Simultaneous Experimental Investigative Approach towards Digital Forensics

by

Victor Basu
B.Tech(CSE), West Bengal University of Technology, 2013

A Project Submitted in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

in the Department of Computer Science

Simultaneous Experimental Investigative Approach towards Digital Forensics

by

Victor Basu
B.Tech(CSE), West Bengal University of Technology, 2013

Supervisory Committee

Dr. Jens Weber, Supervisor
(Department of Computer Science)

Dr. Sudhakar Ganti, Departmental Member
(Department of Computer Science)

Supervisory Committee

Dr. Jens Weber, Supervisor
(Department of Computer Science)

Dr. Sudhakar Ganti, Departmental Member
(Department of Computer Science)

ABSTRACT

Digital forensics is a sub-branch of forensic science which revolves around the acquisition and investigation of information acquired from digital sources, which can often be related to cybercrime. A digital forensic investigation can be associated with a number of scenarios encompassing public and private domains, ranging from evidence related to a civil or criminal case in court to an internal investigation of employees suspected of a data breach within an organization. Understanding the importance of digital forensics has become really important in this day and age with the recent advent of hacking attempts [1] at a number of multinational companies worldwide, whose most prime asset is their data. In addition to safeguarding their sensitive data from being maltreated, companies are also bound to a host of state, local and federal rules and regulations when it comes to preservation of data. This document is a possible representation of investigative approaches adopted by digital forensic engineers to analyze data that is acquired as part of a forensic investigation. A data set of a suspected machine along with a couple of removable storage devices and a cloud storage provider that were used in a data leakage case will be analyzed using a plethora of forensic analysis tools ranging from file carvers, email retrievers to database restoration techniques, to name a few.

Contents

List of Figures

ACKNOWLEDGEMENTS

I would like to thank:

Dr. Jens Weber, for his continuous insight, mentoring and support. I appreciate his inputs and ideas that led to the successful culmination of this research work.

Dr. Sudhakar Ganti, for his encouragement and guidance during this research work.

Wendy Beggs, for her role in providing unconditional support and crucial advice to graduate students.

Family and friends, for lending me boundless support throughout my journey in graduate school.

Don't think twice before hitting that delete button, think twice before creating the data itself. Deleting it doesn't create a void, it gives birth to suspicion. It's never tough to hack the majority of the crowd. All it takes is watching them, listening carefully to what they have to say, their susceptibilities resemble what they are trying to advertise. But just because you can exploit them, doesn't give you the control. As control is just an illusion.

DEDICATION

I dedicate this work to Ma, Nandu and Bhomma for their continuous support during my studies.

Chapter 1

Introduction

In the modern world we are surrounded by smart computing devices all around us, from the fitness trackers on our wrist to the highly powerful smartphones in our pockets. The introduction of these devices in our lives had the primary intention of making our lives easier by acting as digital assistants, but through time people have learned to exploit these devices and tamper with their data and functionality. These devices have also started to act as an aid in the transportation of confidential data, which may otherwise be unauthorized, which in-turn has led to the high demand of experts in the field of digital forensics investigation. In such a fast paced computing world, digital forensic investigative agencies are either leading the way or playing catch up to a new loophole or backdoor discovered in a system. This leads to a continuous process of digital hide and seek between the intruder and the investigator.

Every digital forensic examination starts with identification of acquired data. Before proceeding with the investigation, it is important to identify where the data that is obtained, was stored. Data can be retrieved from smartphones, hard drives, cloud servers, flash drives etc. Data from a crime scene can be acquired in the following ways:

- Dead & Live image acquisition: One of the best practices while acquiring a disk image is to disconnect the power source of the storage medium and access the drive via a forensic workstation where a write blocker is installed. This assures the prevention of any changes being written to the storage medium in question. However certain scenarios might not provide the opportunity for a forensic engineer to cut power to a device, as it may lead to either deletion/corruption of data, which might be triggered by a malicious process planted by the attacker.

In these cases, a live image of the machine which is still in a "powered on" state must be acquired. This is also applicable to cases where a hard drive might be fully encrypted and the forensic investigator might not have access to the decryption key, or if the system resides in a remote location, hence doesn't allow the power to be disconnected from the system or the powering off the system could have a major business impact on an organization. One of the major advantages of live acquisition is that it makes way for live memory capture too, which can help an investigator to find out whether any live processes are malicious in nature and if they were communicating to another remote machine which could supposedly be controlled by an attacker, thus acting as an aid in narrowing down the source.

- Physical & Logical acquisition: Physical imaging of a storage device involves the capture of data in binary form which means accumulation of all 0s and 1s residing within it. This process also captures the deleted and unused space on the drive in question. If the drive underwent a recent format cycle, this process will try to capture the deleted files and their fragments which can be useful in conducting a successful forensic investigation. Hence if the size of the hard drive is 4 TB, the generated size of a physical acquisition will be 4 TB. An issue with this type of imaging is the lack of live memory obtainment. Sometimes running and background processes are intertwined to the generation of malicious data on a storage unit. On the other hand, logical acquisition entails the capture of data that is currently in use or also referred to as active data. Logical extraction of data often requires an application to be installed using administrative/root privileges on the system and is usually smaller in size than its physical counterpart. In this case if only 80 GB of a 4TB hard drive is being used actively, only that portion will be acquired and the deleted file segments will not be captured. This type of acquisition is preferred while working with enormous data sets that can span over petabytes of data. But as a standard physical acquisition is preferred over logical capture as obtainment of crucial data might be skipped as a result of choosing the latter.

- Proprietary acquisition: Embedded systems and analytical generators usually generate this type of data that is associated with a certain organization while meeting industry standards. Dedicated capturing mechanisms are implemented to gather information from a data source which doesn't generate standard data

formats. These can be captured using third party standalone data acquisition tools along with write blockers.

1.1 Motivation and Purpose

An increasing number of attacks are being reported on a daily basis and that too on a global scale. The fraction of identified attacks that are reported can have a minuscule value [2] if and when compared to the number of attacks which can escape detection. This can raise an alarm for security experts at a stage where the breach might pose a larger threat than what it could have posed if it was detected in its infancy. Technical literacy is on the rise, where more and more of the general population is learning the use of digital devices, but what they are not learning is the "proper use" of these devices. The average user needs to be aware of the circumstances they might land up in if they click a link within a well-crafted email containing that malicious link. When the same individual user is part of a multi-national organization, it may lead to widespread infection of the company's systems. If a similar situation pertains to a financial organization, this might quickly escalate to a situation where millions of individuals can be affected on a global scale. This leads to the importance of this report which will not only appeal to an individual without any background in digital forensics by giving him/her an introduction about the same with the help of examples related to a data leakage case but also to the expert user by giving them the nitty gritty details of file carving, extracting emails from the file system, re-constructing databases from deleted snapshots of the same and more. This report is an attempt to explain the usage and comparison of tools and methodologies used in the field of digital forensics with the help of real life examples and not just jumping to a conclusion without making the reader understand the ways which leads an investigator to make an inference.

1.2 Structure of the Project

This section provides an outline of the report and synopses of the content belonging to an associated chapter is summarized as follows:

Chapter 1 provides an introduction to the world of digital forensics, imaging mechanisms, their importance and their necessity in a world that is becoming more

and more digitized every second. It also houses the motivation and purpose behind the pursuit of this project.

Chapter 2 describes the data set that I will be working with, its technical specifications and tools/methods used during extraction of data from the devices confiscated from the suspect.

Chapter 3 outlines information about the approaches that need to be adopted in order to conduct a successful digital forensic investigation related to the data set of choice.

Chapter 4 provides an in-depth review of data that is obtained. This is the section where the forensic investigation is performed. A detailed analysis is presented relating to every aspect of the data set that is dealt with. Along the way, there is also a comparison of various digital forensic tools that are used in the process of uncovering hidden information.

Chapter 5 looks into the investigation of additional removable devices that were confiscated from the suspect and further analysis is conducted on these storage units.

Chapter 6 re-affirms the successful use of tools and methodologies adopted for specific use with the associated data set and also confirms how the experiments were successful in uncovering the process of a data leak.

Chapter 2

Data Set Selection

There are quite a few data sets available on the world wide web that are intended for research purposes related to the field of digital forensics and network security. When it comes to choosing a single data set, one thing that needed to be kept in mind for a project of this stature is the ability to show methods of data extraction on a variety of electronic media, which could help the reader to understand the severity of the current situation of data security. Another thing that needed to be addressed was the use of a data set that provided a scope for future work by making available data captures via different mechanisms. This report is an attempt to provide the reader an insight into the mind of a digital forensics investigator and the approaches one takes to solve a case. The technical details of the data set that was used is described in the following section.

2.1 Background of Data Set in use

After careful consideration, the decision was to use the "Data Leakage Case" data set created by the National Institute of Standards and Technology [3] as part of the Computer Forensic Reference Data Sets (CFReDS) [4]. This is a repository of images extracted from suspected devices, which are made available online, for research purposes. Some of these images are created by NIST, commonly via the Computer Forensics Tool Testing (CFTT) project [5], while others are provided by different organizations. The CFReDS project was funded by the National Institute of Justice along with the NIST Office of Law Enforcement Standards. Documentation related to the available data sets are highly accurate and updated. Every data set is carefully

monitored and any information that pertains to people/organizations in the real world are replaced by random names which are purely fictional. The Data Leakage Case that is being investigated in this report is an excellent example of that.

The primary reason behind the selection of the "Data Leakage Case" is the versatility that this dataset has and its multi-skill holistic nature. It is substantial in its size and is quite a complex set of images revolving around intellectual property theft.

2.2 Case Specific Scenario

This information is provided as part of documentation related to the case. The primary suspect is the manager "*Iaman Informant*" of the technology development division of famous multinational corporation "*OOO*". It is suspected that this manager was contacted by a "*Spy Conspirator*" from a rival company and that this conspirator had offered the manager money in exchange of sensitive information related to an upcoming technology at "*OOO*'. The manager is believed to have made an effort to hide the plans of the data leak. Emails were exchanged between the two by masking them under a business relationship approach. A cloud storage provider was also used to upload a part of the data. As too much data was not being able to be uploaded to the cloud, the conspirator asked the manager to provide the remaining substantially larger amount of data on a portable storage unit, which in this case was a USB flash drive and a CD. These devices were briefly scanned at the security checkpoint, but there was no evidence of any data leak, so they were directly transferred to a digital forensics investigative agency for further investigation. This is where the project started and investigation of the acquired data began.

2.3 Captured Devices

It is very important to gather metadata while the capture of forensic evidence, which can help rule out any inconsistencies and mal-tamper related to the data. The following information about the confiscated devices were provided before starting the investigation.

Personal Computer (PC)	Type	Virtual System
	CPU	1 Processer (2 Core)
	RAM	2,048 MB
	HDD Size	20 GB
	File System	NTFS
	IP Address	10.11.11.129
	Operating System	Microsoft Windows 7 Ultimate (SP1)

Figure 2.1: Details of Informant's PC system

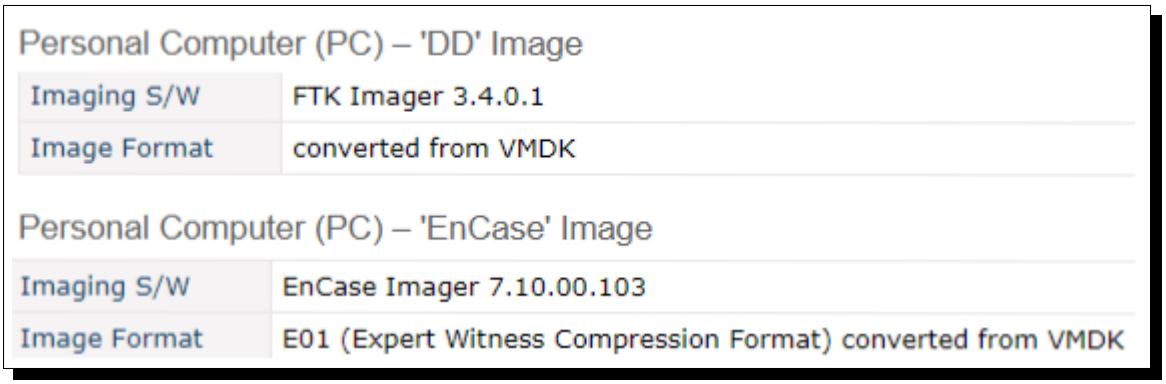
Removable Media #1 (RM#1)*	Type	USB removable storage device
	Serial No.	4C530012450531101593
	Size	4 GB
	File System	exFAT
Removable Media #2 (RM#2)	Type	USB removable storage device
	Serial No.	4C530012550531106501
	Size	4 GB
	File System	FAT32
Removable Media #3 (RM#3)	Type	CD-R
	Size	700 MB
	File System	UDF

Figure 2.2: Details of Removable Media (Flash Drive & CD confiscated from the Informant)

2.4 Information about Acquired Data

Data from the suspected machine was captured using a couple of imaging software(s). One was FTK Imager v3 and the other one was EnCase Imager v7. The image from the former software was converted from a virtual machine disk (VMDK). It is a container which is used to store virtual hard disks that are used to run on virtual environments. In this case the virtual machine disk was converted to a dd image which is sometimes referred to as GNU dd [6]. It is a primitive form of imaging systems and lacks features such as obtaining metadata, on the fly error correction, a

user-friendly UI, which are present in modern tools. In-spite of these drawbacks, it is one of the most robust tools still available to a forensics investigator. FTKImager has a native option of generating these files via a virtual machine disk file, which is located in the working directory of the VM that needs to be converted. Once the proper vmdk file is pointed to, FTKImager can take care of generating a raw (dd) image of the file-system.



Personal Computer (PC) – 'DD' Image

Imaging S/W	FTK Imager 3.4.0.1
Image Format	converted from VMDK

Personal Computer (PC) – 'EnCase' Image

Imaging S/W	EnCase Imager 7.10.00.103
Image Format	E01 (Expert Witness Compression Format) converted from VMDK

Figure 2.3: Details of PC DD and Encase Image

It is always a standard practice to make multiple copies while acquiring data, hence an alternate method in the form of EnCase was used additionally, while imaging the suspected machine. While using EnCase, the file format that was generated was "E01" instead of "dd". While generating E01 images, Encase splits up the entire disk into chunks of 640 MB, as a result of which a single image can often be divided into multiple parts. Although the extension gets changed from E01 to E02, E03, E04, henceforth, the integrity of the filesystem is maintained. Every E01 file consists of a header which stores information about the case. Within the image itself [7], there is a Cyclic Redundancy Check (CRC) after each and every block 64 sectors which translates to 32 KB of data. The advantage of this is that if there is an error within the 32 KB space, it will be picked up by the Cyclic Redundancy Check.

A CRC is originally a hash function which tries to match a value that was generated while creation of a block with the one that is generated while checking the same block for inconsistencies. The footer of an E01 image houses the MD5 value which can be matched with the MD5 value that is generated by another tool. If both values match, it assures that the data has not be tampered with.

One of the removable media's (USB Flash Drive) that was confiscated was also

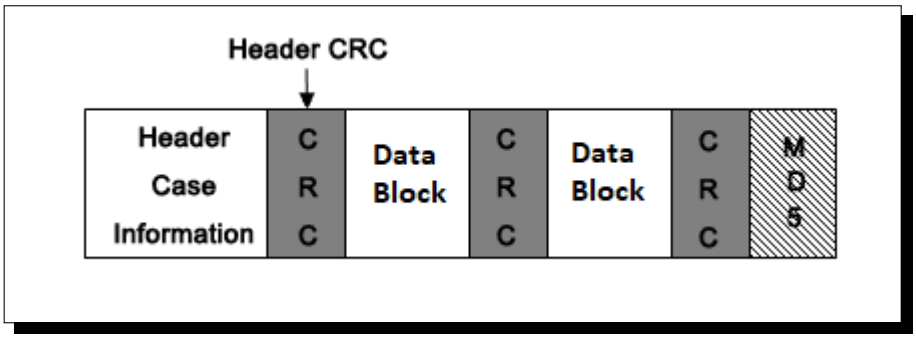


Figure 2.4: E01 file format

imaged using FTK Imager and EnCase Imager and a .dd and .E01 image was generated. In this case though, it was ensured that there was no writing to this flash drive by the use of Tableau USB Bridge T8-R2 [8] developed by Guidance Software [9]. This specific write blocker is built for a wide range of USB devices from USB 3.0, 2.0, 1.1 full speed as well as low speed USB devices. It can also extract information from usb-based media players and cameras. The internal power circuitry is designed in such a way that it can function without the need of an external power source while providing higher USB bus power. There is an LCD panel on device which can display technical information about a certain acquisition via USB and vital information about the device itself.

Removable Media #2 (RM#2) – 'DD' Image

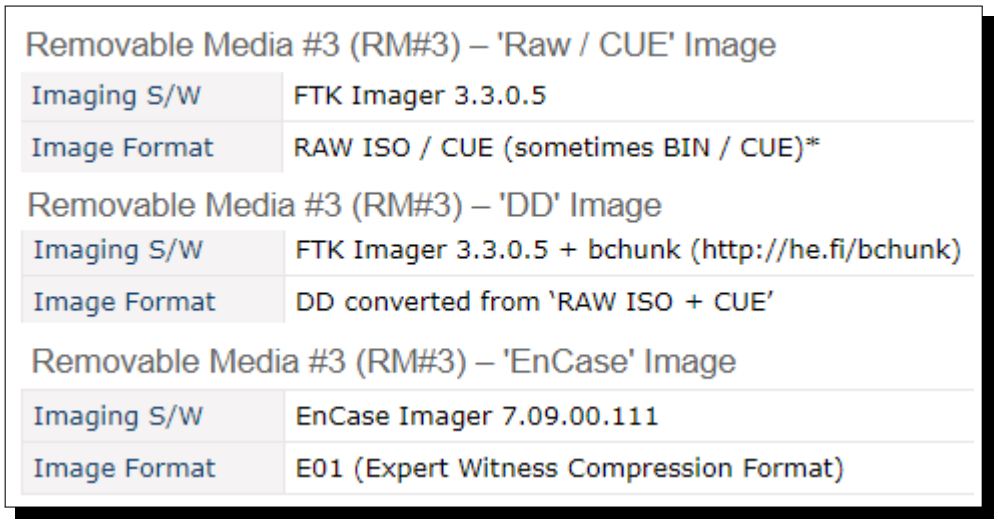
Imaging S/W	FTK Imager 3.3.0.5 (write-blocked by Tableau USB Bridge T8-R2)
Image Format	DD

Removable Media #2 (RM#2) – 'EnCase' Image

Imaging S/W	EnCase Imager 7.09.00.111 (write-blocked by Tableau USB Bridge T8-R2)
Image Format	E01 (Expert Witness Compression Format)

Figure 2.5: Details of Removable Media #2 DD and Encase Image

Figure 2.6 demonstrates the ways in which Removable Media #3 which in this case is the compact disk (CD) that was confiscated from the manager was imaged. Three types of imaging toolsets were used. The primary reason behind the use of multiple methods is to demonstrate the versatility of forensic imaging software available for use.



