have to relate the process with the harmful process and detect it from the process running or already executed into the machine. This sleuthing step might vary from investigator to investigator and even from malicious process to process. Investigator can use the Brute-force method to detect the process that means the investigator will apply one-by-one plugins and try to identify the process. An astute and knowledgeable investigator can try to shorten the duration of the search by executing only the set of commands required to determine the presence of a specific malicious process. But there were hundreds of plugins from where the investigator had to run the required one to view the necessary information of the processes. However, there does not seem to be any standardized procedure.

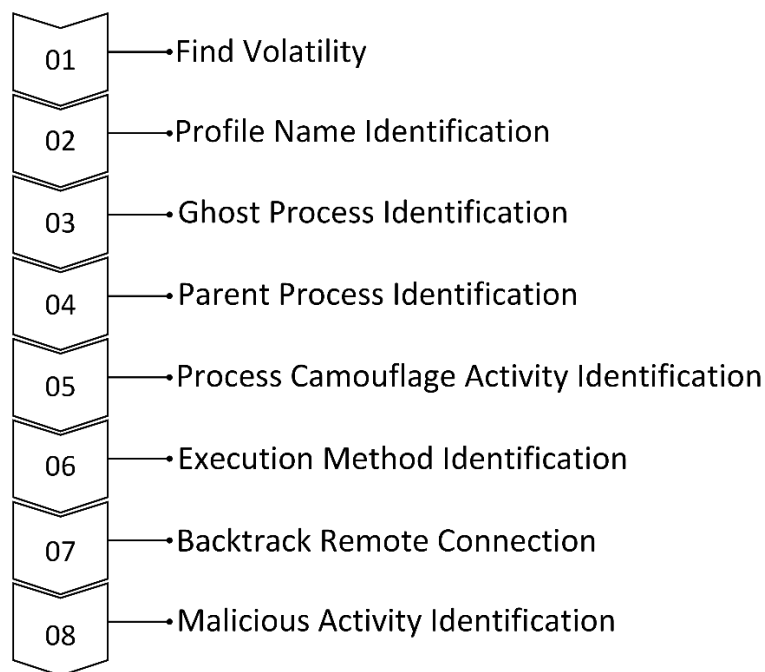


Figure 6.1: Proposed process diagram for identifying the malicious process

This process diagram detects the malicious run or running process into the memory dump captured from the suspected machine. Currently, there are many distinct sorts and varieties

of malicious processes, and the identification of the process might also vary from one instance of that process to another. Therefore, not all of the suggested method's steps may be necessary to detect all malicious processes. Cutting off more than one step from a particular process may be feasible. However, by following the procedures described in the suggested strategy, we can identify practically every malicious activity present at the moment.

6.2.1 Find Volatility

Volatility is a command-line-based tool, so we have to use the command line to execute all the plugins to view the information. After downloading the desired version of the Volatility tool from the Volatility Foundation websites, we will be ready to go for the memory dump analyzing process using the Volatility. To execute our first proposed method steps, we must go to the directory where the software is located and remember the directory address. Open the Windows command prompt or Windows PowerShell. Here we are opening the windows command prompt. Now we have to go to that directory using the `cd` command, where the Volatility software executable file is located.



```
Command Prompt
Microsoft Windows [Version 10.0.19044.2364]
(c) Microsoft Corporation. All rights reserved.

C:\Users\rafid>cd Downloads

C:\Users\rafid\Downloads>cd volatility_2.6_win64_standalone

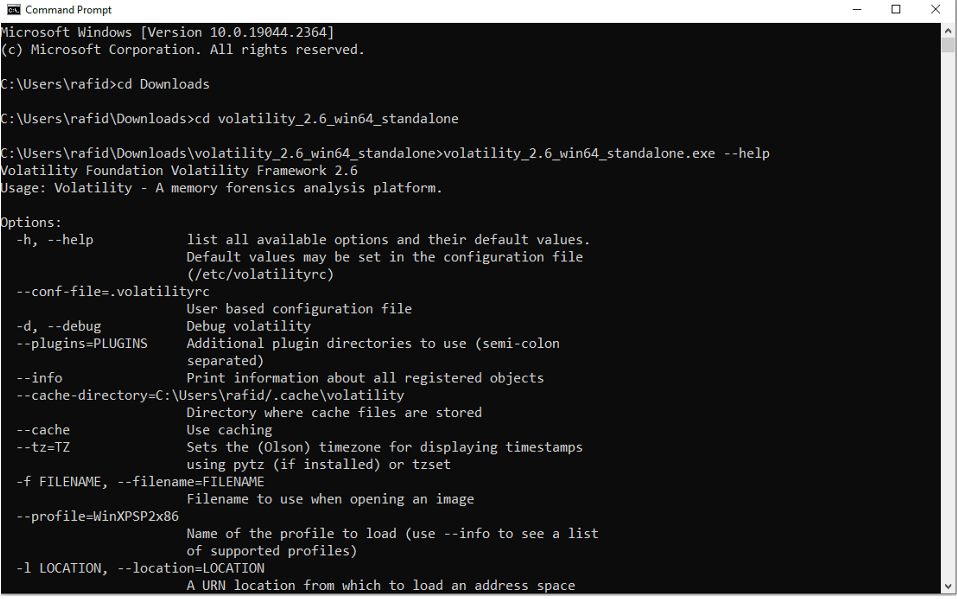
C:\Users\rafid\Downloads\volatility_2.6_win64_standalone>
```

Figure 6.2: Directory of the Volatility tool

When we are in the directory where the executable Volatility file is located, we can use the command prompt to run the tool. Remember that Volatility does not have a graphical user interface (GUI), so if we try to double-click the executable file, it will not work like that. We can run the executable file directly from anywhere, like we can run it from the removable USB drive or removable HDD drive. This feature makes Volatility a portable tool that you can take for remote investigation. In the investigation area, we may not be allowed to take our devices like laptops. So, the portability features of the Volatility help us in that tight situation.

To run the volatility, we have to launch the executable from the stored directory (the directory where the executable file of the Volatility 2.6 is located) in the command prompt. For example, if we write the .exe file name and “—help,” it will run and show all the available plugin options and default values. The command is written below-

“Volatility_2.6_win64_standalone.exe --help”



```
Microsoft Windows [Version 10.0.19044.2364]
(c) Microsoft Corporation. All rights reserved.

C:\Users\rafid>cd Downloads
C:\Users\rafid\Downloads>cd volatility_2.6_win64_standalone
C:\Users\rafid\Downloads\volatility_2.6_win64_standalone>volatility_2.6_win64_standalone.exe --help
Volatility Foundation Volatility Framework 2.6
Usage: Volatility - A memory forensics analysis platform.

Options:
  -h, --help            list all available options and their default values.
                        Default values may be set in the configuration file
                        (/etc/volatilityrc)
  --conf-file=.volatilityrc  User based configuration file
  -d, --debug            Debug volatility
  --plugins=PLUGINS       Additional plugin directories to use (semi-colon
                        separated)
  --info                Print information about all registered objects
  --cache-directory=C:\Users\rafid\.cache\volatility  Directory where cache files are stored
  --cache                Use caching
  --tz=TZ                Sets the (Olson) timezone for displaying timestamps
                        using pytz (if installed) or tzset
  -f FILENAME, --filename=FILENAME  Filename to use when opening an image
  --profile=WinXPSP2x86  Name of the profile to load (use --info to see a list
                        of supported profiles)
  -l LOCATION, --location=LOCATION  A URN location from which to load an address space
```

Figure 6.3: Volatility tool’s help command

The basic command construction of the volatility is the main executable file, then give the name of the memory dump file, the name of the profile we used, and the selected plugin to execute the command.

“Volatility_2.6_win64_standalone.exe -f <File name> --profile <profile name> <plugin>”

One of the examples is given below:

“Volatility_2.6_win64_standalone.exe -f WIN.raw --profile winxpsp2x86 pslist”

In the example, Volatility_2.6_win64_standalone.exe is the executable volatility file, -f stands for the file name, WIN.raw is the memory dump file with the raw extension, winxpsp2x86 is the specific profile name and pslist is the plugin.

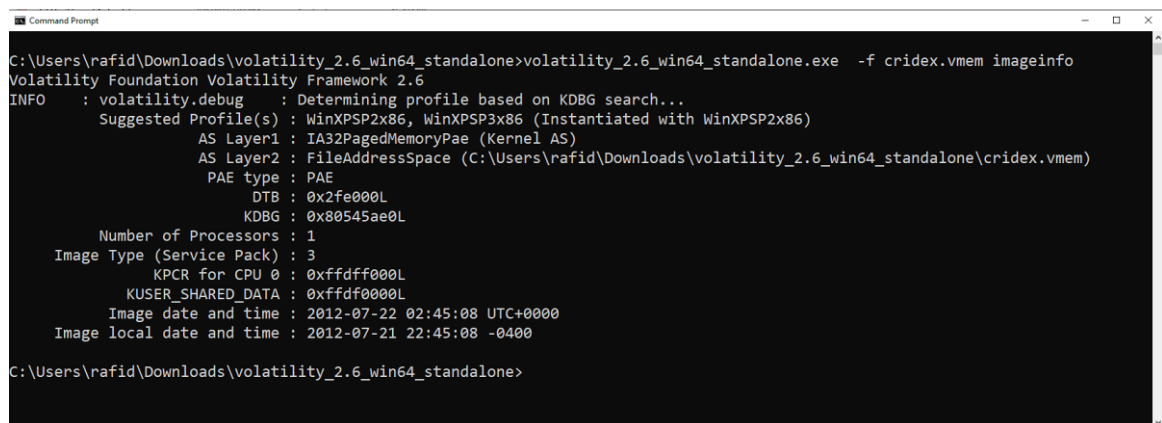
It is important to remember that the suspected memory dump we are investigating must be located in the same directory where the Volatility is.

6.2.2 Profile Name Identification

In this step, we will identify the profile name of the collected memory dump we are investigating. Whenever we write a command using the Volatility, we need a specific profile name for the memory dump. The profile name is nothing but the profile of the device system. So, we have to know the profile name of the system from which we created the memory dump. Without that, we can't use the command, and an error will occur saying, "No suitable address space mapping found". So, knowing the profile name is essential. We have two ways to find the profile name, but the most useful will be learning some extra information about the profile name. We can use the imageinfo plugin to know the profile name and other important information. Using the imageinfo plugin, we can know the suggested profile names, service pack version, number of processors, and much more related information about the suspected device. To do so, we have to write the executable file of volatility, the file name of the memory dump, and imageinfo.

The command is given below-

```
"volatility_2.6_win64_standalone.exe -f cridex.vmem imageinfo"
```



```
C:\Users\rafid\Downloads\volatility_2.6_win64_standalone>volatility_2.6_win64_standalone.exe -f cridex.vmem imageinfo
Volatility Foundation Volatility Framework 2.6
INFO : volatility.debug : Determining profile based on KDBG search...
      Suggested Profile(s) : WinXPSP2x86, WinXPSP3x86 (Instantiated with WinXPSP2x86)
      AS Layer1 : IA32PagedMemoryPae (Kernel AS)
      AS Layer2 : FileAddressSpace (C:\Users\rafid\Downloads\volatility_2.6_win64_standalone\cridex.vmem)
      PAE type : PAE
      DTB : 0x2fe000L
      KDBG : 0x80545ae0L
      Number of Processors : 1
      Image Type (Service Pack) : 3
      KPCR for CPU 0 : 0xffdf000L
      KUSER_SHARED_DATA : 0xffdf000L
      Image date and time : 2012-07-22 02:45:08 UTC+0000
      Image local date and time : 2012-07-21 22:45:08 -0400
C:\Users\rafid\Downloads\volatility_2.6_win64_standalone>
```

Figure 6.4: "imageinfo" command on the Volatility tool

After executing the command, the information will appear on the command line. Here we can see that the machine's operating system is windows XP, the service pack is 3, the number of the processors of the device, the image captured to date and time is 22-07-2012,

and much more information will be needed in our investigation process. It also shows the files address space, where the memory dump file is located in the running device memory. We also found the suggested profile name that we will use in our following plugin. This command determines the recommended profile based on the KDBG search. We can also know the profile name using the KDBG command. The command will be like this-