Removable Media #3 (RM#3) – 'Raw / CUE' Image	
Imaging S/W	FTK Imager 3.3.0.5
Image Format	RAW ISO / CUE (sometimes BIN / CUE)*
Removable Media #3 (RM#3) – 'DD' Image	
Imaging S/W	FTK Imager 3.3.0.5 + bchunk (http://he.fi/bchunk)
Image Format	DD converted from 'RAW ISO + CUE'
Removable Media #3 (RM#3) – 'EnCase' Image	
Imaging S/W	EnCase Imager 7.09.00.111
Image Format	E01 (Expert Witness Compression Format)

Figure 2.6: Details of Removable Media #3 DD and Encase Image

As the third device that was captured was a compact disc, a raw ISO was generated out of it, using FTK Imager. ISO's comprise of data from every sector of the optical disc including its file system. On the other hand a cue/bin file was also generated. This is a binary copy of the entire optical disc. The difference between an ISO and a BIN/CUE file format lies in its size. ISO's are usually 700 MB in size whereas the latter can go upto 800 MB. The reason behind this is along with the files and folders within the disc, it also contains volume attributes, bootable information and any other relevant system specific information. This format is an exact replica of the raw data which is stored sector by sector within a disc. Additionally a CUE file is obtained, which contains metadata about the the disc and tracks in a plain text format. A .dd image was then generated by using the raw iso/cue file(s) as source with the help of FTK Imager along with a tool called bchunk [10]. The latter converts a compact disc image into a set of .iso tracks. bchunck compiles and runs on any platform that integrates an ANSI C compiler. Finally an .E01 file was also generated using EnCase Imager.

Chapter 3

The Approach

The approach taken to solve a forensic investigation is an integral component of a criminal investigation and often affects other stages within the investigation. Approaches vary according to the type of data that needs to be tackled and looked into. For this specific case, the priority will revolve around searching for the data that is suspected of being leaked and then finding the source and destination for the same. Multiple attempts have been theorized to propose a standardized methodology for an investigation, but the data that is being captured has been becoming more and more dynamic in nature. With this, having a standard approach is not feasible and rapid improvisation techniques might require adoption on the fly. Some models can be adopted for certain types of analysis but a single model cannot be applied on a broader category. There needs to be a certain degree of flexibility when it comes to solving digital forensics issues in the modern era.

Seizure, acquisition, analysis and reporting are the four broad categories that every investigation goes through and the two subsections that vary drastically depending upon the type of the data source are acquisition and analysis. Some very popular process models for digital forensic investigation are FORZA - Digital forensics investigation framework, A Hierarchical, Objectives-Based Framework for the Digital Investigations Process, The Advanced Data Acquisition Model (ADAM): A process model for digital forensic practice, The Systematic Digital Forensic Investigation Model (SRDFIM) [11].

3.1 Adopted Approach

The approach that is adopted to conduct investigation upon the associated data set is listed below. It spans through the steps of Identification, Preservation, Collection, Examination & Analysis and culminating in an Inference & Conclusion as can be seen in Figure 3.1 below. The three initial sub-parts have been already discussed in detail in Section 1 and Section 2 while the remainder of the parts will be discussed in the sections to follow. An outline of what each step in this process flow entails is the following.

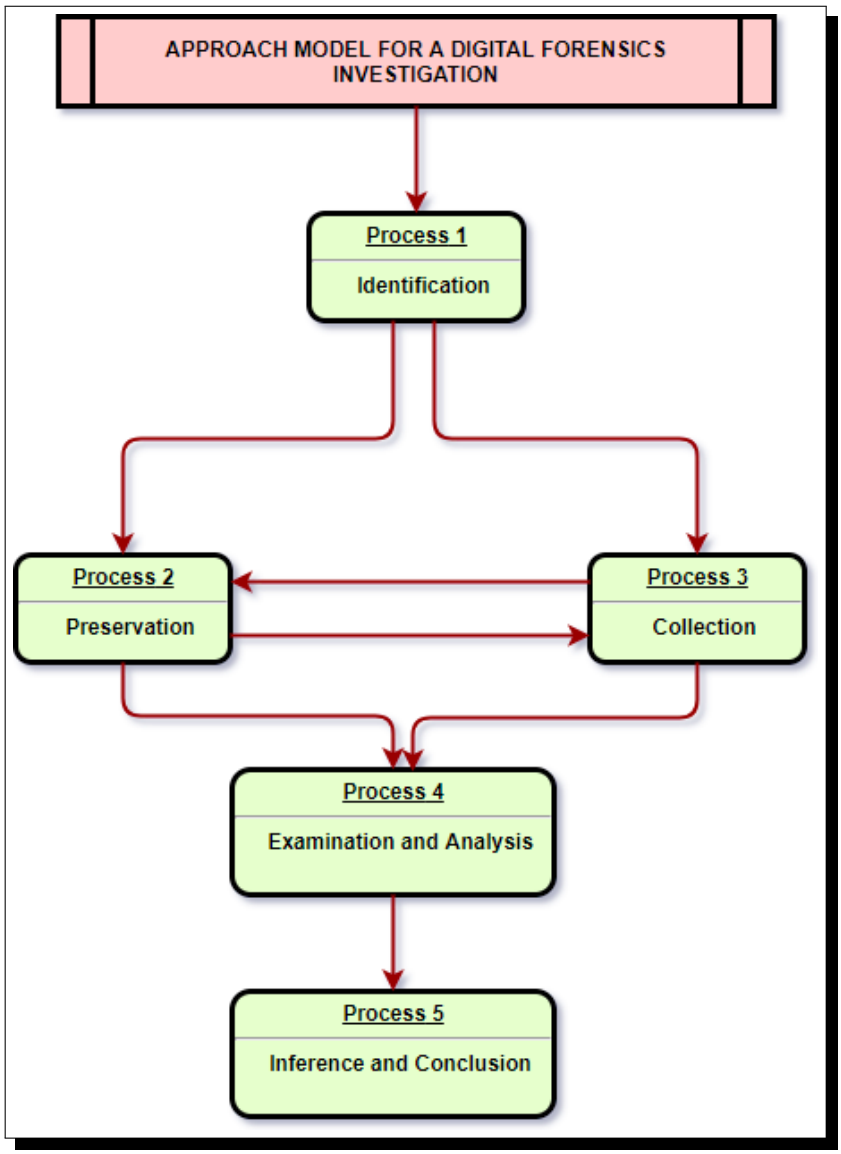


Figure 3.1: Flow of a Digital Forensics Investigation

3.1.1 Identification

The identification phase goes on in parallel with the verification phase. A forensic investigation is usually conducted as and when an incident is reported and needs to be investigated. The first step in such a scenario is to verify whether the incident that is being reported has indeed occurred or whether it is a false positive. The scope and extent of the issue should then be assessed which helps in finding the manpower with the type of skill set required to conduct a thorough and successful investigation. If the scenario is such where a business organization system is being affected, the preliminary approach would be to take the system offline, which means cutting off any internet/intranet access to the system, while keeping the system powered on, in case a live image of the system needs to be acquired and shutting off the system might kill infected processes which can lead to the source of the attacker. This first step in the process flow helps in determining the nature of the threat and the characteristics of the data that needs to be worked upon, which leads to the identification of the best possible approach that should be taken in an investigation.

3.1.2 Preservation

This section is extremely critical even though it is before the start of actual examination of data. If there are any inconsistencies in this phase, it might jeopardize the entire fate of the investigation. Data integrity is of utmost importance to the investigator and critical attention must be paid to avoid any external interference with the system that is the victim of an attack. The evidence is securely stored and packaged which avoids any issues that are related to handling and transportation of it. A case management workflow should be setup along with ensuring that proper custody of the evidence is obtained and also synchronization of time is conducted while capturing the data.

3.1.3 Collection

The collection phase involves the extraction of data from the devices confiscated by the use of approved and authorized software & hardware. The primary motive of this phase is lossless collection of extracted data. Describing the systems under investigation is also part of this phase. Outlining where the system is located within the network of the organization, file system format and type of the hard disk drives,

amount of RAM, where the system was acquired from and also the users role within the organization. Most of the low level data is addressed in this phase although recovery of deleted files, reverse engineering, and file carving could be applied within this phase. Both volatile and non-volatile acquisition must be pursued at this point, although both data dumps might not be used during the investigation after the following phase of examination is conducted. As the nature of volatile data is very different from that of its non-volatile counterpart, there should be a specific plan of prioritization in this phase, where in volatile data should be captured at the very beginning. Artifacts such as running processes in the RAM, user logins and session information, files that are currently in open state and being used by certain processes, networks that the machine is connected to, logging information. Tools that don't alter these important information should be used in accordance to the investigative agency's policies.

Once the volatile data is aggregated, the non-volatile data such as hard disk drives, flash drives can be captured next which comprise of non-volatile data and which don't change after a system goes through power cycles. A write blocker should always be used while capturing non-volatile data as making changes to the source of the data is highly undesirable as it may have a direct impact on the examination and analysis of the evidence. It is also a great idea to make multiple copies of this data and store the data in multiple formats as described in Section 2. This ensures the secure collection of the data and negates the presence of any inconsistencies that might arise after the investigation has started. MD5 [12] and SHA-1(Secure Hash Algorithm 1) [13] checksum(s) of the captured data should also be recorded along with proper documentation of tools and methods used to make copies of the data. This further rules out any chances of disparity of the source and the copy of the data that is under investigation.

3.1.4 Examination & Analysis

The first step in the Examination phase is to confirm that the working data set is not contaminated and this can be confirmed by using the hashing functions and making sure the values match to those of the original data. The analysis phase follows after the examination phase and this is where the technical details of the operation are conducted. A reconstruction of the crime scene is setup first in certain

cases, where the investigator has to setup a similar environment which mirrors that of the crime scene. Once this is complete, the next step is the most vital part of the investigation which entails file carving of both system and personal files, deleted data, email analysis forensics, data mining, database reconstruction, operating system level forensics(Windows in this case), web browser forensics and a user behavior inspection and study. The following figure(3.2) gives a better understanding of the areas of digital forensics investigation that will be explored during the examination and analysis phases of this project.

Understanding Types of Data Leakage	- Storage devices > HDD (Hard DiskDrive), SSD (Solid State Drive) > USB flash drive, Flash memory cards > CD/DVD (with Optical Disk Drive) - Network Transmission > File sharing, Remote Desktop Connection > E-mail, SNS (Social Network Service) > Cloud services, Messenger
Windows Forensics	- Windows event logs - Opened files and directories - Application (executable) usage history - CD/DVD burning records - External devices attached to PC - Network drive connection traces - System Caches - Windows Search databases - Volume Shadow Copy
File System Forensics	- FAT, NTFS, UDF - Metadata (NTFS MFT, FAT Directory entry) - Timestamps - Transaction logs (NTFS)
Web Browser Forensics	- History, Cache, Cookie - Internet usage history (URLs, Search Keywords...)
E-mail Forensics	- MS Outlook file examination - E-mails and attachments
Database Forensics	- MS Extensible Storage Engine (ESE) Database - SQLite Database
Deleted Data Recovery	- Metadata based recovery - Signature & Content based recovery (aka Carving) - Recycle Bin of Windows - Unused area examination
User Behavior Analysis	- Constructing a forensic timeline of events - Visualizing the timeline

Figure 3.2: Digital Forensics Investigation Areas Explored during the Analysis Phase

3.1.5 Inference & Conclusion

This is the final phase of the investigation where evidence that is carved out in the penultimate phases of examination and analysis are presented in front of the jury in charge of a specific digital forensic investigation case. Various ways in which data was recovered is explained in this step and recommendations are proposed in improvement of security policies and guidelines of an organization which can curb attacks of a similar nature in the future. Along with the case in hand, it is the investigator's responsibility to submit any additional drawbacks that are discovered in an organization's systems, for e.g. loopholes within policies, backdoors within the network etc. During the final juncture of this phase, the focus is on reflection and betterment of the methodologies adopted during a particular digital forensic investigation.

This chapter explained the various processes adopted during a typical forensic investigation and the approach model that is followed when dealing with a data set of this nature.

Chapter 4

Examination and Analysis

This is the penultimate and arguably the most critical phase of a digital forensic investigation upon which the success or failure of an operation rests. In this phase, this report will outline all possible actions taken to conduct investigation upon the data set in question, the tools that were used in uncovering encrypted and hidden data, how snapshot databases were recovered from browser history, tools used to conduct email forensics and recreate exchange of communication via emails with a proper timeline.

To lend flexibility to the reader, both Linux and Windows environments were used to conduct experiments. The OS of choice for Windows was Windows 10 and for Linux, the Kali distribution was used. The latter was used in most cases as it is a distribution which has been designed with a specific aim of acting as an aid to digital forensics testing [14]. It has over 600 pre-installed tools related to wireless/wired attacks, web application attacks, information gathering, vulnerability analysis, access control and password hacking, reporting tools, reverse engineering, hardware hacking, spoofing and sniffing, stress testing, system services, social engineering, exploitation tools and forensic tools [15]. A couple of tools that this report will outline are Autopsy and Foremost, which fall under the umbrella of forensics tools within Kali. Some of the categories and tools available in a specific category is listed in Figure 4.1 below.

Figure 4.1: Categories of Digital Forensics & Penetration Testing tools in Kali

4.1 Verification of Integrity of Data Set

Before starting any forensic investigation it is very important to verify the integrity of the data, in order to ensure that the data hasn't been tampered with. When the data was captured in the first phase of the investigation, an SHA1 sum was generated for each part of the data. The SHA1 values of the available data must be re-generated and checked against that of the original data and if there are no indications of any change in values, the investigation can further proceed.

This can be achieved by using the command *sha1sum* followed by the name of

the file which needs an SHA1 sum to be generated. So if the name of the file is filename.db, the command would be *sha1sum filename.db*. This will generated the SHA1 sum in the following line of the terminal.

Figure 4.2: Matching SHA1 sum(s) of part 1 of the zipped .DD image from suspected system

Re-calculated SHA1 sums of all 12 files can be seen in Figures 4.2 to 4.13 and the figures will also depict that the re-generated values match the original SHA1 sums.

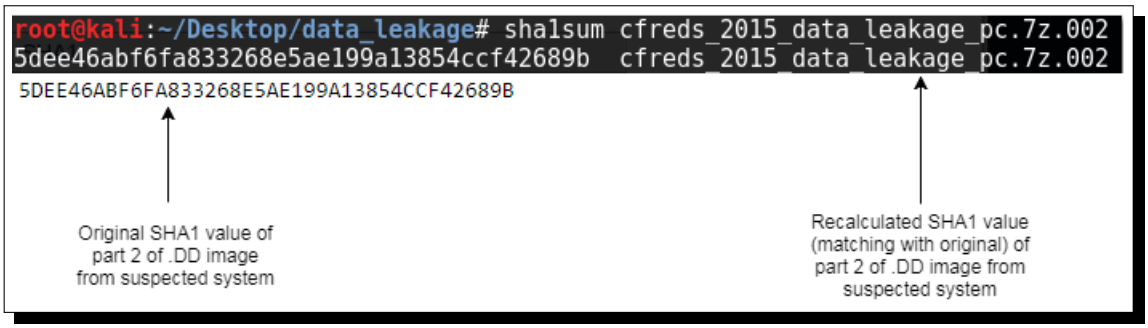


Figure 4.3: Matching SHA1 sum(s) of part 2 of the zipped .DD image from suspected system

Once the data for the compressed parts of the .dd image of the primary suspect system completed successfully, the next choice was the .E01 to .E04 images generated by the use of Encase Imager, and verify its consistency.

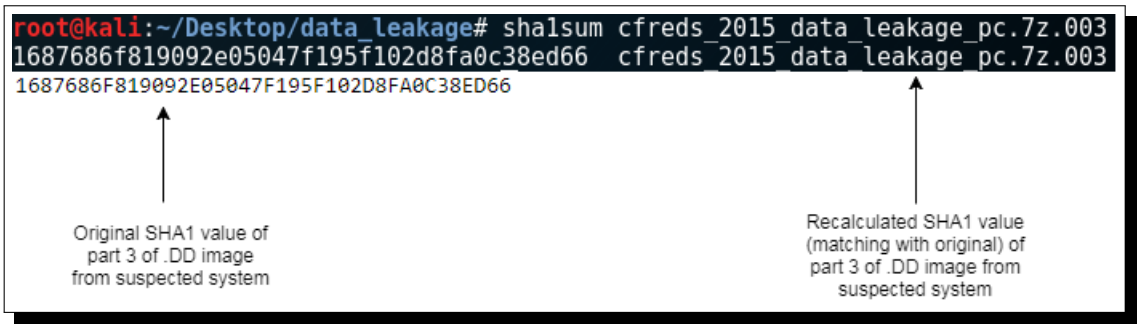


Figure 4.4: Matching SHA1 sum(s) of part 3 of the zipped .DD image from suspected system

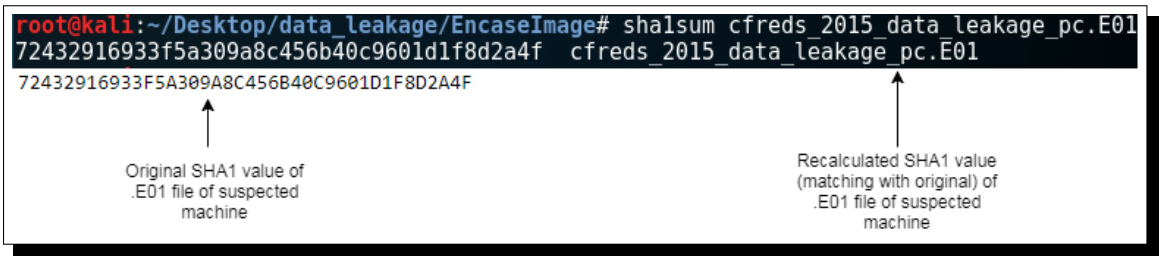


Figure 4.5: Matching SHA1 sum(s) of .E01 image from suspected system

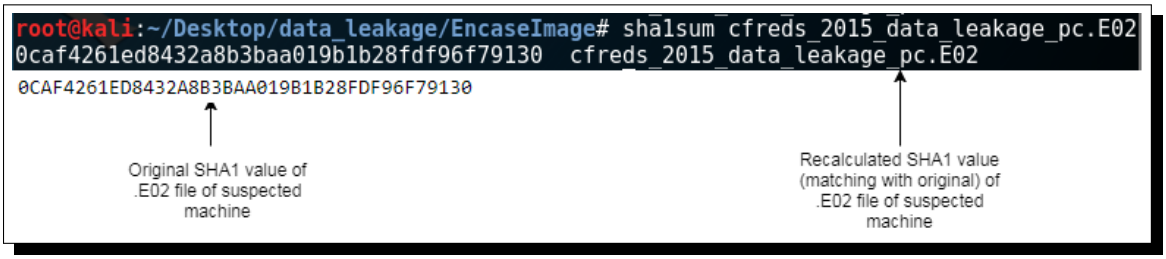


Figure 4.6: Matching SHA1 sum(s) of .E02 image from suspected system

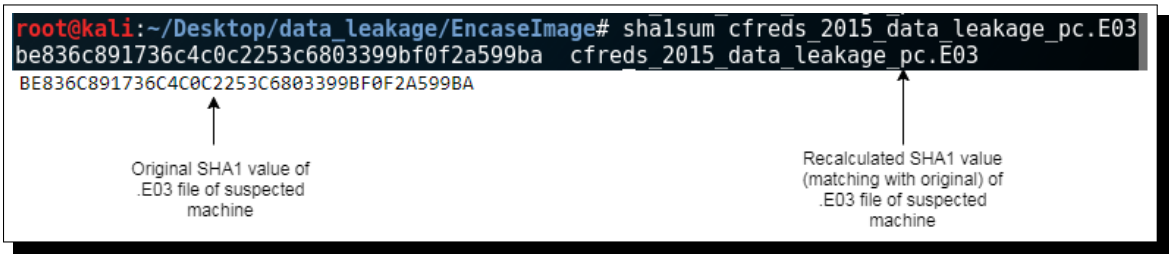


Figure 4.7: Matching SHA1 sum(s) of .E03 image from suspected system

Once all the SHA1 sums of the images obtained via Encase Imager is obtained, the next step was to verify the integrity of the images captured for the two sets of