`“Volatility_2.6_win64_standalone.exe -f WIN.raw kdbg”`

Here one thing has to be noted as a memory dump investigation using the volatility; every memory is not supported by the volatility tool. That means we have a limitation of the operating system of the memory dump. There is also a list of operating system volatility framework support. If we use any other operating system which is not on the list, then we can't use the volatility framework. The memory dump we are using is a windows XP service pack 3 machine in our project demonstration, which is supported by the framework. Another memory dump we used to double-check the proposed method is Windows 11, which is volatility-supported. There is also a limitation of memory dump format support in the volatility tool. Suppose we create a memory dump, and the format of the memory dump is not supported by the volatility tool. In that case, we cannot make the investigation process with the memory dump using volatility. So, we must be careful when creating the memory dump from the suspected computer or device. The supported formats or extensions of the Volatility framework is given below-

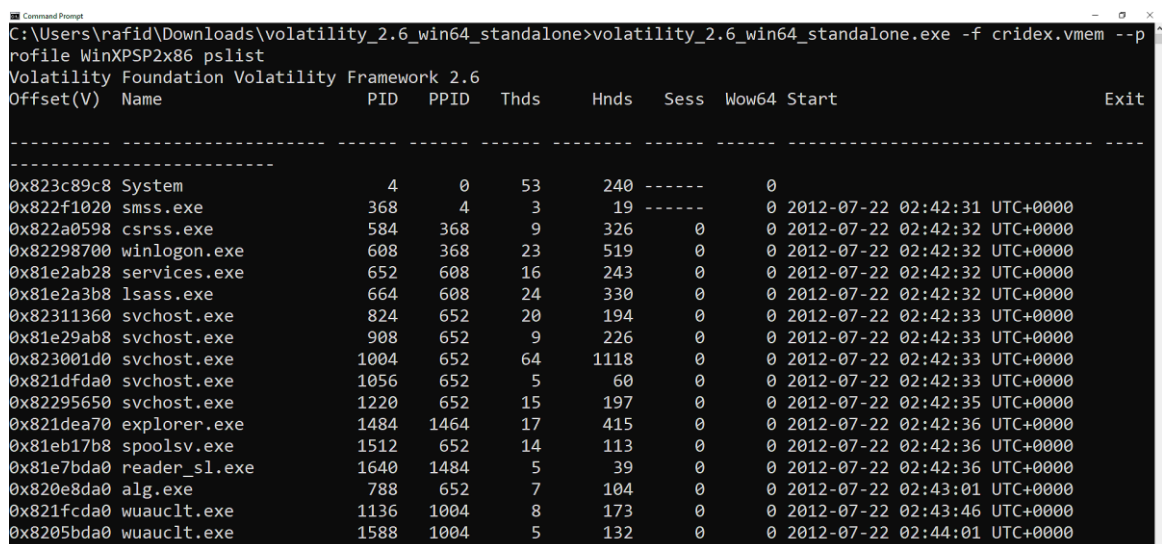
- Raw Physical Memory
- Firewire (IEEE 1394 standards)
- EWF
- 32-bit and 64-bit windows crash dump
- 32-bit and 64-bit windows hibernation
- 32-bit and 64-bit MachO
- VirtualBox core dump
- VMware snapshot and saved state
- HPAK format (FirstDump)
- QEMU dump

In our project, We will use a capture memory dump called “crindex.vmem” which is a .vmem extension, and we will use the volatility framework for the malicious process identification. The Volatility framework supports the .vmem extension is how we can use that memory dump extension. We can use this process if the framework doesn't support the extension.

6.2.3 Ghost process identification

After successfully finding the profile information in the previous steps, we can execute the ghost process identification step to identify the malicious activities from the collected memory dump. In this step, we would try to see which processes were running when the memory was captured and from those processes there are any suspected malicious processes or not. To do that, we will use the plugin called “pslist”. “Pslist” will show all the running processes of the machine from the memory dump. It also shows us the parent process ID, process ID, threads information, handles information, and start-exit time and date of the processes. The command is given below-

```
“volatility_2.6_win64_standalone.exe -f cridex.vmem --  
profile WinXPSP2x86 pslist”
```

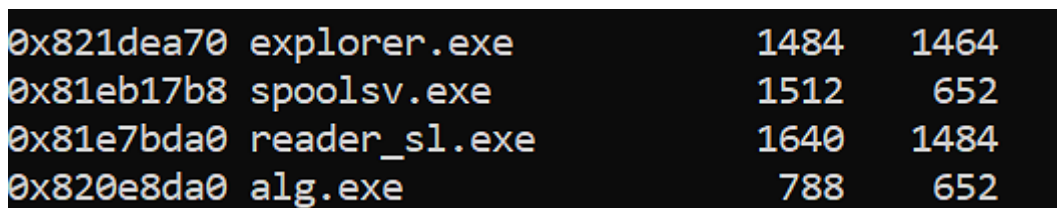


```
C:\Users\rafid\Downloads\volatility_2.6_win64_standalone>volatility_2.6_win64_standalone.exe -f cridex.vmem --p  
rofile WinXPSP2x86 pslist  
Volatility Foundation Volatility Framework 2.6  
Offset(V)  Name                PID  PPID  Thds  Hnds  Sess  Wow64  Start                Exit
```

Offset(V)	Name	PID	PPID	Thds	Hnds	Sess	Wow64	Start	Exit
0x823c89c8	System	4	0	53	240	-----	0		
0x822f1020	smss.exe	368	4	3	19	-----	0	2012-07-22 02:42:31 UTC+0000	
0x822a0598	csrss.exe	584	368	9	326	0	0	2012-07-22 02:42:32 UTC+0000	
0x82298700	winlogon.exe	608	368	23	519	0	0	2012-07-22 02:42:32 UTC+0000	
0x81e2ab28	services.exe	652	608	16	243	0	0	2012-07-22 02:42:32 UTC+0000	
0x81e2a3b8	lsass.exe	664	608	24	330	0	0	2012-07-22 02:42:32 UTC+0000	
0x82311360	svchost.exe	824	652	20	194	0	0	2012-07-22 02:42:33 UTC+0000	
0x81e29ab8	svchost.exe	908	652	9	226	0	0	2012-07-22 02:42:33 UTC+0000	
0x823001d0	svchost.exe	1004	652	64	1118	0	0	2012-07-22 02:42:33 UTC+0000	
0x821dfda0	svchost.exe	1056	652	5	60	0	0	2012-07-22 02:42:33 UTC+0000	
0x82295650	svchost.exe	1220	652	15	197	0	0	2012-07-22 02:42:35 UTC+0000	
0x821dea70	explorer.exe	1484	1464	17	415	0	0	2012-07-22 02:42:36 UTC+0000	
0x81eb17b8	spoolsv.exe	1512	652	14	113	0	0	2012-07-22 02:42:36 UTC+0000	
0x81e7bda0	reader_sl.exe	1640	1484	5	39	0	0	2012-07-22 02:42:36 UTC+0000	
0x820e8da0	alg.exe	788	652	7	104	0	0	2012-07-22 02:43:01 UTC+0000	
0x821fcd00	wuauclt.exe	1136	1004	8	173	0	0	2012-07-22 02:43:46 UTC+0000	
0x8205bda0	wuauclt.exe	1588	1004	5	132	0	0	2012-07-22 02:44:01 UTC+0000	

Figure 6.5: “pslist” command on the Volatility tool

We can see all the processes running on that machine when the memory is captured using the command. If we deeply go through all the processes, we can see some processes look suspicious or say we don’t see those processes very often.



0x821dea70	explorer.exe	1484	1464
0x81eb17b8	spoolsv.exe	1512	652
0x81e7bda0	reader_sl.exe	1640	1484
0x820e8da0	alg.exe	788	652

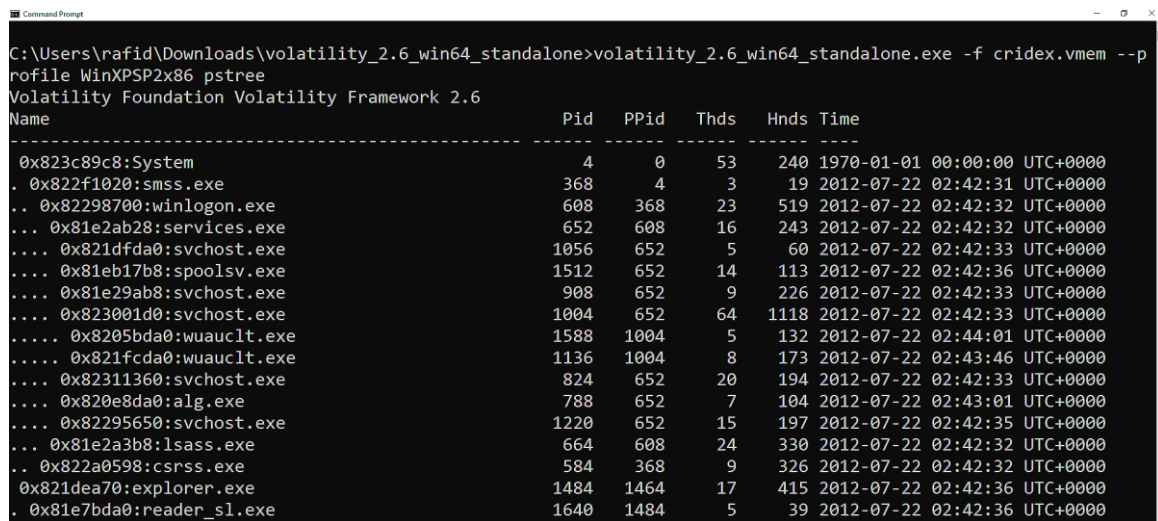
Figure 6.6: Suspicious processes viewing on the Volatility tool

Here we can see a process named “exporer.exe”, “reader_sl.exe” and “alg.exe” are some processes that we don’t regularly see in the process list. So, these processes can be malicious. To know whether those processes are malicious, we must conduct more investigation.

6.2.4 Parent process identification

In this step, we will try to see the relation of parent and child processes, which parent and child process they have (previously selected process from step 3). To do so, we can use a plugin called “pstree”. “pstree” command shows the running processes in a tree format. This command extracts the process information like the “pslist” command. But using the “pstree” command, we can see all the parent processes and, in those parents, processes which processes have child processes running and how many child processes are running in a parent process. The command is given below-

```
“volatility_2.6_win64_standalone.exe -f cridex.vmem --
profile WinXPSP2x86 pstree”
```



Name	Pid	PPid	Thds	Hnds	Time
0x823c89c8:System	4	0	53	240	1970-01-01 00:00:00 UTC+0000
. 0x822f1020:smss.exe	368	4	3	19	2012-07-22 02:42:31 UTC+0000
.. 0x82298700:winlogon.exe	608	368	23	519	2012-07-22 02:42:32 UTC+0000
... 0x81e2ab28:services.exe	652	608	16	243	2012-07-22 02:42:32 UTC+0000
.... 0x821dfda0:svchost.exe	1056	652	5	60	2012-07-22 02:42:33 UTC+0000
.... 0x81eb17b8:spoolsv.exe	1512	652	14	113	2012-07-22 02:42:36 UTC+0000
.... 0x81e29ab8:svchost.exe	908	652	9	226	2012-07-22 02:42:33 UTC+0000
.... 0x823001d0:svchost.exe	1004	652	64	1118	2012-07-22 02:42:33 UTC+0000
..... 0x8205bda0:wuauc1t.exe	1588	1004	5	132	2012-07-22 02:44:01 UTC+0000
..... 0x821fcd00:wuauc1t.exe	1136	1004	8	173	2012-07-22 02:43:46 UTC+0000
.... 0x82311360:svchost.exe	824	652	20	194	2012-07-22 02:42:33 UTC+0000
.... 0x820e8da0:alg.exe	788	652	7	104	2012-07-22 02:43:01 UTC+0000
.... 0x82295650:svchost.exe	1220	652	15	197	2012-07-22 02:42:35 UTC+0000
.. 0x81e2a3b8:lsass.exe	664	608	24	330	2012-07-22 02:42:32 UTC+0000
.. 0x822a0598:csrss.exe	584	368	9	326	2012-07-22 02:42:32 UTC+0000
0x821dea70:explorer.exe	1484	1464	17	415	2012-07-22 02:42:36 UTC+0000
. 0x81e7bda0:reader_sl.exe	1640	1484	5	39	2012-07-22 02:42:36 UTC+0000

Figure 6.7: “pstree” command on Volatility

By executing the “pstree” command, we can view all the processes in the parent-child relation tree form. Here we can see that alg.exe is the child of the system process. So that process was run by the system. But the process reader_sl.exe is the child process of the explorer.exe parent process. That process can be malicious. We will do more investigation into that process.