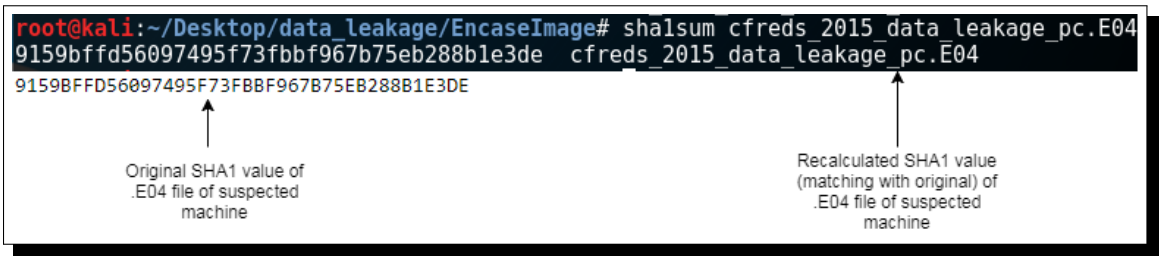


Figure 4.8: Matching SHA1 sum(s) of .E04 image from suspected system

removable media, namely the USB flash drive and the compact disc. The re-calculated checksums of the same follow:

Figure 4.9: Recalculation of checksum from .DD image of flash drive

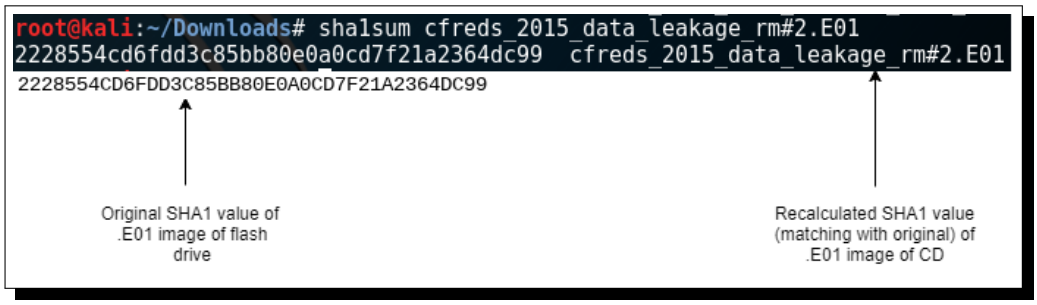


Figure 4.10: Recalculation of checksum from .E01 image of flash drive

One thing that can be inferred at this point of the project is that the images obtained have not been altered in any way since their capture and that they are ready for undergoing investigation. Imaging options provided by Forensics Tools like FTK Imager, Encase Imager and imaging while maintaining integrity of data provided by write blockers such as Tableau T8-R2 are quite reliable in their functionality.

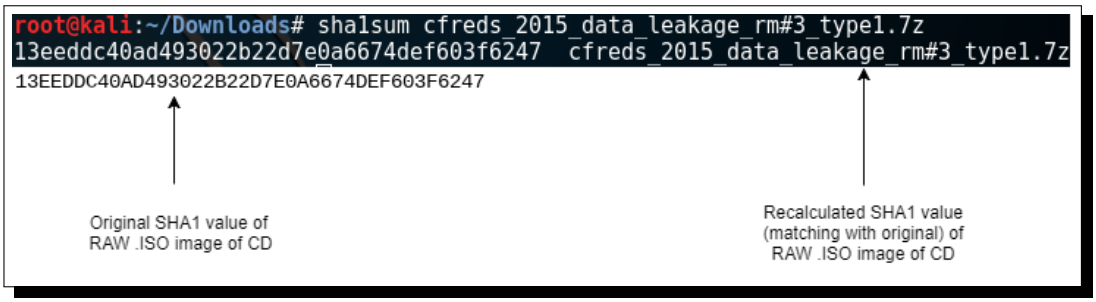


Figure 4.11: Recalculation of checksum from .ISO image of disc

Figure 4.12: Recalculation of checksum from .DD image of disc

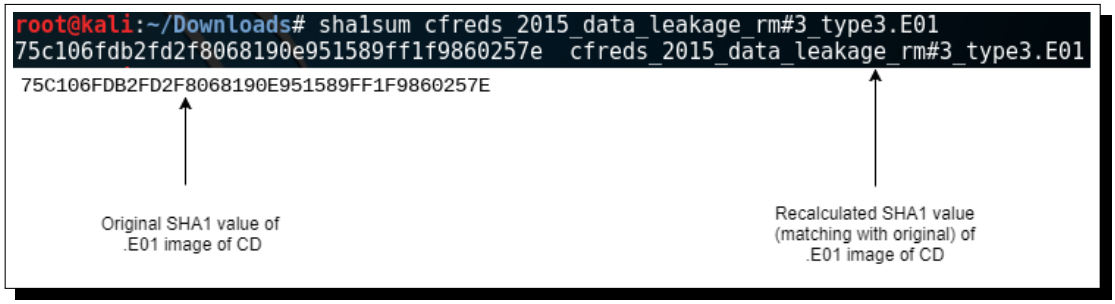


Figure 4.13: Recalculation of checksum from .E01 image of disc

4.2 Analysis of Suspected System

After successful completion of the checksum verification stage, the image from the primary machine was inspected. The Sleuth Kit and Autopsy [16] toolkit was used in this phase. The toolkit was used both in Windows and Linux environments for a variety of reasons ranging from metadata collection on the former and raw data extraction from the latter. The first thing to verify is the operating system that is being dealt with as that provides information about its file structure, location of installation directories, system paths, temp application data location etc. When the .dd image of the system was loaded into Autopsy after creating a new case and adding

the file as an evidence to the case, the following details were uncovered.

4.2.1 Basic System Information

The system was running Windows 7 Ultimate Service Pack 1 and one of the user accounts on this computer was that of the suspected employee as can be seen in the Figure 4.14 below.

Source File	Name	Domain	Version	Processor Architecture	Temporary Files Directory	Data Source	Program Name	Date/Time	Path	Owner
SYSTEM	INFORMANT-PC		Windows_NT	AMD64	%SystemRoot%\TEMP	cfreds_2015_data_leakage_pc.dd				
SYSTEM	INFORMANT-PC		Windows_NT	AMD64	%SystemRoot%\TEMP	cfreds_2015_data_leakage_pc.dd				
SOFTWARE						cfreds_2015_data_leakage_pc.dd	Windows 7 Ultimate Service Pack 1	2015-03-22 14:34:26 PDT	C:\Windows	informant
SOFTWARE						cfreds_2015_data_leakage_pc.dd	Windows 7 Ultimate Service Pack 1	2015-03-22 14:34:26 PDT	C:\Windows	informant

Figure 4.14: Log-in information of suspect on the machine

Some of the other user accounts that were associated to the same machine are listed in the Figure 4.15 below. This list was generated as a result of one of the following users logging into this machine as these are all organizational accounts and can be used to log into any system that is configured to be a part of the same workspace. The account which is suspected of the data leak is highlighted in the Figure(4.15) below. The reason behind there being two occurrences of the account is due to the login using that account at different times.

Source File	Username	User ID
SOFTWARE	systemprofile	S-1-5-18
SOFTWARE	LocalService	S-1-5-19
SOFTWARE	NetworkService	S-1-5-20
SOFTWARE	informant	S-1-5-21-2425377081-3129163575-2985601102-1000
SOFTWARE	admin11	S-1-5-21-2425377081-3129163575-2985601102-1001
SOFTWARE	temporary	S-1-5-21-2425377081-3129163575-2985601102-1003
SOFTWARE	systemprofile	S-1-5-18
SOFTWARE	LocalService	S-1-5-19
SOFTWARE	NetworkService	S-1-5-20
SOFTWARE	informant	S-1-5-21-2425377081-3129163575-2985601102-1000
SOFTWARE	admin11	S-1-5-21-2425377081-3129163575-2985601102-1001
SOFTWARE	temporary	S-1-5-21-2425377081-3129163575-2985601102-1003

Figure 4.15: Associated User-ID[belonging to a user group] of the suspect

4.2.2 SAM Hive Exploration

Once this list was populated, the SAM and SYSTEM files located in the Windows registry filesystem were extracted via Autopsy and copied over to Kali. SAM is an acronym for Security Account Manager [17]. When a user logs into a Windows system, an encrypted hash of the password is generated and stored within this file and when the user re-logins, the password is checked against this hash for verification instead of using plain text format. In order to reverse engineer the password of the user accounts within the system, the NT4-hashes were obtained by using the command as displayed in the Figure 4.16 below.

```
root@kali:~/Desktop# samdump2 -o samsystemnthashes.txt SYSTEM SAM
```

Figure 4.16: Dumping of NT-4 hashes

Once the newly generated file is looked into, the NT4 hashes for the various user accounts within the system can be located. The primary objective at this point is to obtain the password for the user account under suspicion.

```
root@kali:~/Desktop# cat samsystemnthashes.txt
*disabled* Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59
d7e0c089c0:::
*disabled* Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089
c0:::
informant:1000:aad3b435b51404eeaad3b435b51404ee:9e3d31b073e60bfd7b07978d6f914d0a:::
admin11:1001:aad3b435b51404eeaad3b435b51404ee:21759544b2d7efccc978449463cf7e63:::
ITechTeam:1002:aad3b435b51404eeaad3b435b51404ee:75ed0cb7676889ab43764a3b7d3e6943:::
temporary:1003:aad3b435b51404eeaad3b435b51404ee:1b3801b608a6be89d21fd3c5729d30bf:::
```

Figure 4.17: View of NT-4 hashes obtained from the system

Once the hashes had been obtained, they had to be reverse engineered to obtain plain text passwords. A lot of tools are available for this purpose, but "John the Ripper" [18], an open source utility used for testing and re-calculation of passwords was used. It combines a number of password cracking plugins into one package and auto detects password hash types to decrypt them. The command line variant of the tool can be seen in the following Figure 4.18. Once it was started , it creates combinations on the fly and tests them against a password hash. This can take an unknown length of time and can span across months at times, depending upon the complexity and length of the password. The next feasible option was to run the hash set against a collection of generated passwords from a well maintained dictionary. In

the latter part of the Figure 4.18, it can be seen that not only was the format of the generated hashes specified, but also an updated and further modified dictionary was used and upon checking the time it would take to decrypt the hash, it was indicated to be approximately 14 hours. Although the final time it took was approximately 27 hours.

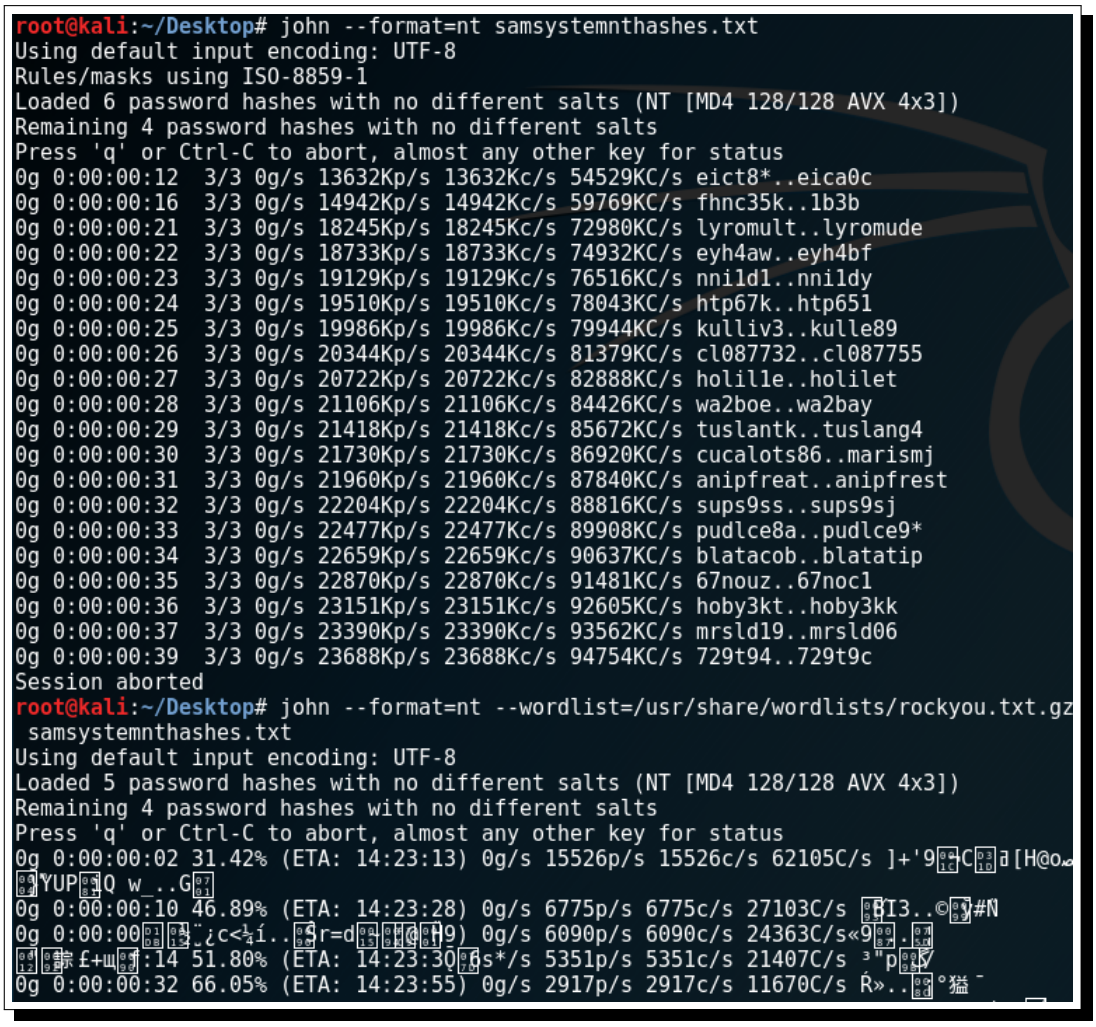


Figure 4.18: Starting JohnTheRipper

After running for quite a while as indicated earlier, the password of the suspected user was finally revealed as can be seen in the Figure 4.19 below. This password can now be used if there is any further investigation required to be conducted on a live version of the system as part of future work. A live version of the system might be able to be produced by converting the dd image to a vmdk and trying to boot it up. Although the volatile memory in this instance would be lost but this is an alternate

approach that can be adopted to conduct further analysis. Another way of achieving this in a lesser amount of time is the use of rainbow tables[19]. They are built out of a chaining mechanism. Passwords are decrypted by using their hash value, like above, but each link in a chain is compared with the last value of the chain and a match is obtained. The chain undergoes re-building, while preserving the values of the reduction and hashing function, which can be utilized to reverse engineer a system's password. A more in-depth look into their use can be part of future work on this project.

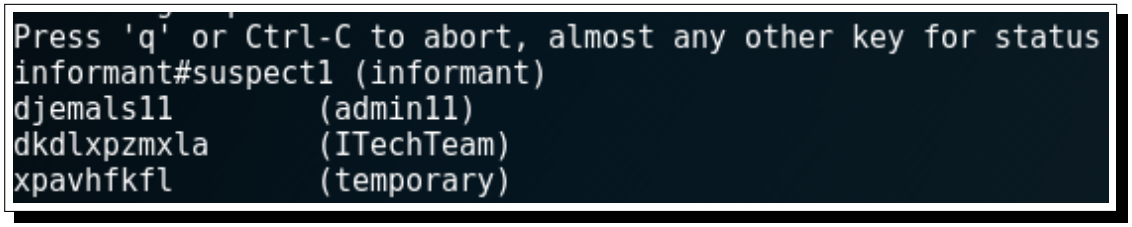


Figure 4.19: Uncovering of passwords from NT-4 hashes

Another important evidence that is crucial in an investigation of this nature is to find out the users who were logged in the final session on a suspected system. To gather this information, another open source utility called RegRipper [20] was used. Login information is usually located within the SAM file of a system and a plugin called samparse [21] could be used in conjuction with RegRipper to find out all users who were logged into the system. Although users can login in remotely to a system, their login count in the findings is not calculated. Only users who physically logged into the system and not via a network or remote access is another consideration that was made. Regripper is a tool developed in Perl which is used is parsing keys from a Windows registry hive.

On the other hand samparse is one of the plugins which decrypts user/group belonging information from a SAM hive. The usage and output of RegRipper is depicted in the Figure 4.20 above and also a code snippet from samparse, which depicts the section, where it configures the plugin to extract information from the hive source as can be seen in Figure 4.21.

After the tool runs successfully it generates output in plain text format and after post processing of that data, the output can be seen in the Figure 4.22 below which depicts the 3 major accounts that had a login count of more than 0, which means that the system was accessed physically on location and not via a remote client. Out of the 3 user accounts, the one that is under suspicion was the one that was last used

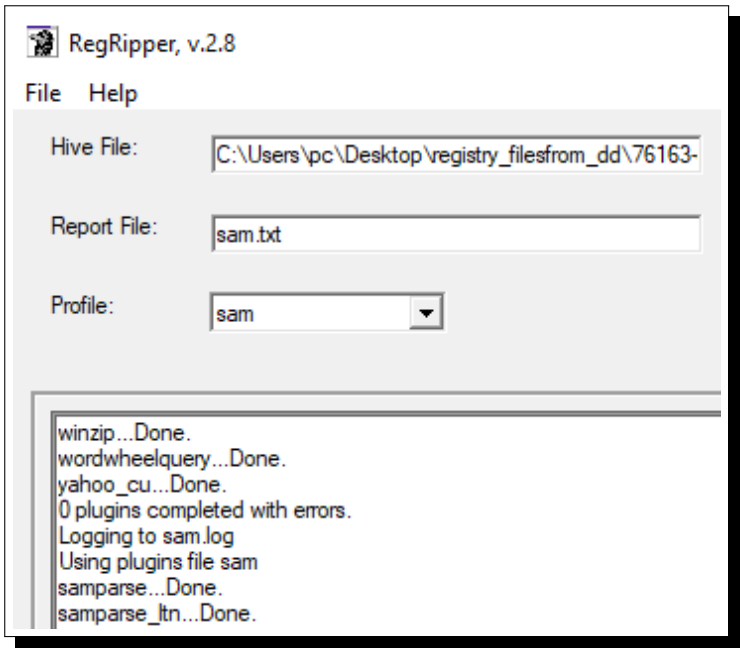


Figure 4.20: RegRipper running on SAM

to log in to the machine which validates the fact that there were no changes made by any other user account to this system. Any changes made to the system were by the last logged on user and any changes to the file system can now be tracked according to the time that they were modified as the last login time has now be found out.

Another consideration that is of vital importance is the verification of the final time that the computer was shutdown. In order to find this information, RegRipper was used again on the SAM file and the output can be seen in the Figure 4.23 below which depicts that the final shutdown was done by a member of the Administrators group at and the suspected user is highlighted in the same. When this data is compared to the Figure 4.22 shown above in terms of timeline, it can be inferred that the suspected user was the last user to have accessed the system.

```
my %config = (hive          => "SAM",
              hivemask      => 2,
              output        => "report",
              type          => "Reg",
              output        => "report",
              category      => "",
              osmask        => 63, #XP - Win8
              hasShortDescr => 1,
              hasDescr      => 0,
              hasRefs       => 1,
              version       => 20120925);

sub getConfig{return \%config}

sub getShortDescr {
  return "Parse SAM file for user & group mbrshp info";
}
sub getDescr{}
sub getRefs {
  my %refs = ("Well-known SIDs" => "http://support.microsoft.com/kb/243330");
  return %refs;
}
sub getHive {return $config{hive};}
sub getVersion {return $config{version};}

my $VERSION = getVersion();

my %acb_flags = (0x0001 => "Account Disabled",
                 0x0002 => "Home directory required",
                 0x0004 => "Password not required",
                 0x0008 => "Temporary duplicate account",
                 0x0010 => "Normal user account",
                 0x0020 => "MNS logon user account",
                 0x0040 => "Interdomain trust account",
                 0x0080 => "Workstation trust account",
                 0x0100 => "Server trust account",
                 0x0200 => "Password does not expire",
                 0x0400 => "Account auto locked");

my %types = (0xbc => "Default Admin User",
             0xd4 => "Custom Limited Acct",
             0xb0 => "Default Guest Acct");
```

Figure 4.21: Configuration of samparse to extract user information from SAM hive and formatting of raw data

4.2.3 List of suspicious applications installed on the system

A list of installed applications on this system was also recovered and after analyzing the list, three applications were found to be suspicious in nature, that resided on the system. They are listed in the Figure 4.24 below. Two out of the three programs

```
Username        : informant [1000]
Full Name       :
User Comment    :
Account Type    : Default Admin User
Account Created : Sun Mar 22 14:33:54 2015 Z
Name            :
Password Hint   : IAMAN
Last Login Date : Wed Mar 25 14:45:59 2015 Z
Pwd Reset Date  : Sun Mar 22 14:33:54 2015 Z
Pwd Fail Date   : Wed Mar 25 14:45:43 2015 Z
Login Count     : 10

Username        : admin11 [1001]
Full Name       : admin11
User Comment    :
Account Type    : Default Admin User
Account Created : Sun Mar 22 15:51:54 2015 Z
Name            :
Last Login Date : Sun Mar 22 15:57:02 2015 Z
Pwd Reset Date  : Sun Mar 22 15:52:10 2015 Z
Pwd Fail Date   : Sun Mar 22 15:53:02 2015 Z
Login Count     : 2

Username        : temporary [1003]
Full Name       : temporary
User Comment    :
Account Type    : Custom Limited Acct
Account Created : Sun Mar 22 15:53:01 2015 Z
Name            :
Last Login Date : Sun Mar 22 15:55:57 2015 Z
Pwd Reset Date  : Sun Mar 22 15:53:11 2015 Z
Pwd Fail Date   : Sun Mar 22 15:56:37 2015 Z
Login Count     : 1
```

Figure 4.22: In-detail login information of suspect via RegRipper

```
Group Name    : Administrators [4]
LastWrite     : Sun Mar 22 15:52:30 2015 Z
Group Comment : Administrators have complete and u
Users :
  S-1-5-21-2425377081-3129163575-2985601102-1001
  S-1-5-21-2425377081-3129163575-2985601102-1000
  S-1-5-21-2425377081-3129163575-2985601102-500
  S-1-5-21-2425377081-3129163575-2985601102-1002
```

```
Profile: sam

Logging to sam.log
Using plugins file sam
samparse...Done.
samparse_ltn...Done.
1 plugins completed with errors.
```

Figure 4.23: Final user to login to the suspected system

were related to cloud storage and they were Google Drive [22] and iCloud [23]. These might have been used to upload sensitive and confidential files to the cloud which could then be accessed from any corner of the world. The same could have also been used to download malicious files within the system which could lead to the spread of a virus within the organization's systems, although the latter is not the case in hand here.

SOFTWARE	iCloud v.4.0.6.28	2015-03-23 20:01:54 PDT	cfreds_2015_data_leakage_pc.dd
SOFTWARE	Google Drive v.1.20.8672.3137	2015-03-23 20:02:46 PDT	cfreds_2015_data_leakage_pc.dd
SOFTWARE	Eraser 6.2.0.2962 v.6.2.2962	2015-03-25 14:57:31 PDT	cfreds_2015_data_leakage_pc.dd

Figure 4.24: Preliminary list of suspicious applications installed on the system

The final suspicious application, at this point of the analysis is called Eraser [24]. Although it has a generic name, but after looking for the specific version found in Autopsy it was discovered that it is an advanced security application that helps in elimination of data from a hard drive by conducting multiple writes with varying patterns which can be fetched from a data source as can be seen in the Figure 4.25 below, in order to avoid carving and reverse engineering. According to the information obtained from the official website of Eraser, it seems to have addressed the issue of files remaining in the system even after deletion due to the use of data encoding and write cache. This could prove the point that the suspected employee was indeed trying to upload files in the cloud and then remove any sources of that file from the system, in order to avoid detection. It should be noted here that RegRipper used in the previous subsection can also be used in this case to parse Windows registry to obtain the software hive and discover all installed applications.

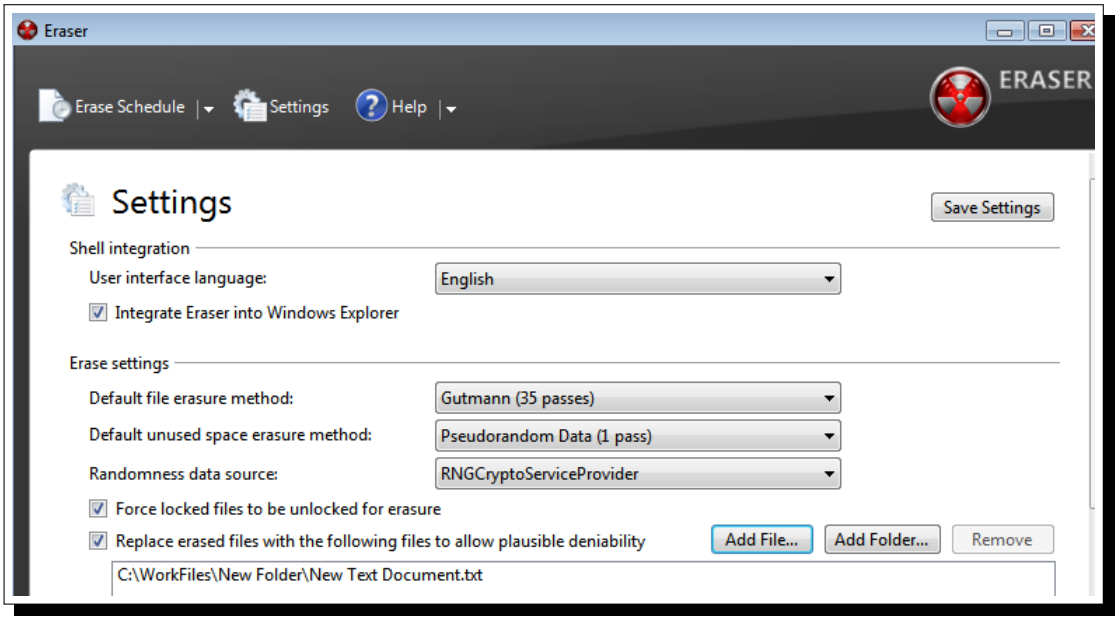


Figure 4.25: Eraser UI

Additionally, a suspicious tool that was installed and is suspected to have aided in anti-forensic actions is CCleaner [25]. This was also gathered via the presence of the installer in the downloads and later via RegRipper about its installation on the system. This tool is suspected of overwriting a files content with random characters. The number of times each file is overwritten can be altered according to the users' choice. The choice to perform a simple one pass data write over a file is also available.

Even at its strictest settings, it will leave some traces of a file in the pagefile of the system. Files residing in Volume Shadow Copies can help in determining which files have been overwritten. Files overwritten via this tool can be carved out by adopting certain techniques which are discussed in the following chapters. Presence of CCleaner was discovered via running RegRipper on the registry hive(NTUSER.DAT). The Figure 4.26 below depicts its UI and also the version installed on the system, which was later uninstalled by the suspect.

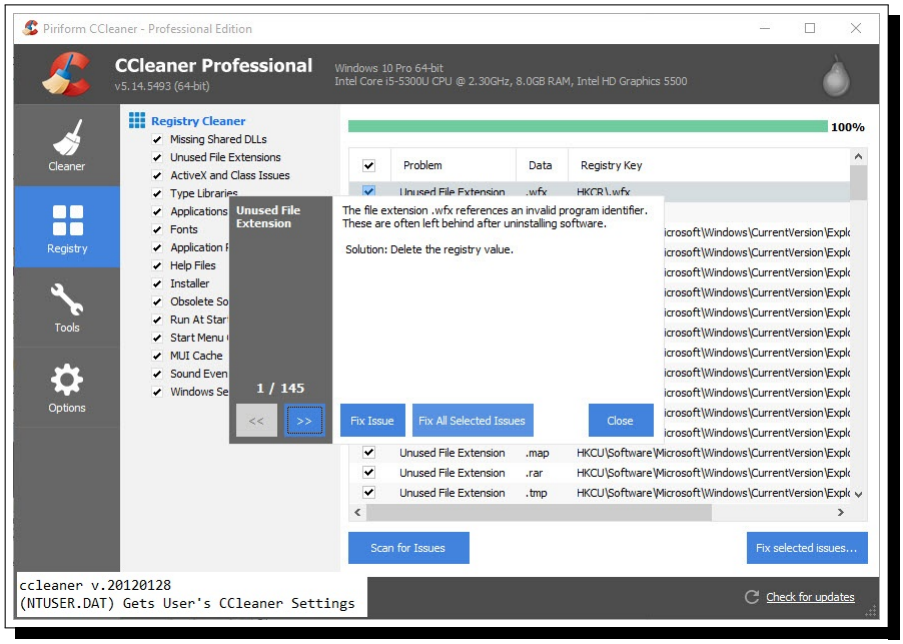
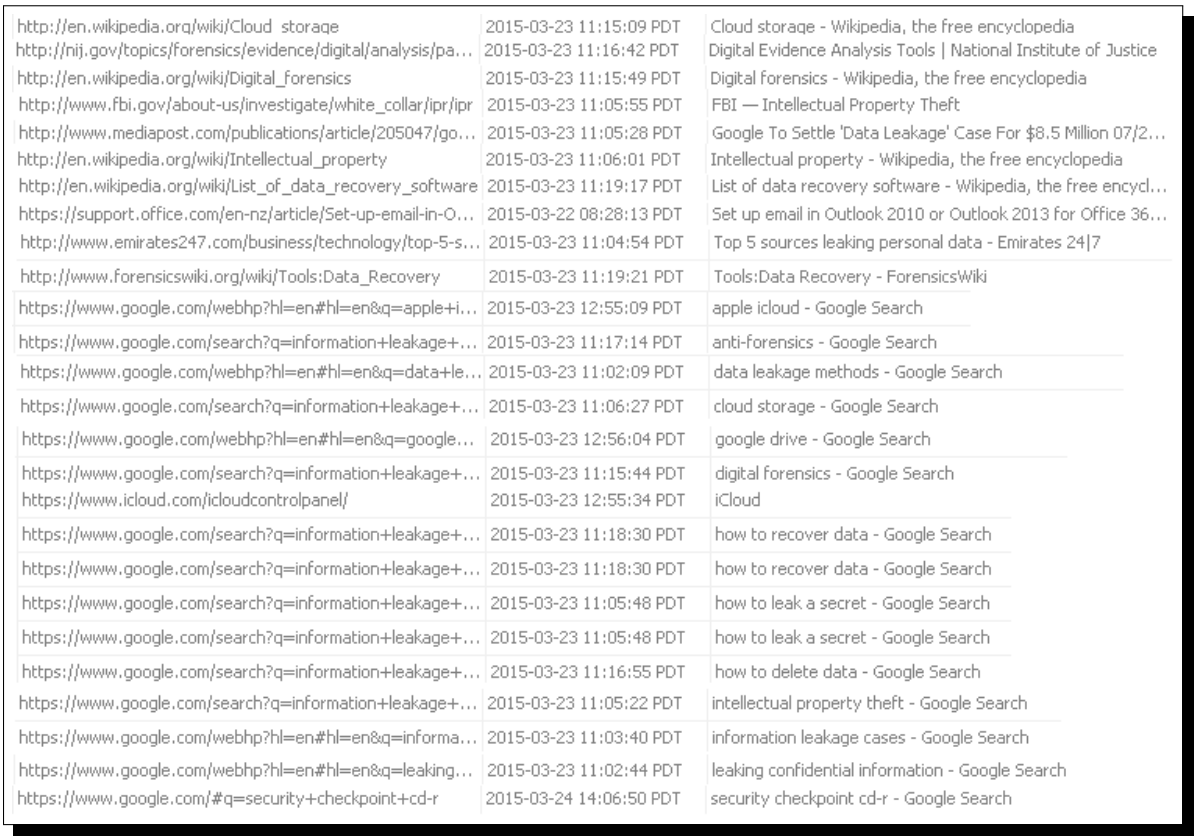


Figure 4.26: CCleaner UI and uninstall information from RegRipper output

4.2.4 Browser History Analysis

Browser history analysis is becoming an increasingly popular trend in the world of digital forensics. Although browsing history can be cleared and is quite a common practice on workstations, it is not a popular occurrence within mobile browsers. Although, even if the data is cleared from within the browser, there are remains of it that can be extracted and analyzed for forensic purposes. Autopsy was the first tool that was used to delve into the details of browser history of the suspected user. The details that were found is depicted in the Figure 4.27 below, by skimming out the search terms that are relevant to the scenario and not all search terms that were found out. It is evident that the user was looking for information about cloud storage platforms which led the search results to iCloud and Google Drive. The next popular terms were related to digital forensics, evidence analysis software, IP theft, data recovery software, ways to leak personal information, data leakage procedures, anti-forensics, leaking of confidential information, recovery process of deleted data, ways to permanently delete data, legal cases related to data leakage in the past and information about passing security checkpoints while having non-volatile storage and bypassing security. These terms could definitely reflect the users' interest in a data leakage case and ways to avoid detection. This is a definite trigger at this point in the investigation. Data like this supplements any concrete information that is uncovered, thereby strengthening the reliability of a forensic engineer's findings.

An alternate method would be to extract the information from the raw data format itself. Chrome stores its history in the the History file which is a SQLite database. The file was recovered and exported via Autopsy from the location `/img\_cfreds\_2015\_data\_leakage\_pc.dd/vol\_vol3/Users/informant/AppData/Local/Google/Chrome/User Data/Default/History`. Once the history file is loaded into a SQL client a simple SQL query as depicted in the figure below can be executed to gather the same information as depicted in Figure 4.28 below.



http://en.wikipedia.org/wiki/Cloud_storage	2015-03-23 11:15:09 PDT	Cloud storage - Wikipedia, the free encyclopedia
http://nij.gov/topics/forensics/evidence/digital/analysis/pa...	2015-03-23 11:16:42 PDT	Digital Evidence Analysis Tools | National Institute of Justice
http://en.wikipedia.org/wiki/Digital_forensics	2015-03-23 11:15:49 PDT	Digital forensics - Wikipedia, the free encyclopedia
http://www.fbi.gov/about-us/investigate/white_collar/ipr/ipr	2015-03-23 11:05:55 PDT	FBI — Intellectual Property Theft
http://www.mediapost.com/publications/article/205047/go...	2015-03-23 11:05:28 PDT	Google To Settle 'Data Leakage' Case For $8.5 Million 07/2...
http://en.wikipedia.org/wiki/Intellectual_property	2015-03-23 11:06:01 PDT	Intellectual property - Wikipedia, the free encyclopedia
http://en.wikipedia.org/wiki/List_of_data_recovery_software	2015-03-23 11:19:17 PDT	List of data recovery software - Wikipedia, the free encycl...
https://support.office.com/en-nz/article/Set-up-email-in-O...	2015-03-22 08:28:13 PDT	Set up email in Outlook 2010 or Outlook 2013 for Office 36...
http://www.emirates247.com/business/technology/top-5-s...	2015-03-23 11:04:54 PDT	Top 5 sources leaking personal data - Emirates 24|7
http://www.forensicswiki.org/wiki/Tools:Data_Recovery	2015-03-23 11:19:21 PDT	Tools:Data Recovery - ForensicsWiki
https://www.google.com/webhp?hl=en#hl=en&q=apple+i...	2015-03-23 12:55:09 PDT	apple icloud - Google Search
https://www.google.com/search?q=information+leakage+...	2015-03-23 11:17:14 PDT	anti-forensics - Google Search
https://www.google.com/webhp?hl=en#hl=en&q=data+le...	2015-03-23 11:02:09 PDT	data leakage methods - Google Search
https://www.google.com/search?q=information+leakage+...	2015-03-23 11:06:27 PDT	cloud storage - Google Search
https://www.google.com/webhp?hl=en#hl=en&q=google...	2015-03-23 12:56:04 PDT	google drive - Google Search
https://www.google.com/search?q=information+leakage+...	2015-03-23 11:15:44 PDT	digital forensics - Google Search
https://www.icloud.com/icloudcontrolpanel/	2015-03-23 12:55:34 PDT	iCloud
https://www.google.com/search?q=information+leakage+...	2015-03-23 11:18:30 PDT	how to recover data - Google Search
https://www.google.com/search?q=information+leakage+...	2015-03-23 11:18:30 PDT	how to recover data - Google Search
https://www.google.com/search?q=information+leakage+...	2015-03-23 11:05:48 PDT	how to leak a secret - Google Search
https://www.google.com/search?q=information+leakage+...	2015-03-23 11:05:48 PDT	how to leak a secret - Google Search
https://www.google.com/search?q=information+leakage+...	2015-03-23 11:16:55 PDT	how to delete data - Google Search
https://www.google.com/search?q=information+leakage+...	2015-03-23 11:05:22 PDT	intellectual property theft - Google Search
https://www.google.com/webhp?hl=en#hl=en&q=informa...	2015-03-23 11:03:40 PDT	information leakage cases - Google Search
https://www.google.com/webhp?hl=en#hl=en&q=leaking...	2015-03-23 11:02:44 PDT	leaking confidential information - Google Search
https://www.google.com/#q=security+checkpoint+cd-r	2015-03-24 14:06:50 PDT	security checkpoint cd-r - Google Search

Figure 4.27: Relevant browser history found via Autopsy

Another important feature of using a db client to access information within the history file is to find unique files that were downloaded via the browser. The SQL query used and the data found is shown in the Figure 4.29 below.