0x821dea70:explorer.exe	1484	1464	17
. 0x81e7bda0:reader_sl.exe	1640	1484	5

Figure 6.8: Processes parent-child view on Volatility

6.2.5 Process camouflage activity identification

This step will determine if that particular process wants to hide something. For this purpose, we will use the plugin called “psxview”. This plugin is used to find the hidden process that various process list. The plugin “psxview” make a comparison with the live or running process mentioned in the “psActiveProcessHead” and possibly another source into the imaged memory. If any false value is found in the Column, then we can ensure the process was not found in that particular place. The command is given below-

“volatility_2.6_win64_standalone.exe -f cridex.vmem --
profile WinXPSP2x86 psxview”

```

C:\Users\rafid\Downloads\volatility_2.6_win64_standalone>volatility_2.6_win64_standalone.exe -f cridex.vmem
--profile WinXPSP2x86 psxview
Volatility Foundation Volatility Framework 2.6
Offset(P)  Name                PID  pslist  psscan  thrdproc  pspcid  csrss  session  deskthrd  ExitTime
-----
0x02498700 winlogon.exe         608  True    True    True      True    True   True     True
0x02511360 svchost.exe          824  True    True    True      True    True   True     True
0x022e8da0 alg.exe             788  True    True    True      True    True   True     True
0x020b17b8 spoolsv.exe          1512 True    True    True      True    True   True     True
0x0202ab28 services.exe       652  True    True    True      True    True   True     True
0x02495650 svchost.exe          1220 True    True    True      True    True   True     True
0x0207bda0 reader_sl.exe 1640 True    True    True      True    True   True     True
0x025001d0 svchost.exe          1004 True    True    True      True    True   True     True
0x02029ab8 svchost.exe          908  True    True    True      True    True   True     True
0x023fcd00 wuauclt.exe        1136 True    True    True      True    True   True     True
0x0225bda0 wuauclt.exe        1588 True    True    True      True    True   True     True
0x0202a3b8 lsass.exe            664  True    True    True      True    True   True     True
0x023dea70 explorer.exe      1484 True    True    True      True    True   True     True
0x023dfda0 svchost.exe          1056 True    True    True      True    True   True     True
0x024f1020 smss.exe           368  True    True    True      True    False  False    False
0x025c89c8 System              4    True    True    True      True    False  False    False
0x024a0598 csrss.exe           584  True    True    True      True    False  True     True
C:\Users\rafid\Downloads\volatility_2.6_win64_standalone>

```

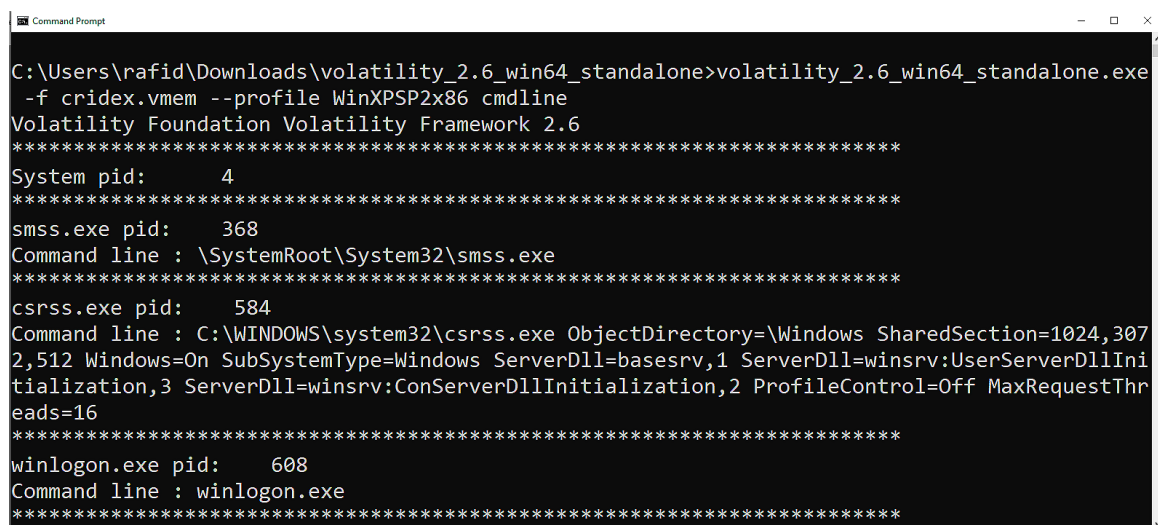
Figure 6.9: “psxview” command on Volatility

After executing the command, we can see that all the process values are true in the column pslist and psscan fields. So we can ensure that no processes are trying to hide something. If a particular process hides any information, then the value of the pslist and psscan will be false. This step intends to identify the suspected process of hidden information. As the alleged process selected in the previous steps did not try to hide anything, we have to do further investigation.

6.2.6 Execution method identification

In this step, we will try to see which command was run in the command prompt until the memory of the suspect device is captured. To do so, we have to use a plugin called “cmdline”. “cmdline” command displays process command-line arguments. Using that plugin, we can see all the commands written in the prompt. The command is given below-

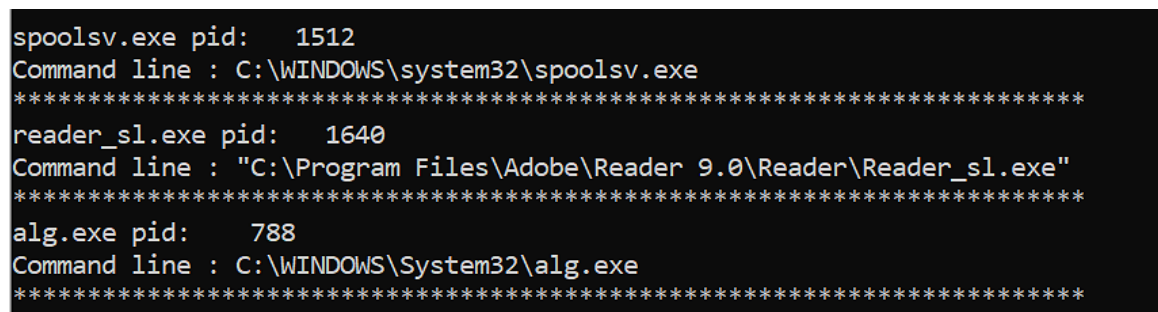
```
“volatility_2.6_win64_standalone.exe -f cridex.vmem --  
profile WinXPSP2x86 cmdline”
```



```
Command Prompt
C:\Users\rafid\Downloads\volatility_2.6_win64_standalone>volatility_2.6_win64_standalone.exe
-f cridex.vmem --profile WinXPSP2x86 cmdline
Volatility Foundation Volatility Framework 2.6
*****
System pid:      4
*****
smss.exe pid:    368
Command line : \SystemRoot\System32\smss.exe
*****
csrss.exe pid:   584
Command line : C:\WINDOWS\system32\csrss.exe ObjectDirectory=\Windows SharedSection=1024,307
2,512 Windows=On SubSystemType=Windows ServerDll=basesrv,1 ServerDll=winsrv:UserServerDllIni
tialization,3 ServerDll=winsrv:ConServerDllInitialization,2 ProfileControl=Off MaxRequestThr
eads=16
*****
winlogon.exe pid: 608
Command line : winlogon.exe
*****
```

Figure 6.10: “cmdline” command on Volatility

After executing the command, we can see many commands in the command line. We have to identify whether the suspected reader_sl.exe was there or not. We can see that process is in the command line. The process ID of that particular process reader_sl.exe is 1640. We have to investigate using the individual process ID to find malicious activity.

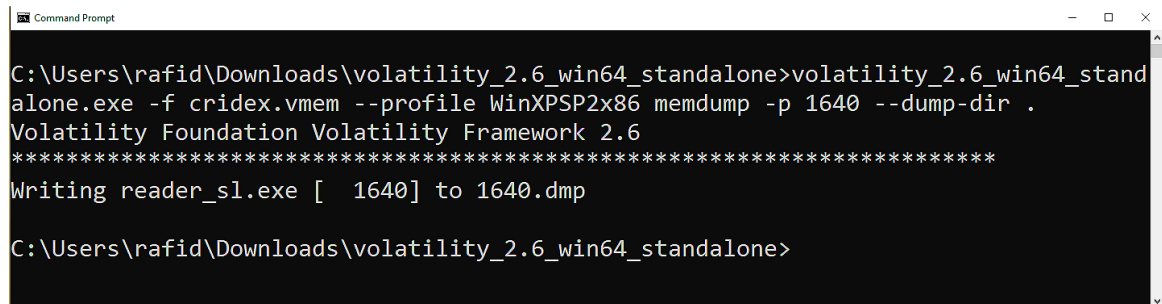


```
spoolsv.exe pid: 1512
Command line : C:\WINDOWS\system32\spoolsv.exe
*****
reader_sl.exe pid: 1640
Command line : "C:\Program Files\Adobe\Reader 9.0\Reader\Reader_sl.exe"
*****
alg.exe pid: 788
Command line : C:\WINDOWS\System32\alg.exe
*****
```

Figure 6.11: Finding process ID on Volatility

Now we will create a dump file of that particular reader_sl.exe process using the process ID 1640 to investigate that process. To do that, we will need to write a command which is given below-

```
“volatility_2.6_win64_standalone.exe -f cridex.vmem --  
profile WinXPSP2x86 memdump -p 1640 --dump-dir . ”
```



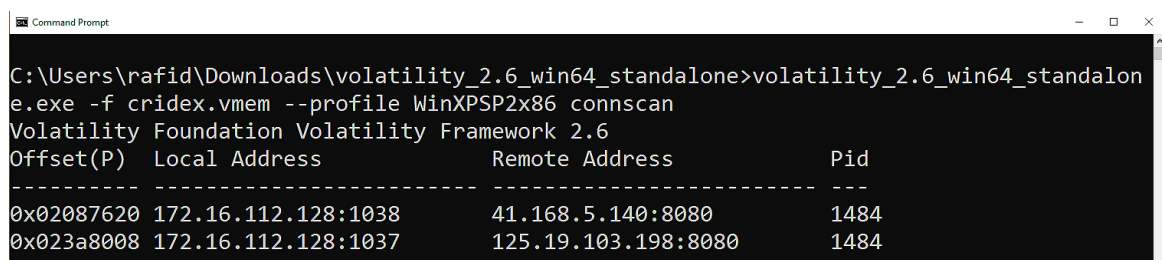
```
Command Prompt  
C:\Users\rafid\Downloads\volatility_2.6_win64_standalone>volatility_2.6_win64_standalone.exe -f cridex.vmem --profile WinXPSP2x86 memdump -p 1640 --dump-dir .  
Volatility Foundation Volatility Framework 2.6  
*****  
Writing reader_sl.exe [ 1640] to 1640.dmp  
C:\Users\rafid\Downloads\volatility_2.6_win64_standalone>
```

Figure 6.12: Creating process dump using process ID on Volatility

6.2.7 Backtrack the remote connection

In this step, we will determine if there are any established connections with the suspected device. To do so, we have to use a plugin called “connscan”. This plugin shows us all the connections made with the particular supposed machine. The command is given below-

```
“volatility_2.6_win64_standalone.exe -f cridex.vmem --  
profile WinXPSP2x86 connscan”
```



```
Command Prompt  
C:\Users\rafid\Downloads\volatility_2.6_win64_standalone>volatility_2.6_win64_standalone.exe -f cridex.vmem --profile WinXPSP2x86 connscan  
Volatility Foundation Volatility Framework 2.6  
Offset(P) Local Address Remote Address Pid  
-----  
0x02087620 172.16.112.128:1038 41.168.5.140:8080 1484  
0x023a8008 172.16.112.128:1037 125.19.103.198:8080 1484
```

Figure 6.13: “connscan” command on Volatility

Here we can see that there were connections made with that suspected machine. Two connections were made. From here, we can find two remote IP address which was accessed. Now we will determine whether the process reader_sl.exe is used to make those connections. For that, if we open the process 1640 memory dump file using notepad or any other text editor software and search that particular IP address into the dump file, we can

ensure that the remote IP address was connected through that 1640 process or not. To do that, we have to go to the directory where the process dump file of the process ID 1640 is located. After going to the directory, we have to right-click into the 1640 process memory dump file and click the “Open with” option.

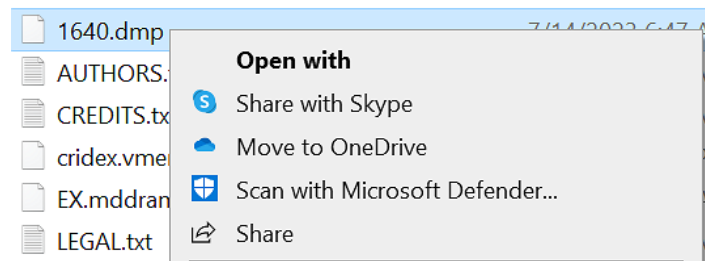


Figure 6.14: Opening process dump on Windows

After clicking the “Open with” option, we will see how we want to open the process dump file. Here we will select the notepad option to open the process dump file. Anyone who wants to open the process dump file with any other text editor software can do it by selecting that specific text editor software in the “more apps” option.

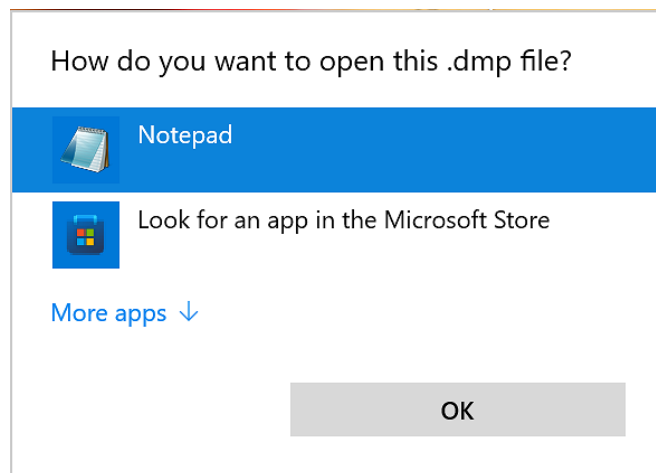


Figure 6.15: Opening process dump in Notepad

After opening the process dump with the notepad, we now have to search if this particular process made the network connection or if we can say that this specific process is responsible for the remote network connection. To do that, we have to press the “Control+F” key to open the search option of the notepad. After that, in the “find what” field, we have to write that network IP address which was found after we use the “connscan” plugin command and click the “Find Next” button.

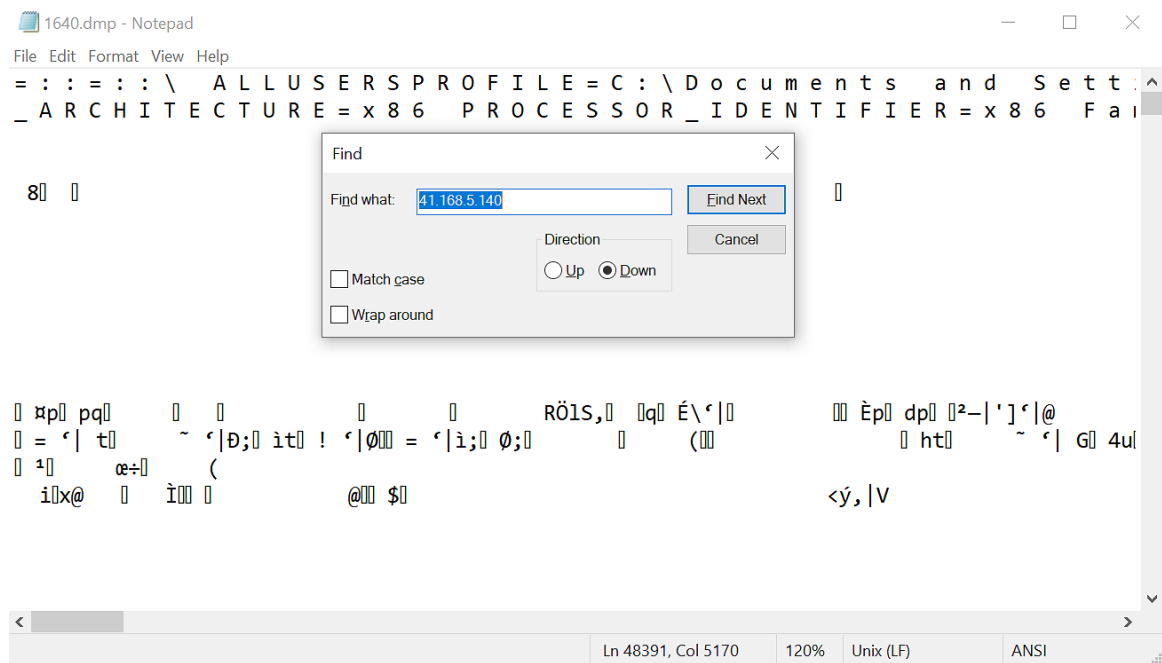


Figure 6.16: Searching IP address on the dumped process

After clicking the “Find Next” button, if that address is in that particular process memory dump file, it will show on our screen.

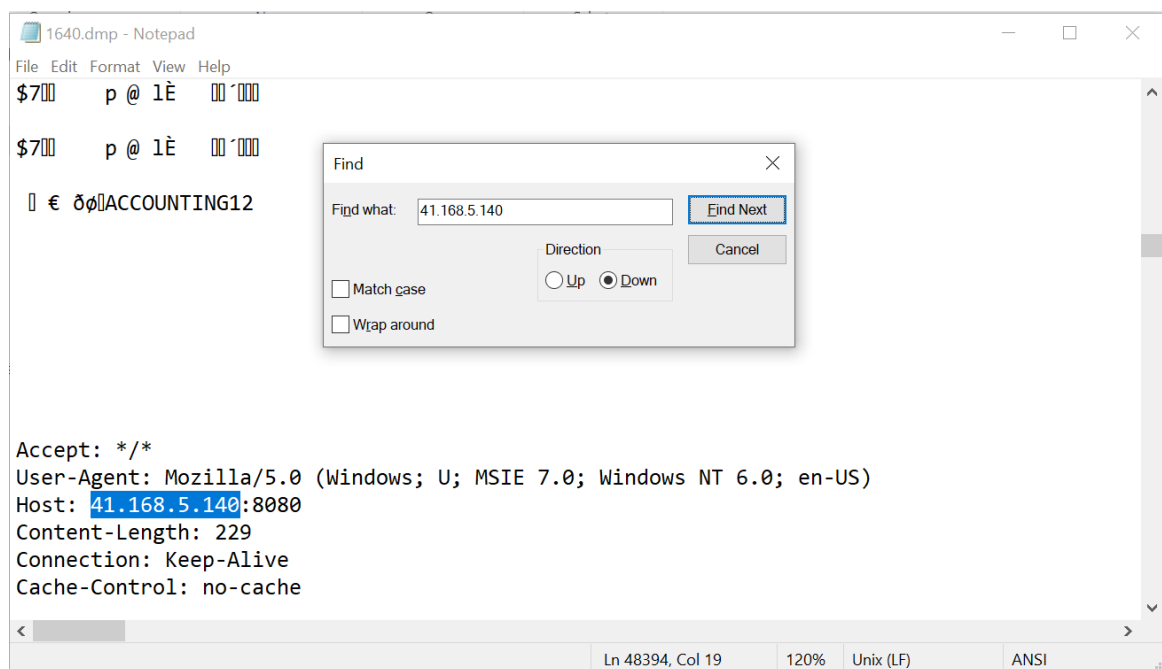


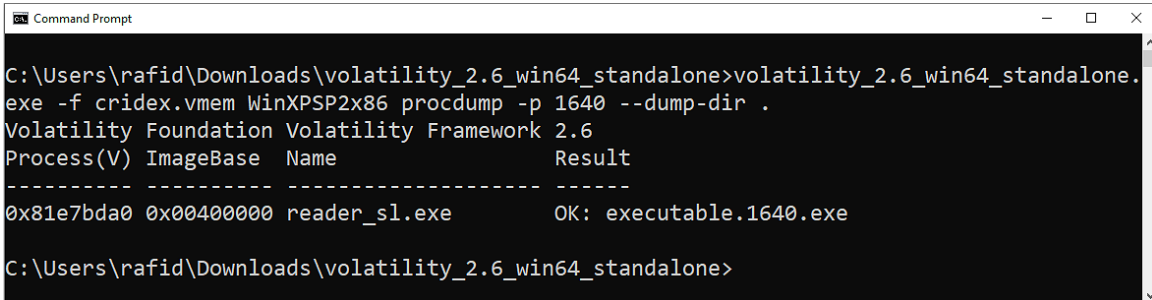
Figure 6.17: Identifying IP address on the dumped process

Here we can see that a particular 1640 process was made that connection with the remote IP 41.168.5.140. So now we are sure the process is connected with that remote IP address using the suspected computer.

6.2.8 Malicious activity identification

In this step, we will finally investigate whether or not the suspected processes are malicious and what types of maliciousness the processes have. In all the previous actions, we verified the selected or suspected process had potential maliciousness and that the involvement was there. In the following phase, we'll attempt to determine the type of maliciousness the process has. To do this, we must create an executable file for the processes reader_sl. To make that, we have to write the command below-

```
“volatility_2.6_win64_standalone.exe -f cridex.vmem --  
profile WinXPSP2x86 procdump -p 1640 --dump-dir . ”
```



```
Command Prompt  
C:\Users\rafid\Downloads\volatility_2.6_win64_standalone>volatility_2.6_win64_standalone.  
exe -f cridex.vmem WinXPSP2x86 procdump -p 1640 --dump-dir .  
Volatility Foundation Volatility Framework 2.6  
Process(V) ImageBase Name Result  
-----  
0x81e7bda0 0x00400000 reader_sl.exe OK: executable.1640.exe  
C:\Users\rafid\Downloads\volatility_2.6_win64_standalone>
```

Figure 6.18: Creating executable process using process ID on Volatility

After executing the command, the executable file of that particular process with the process ID 1640 will be created into the current directory of the computer from where we are investigating the memory dump of the suspected computer or device.

Afterward, we must go to the “virus total” website and upload the executable file. To do that, we have to open any browser from our computer. In our project, we used the chrome browser to do the task. After opening the browser, we must go to the “virus total” website. On the webpage, we will see a “Choose file” button. We have to click that button and select the reader_sl executable file.



Analyse suspicious files, domains, IPs and URLs to detect malware and other breaches, automatically share them with the security community.

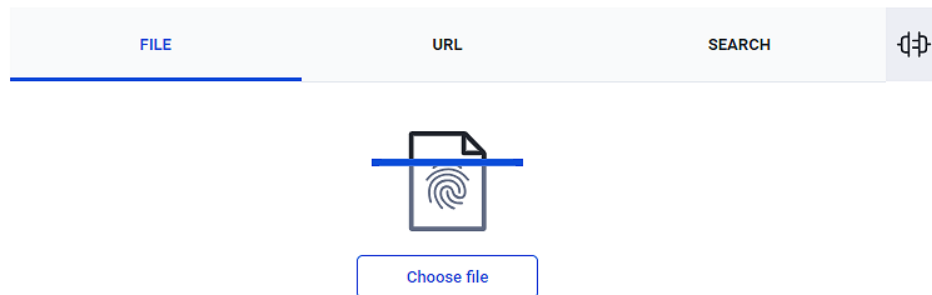


Figure 6.19: Virustotal websites main web page

<https://www.virustotal.com/gui/home/upload> (accessed Jul. 14, 2023)

After successfully uploading the .exe file to the website, many vendors maliciously ensured the process. How the process has been malicious, and another report can be found here. From the webpage, we can see that 35 security vendors have selected that particular file as malicious. There is also information on what type of malware that process can be and the expected behavior of the process that will make it into the device.