This re-confirms the fact that the above 2 cloud platforms as indicated in the findings were indeed downloaded and installed on the system. Another tool that can be used alongside the ones used above is called Hindsight [26]. It is an open source application that helps in analyzing browsers via reversing their caching mechanism. It can parse a users' download and browsing history, site specific preferences, saved passwords, auto fill recommendations, cookies (HTML5 and HTTP). It can also create a timeline of captured data and can be exported in a csv file or in the form of a report with GUI results. A screenshot of the application used in conjunction with the data set in use is depicted in the Figure 4.30 below. The command line usage is as following: `C:\tools\hindsight\hindsight.py -i "C:\Users\PC\Desktop\Temp\temp\Export\Google\9089-Google\Chrome\User Data\Default" -o source\_case\_data`

There are multiple other parameters which can be used in conjunction with the

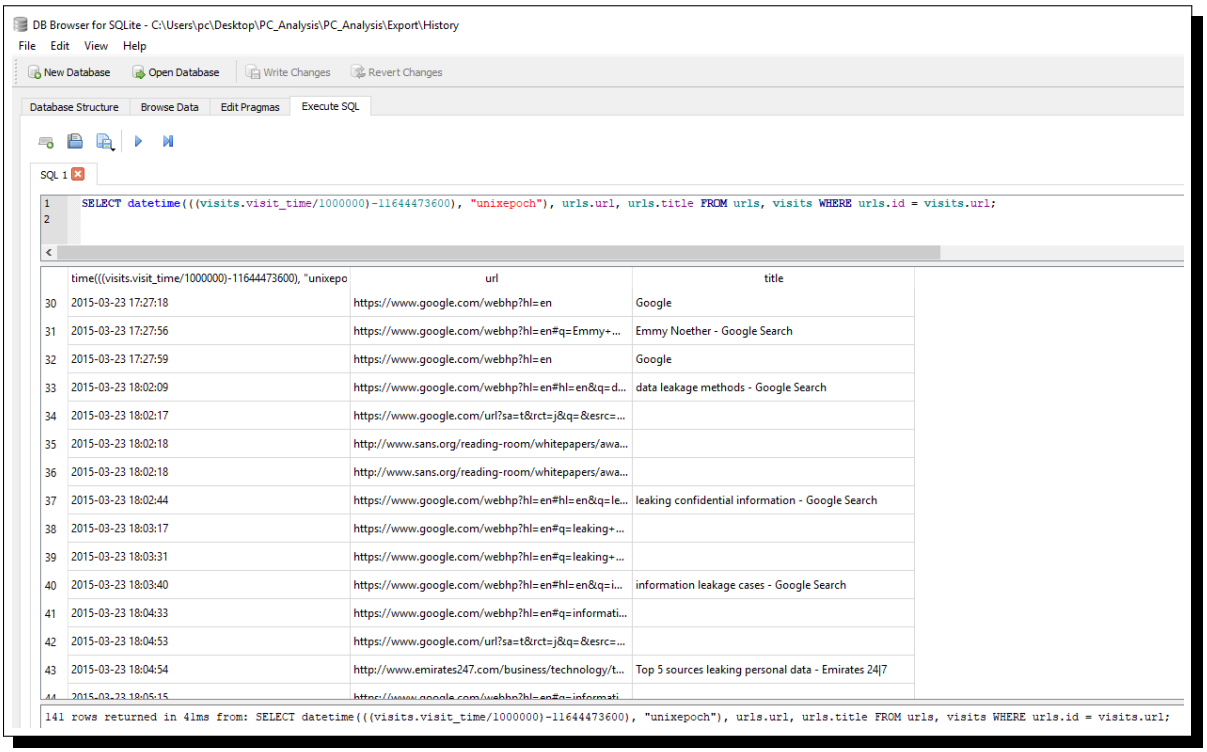


Figure 4.28: Relevant browser history found via DB Browser for SQlite on Chrome History DB I

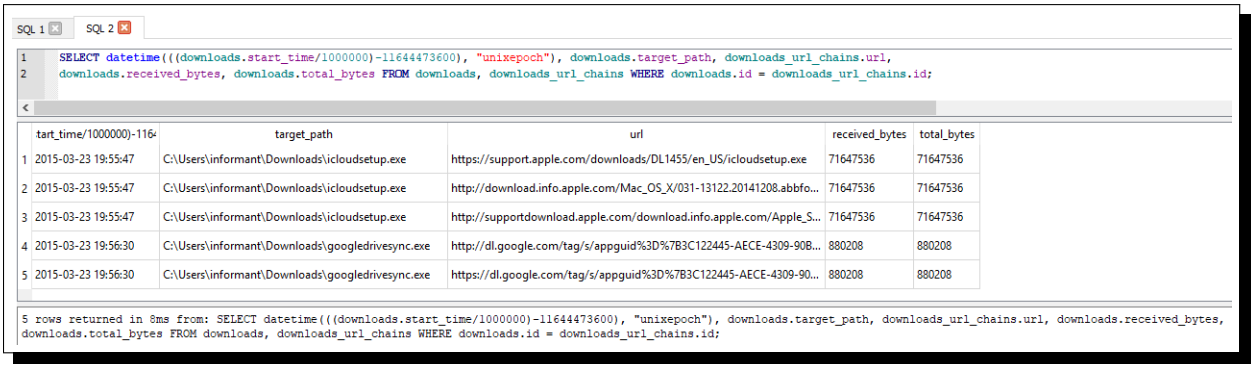


Figure 4.29: Relevant browser history found via DB Browser for SQlite on Chrome History DB II

Hindsight and they are listed in the Figure 4.31 below:

Along with forensics of the Google Chrome browser, it is important to look into any other browser that were installed on the system. As it was earlier found out that Internet Explorer 11 was additionally installed on the system, forensic investigation of the same needed to be conducted. To do so, BrowsingHistoryView v2.10

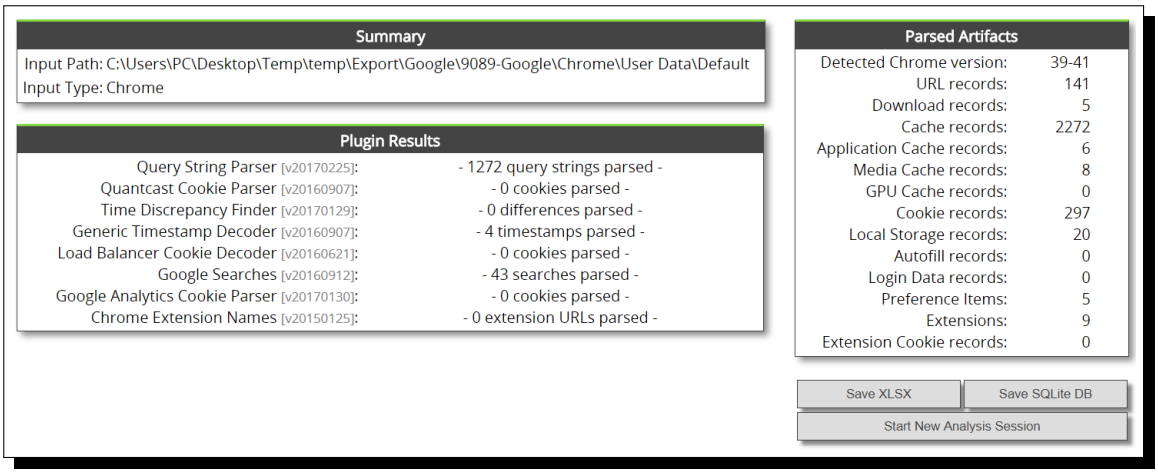
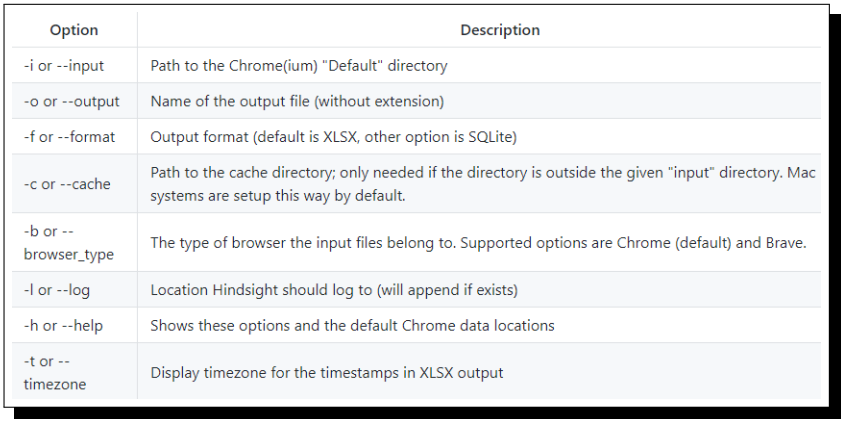


Figure 4.30: Hindsight analysis of Chrome artifacts



Option	Description
-i or --input	Path to the Chrome(ium) "Default" directory
-o or --output	Name of the output file (without extension)
-f or --format	Output format (default is XLSX, other option is SQLite)
-c or --cache	Path to the cache directory; only needed if the directory is outside the given "input" directory. Mac systems are setup this way by default.
-b or --browser_type	The type of browser the input files belong to. Supported options are Chrome (default) and Brave.
-l or --log	Location Hindsight should log to (will append if exists)
-h or --help	Shows these options and the default Chrome data locations
-t or --timezone	Display timezone for the timestamps in XLSX output

Figure 4.31: Additional options within Hindsight

[27] was utilized. The cache of the history of Internet Explorer is usually preserved in the following location : "AppData/Local/Microsoft/Windows/WebCache/WebCacheV01.dat". After extraction of this file, it was fed into the tool and a quick filter on searches resulted in the following:

Significant keywords deduced from the analysis were "windows system artifacts", "investigation on windows machine", "file sharing and tethering", "forensic email investigation" "external device forensics", "email investigation", "cd burning method in windows", "anti forensics tools" etc. This clearly indicates that the suspected was aware of his/her actions and was looking for ways to delete records and also finding consequences of his/her actions. The suspect was also looking at ways to transfer data via devices that are not connected to a network. This is made possible via transmission of heat [28], but the speed is extremely limited to a reliable transfer rate

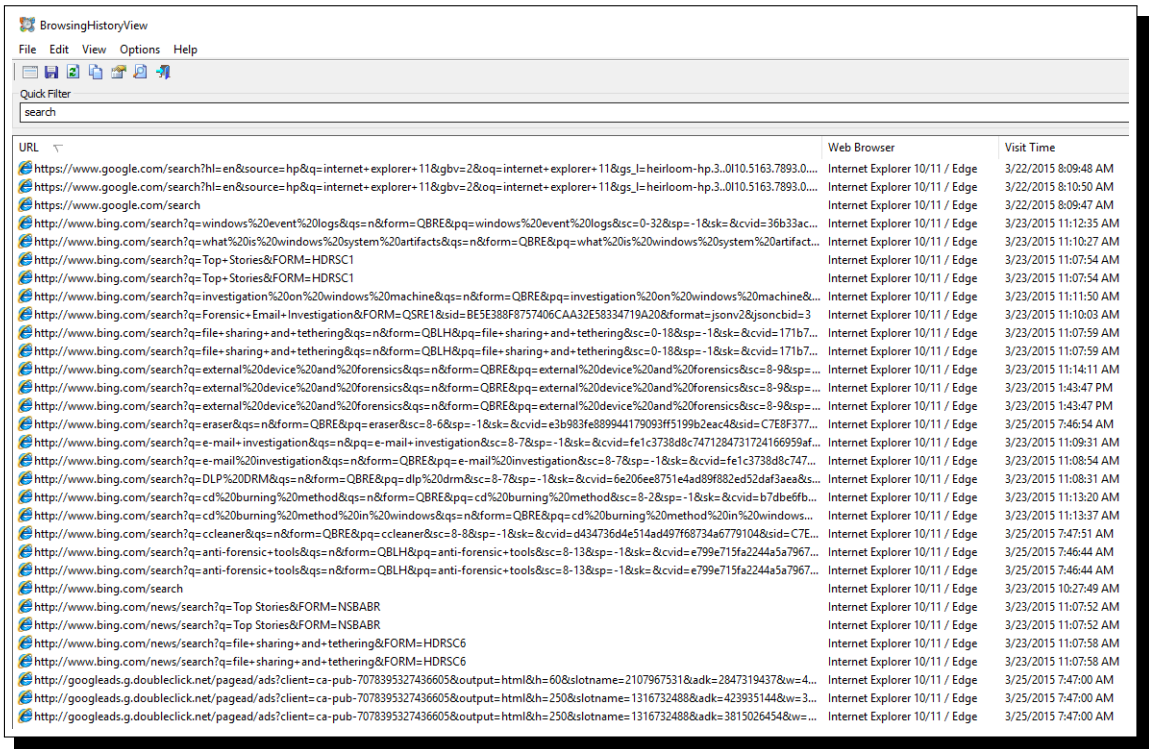
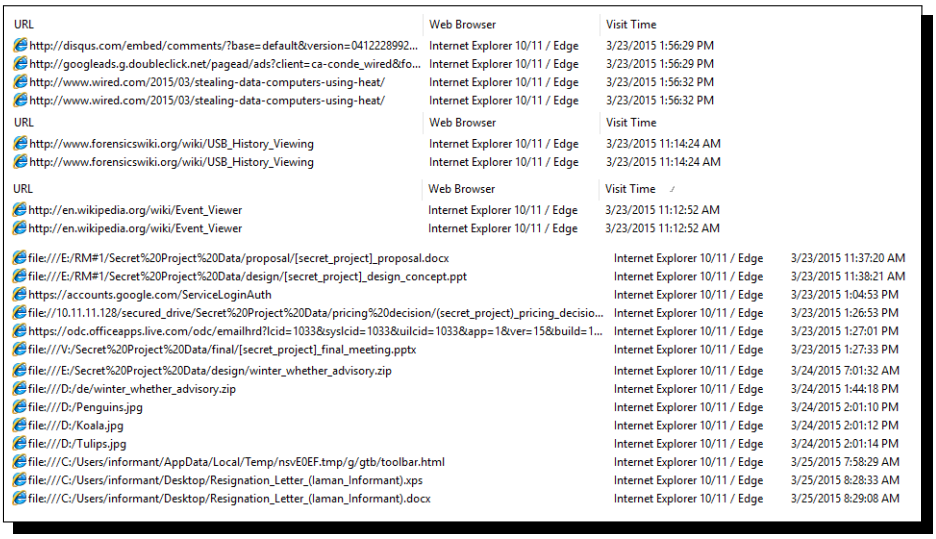


Figure 4.32: Use of BrowsingHistoryViewer to find history related to Internet Explorer I

of 8 bits over an hour, which is extremely slow, while considering large files. Details of USB analysis and actions performed on a flash drive was being researched from a forensic standpoint. Working mechanisms of the Windows Event Viewer, which creates logs for application and system messages like errors, job triggers, information messages, system exceptions and warnings, was also looked into. Additional pieces of information along with the aforementioned data is represented in the concise Figure 4.33 below.