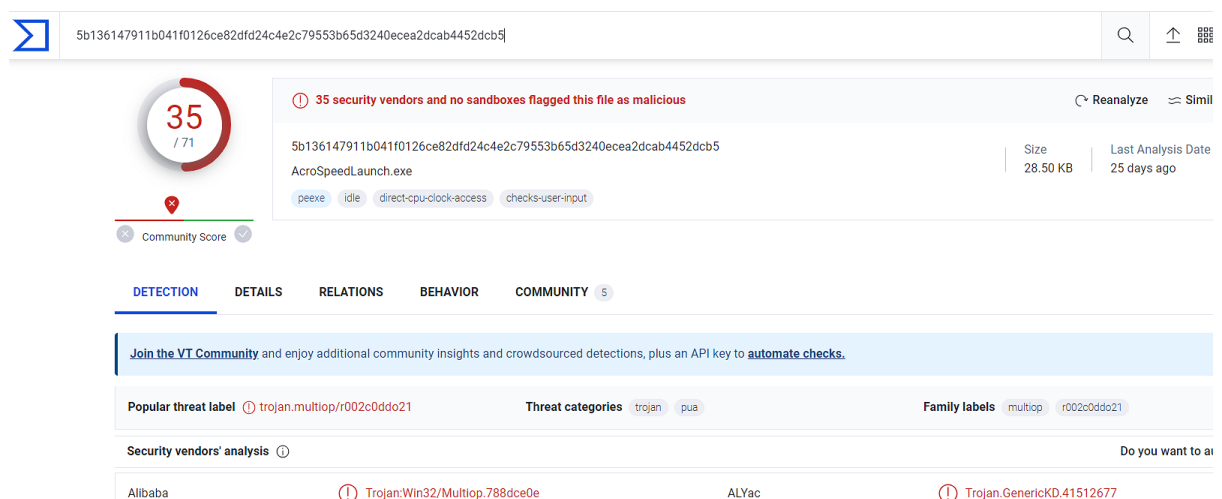


Figure 6.20: Virustotal websites result of the process

https://www.virustotal.com/gui/search/reader_sl (accessed Jul. 14, 2023)

Chapter 7

Malicious activity identification from secondary memory dump

Autopsy can be a useful memory forensic toolkit for investigating the secondary memory image or memory dump created in the acquisition phase. From the secondary memory of the suspected computer or device, the secondary memory dump or image was created, using this secondary memory dump of the suspected computer or device, we will start the investigation process and try to find the malicious activity done using or within the suspected computer or device. An Autopsy is mainly used by the corporate investigator, defense departments, and law enforcement departments to detect what unethical actions were taken from the suspected computer or device. It can also use for recovering elements from devices. What Autopsy makes most useable is that it has a GUI (Graphical User Interface) not a command-line based toolkit. Unlike command-line-oriented tools, where plugins must be executed through the command prompt, we have the option of using the interface to do our investigation quickly and conveniently. Additionally, we do not need to remember the instruction to carry them out. As an operating system, we are using windows OS to install and operate the Autopsy digital forensic software. As Autopsy has the graphical interface mode, we can use the software directly without using the command line or PowerShell of the windows operating system. To use the Autopsy software, we have to first download the software from Internet (www.Autopsy.com) and install the software to our machine.

7.1 Accumulation of Materials

To identify malicious activities, we require a memory dump that records all of the malicious activities running on the system before the generation of the memory dump. After that, the investigator will examine the memory dump and use Autopsy plugins to determine what malicious actions were taken using the device. The memory dump may also be retrieved using a different method. We may perform malicious activities on a device and then generate the memory dump by utilizing the Image-generating tools discussed earlier in this work. In the present situation, there is no pre-created memory dump that has all the possible malicious activities within it. By searching the Internet, we can collect very few secondary

memory dumps as the resources are limited in this area. But they need to be more prosperous. Sure, of them carry out certain sorts of malicious activities, but none of them carry out every form of malicious activity that might exist. Thus, we must resort to generating a memory dump that includes just about every malicious activity that might occur. Once we had generated as many malicious processes as possible inside the device, we created a secondary memory Image. As an operating system, we used Windows to create those malicious activities, and we used a virtual machine to run the system. For capturing the memory, we used a Windows-based free software (discussed in the paper's early chapters).

7.2 Proposed Method

To identify the malicious activities from the secondary memory dump, investigators go through different information that Autopsy offers and try to relate and detect the malicious process. Autopsy mainly automatically generates all the information from the memory dump. Investigators have to view the information that Autopsy generated and try to detect malicious activities. Autopsy, however, produces enormous amounts of data. It will therefore require much time and effort to determine whether any malicious action was taken. All the options that the Autopsy digital forensic tool offers are shown in the table with the direction of using to detect malicious activities from the memory dump.

Table 7.1: Options that Autopsy offers and directions table

Options of Autopsy	The direction of using the options
Search evidence using extensions	Views⇒ By Extensions⇒ Search
Search evidence using File types	Views⇒ File Types⇒ Search
Search evidence from deleted files	Views⇒ Deleted Files⇒ File System⇒ Search
Search evidence from applications	Views⇒ By MIME Type⇒ Application⇒ Search
To know the Operating system version.	Data Source⇒ Select Source⇒ Types
To know all the events on a specific Date.	Data Source⇒ Select Source⇒ Timeline⇒ View in Timeline
To know the password hints.	Keyword Search⇒ Search hint⇒ RagRipper
Website visited	Results⇒ Extracted Content⇒ Web Search
Downloaded files from the internet	Results⇒ Extracted Content⇒ Web Downloads
Find the Sticky Note	Data Source⇒ Windows Volume ⇒ Users⇒ Informant⇒ AppData⇒ Roaming⇒ Microsoft⇒ Sticky Notes⇒ .snt file

Recent Documents accessed	Results⇒ Extracted Content⇒ Recent Documents
Recycle bin files	Results⇒ Extracted Content⇒ Recycle Bin
Email Address Logged in	Results⇒ Keyword Hits⇒ Email Address
Encrypted files	Results⇒ Extracted Content⇒ Encryption Suspected.
USB Attached information	Results⇒ Extracted Content⇒ USB Device Attached

If we follow a certain method to identify the malicious activity inside the memory dump, it will be much simpler to identify and detect. This will be the case since the method has been specified. However, there does not seem to be any standardized method. Finding malicious activity is made much simpler with a methodical approach, saving time, energy, and resources. This project aims to develop and demonstrate a method for quickly and accurately identifying malicious activities in a compiled memory dump. Using Autopsy, our proposed method can quickly and easily search through a memory dump for any traces of malicious activity. The proposed method is a static approach as we are investigating the pre-generated information and trying to identify the malicious activities. The proposed process diagram for identifying the malicious activities from suspected secondary memory dump using the Autopsy software is shown in Figure 7.1.

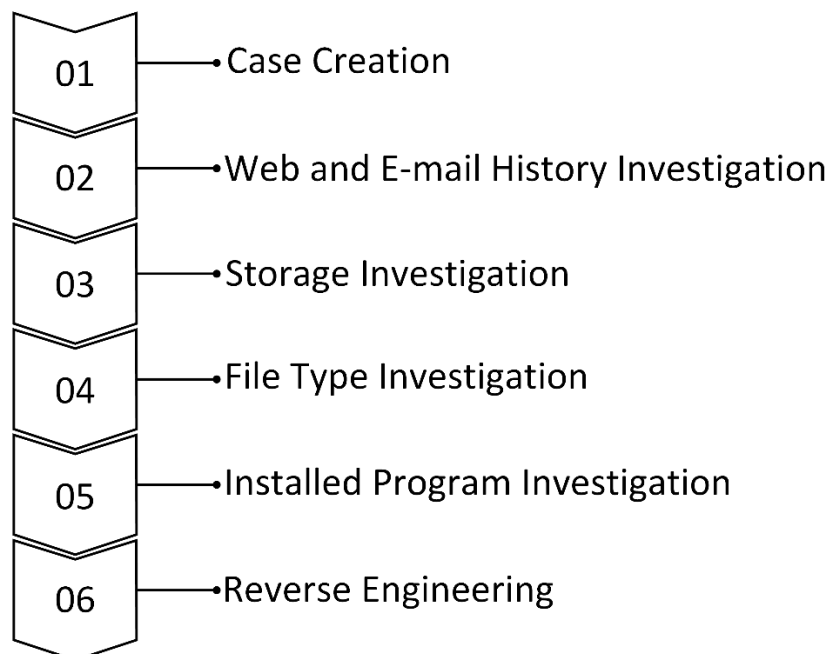


Figure 7.1: Proposed process diagram for identifying the malicious activities

This process diagram is designed to detect the malicious activities in the memory dump captured from the suspected machine. Currently, there are many distinct sorts and varieties of malicious activities, and the identification of the activity might also vary from one instance of that activity to another. Therefore, none of the suggested method's steps will be necessary to detect all malicious activity. It may be feasible to cut off more than one step entirely from a particular activity. However, by following the procedures described in the suggested strategy, we can identify practically every malicious activity that is present at the moment.

7.2.1 Case creation

As we are already downloaded and installed the Autopsy software into the device in this first step of the proposed method, we will create a case for our particular secondary memory dump to investigate. Appropriately creating a case can be used in the documentation, and proper documentation is needed for a pleasant investigation as the documentation will be present in the court. To create a case, the first thing that has to be done is to launch the Autopsy program. After opening the software, we will see there are two options. One is we can open a previous case that we investigated previously, and another one is we can create a new fresh case to make the investigation. As we are investigating a current case, we will select the new case option to open the interface for investigation using Autopsy.



Figure 7.2: Opening a new case in Autopsy Digital Forensic Software

After selecting the new case option, we have to give some case information which is the name of the case and the directory of the case where case data will be stored. If we want to continue the investigation in the future, then we can open the case in Autopsy and continue

the investigation process. For documentation and verification in the future, we also need this case information. After giving all the information, we need to click the “Next” Button.

New Case Information

Steps

1. **Case Information**
2. Optional Information

Case Information

Case Name: Searching For Malicious Activitys

Base Directory: C:\Users\rafid\Documents

Case Type: ☒ Single-user ☐ Multi-user

Case data will be stored in the following directory:
C:\Users\rafid\Documents\Searching For Malicious Activitys

Next > Finish Cancel Help

Figure 7.3: Information fill-up for a new case in Autopsy Digital Forensic Software

After clicking the “Next” button, the next step will be shown on the screen, where we have to give a few more case-related information, but this part is optional. If we don’t want to fill in that information, we can click the “Finish” button.

New Case Information

Steps

1. Case Information
2. **Optional Information**

Optional Information

Case

Number: 1

Examiner

Name: Rafid

Phone: 01982279776

Email:

Notes:

Organization

Organization analysis is being done for: Manage Organizations

< Back Next > Finish Cancel Help

Figure 7.4: Optional information fill up for new case in Autopsy Digital Forensic Software

After clicking the “Finish” button, the next step will be shown on the screen where we have to select the option from which types of data source we will add to the Autopsy for investigation. As we will investigate the secondary memory dump to find the malicious activities using this Autopsy software, we will select the “Disk Image or VM Files” from the options and click the “Next” button to go to the next steps.

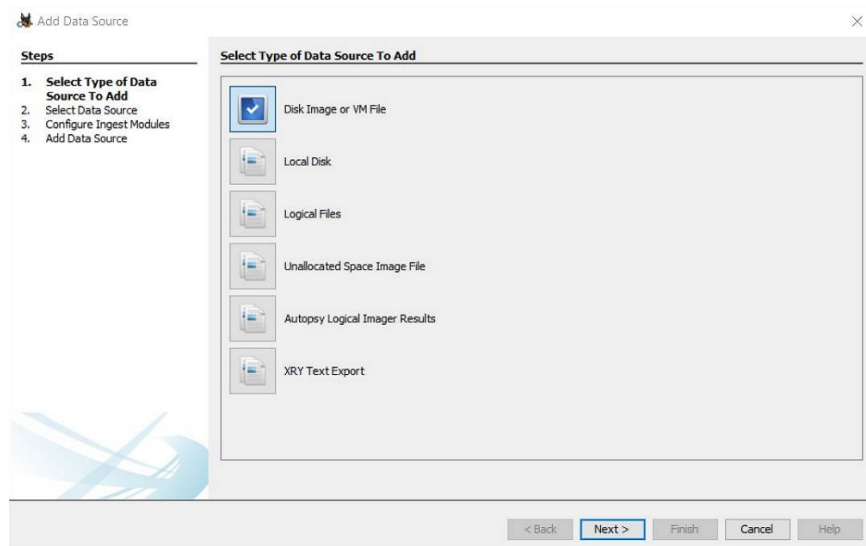


Figure 7.5: Selection of secondary memory dump in Autopsy Digital Forensic Software

After clicking the “Next” button, the next step will be shown on the screen where we have to give the memory dump source path, where the dumped of the memory was located. We will click the “Browse” button and select the location path of the memory dump file. After selecting the source path location of the memory dump file, we will click the “Next” button to go further.