4.2.5 Email Forensics

Email exchanges stored within a mail client can be extracted in a variety of ways. This report will outline two possible approaches that are commonly used and have a high success rate in recovering messages in the form of email that were sent and/received using an account that is linked to a mail client installed natively on the system. Often times, organizations and individuals opt for the option to store a local copy of emails exchanged, calendar information, contacts etc. related to a mail client locally on the



URL	Web Browser	Visit Time
http://disqus.com/embed/comments/?base=default&version=0412228992...	Internet Explorer 10/11 / Edge	3/23/2015 1:56:29 PM
http://googleads.g.doubleclick.net/pagead/ads?client=ca-conde_wired&fo...	Internet Explorer 10/11 / Edge	3/23/2015 1:56:29 PM
http://www.wired.com/2015/03/stealing-data-computers-using-heat/	Internet Explorer 10/11 / Edge	3/23/2015 1:56:32 PM
http://www.wired.com/2015/03/stealing-data-computers-using-heat/	Internet Explorer 10/11 / Edge	3/23/2015 1:56:32 PM

URL	Web Browser	Visit Time
http://www.forensicswiki.org/wiki/USB_History_Viewing	Internet Explorer 10/11 / Edge	3/23/2015 11:14:24 AM
http://www.forensicswiki.org/wiki/USB_History_Viewing	Internet Explorer 10/11 / Edge	3/23/2015 11:14:24 AM

URL	Web Browser	Visit Time
http://en.wikipedia.org/wiki/Event_Viewer	Internet Explorer 10/11 / Edge	3/23/2015 11:12:52 AM
http://en.wikipedia.org/wiki/Event_Viewer	Internet Explorer 10/11 / Edge	3/23/2015 11:12:52 AM

file:///E:/RM#1/Secret%20Project%20Data/proposal/[secret_project]_proposal.docx	Internet Explorer 10/11 / Edge	3/23/2015 11:37:20 AM
file:///E:/RM#1/Secret%20Project%20Data/design/[secret_project]_design_concept.ppt	Internet Explorer 10/11 / Edge	3/23/2015 11:38:21 AM
https://accounts.google.com/ServiceLoginAuth	Internet Explorer 10/11 / Edge	3/23/2015 1:04:53 PM
file://10.11.11.128/secured_drive/Secret%20Project%20Data/pricing%20decision/(secret_project)_pricing_decisio...	Internet Explorer 10/11 / Edge	3/23/2015 1:26:53 PM
https://odc.officeapps.live.com/odc/emailhrd?lcid=1033&syslcid=1033&uilcid=1033&app=1&ver=15&build=1...	Internet Explorer 10/11 / Edge	3/23/2015 1:27:01 PM
file:///V:/Secret%20Project%20Data/final/[secret_project]_final_meeting.pptx	Internet Explorer 10/11 / Edge	3/23/2015 1:27:33 PM
file:///E:/Secret%20Project%20Data/design/winter_whether_advisory.zip	Internet Explorer 10/11 / Edge	3/24/2015 7:01:32 AM
file:///D:/de/winter_whether_advisory.zip	Internet Explorer 10/11 / Edge	3/24/2015 1:44:18 PM
file:///D:/Penguins.jpg	Internet Explorer 10/11 / Edge	3/24/2015 2:01:10 PM
file:///D:/Koala.jpg	Internet Explorer 10/11 / Edge	3/24/2015 2:01:12 PM
file:///D:/Tulips.jpg	Internet Explorer 10/11 / Edge	3/24/2015 2:01:14 PM
file:///C:/Users/informant/AppData/Local/Temp/nsvE0EF.tmp/g/gtb/toolbar.html	Internet Explorer 10/11 / Edge	3/25/2015 7:58:29 AM
file:///C:/Users/informant/Desktop/Resignation_Letter_(laman_Informant).xps	Internet Explorer 10/11 / Edge	3/25/2015 8:28:33 AM
file:///C:/Users/informant/Desktop/Resignation_Letter_(laman_Informant).docx	Internet Explorer 10/11 / Edge	3/25/2015 8:29:08 AM

Figure 4.33: Use of BrowsingHistoryViewer to find history related to Internet Explorer II

system in the form of .pst or .ost files. The primary difference between the two are that, the former is a local copy of ones' email information from an Exchange server, thus removing that information from the Exchange storage, where as the latter is used for fetching individual information from a local system, when an Exchange server is offline. Another difference is in their syncing processes. The .pst format might not necessarily delete the backup from the central store, but while using .ost file type, any changes made locally is also reflected in the central repo, which helps in maintaining uniformity while accessing information about an account from a variety of devices. In this specific case, the .ost file was found to reside in the location that is displayed in the Figure 4.34 below along with the time stamp of when it was last modified.

Once the file was recovered via Autopsy, the next step was to explore its contents. The tool of choice was PST Viewer V8 [29] in this case. Once the .ost file was loaded into PST Viewer, the core contents of every message that was exchanged can be uncovered. The total number of messages exchanged between the suspect and the conspirator can be seen in the Figure 4.35 below.

To find the contents of each of the email exchanges listed above, the messages need to be exported into a format of any choice ranging across text, pdf, html, csv, xls, doc etc. Once the messages were exported, a diagram (Figure 4.38)was generated by timeline analysis of each message and the figure can be seen below. The conversations definitely point to the fact that a data leak was plotted between 2 individuals, one

/img_cfreds_2015_data_leakage_pc.dd/vol_vol3/Users/informant/AppData/Local/Microsoft/Outlook

Table | Thumbnail

Name	Modified Time	Change Time
[current folder]	2015-03-25 08:11:47 PDT	2015-03-25 08:11:47 PDT
[parent folder]	2015-03-23 10:29:57 PDT	2015-03-23 10:29:57 PDT
Offline Address Books	2015-03-22 08:50:21 PDT	2015-03-22 08:50:21 PDT
RoamCache	2015-03-23 12:29:29 PDT	2015-03-23 12:29:29 PDT
fc39fbc8c85bcb43816b40b7d4c72f22 - Autodiscover.xml	2015-03-25 07:41:36 PDT	2015-03-25 07:41:36 PDT
iaman.informant@nist.gov.ost	2015-03-25 08:11:47 PDT	2015-03-25 08:11:47 PDT
mapisvc.inf	2015-03-25 07:41:03 PDT	2015-03-25 07:41:03 PDT
~iaman.informant@nist.gov.ost.tmp	2015-03-25 07:41:04 PDT	2015-03-25 07:41:04 PDT
~iaman.informant@nist.gov.ost.tmp	0000-00-00 00:00:00	0000-00-00 00:00:00

Figure 4.34: Recovery of .ost file related to the suspected user

From	To	Subject	Received time
spy	iaman	Hello, Iaman	03/23/2015 10:29:29 AM
iaman	spy	RE: Hello, Iaman	03/23/2015 11:44:00 AM
spy	iaman	Good job, buddy.	03/23/2015 12:15:00 PM
spy	iaman	RE: Good job, buddy.	03/23/2015 12:20:41 PM
spy	iaman	Important request	03/23/2015 12:26:23 PM
iaman	spy	RE: Important request	03/23/2015 12:27:00 PM
iaman	iaman	Synchronization Log:	03/23/2015 12:57:30 PM
spy	iaman	RE: It's me	03/23/2015 01:41:22 PM
spy	iaman	Last request	03/24/2015 06:25:59 AM
iaman	spy	RE: Last request	03/24/2015 06:35:00 AM
iaman	spy	RE: Watch out!	03/24/2015 12:34:00 PM
iaman	spy	Done	03/24/2015 02:05:00 PM
iaman	iaman	Synchronization Log:	03/25/2015 08:01:49 AM
iaman	iaman	Synchronization Log:	03/25/2015 08:01:55 AM

Figure 4.35: List of emails in PST Viewer

of them was within the organization and is suspected of this breach and the other person who was outside of the organization and works for a competing company.

Within the conversation which is depicted with a timeline in the Figure 4.38, there were 3 synchronization logs that were sent by the mail server. The first of them was captured at 13:41 on March 23rd, 2015, which was shortly after the first file was sent

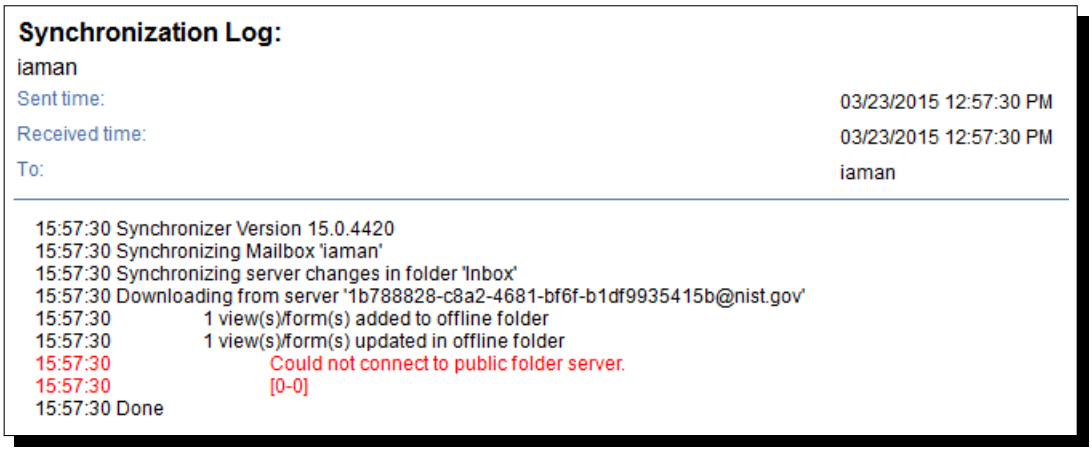
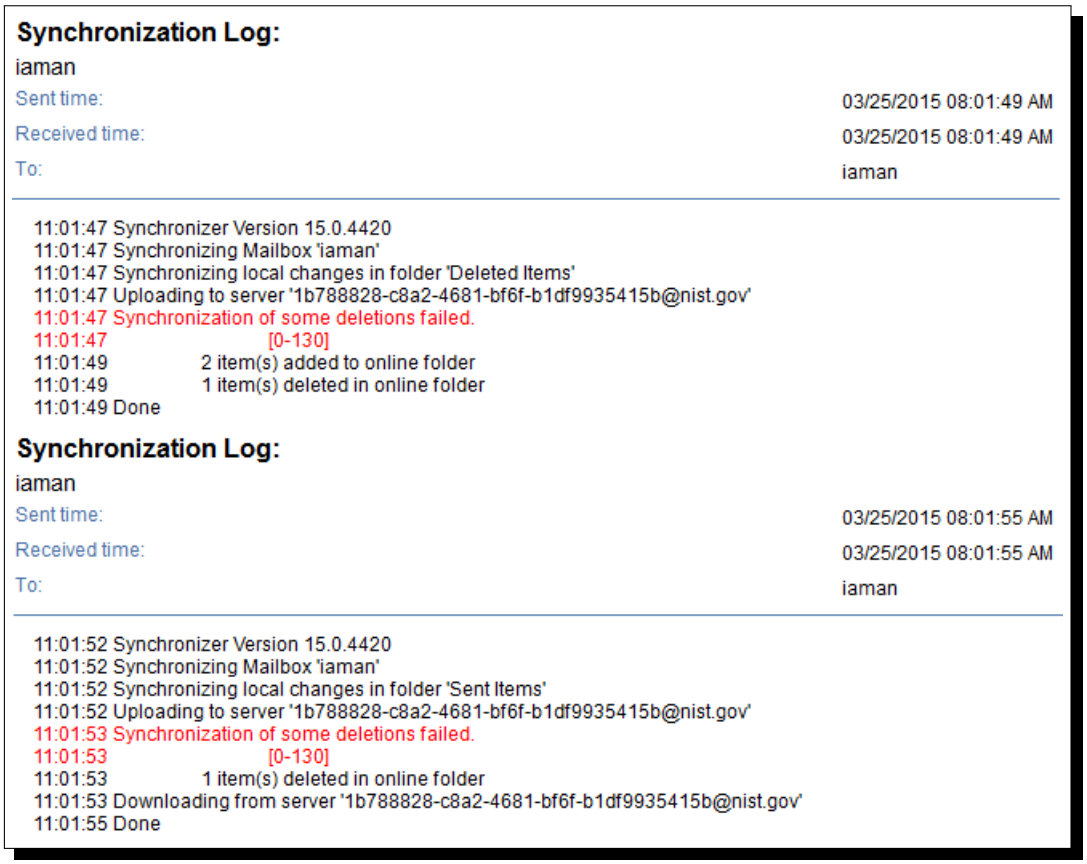


Figure 4.36: Error reported in first Sync Log from Exchange Server

by the source to the destination and after acknowledgment of receipt of the file, it was deleted from the exchange server. This could have been possibly done by the informant. At this point the investigator does not have access to the mail server, so this cannot be further investigated. But if additional evidence is required at a future stage of the investigation, activities in the exchange server can be examined. The screenshot (Figure 4.36) confirms that there was one view to the public folder which is basically a container for an attachment, but after deletion of the file, its presence in the same public folder might not be verified.

The following two synchronization logs are depicted in the Figure 4.37 below, which proves the point that two files were uploaded to the public folder of the exchange sever by the informant. Once of the files were downloaded by the conspirator, it was deleted by the insider, hence there were synchronization failures. This can be used as a red flag by the IT Security division of a company to look into, to deduce whether it is a matter of serious concern or otherwise.

An important thing to mention here is that Microsoft Outlook could also be used to import the .ost file and look at the information exchange. It might house additional and extraneous data which might require cleansing before it can be presented as evidence. But PST Viewer allows an investigator to look into a plethora of file formats ranging from .msg, .pst(this case), .ost, and .eml files. Exporting options are available from plain text, html, pdf, csv to name a few. Data can be generated in a way so that further processing can be done with ease. Notable mentions here which could be used for similar purposes would be SQL MDF, SQL LDF, Exchange EDB, DBX Viewer.



Synchronization Log:

iaman

Sent time: 03/25/2015 08:01:49 AM

Received time: 03/25/2015 08:01:49 AM

To: iaman

11:01:47 Synchronizer Version 15.0.4420
11:01:47 Synchronizing Mailbox 'iaman'
11:01:47 Synchronizing local changes in folder 'Deleted Items'
11:01:47 Uploading to server '1b788828-c8a2-4681-bf6f-b1df9935415b@nist.gov'
11:01:47 Synchronization of some deletions failed.
11:01:47 [0-130]
11:01:49 2 item(s) added to online folder
11:01:49 1 item(s) deleted in online folder
11:01:49 Done

Synchronization Log:

iaman

Sent time: 03/25/2015 08:01:55 AM

Received time: 03/25/2015 08:01:55 AM

To: iaman

11:01:52 Synchronizer Version 15.0.4420
11:01:52 Synchronizing Mailbox 'iaman'
11:01:52 Synchronizing local changes in folder 'Sent Items'
11:01:52 Uploading to server '1b788828-c8a2-4681-bf6f-b1df9935415b@nist.gov'
11:01:53 Synchronization of some deletions failed.
11:01:53 [0-130]
11:01:53 1 item(s) deleted in online folder
11:01:53 Downloading from server '1b788828-c8a2-4681-bf6f-b1df9935415b@nist.gov'
11:01:55 Done

Figure 4.37: Consequent couple of errors in Sync log

4.2.6 File System Analysis

This is undoubtedly the utmost vital phase of the analysis as the data that is suspected of being leaked by an insider will be explored and carved out. But before looking into the native file system itself, it is important to explore the results that were found out during the email forensics phase a bit further. There were a couple of Google Drive links that were shared by the source with the destination in one of the emails'. It is important to look into the files as they were supposed to be the sample data that was first shared but due to the larger size of the other files, the option to share data via cloud storage was decided against. The following steps were performed within Kali to demonstrate an alternate analysis methodology. The first link lead to the download of what was masked to be an audio file in .mp3 format. Although the file size was that of an usual .mp3 file, sharing of this file with the supposed conspirator didn't make much sense, which is why the file had to be explored further. The first thing that needed to be checked was the verification of whether the file was indeed

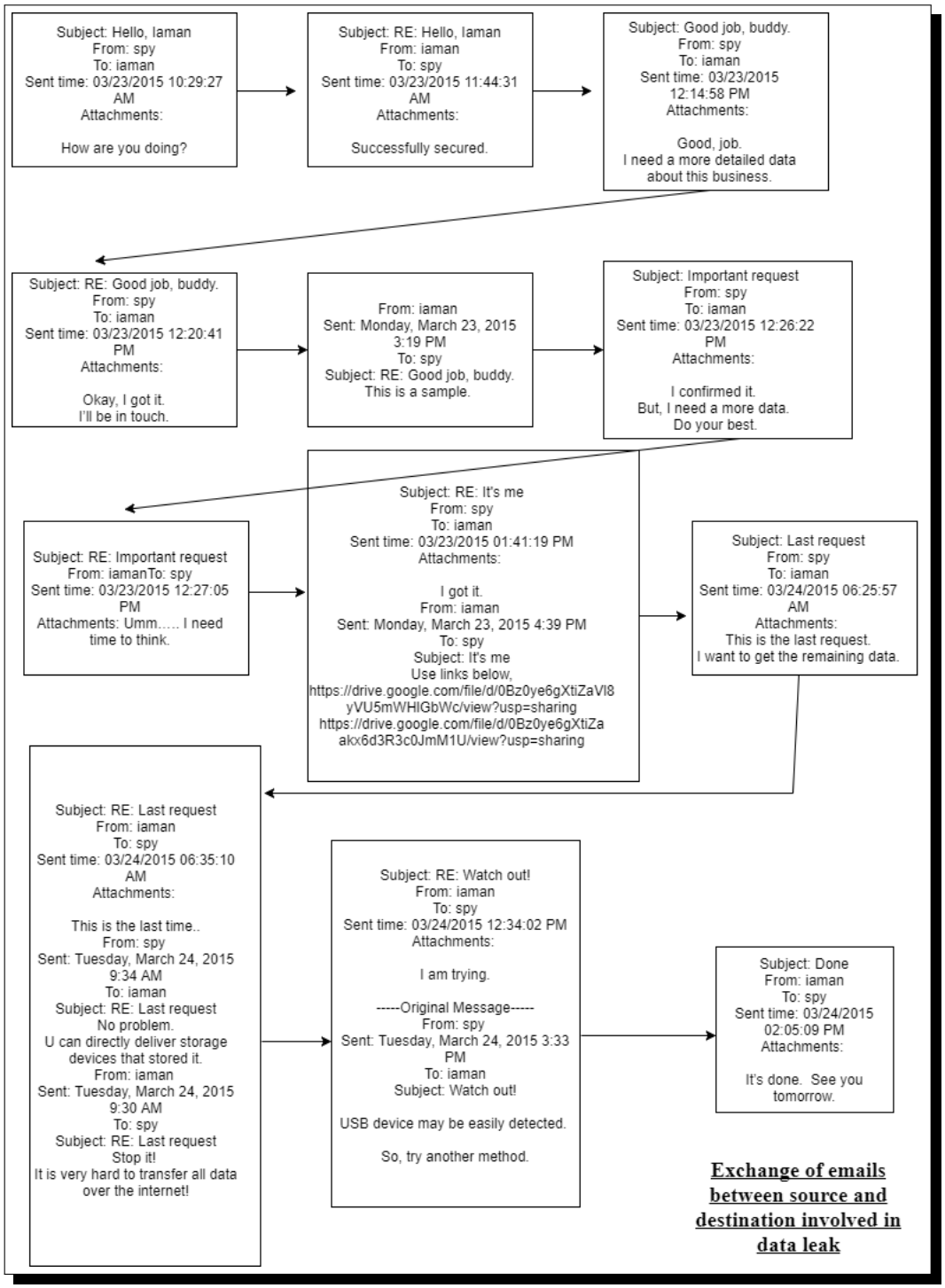


Figure 4.38: Entire list of emails exchanged between the internal suspect and the conspirator

an .mp3 file or not. To do this a Hex editor was used to check the header of the file and it was found out to have a PK [30] header signature. PK is the initials of Phil Katz, the co-founder of the .zip format and the developer of PKZIP. So now that it was uncovered that the file was a zipped archive, it was renamed with an extension of .zip.

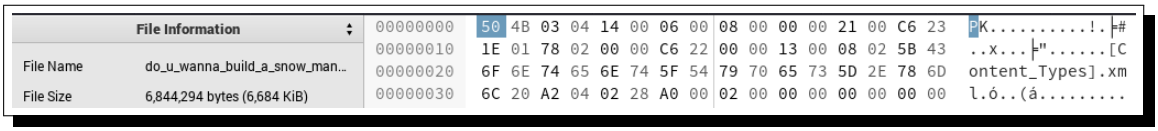


Figure 4.39: Header information of a modified file via HexEditor

Upon extracting the file after renaming it, its contents were a bunch of files and folders within which didn't give any further clues, apart from the fact of the presence of a folder called ppt within, which could've meant that the file was a .ppt file. The question that arises at this point was how to merge all the files in it and restore the file to its originally intended phase. There had to be some sort of reverse engineering performed on it to recover the file.

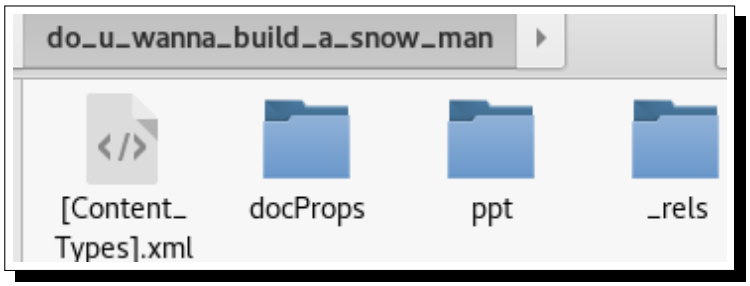


Figure 4.40: Contents of zipped file

Upon opening up one of the .xml files, as can be seen in the Figure 4.41 below, it was found out that the document was an Open Office document, of presentation type and there were a few slides in it.

The files and folders within the zipped archive was then zipped up again but in a way so that it became a readable format. The command line was used to do this and the experimental command "zip -r ppt_from_gdrive.ppt *" was used to obtain the original .ppt file. The contents listed in the xml file were deflated as can be seen in the Figure 4.42 from the terminal capture below and a new .ppt was created.

```
<Types xmlns="http://schemas.openxmlformats.org/package/2006/content-types">
  <Default Extension="png" ContentType="image/png"/>
  <Default Extension="jpeg" ContentType="image/jpeg"/>
  <Default Extension="rels" ContentType="application/vnd.openxmlformats-package.relationships+xml"/>
  <Default Extension="xml" ContentType="application/xml"/>
  <Override PartName="/ppt/presentation.xml" ContentType="application/vnd.openxmlformats-
  officedocument.presentationml.presentation.main+xml"/>
  <Override PartName="/ppt/slideMasters/slideMaster1.xml" ContentType="application/vnd.openxmlformats-
  officedocument.presentationml.slideMaster+xml"/>
  <Override PartName="/ppt/slideMasters/slideMaster2.xml" ContentType="application/vnd.openxmlformats-
  officedocument.presentationml.slideMaster+xml"/>
  <Override PartName="/ppt/slides/slide1.xml" ContentType="application/vnd.openxmlformats-
  officedocument.presentationml.slide+xml"/>
  <Override PartName="/ppt/slides/slide2.xml" ContentType="application/vnd.openxmlformats-
  officedocument.presentationml.slide+xml"/>
  <Override PartName="/ppt/slides/slide3.xml" ContentType="application/vnd.openxmlformats-
  officedocument.presentationml.slide+xml"/>
  <Override PartName="/ppt/slides/slide4.xml" ContentType="application/vnd.openxmlformats-
  officedocument.presentationml.slide+xml"/>
  <Override PartName="/ppt/slides/slide5.xml" ContentType="application/vnd.openxmlformats-
  officedocument.presentationml.slide+xml"/>
  <Override PartName="/ppt/slides/slide6.xml" ContentType="application/vnd.openxmlformats-
  officedocument.presentationml.slide+xml"/>
  <Override PartName="/ppt/slides/slide7.xml" ContentType="application/vnd.openxmlformats-
  officedocument.presentationml.slide+xml"/>
  <Override PartName="/ppt/slides/slide8.xml" ContentType="application/vnd.openxmlformats-
  officedocument.presentationml.slide+xml"/>
  <Override PartName="/ppt/slides/slide9.xml" ContentType="application/vnd.openxmlformats-
  officedocument.presentationml.slide+xml"/>
  <Override PartName="/ppt/slides/slide10.xml" ContentType="application/vnd.openxmlformats-
  officedocument.presentationml.slide+xml"/>
  <Override PartName="/ppt/slides/slide11.xml" ContentType="application/vnd.openxmlformats-
  officedocument.presentationml.slide+xml"/>
  <Override PartName="/ppt/slides/slide12.xml" ContentType="application/vnd.openxmlformats-
  officedocument.presentationml.slide+xml"/>
  <Override PartName="/ppt/slides/slide13.xml" ContentType="application/vnd.openxmlformats-
  officedocument.presentationml.slide+xml"/>
  <Override PartName="/ppt/slides/slide14.xml" ContentType="application/vnd.openxmlformats-
  officedocument.presentationml.slide+xml"/>
  <Override PartName="/ppt/slides/slide15.xml" ContentType="application/vnd.openxmlformats-
  officedocument.presentationml.slide+xml"/>
  <Override PartName="/ppt/slides/slide16.xml" ContentType="application/vnd.openxmlformats-
```

Figure 4.41: XML file explaining the contents of the Open Office document

```
root@kali:~/Downloads/drive/do_u_wanna_build_a_snow_man# zip -r ppt_from_gdrive.ppt
*
  adding: [Content_Types].xml (deflated 93%)
  adding: docProps/ (stored 0%)
  adding: docProps/core.xml (deflated 48%)
  adding: docProps/app.xml (deflated 64%)
  adding: docProps/thumbnail.jpeg (deflated 7%)
  adding: ppt/ (stored 0%)
  adding: ppt/presProps.xml (deflated 52%)
  adding: ppt/viewProps.xml (deflated 63%)
  adding: ppt/slideLayouts/ (stored 0%)
  adding: ppt/slideLayouts/slideLayout18.xml (deflated 69%)
  adding: ppt/slideLayouts/slideLayout16.xml (deflated 77%)
  adding: ppt/slideLayouts/slideLayout20.xml (deflated 77%)
  adding: ppt/slideLayouts/slideLayout10.xml (deflated 70%)
  adding: ppt/slideLayouts/slideLayout22.xml (deflated 73%)
  adding: ppt/slideLayouts/slideLayout7.xml (deflated 63%)
```

Figure 4.42: Re-zipping of files into an appropriate format

The contents of the file as can be seen in Figure 4.43 seemed to be excerpts from a secret project of some sort which was mentioned within the case information about being leaked. This was just the beginning of the file analysis and definitely a positive result leading the way to further investigation.

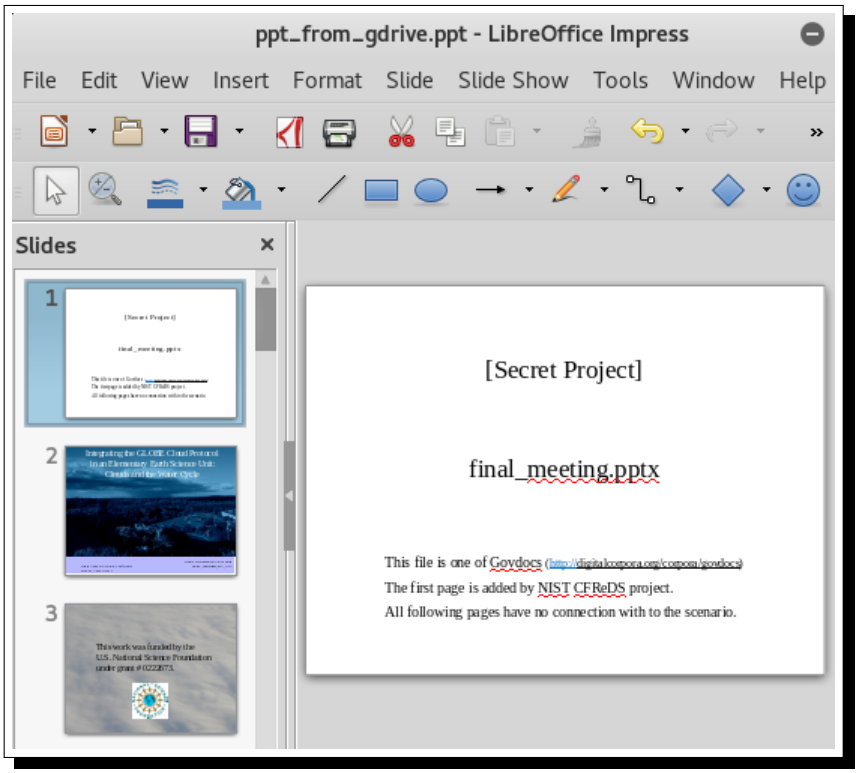


Figure 4.43: One of the recovered Powerpoint presentations'

The second link from Google Drive was the source to a .jpg file, which upon examination with a HEX editor was found out to be a pk zip file again. After renaming it to .zip and extracting the contents, it seemed to house an excel sheet in open office format. The command that was used to deflate the objects according to the source XML and restoring the original file was " zip -r xl_from_gdrive.xls *". Once this was done, it was observed that the file contained pricing decisions of the organizations "secret project". Accounting information for a company is highly sensitive and leakage of the same can result in substantial financial losses. The Figure 4.44 below depicts the restoration of the second file along with a section of the contents of the spreadsheet that was recovered.

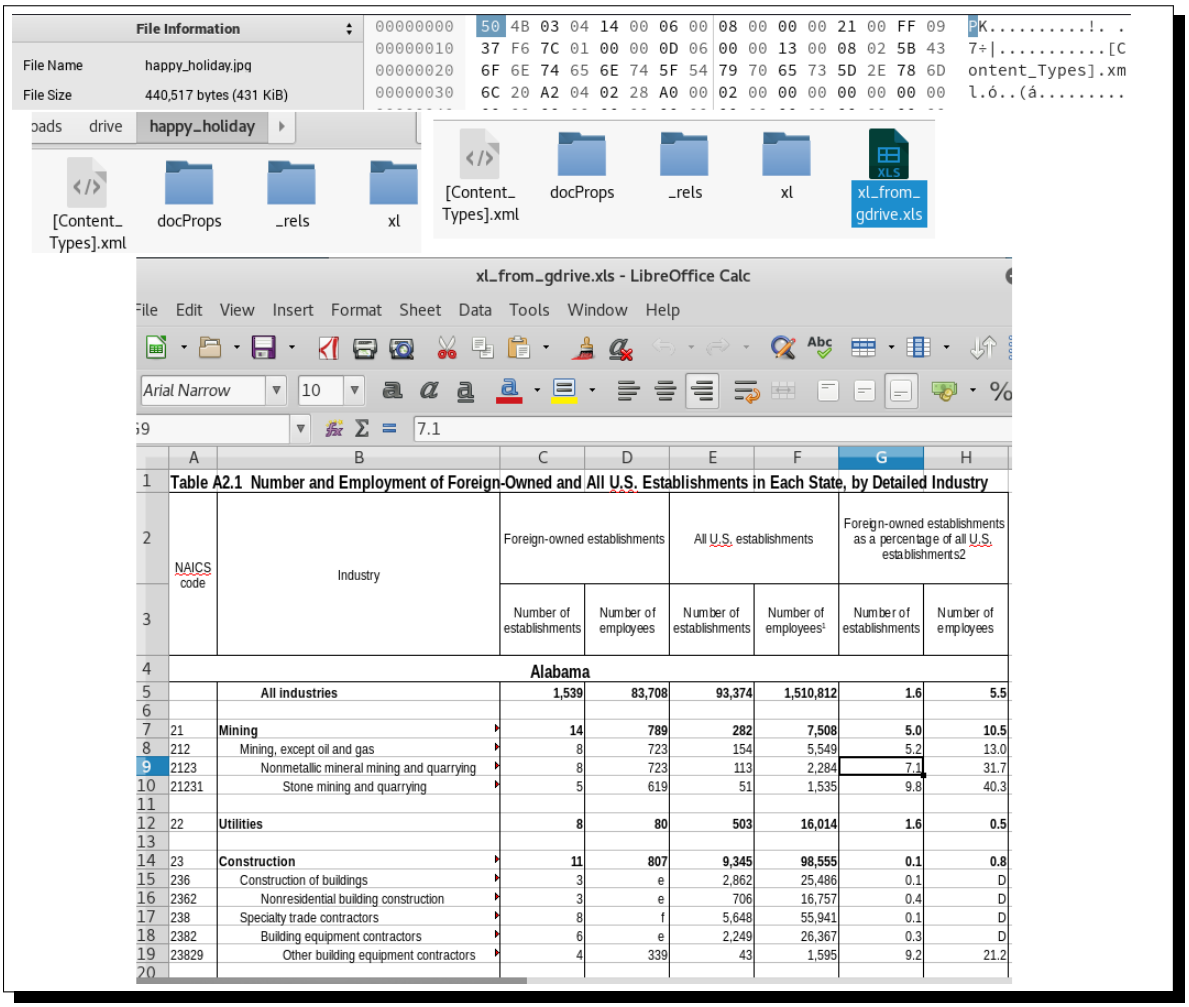


Figure 4.44: Recovery of a Spreadsheet

One important consideration here is that although they were part of the communication, there isn't evidence of uploading these files by the insider to the cloud storage platform. To prove this, the section will be visited at a later stage in this report while exploring database forensics.

Post analysis of the links shared in one of the email messages, the file system of the computer needed to be examined. A list of directories that were traversed by the user needed to be obtained to make sure that the files were accessed by the suspect. In order to do this the associated Windows Shellbags needed to be examined. Shellbags have been a part of Windows operating systems since Windows XP, but they have not been regarded as having a high potential in forensic investigations, up until recently. Shellbags not only contain information about local storage and the views

associated with it, but also about the removable devices and shared network drives associated with a system. Shellbags can help in root cause analysis of investigations related to spreading of malware, remote snooping from foreign machines via RDP. It also holds crucial information about directory traversal within a specified system. This information is usually stored in an encrypted format within the UsrClass.dat file which is located under "/username/AppData/Local/Microsoft/Windows" folder in Windows 7 and 10 operating systems. Once the file was extracted using Autopsy, it was first examined in a Windows environment and then transferred over to Kali for further investigation. The first tool that was used is called ShellBags Explorer [31]. The absence of relevant data from Shellbags might indicate the removal of certain entries from an anti-forensic standpoint, although the user did install Eraser, it is not capable enough to delete this entries from the system. The registry key that is of primary importance here is "HKEY_CURRENT_USER_Software_Classes_Local Settings_Software_Microsoft_Windows_Shell_BagMRU". Using SBE, an offline registry hive can be examined and that is exactly what was done in this scenario. The following Figure 4.45 depicts the initial analysis which shows the presence of a directory called "Secret Project Data" which contains 3 subfolders called design, pricing decision, technical review, projects, final etc. Each directory has an individual MRU position which can be used to trace back its traversal.

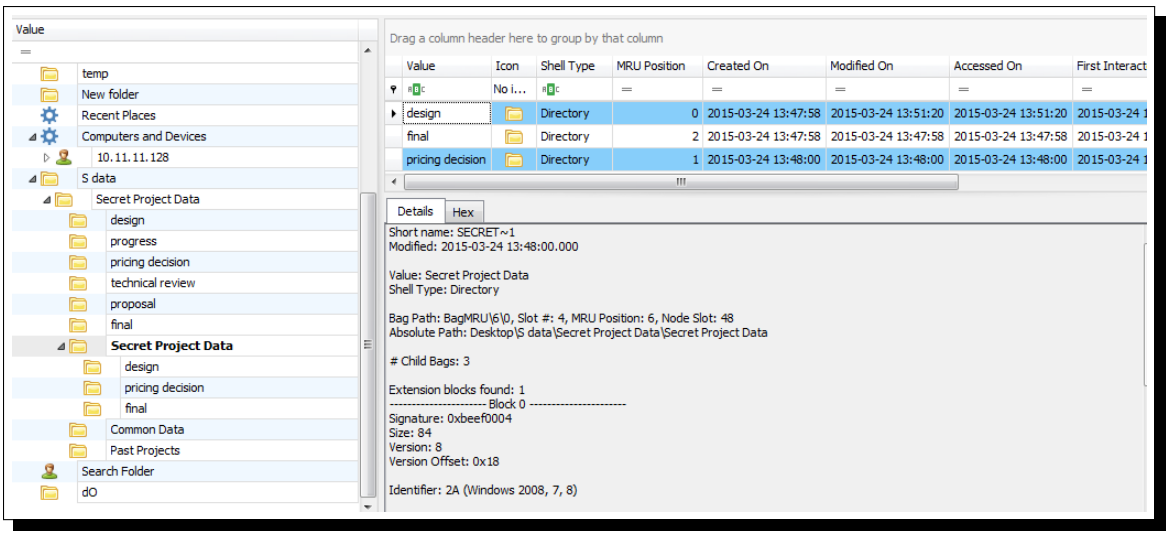


Figure 4.45: Use of ShellBag Explorer to find local directory traversal

Another interesting artifact that was uncovered at this point was the presence of a shared network drive as indicated in the Figure 4.46 below. It is evident from the structure of the traversed paths that certain folders were copied over from the network drive to the local hard disk drive of the machine. This can be verified in the next stage when those locations are investigated for remains of any files that were suspected of being leaked.

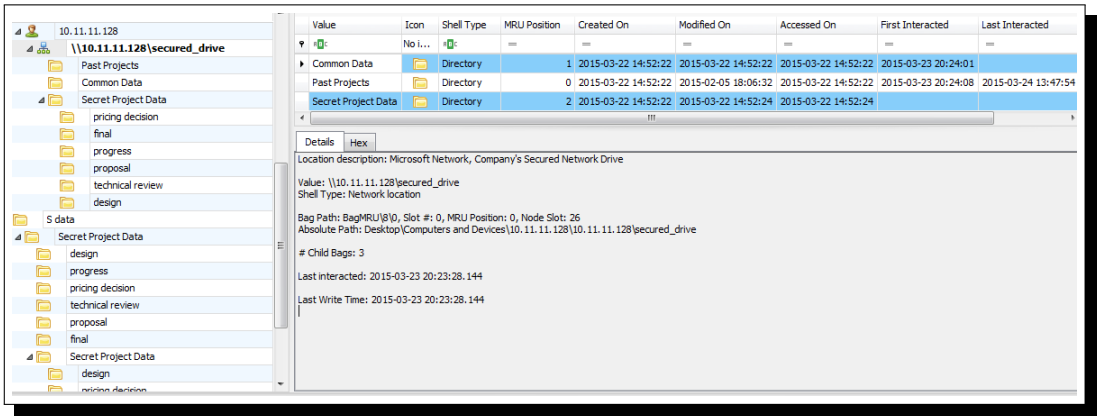


Figure 4.46: Use of ShellBag Explorer to find directory traversal in Shared Network Drive

The final piece of information uncovered via this tool is the different drives that were accessed via this system. Out of the various drives, it can be noticed that drives D: and E: are of special interest. The former was used to burn a bunch of files that are supposed to be of high importance to a CD ROM, where as the latter houses a link to a removable device which could be the flash drive and it contains one zipped file which will be investigated in the latter sections.

This tool was highly useful in lending an initial insight into the traversed system paths, but in order to access those same locations from the images, actual paths needed to be recovered. To do this, the UsrClass.dat file is transferred over to Kali. An opensource cross-platform tool called shellbags [32] was used in this case to find the absolute paths traversed within the suspected system. A csv file was generated by using the command as indicated in the Figure 4.48 below.