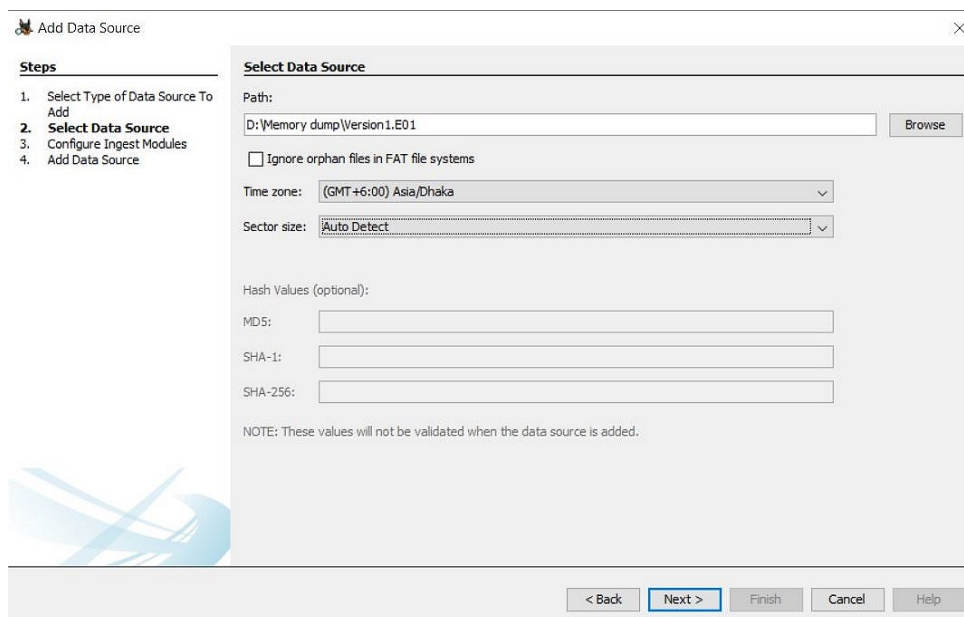


Figure 7.6: Data source selection in Autopsy Digital Forensic Software

After clicking the “Next” button, the next step will be shown on the screen where we have to select the modules for what we will investigate into the machine in which we created the memory dump. All the modules will be pre-selected so that we will click the “Next” button. Next, we will click the “Finish” button. After clicking the button now, all the modules will start loading. Loading the memory dump file into the Autopsy software will take some time. After successfully loading the memory dump files into Autopsy, the environment is now ready to investigate the malicious activities from the memory dump.

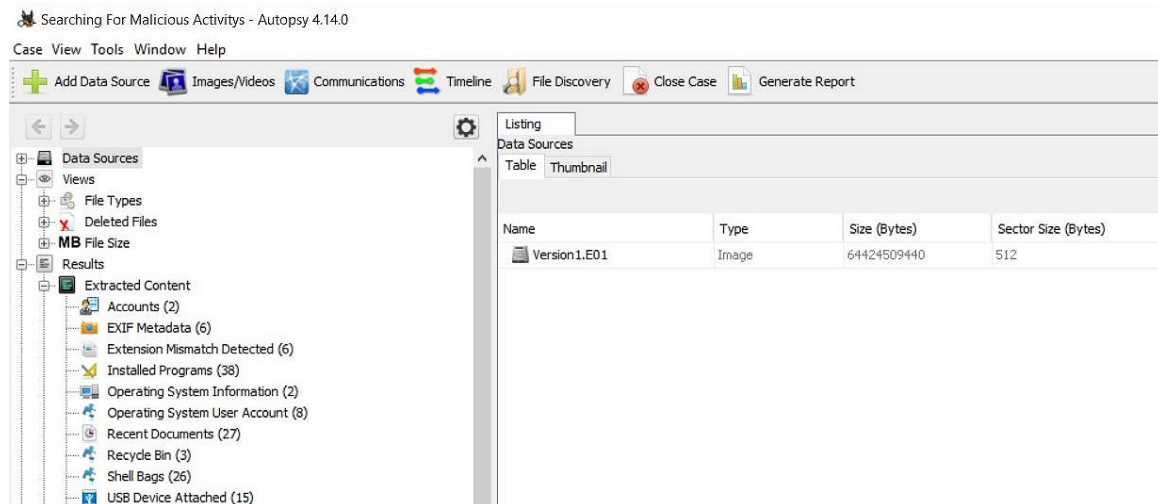


Figure 7.7: Dashboard of Autopsy Digital Forensic Software

Now looking at the left side of the Autopsy software dashboard, we will see a data source option. If we expand that, we will see the partition of the suspected machine disk same as the Windows system shows. We can see how many partitions were there and what files are located there in the partition. We can investigate particular files from here if we detect any file which can be malicious or responsible for malicious activities.

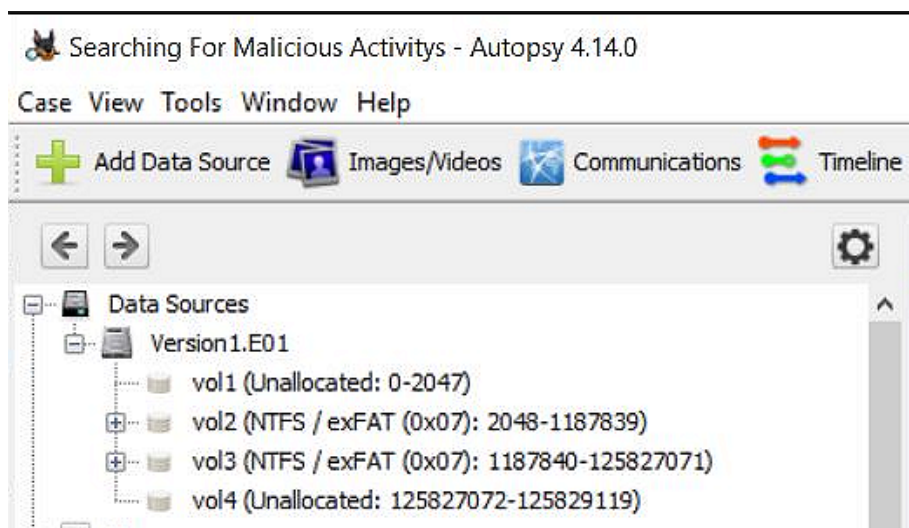
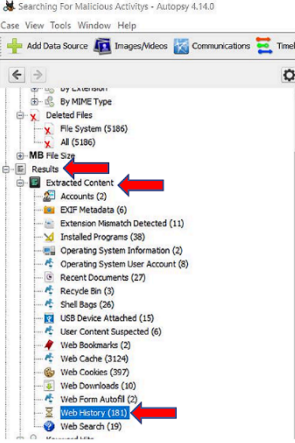


Figure 7.8: Partition viewing in Autopsy Digital Forensic Software

7.2.2 Web and E-mail History investigation

In this step of the proposed method, we will try to find out how many and which website was visited using this machine. To do that, we have to expand the “results” option from the lower left side of the dashboard and again expand the “Extracted content”. The Analyzer of the autopsy extracts some juicy information and shows it in the “Extracted content” section. From here, we will click on the “Web History” section. After clicking the “Web History” section, Autopsy will show all the websites visited using the suspected machine.



Source File	URL	Date Access	Title	Prog
History	https://mail.yahoo.com/	2022-08-04	(1 unread) - prottoycsChrc	
History	https://mail.yahoo.com/	2022-08-04	prottoycse1@yahoo.cChrc	
History	https://mail.yahoo.com/	2022-08-04	prottoycse1@yahoo.cChrc	
History	https://accounts.google	2022-08-04	Inbox (69) - prottoycsChrc	
History	https://www.google.cor	2022-08-04	outlook login - GoogleChrc	
History	https://mail.google.com	2022-08-04	Inbox (69) - prottoycsChrc	
History	https://nijhoom.com/ba	2022-08-04	Bangladesh Holidays Chrc	
History	https://www.google.cor	2022-08-04	location of lalbagh forChrc	
History	https://login.live.com/lc	2022-08-04	Sign in to your MicrosChrc	
History	https://www.google.cor	2022-08-04	Lalbagh - Google MapChrc	
History	https://www.google.cor	2022-08-04	Lalbagh - Google MapChrc	

Figure 7.9: Web history viewing in Autopsy Digital Forensic Software

Here we can see that Yahoo and Google Mail were logged in using the machine. Also, using those emails, some emails were sent, and others were received. If we check the title column, then we can detect prottoycse1@yahoo.com email ID was logged in and composed a mail using the suspected machine.

Source	URL	Date Access	Title	Program	Data Source
History	https://mail.yahoo.com/	2022-08-04	prottoycse1@yahoo.com	Chrome	Version1.E01
History	https://mail.yahoo.com/	2022-08-04	prottoycse1@yahoo.com	Chrome	Version1.E01
History	https://mail.yahoo.com/	2022-08-04	(1 unread) - prottoycse1	Chrome	Version1.E01

Figure 7.10: Logged-in email address viewing in Autopsy Digital Forensic Software

Also, we can see that another email ID prottoycse2@gmail.com checked the inbox mail and then composed a mail to send it. One of the emails has a subject called “About the recent task”.

History	https://mail.google.com/	2022-08-04	about the recent task - p	Chrome Version1.E01
History	https://accounts.google.c	2022-08-04	Google Accounts	Chrome Version1.E01
History	https://mail.google.com/	2022-08-04	Inbox (69) - prottoyce2	Chrome Version1.E01

Figure 7.11: Sending email using the victim's machine

Then google chrome was used to search the Lalbagh fort location, History, Opening time, and Ticket Price information. Also, google Maps was used to see the information from Kalabagan to Lalbagh. From that information, we can see that someone is interested in knowing about the Lalbagh fort. That is different from what we see often. We need to investigate further to identify more information.

URL	Date Access	Title	Program	Data Source
https://www.google.com	2022-08-04	lalbagh fort location - Gc	Chrome	Version1.E01
https://www.google.com	2022-08-04	lalbagh fort location - Gc	Chrome	Version1.E01
https://nijhoom.com/lalb	2022-08-04	Lalbagh Fort: Location, H	Chrome	Version1.E01
https://nijhoom.com/bar	2022-08-04	Bangladesh Holidays B	Chrome	Version1.E01
https://www.google.com	2022-08-04	location of lalbagh fort -	Chrome	Version1.E01
https://www.google.com	2022-08-04	location of lalbagh fort -	Chrome	Version1.E01
https://www.google.com	2022-08-04	Lalbagh - Google Maps	Chrome	Version1.E01

Figure 7.12: Visited websites and locations using the victim machine

7.2.3 Storage investigation

This step will search the storage section to identify any suspected behavior. In the previous step, we learned that the device was used to search the Lalbagh fort information, and also, the map was used to see the direction. So, we will find out if any images and videos are there related to Lalbagh Fort, as we previously found there through searching and location-finding events of Lalbagh Fort there. To do that, we have to expand the Views option, and then we have to expand the File Types option. From there, we have to expand the by Extension option, and after expanding the by Extension option, we will see all the file types like images, videos, audio, and many more file types are there. As we will search for images and videos from there, we will open the images and videos option. We have to search for Lalbagh Fort related images and videos by viewing all the images and videos one by one.

Name	Modified	Time	Access	Size	Flags(Dir)	Known	Location	Extension
DSC02423.JPG	2014-04-18	2022-08-04		6657008	Allocated	unknown	/img_Versio	jpg
DSC02426.JPG	2014-04-18	2022-08-04		6624726	Allocated	unknown	/img_Versio	jpg
DSC02432.JPG	2014-04-18	2022-08-04		6732267	Allocated	unknown	/img_Versio	jpg
DSC02547.JPG	2014-05-22	2022-08-04		6759447	Allocated	unknown	/img_Versio	jpg
DSC02548.JPG	2014-05-22	2022-08-04		6714854	Allocated	unknown	/img_Versio	jpg
DSC02554.JPG	2014-05-22	2022-08-04		6664186	Allocated	unknown	/img_Versio	jpg

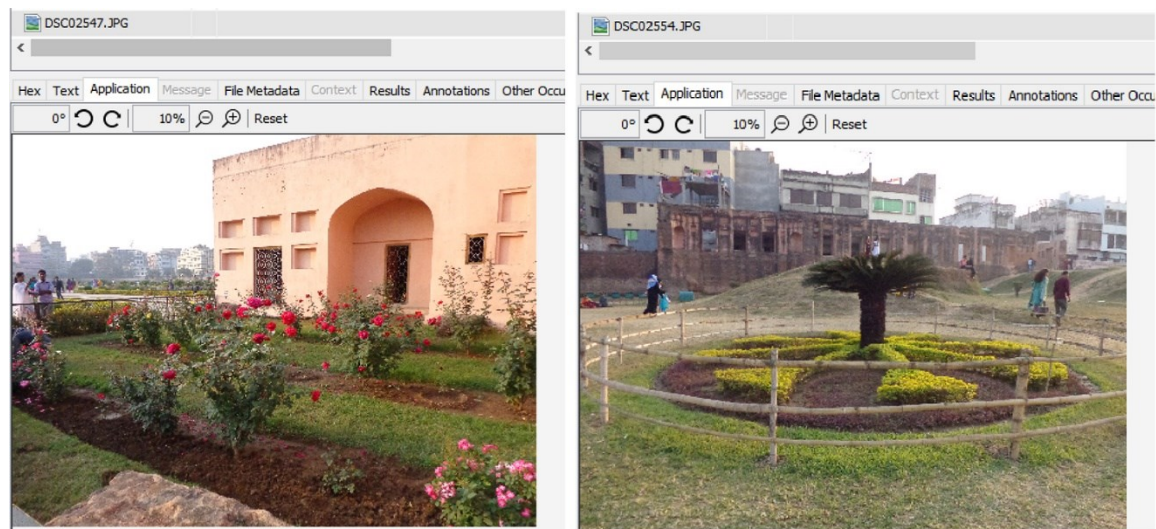


Figure 7.13: Images downloaded into the victim's machine

Here we can see some of the pictures of the Lalbagh Fort present in the memory dump of the suspected computer or device. If we look into the date and time of the picture, we can see that all of these pictures were saved into the machine on the same date, and the time was within the occurring incidence period. With the Lalbagh Fort picture, we can also see some laptop pictures were there at the same time and date. This laptop picture may be important for our investigation. We must investigate more to determine if those pictures are related to malicious activities.

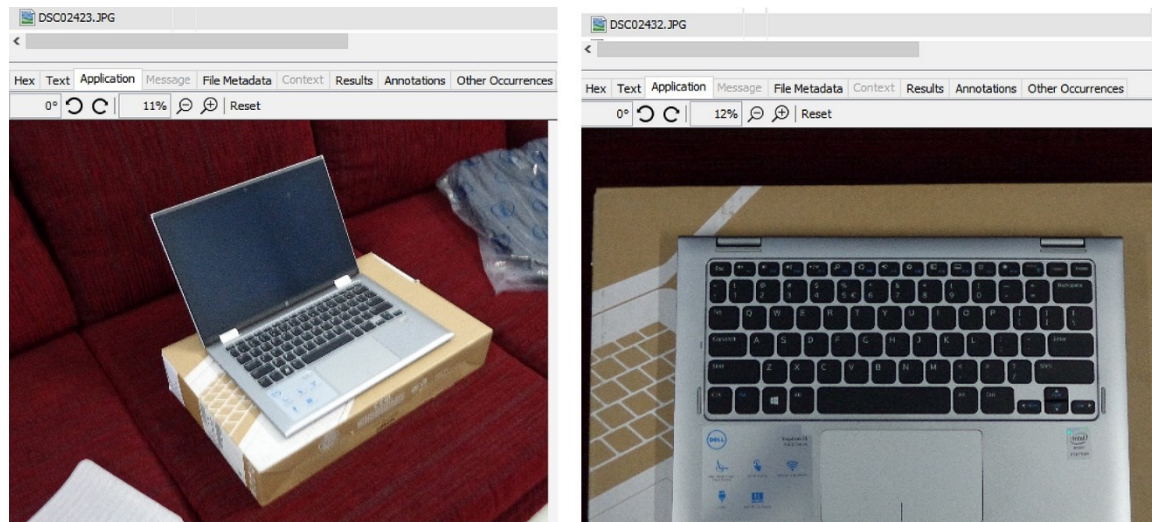


Figure 7.14: More images downloaded in incident occurring time

7.2.4 File type investigation

In this step, we will try to find whether or not the previously selected suspicious list files are trying to hide some information. All images, audio, video, or text files have some specific file extension type. For example, if we consider a video file, then the file extension type can be .mp4 or .mov, and many more. But suppose we see a video file with a .bmp extension. In that case, a particular video file may have some hidden information because a video file with a .bmp extension is not what we see often. So, if we find any file with an uncommon extension, we can select the file as suspicious. As we found some pictures in the memory dump related to Lalbagh Fort that was previously searched using the machine, we will now try to find out if those pictures are trying to hide something. If we expand the Result option, then “Extracted Content,” and then Extension Mismatch Detected option, we can see the same pictures are here, which we found suspicious. These pictures used to be in image format or JPG extension, but they are in BMP extension. So, we are selecting those pictures as our suspicious list and trying to gather more information about those pictures.

Source File	Extension	MIME Type	Data Source
DSC02547.JPG	jpg	image/bmp	Version1.E01
DSC02554.JPG	jpg	image/bmp	Version1.E01
DSC02423.JPG	jpg	image/bmp	Version1.E01
DSC02432.JPG	jpg	image/bmp	Version1.E01
DSC02547.JPG	jpg	image/bmp	Version1.E01
DSC02554.JPG	jpg	image/bmp	Version1.E01
DSC02423.JPG	jpg	image/bmp	Version1.E01
DSC02547.JPG	jpg	image/bmp	Version1.E01
DSC02554.JPG	jpg	image/bmp	Version1.E01

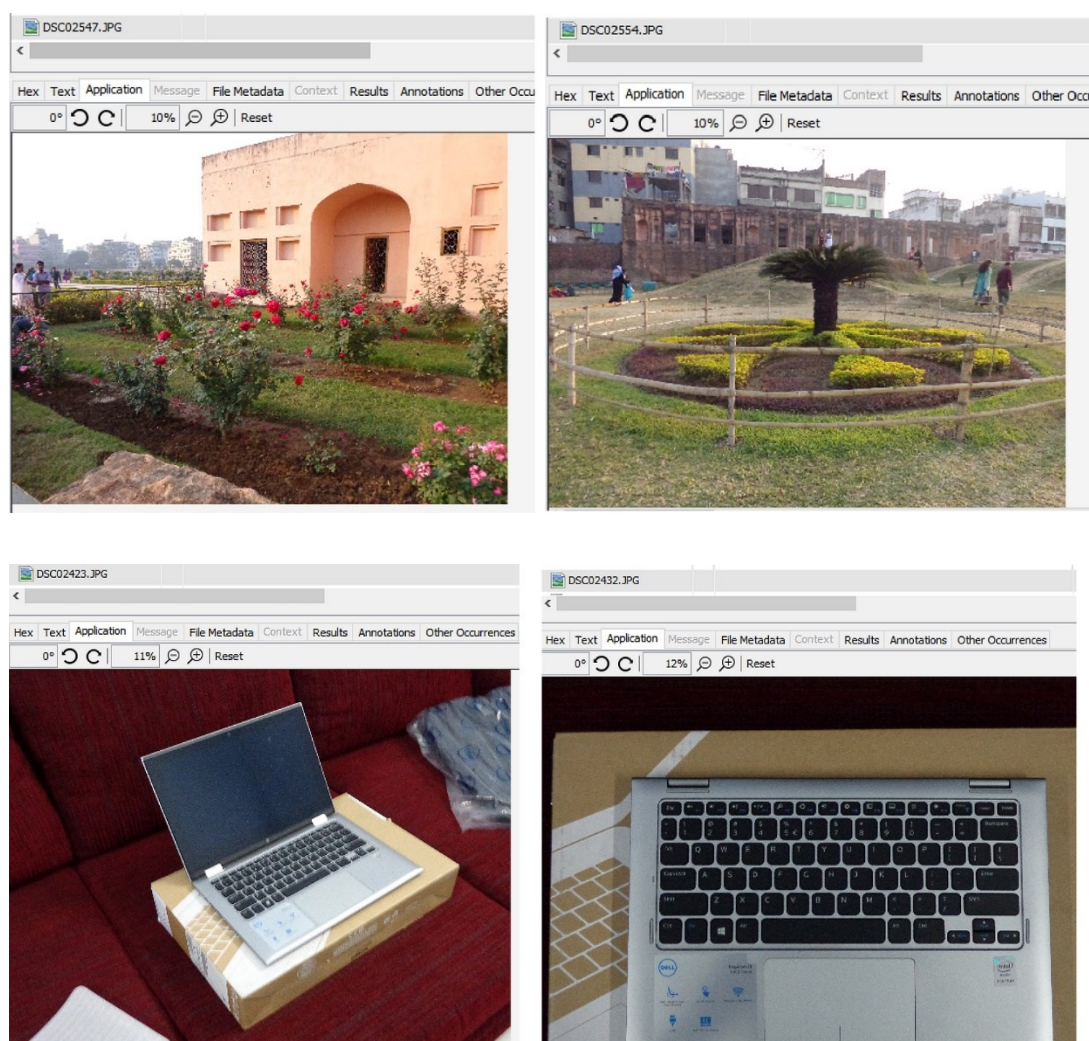


Figure 7.15: Mismatched Extension images

7.2.5 Installed Program investigation

In this step, we will determine if any outside the installed program has any involvement with the suspicious list file. In our questionable list, if any extension mismatched file is present, there is a high possibility that the file was converted using software installed on

the machine. The installed program may have included some information hidden in the file and, after that, converted the file so that no one could find out what information was hidden in the particular file. So, we have to investigate the involvement of the installed program. As we identified those picture's format was mismatched now, we will investigate why this mismatch occurred. As we see, the pictures are usually saved into the machine in Image format, but in this machine, some of the pictures which are also in our malicious list are in a different form. That can be done by using any software or someone manually converting the image format or extension intentionally. Now we will try to find out if any software or process did this format change or not. To do that, we have to go to the "Installed Programs" section in the same "Extracted content" option. If we go through all the software installed into the device, we can see there is software installed into the device called "Open Stego". The Open Stego is a Steganography software. Steganography is a hiding information technology that is used to hide some secret information in ordinary files, images, text files, or audio that is non-secret files. These types of files where some information is embedded are called carrier files. These types of files can be carried or transferred using the flash drive or cloud drive or can be attached with the mail. Data or information which was hidden in those carrier files can be extracted using the same software anytime. Also, we can see that the date and time are the same as when the incident happened.

Source File	Program Name	Date/Time	Data Source
SOFTWARE	OpenStego v0.6.1	2022-08-04 18:18:47	Version1.E01

Figure 7.16: Software installed into the suspected machine

7.2.6 Reverse engineering

In this step, we will try to find the malicious activities by reverse engineering. We will try to backtrack the process and determine the presence of malicious activities and the types of malicious activities. We found software installed into the suspected machine at the same time and date of the incident. From the previous investigations, we found some images which are on our malicious list. Now we will try to find the hidden messages which were hidden in the carrier files using Steganography. To do that, we will download the same software from the internet and install it into our machine to investigate those images. After that, we will extract those selected images into our computer. To do that, we will go to the same "Images" section where we found those images and right-click on a selected image. After that, we have to click the Extract Files option and give the destination path to save

the image into our device. The file will be saved on our machine when we click the save button.

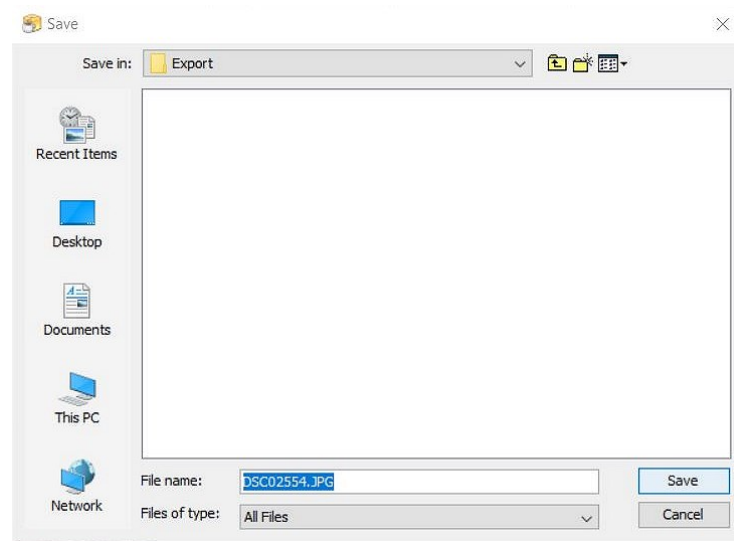


Figure 7.17: Save files into the computer disk which is used in the investigation

After saving the file into our machine now, we will open the software which we previously installed to investigate the file. We will click the “Open Image” button to open those selected images and go to the same directory where we saved those images. After that, we will click the “Open” button.

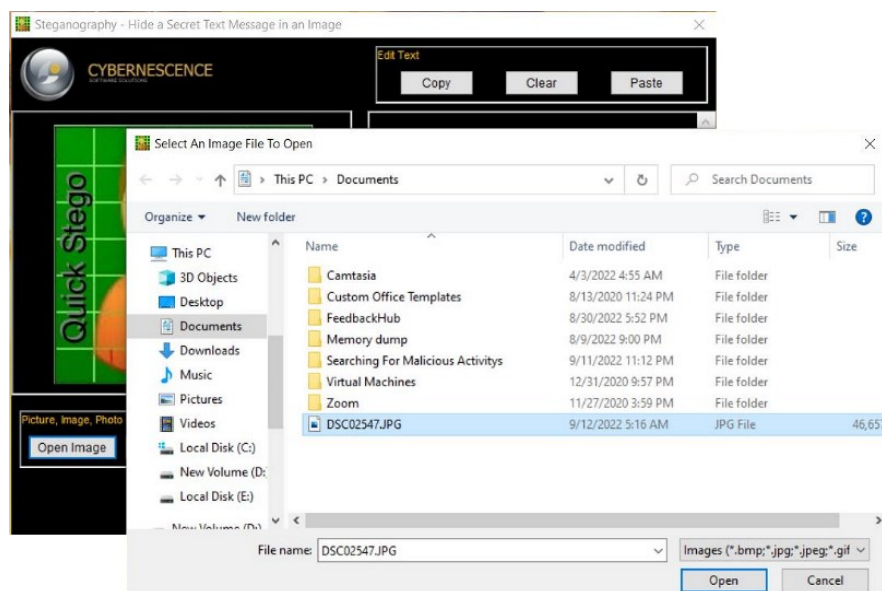


Figure 7.18: File opening using Steganography software

After clicking the opening button picture will load into the software. Now we will click the “Get Text” button. When we click the “Get Text” button, the hidden text of that image will be shown on our screen.

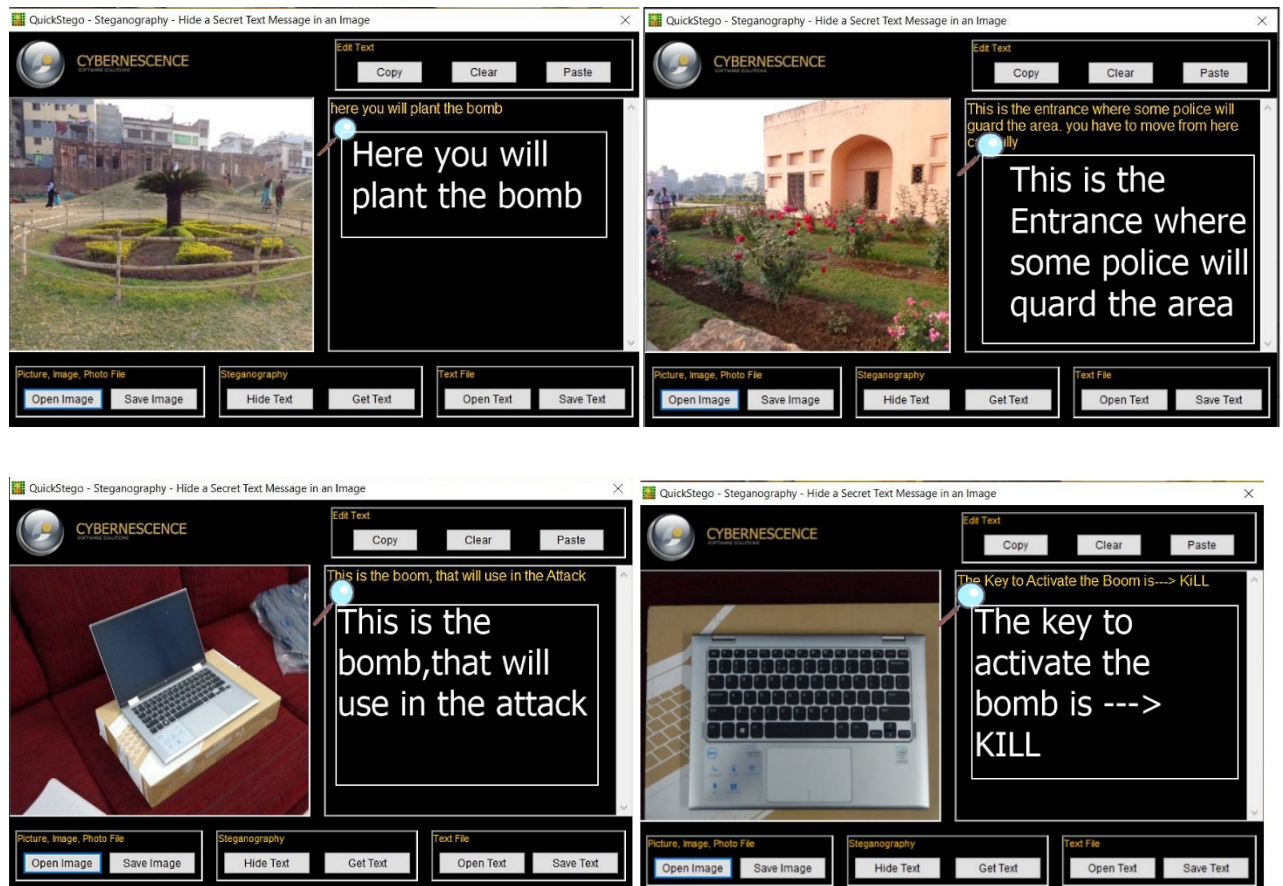


Figure 7.19: Extracting hidden information using Steganography software

Here we can see that all of those four pictures have some information hidden in those pictures. By extracting the information which was hidden in those picture files now, we came to know that there was a terrorist attack planned on the Lalbagh Fort and the Laptop would be the bomb. To activate the bomb, the keys “KILL”.

This is how we investigate the memory dump captured from the suspected machine’s secondary memory. We must remember that not all memory dumps will be identical to our suspected memory dumps. There will be a different scenario for every memory dump, and the investigation process will be different for every memory dump because of the different types of evidence present in the memory dump. But more or less, we will investigate everything shown in this project. Using our proposed method steps, we can easily detect and identify malicious activities by conducting those 6 steps. It will take fewer steps if the scenario is different, but almost all the malicious activities can be found using the proposed method.

Chapter 8

Future works

We explained some features of two different software in this project for identifying malicious activities from two different memory dumps (Primary memory dump or main memory dump and secondary memory dump), which are captured from a suspected device. We collected the main memory dump from the Internet source. Because as we are trying to find out malicious activities, those malicious activities can be very harmful to our personal computers. Creating malware is also impossible for us as we are not experts in that, and it will be unethical too. So, we had to rely on the Internet-sourced memory dump, which professionals created to practice memory forensics. But collecting those memory dumps was very challenging because there are very limited memory dumps of our related subject available in the present. From that few options, some of them are already removed, or sources have been deleted already. So, from those fewer options, very few were available for download and investigation. Also, capturing time can be high with the capacity of RAM. If we take a regular RAM and try to create a memory dump, it will take a very long time to build that memory dump and the investigation. We investigated and tracked some malicious activities from the downloaded memory dump. It is possible to collect some more memory dump. In that case, we can make different investigations not done in our investigation and find different results. So, in the future, we will try to collect more memory dumps and investigate to discover some unexplored malicious activities. The same goes for the secondary memory dump, but on the Internet, we were still looking for a suitable memory dump that would tell us a true story of an incident. So, we had no option but to create a malicious secondary memory dump or disk image and investigate that captured memory dump. But in our captured memory dump, we can't cover all the malicious activities as we can't simulate them. If we find more memory dumps in the future, we will try different types of investigations and might get different results.

Chapter 9

Conclusions

Presently, the dependency on computers, smart devices, and Internet usage is very high. Users are downloading content from the Internet to their devices. With those downloaded files, users also download the malware without knowing of its presence. As malware is very smart to hide the traces, it isn't trivial to discover its presence on devices. When it affects the user's files and makes harmful activities to the user's device only that time, the user knows about the malware. But it will be too late. So, early detection of the malware programs or processes is critical to prevent data loss or unknown attacks. Memory analysis plays an important role in those cases. Also, for knowing the activities of the device used by attackers, memory forensic investigation is critical to detect, track, identify and make evidence of the attacks that have already been done or will be done. For professionals, memory investigation in an efficient way is needed to reduce time and make investigation faster than before. By using efficient software and tools, the investigator can do their job more quickly than the traditional ways. Brute-force searching for malicious activities can take a lot of effort and be time-consuming. For malicious activities finding, if we use mentioned tools and methods, it will be much faster than Brute-force searching. So, using the right tools for the correct memory dump is mandatory for more rapid progress, as professionals must do their work as fast as possible.

References

- [1] Cyberattacks 2021: Statistics from the Last Year. 2022. Available online: <https://spanning.com/blog/cyberattacks-2021-phishingransomware-data-breach-statistics/> (Last accessed on 14 July 2023).
- [2] Latzo, T.; Palutke, R.; Freiling, F. A universal taxonomy and survey of forensic memory acquisition techniques. *Digit. Investig.* 2019, 28, 56–69.
- [3] D. R. Tobergte and S. Curtis, *The Art of Memory Forensics*, vol. 53, no. 9. 2013.
- [4] I. Korkin and I. Nesterov, *Applying Memory Forensics to Rootkit Detection*, 2015.
- [5] H. Nyholm et al., “The Evolution of Volatile Memory Forensics,” *J. Cybersecurity Priv.*, vol. 2, no. 3, pp. 556–572, 2022.
- [6] A. Aljaedi, D. Lindskog, P. Zavarisky, R. Ruhl, and F. Almari, “Comparative analysis of volatile memory forensics, live response vs. memory imaging,” *Proc. - 2011 IEEE Int. Conf. Privacy, Secur. Risk Trust IEEE Int. Conf. Soc. Comput. PASSAT/SocialCom 2011*, pp. 1253–1258, 2011.
- [7] A. S. Bozkir, E. Tahillioglu, M. Aydos, and I. Kara, “Catch them alive: A malware detection approach through memory forensics, manifold learning and computer vision,” *Comput. Secur.*, vol. 103, 2021.
- [8] R. B. van Baar, W. Alink, and A. R. van Ballegooij, “Forensic memory analysis: Files mapped in memory,” *Digit. Investig.*, vol. 5, no. SUPPL., pp. 52–57, 2008.
- [9] K. M. A. Kamal, M. Alfadel, and M. S. Munia, “Memory forensics tools: Comparing processing time and left artifacts on volatile memory,” *IWCI 2016 - 2016 Int. Work. Comput. Intell.*, no. December, pp. 84–90, 2017.
- [10] K. Ghazinour, D. M. Vakharia, K. C. Kannaji, and R. Satyakumar, “A study on digital forensic tools,” *IEEE Int. Conf. Power, Control. Signals Instrum. Eng. ICPSI 2017*, pp. 3136–3142, 2018.
- [11] C. Easttom, *ALL IN ONE CCFP Certified Cyber Forensics Professional Certification*. 2015.
- [12] J. Oh, S. Lee, and S. Lee, “Advanced evidence collection and analysis of web browser activity,” *Digit. Investig.*, vol. 8, no. SUPPL., pp. S62–S70, 2011.

- [13] D. N. Patil and B. B. Meshram, "Web browser analysis for detecting user activities," *Adv. Intell. Syst. Comput.*, vol. 707, no. July, pp. 279–291, 2019.
- [14] A. Almutairi et al., "Evaluation of autopsy and volatility for cybercrime investigation a forensic lucid case study," *Int. J. Digit. Crime Forensics*, vol. 12, no. 1, pp. 58–89, 2020.
- [15] J. Kävrestad, *Fundamentals of Digital Forensics: Theory, Methods, and Real-Life Applications*. Cham: Springer International Publishing, 2020.
- [16] D. Rahevar, "Study on Live analysis of Windows Physical Memory," *IOSR J. Comput. Eng.*, vol. 15, no. 4, pp. 76–80, 2013.
- [17] VöMel, S.; Freiling, F.C. A Survey of Main Memory Acquisition and Analysis Techniques for the Windows Operating System. *Digit. Investig.* **2011**, 8, 3–22.
- [18] D. Dayalamurthy, "Forensic memory dump analysis and recovery of the artefacts of using tor bundle browser-the need," *Proc. 11th Aust. Digit. Forensics Conf. ADF 2013*, pp. 71–83, 2014.
- [19] D. M. Rathod, "WEB BROWSER FORENSICS: GOOGLE CHROME," *International Journal of Advanced Research in Computer Science*, vol. 8, no. 7, pp. 896–899, Aug. 2017.
- [20] A. C. Popescu and H. Farid, "Statistical Tools for Digital Forensics," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 3200, pp. 128–147, 2004.
- [21] F. E. Salamh, M. M. Mirza, and U. Karabiyik, "Uav forensic analysis and software tools assessment: Dji phantom 4 and matrice 210 as case studies," *Electron.*, vol. 10, no. 6, pp. 1–14, 2021.
- [22] J. Bellizzi, M. Vella, C. Colombo, and J. Hernandez-Castro, *Responding to Targeted Stealthy Attacks on Android Using Timely-Captured Memory Dumps*, vol. 10. 2022.