Once the .csv file was generated, the absolute paths were recovered and it matched with the suspected paths from the MRUs discovered in SBE. But the timeline was not generated in an appropriate manner, hence a tool called mactime was used to generate an ASCII timeline of the data which is dependent on the output of the native “fls” tool. The list of directories traversed and sorted by time is represented

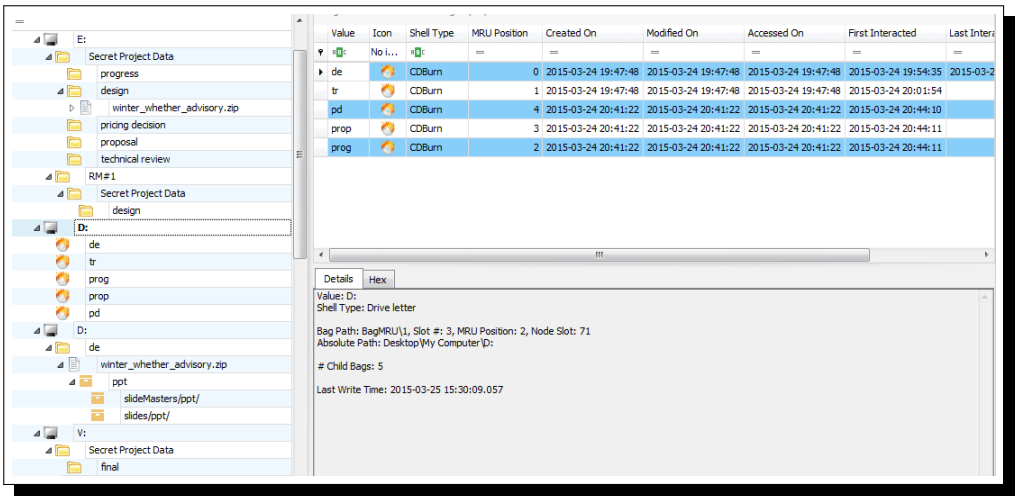
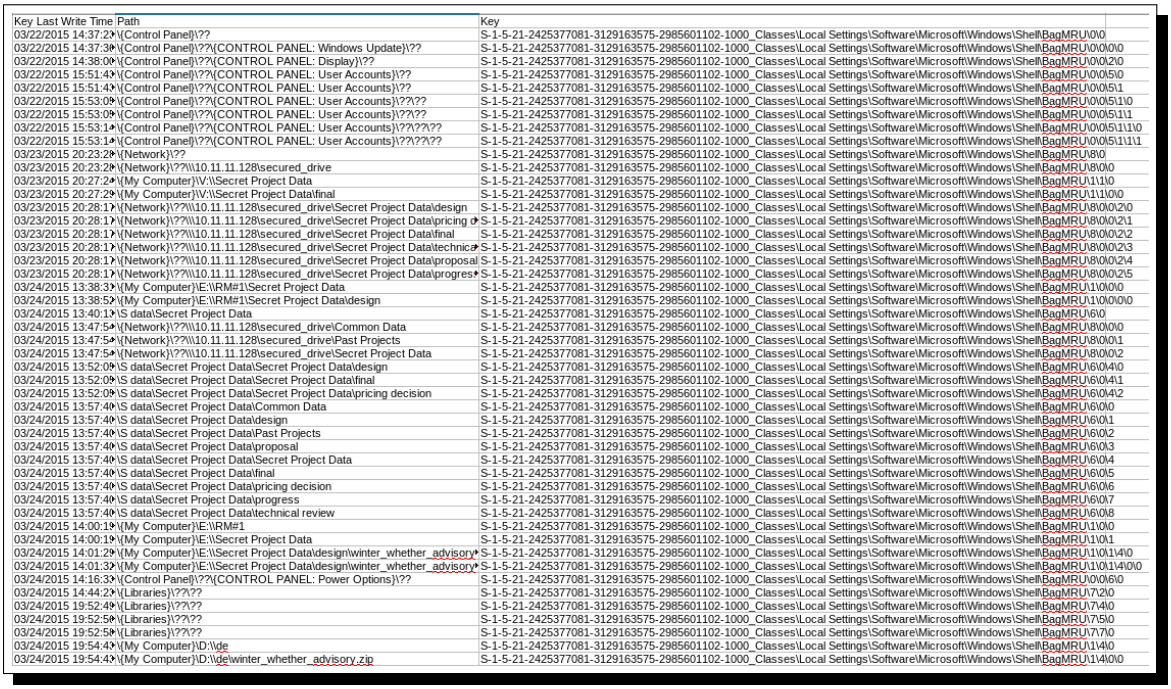


Figure 4.47: Use of ShellBag Explorer to find burning of files/folders to a compact disc

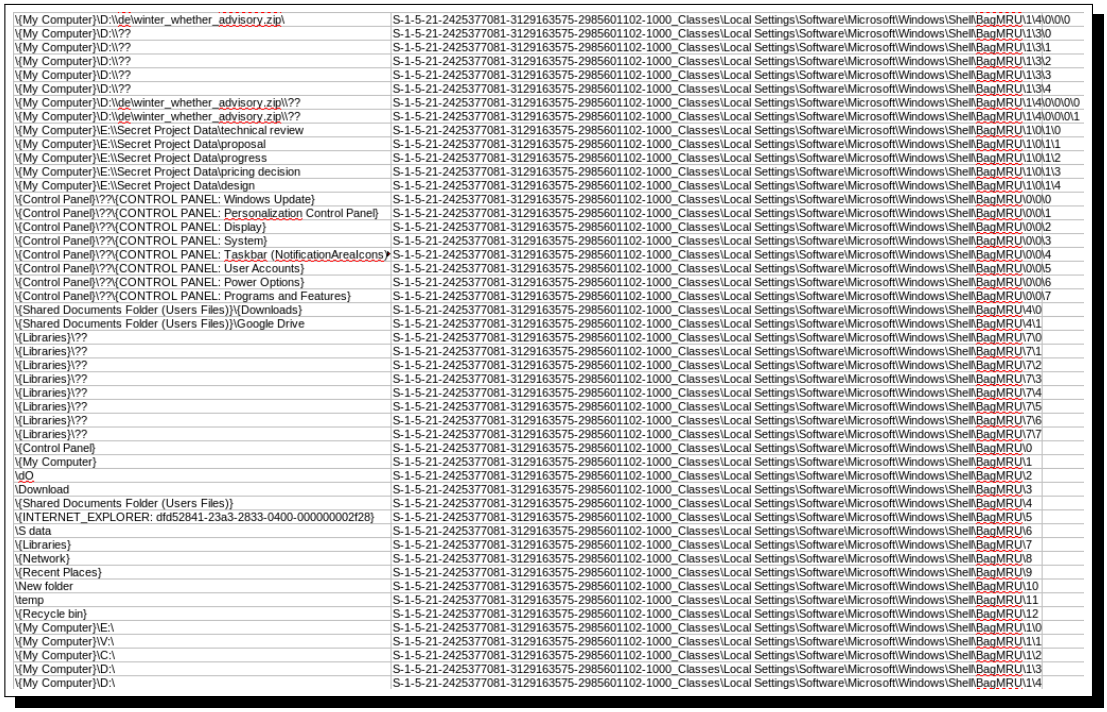
Figure 4.48: Use of ShellBags on UsrClass.dat

in the following two Figures (4.49, 4.50) below.



Key Last Write Time	Path	Key
03/22/2015 14:37:23	\{Control Panel}\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0\0
03/22/2015 14:37:39	\{Control Panel}\??{CONTROL PANEL: Windows Update}\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0\0\0\0
03/22/2015 14:38:00	\{Control Panel}\??{CONTROL PANEL: Display}\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0\0\2\0
03/22/2015 15:51:43	\{Control Panel}\??{CONTROL PANEL: User Accounts}\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0\0\5\0
03/22/2015 15:51:43	\{Control Panel}\??{CONTROL PANEL: User Accounts}\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0\0\5\1
03/22/2015 15:53:09	\{Control Panel}\??{CONTROL PANEL: User Accounts}\??\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0\0\5\1\0
03/22/2015 15:53:09	\{Control Panel}\??{CONTROL PANEL: User Accounts}\??\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0\0\5\1\1
03/22/2015 15:53:13	\{Control Panel}\??{CONTROL PANEL: User Accounts}\??\??\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0\0\5\1\1\0
03/22/2015 15:53:13	\{Control Panel}\??{CONTROL PANEL: User Accounts}\??\??\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0\0\5\1\1\1
03/23/2015 20:23:28	\{Network}\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\8\0
03/23/2015 20:23:28	\{Network}\??\\10.11.11.128\secured_drive	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\8\0\0
03/23/2015 20:27:29	\{My Computer}\V:\\Secret Project Data	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\1\0
03/23/2015 20:27:29	\{My Computer}\V:\\Secret Project Data\final	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\1\0\0
03/23/2015 20:28:17	\{Network}\??\\10.11.11.128\secured_drive\Secret Project Data\design	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\8\0\0\2\0
03/23/2015 20:28:17	\{Network}\??\\10.11.11.128\secured_drive\Secret Project Data\pricing d	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\8\0\0\2\1
03/23/2015 20:28:17	\{Network}\??\\10.11.11.128\secured_drive\Secret Project Data\final	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\8\0\0\2\2
03/23/2015 20:28:17	\{Network}\??\\10.11.11.128\secured_drive\Secret Project Data\technica	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\8\0\0\2\3
03/23/2015 20:28:17	\{Network}\??\\10.11.11.128\secured_drive\Secret Project Data\proposal	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\8\0\0\2\4
03/23/2015 20:28:17	\{Network}\??\\10.11.11.128\secured_drive\Secret Project Data\progres	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\8\0\0\2\5
03/24/2015 13:38:33	\{My Computer}\E:\\RM#1\Secret Project Data	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\0\0\0
03/24/2015 13:38:52	\{My Computer}\E:\\RM#1\Secret Project Data\design	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\0\0\0\0
03/24/2015 13:40:13	\S data\Secret Project Data	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\6\0
03/24/2015 13:47:54	\{Network}\??\\10.11.11.128\secured_drive\Common Data	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\8\0\0\0
03/24/2015 13:47:54	\{Network}\??\\10.11.11.128\secured_drive\Past Projects	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\8\0\0\1
03/24/2015 13:47:54	\{Network}\??\\10.11.11.128\secured_drive\Secret Project Data	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\8\0\0\2
03/24/2015 13:52:09	\S data\Secret Project Data\Secret Project Data\design	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\6\0\4\0
03/24/2015 13:52:09	\S data\Secret Project Data\Secret Project Data\final	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\6\0\4\1
03/24/2015 13:52:09	\S data\Secret Project Data\Secret Project Data\pricing decision	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\6\0\4\2
03/24/2015 13:57:40	\S data\Secret Project Data\Common Data	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\6\0\0
03/24/2015 13:57:40	\S data\Secret Project Data\design	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\6\0\1
03/24/2015 13:57:40	\S data\Secret Project Data\Past Projects	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\6\0\2
03/24/2015 13:57:40	\S data\Secret Project Data\proposal	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\6\0\3
03/24/2015 13:57:40	\S data\Secret Project Data\Secret Project Data	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\6\0\4
03/24/2015 13:57:40	\S data\Secret Project Data\final	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\6\0\5
03/24/2015 13:57:40	\S data\Secret Project Data\pricing decision	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\6\0\6
03/24/2015 13:57:40	\S data\Secret Project Data\progress	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\6\0\7
03/24/2015 13:57:40	\S data\Secret Project Data\technical review	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\6\0\8
03/24/2015 14:00:19	\{My Computer}\E:\\RM#1	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\0\0
03/24/2015 14:00:19	\{My Computer}\E:\\Secret Project Data	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\0\1
03/24/2015 14:01:29	\{My Computer}\E:\\Secret Project Data\design\winter_whether_advisory.z	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\0\1\4\0
03/24/2015 14:01:32	\{My Computer}\E:\\Secret Project Data\design\winter_whether_advisory.z	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\0\1\4\0\0
03/24/2015 14:16:33	\{Control Panel}\??{CONTROL PANEL: Power Options}\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0\0\6\0
03/24/2015 14:44:23	\{Libraries}\??\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\7\2\0
03/24/2015 19:52:49	\{Libraries}\??\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\7\4\0
03/24/2015 19:52:50	\{Libraries}\??\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\7\5\0
03/24/2015 19:52:50	\{Libraries}\??\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\7\7\0
03/24/2015 19:54:43	\{My Computer}\D:\\de	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\4\0
03/24/2015 19:54:43	\{My Computer}\D:\\de\winter_whether_advisory.zip	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\4\0\0

Figure 4.49: Traversed Directories I



\{My Computer}\D:\\de\winter_whether_advisory.zip\	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\4\0\0\0
\{My Computer}\D:\\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\3\0
\{My Computer}\D:\\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\3\1
\{My Computer}\D:\\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\3\2
\{My Computer}\D:\\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\3\3
\{My Computer}\D:\\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\3\4
\{My Computer}\D:\\de\winter_whether_advisory.zip\\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\4\0\0\0\0
\{My Computer}\D:\\de\winter_whether_advisory.zip\\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\4\0\0\0\1
\{My Computer}\E:\\Secret Project Data\technical review	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\0\1\0
\{My Computer}\E:\\Secret Project Data\proposal	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\0\1\1
\{My Computer}\E:\\Secret Project Data\progress	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\0\1\2
\{My Computer}\E:\\Secret Project Data\pricing decision	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\0\1\3
\{My Computer}\E:\\Secret Project Data\design	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\0\1\4
\{Control Panel}\??{CONTROL PANEL: Windows Update}	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0\0\0
\{Control Panel}\??{CONTROL PANEL: Personalization Control Panel}	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0\0\1
\{Control Panel}\??{CONTROL PANEL: Display}	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0\0\2
\{Control Panel}\??{CONTROL PANEL: System}	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0\0\3
\{Control Panel}\??{CONTROL PANEL: Taskbar (NotificationAreaIcons)	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0\0\4
\{Control Panel}\??{CONTROL PANEL: User Accounts}	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0\0\5
\{Control Panel}\??{CONTROL PANEL: Power Options}	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0\0\6
\{Control Panel}\??{CONTROL PANEL: Programs and Features}	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0\0\7
\{Shared Documents Folder (Users Files)}\{Downloads}	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\4\0
\{Shared Documents Folder (Users Files)}\Google Drive	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\4\1
\{Libraries}\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\7\0
\{Libraries}\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\7\1
\{Libraries}\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\7\2
\{Libraries}\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\7\3
\{Libraries}\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\7\4
\{Libraries}\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\7\5
\{Libraries}\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\7\6
\{Libraries}\??	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\7\7
\{Control Panel}	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\0
\{My Computer}	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1
\dO	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\2
\Download	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\3
\{Shared Documents Folder (Users Files)}	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\4
\{INTERNET_EXPLORER: dfd52841-23a3-2833-0400-000000002f28}	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\5
\S data	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\6
\{Libraries}	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\7
\{Network}	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\8
\{Recent Places}	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\9
\New folder	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\10
\temp	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\11
\{Recycle bin}	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\12
\{My Computer}\E:\	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\0
\{My Computer}\V:\	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\1
\{My Computer}\C:\	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\2
\{My Computer}\D:\	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\3
\{My Computer}\D:\	S-1-5-21-2425377081-3129163575-2985601102-1000_Classes\Local Settings\Software\Microsoft\Windows\Shell\BagMRU\1\4

Figure 4.50: Traversed Directories II

The next step is to take a look into the paths that were recovered previously and look for the files in those locations. To do this Autopsy was used, but this time in a Linux(Kali) based environment. Similar to the setup in Windows, a new case had to be setup after running Autopsy via the terminal and open the default url : http://localhost:9999/autopsy. Autopsy in a Linux based environment is accessed via a web browser. The first directory that was investigated was the desktop of the user under suspicion. Here there was a resignation letter that was recovered, the contents of which can be seen in the Figure 4.51 below. A .docx and a .xps version of the file was found which indicates that this file could have been printed from this location, as the suspect had an intention of leaving the organization.

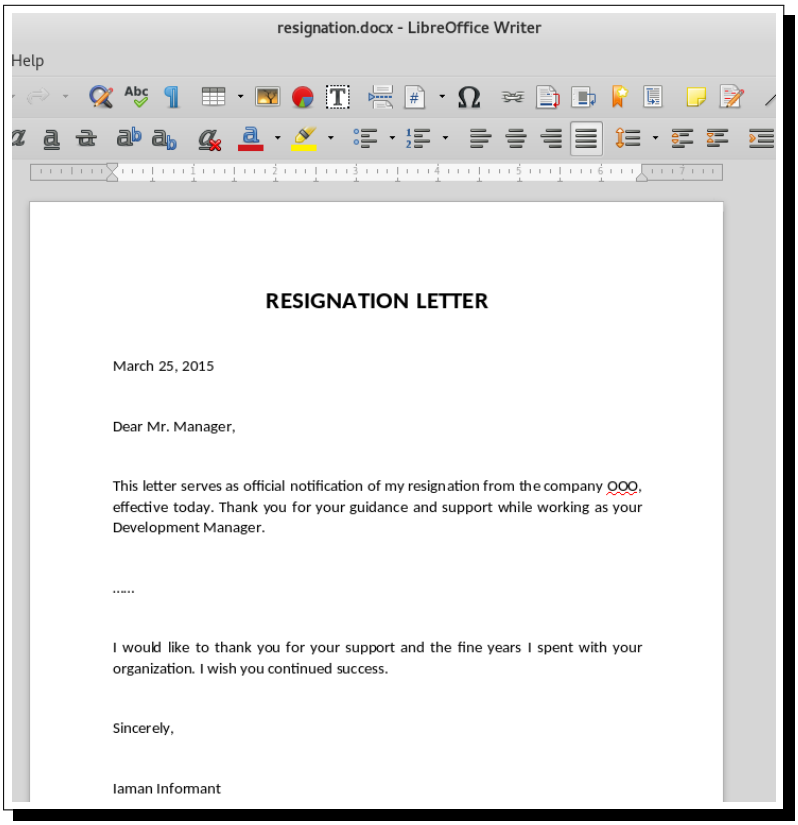


Figure 4.51: Recovery of suspect's resignation letter

The extraction of data via Autopsy is quite interesting and is discussed in this section. Upon accessing the metadata information of the listed file, the information that was gathered is shown in the Figure 4.52 below. The information that is obtained after analysis of the metadata is that the cluster size is 4096, the cluster starts at 826. The next question was how many clusters needed to be extracted? This information

was calculated by dividing the Allocated size by the Cluster size.

$$Number\,of\,Clusters(required\,for\,extraction) = \frac{Allocated\,size}{Cluster\,size}$$

which in this case translates to be:

$$\frac{12288}{4096} = 3$$

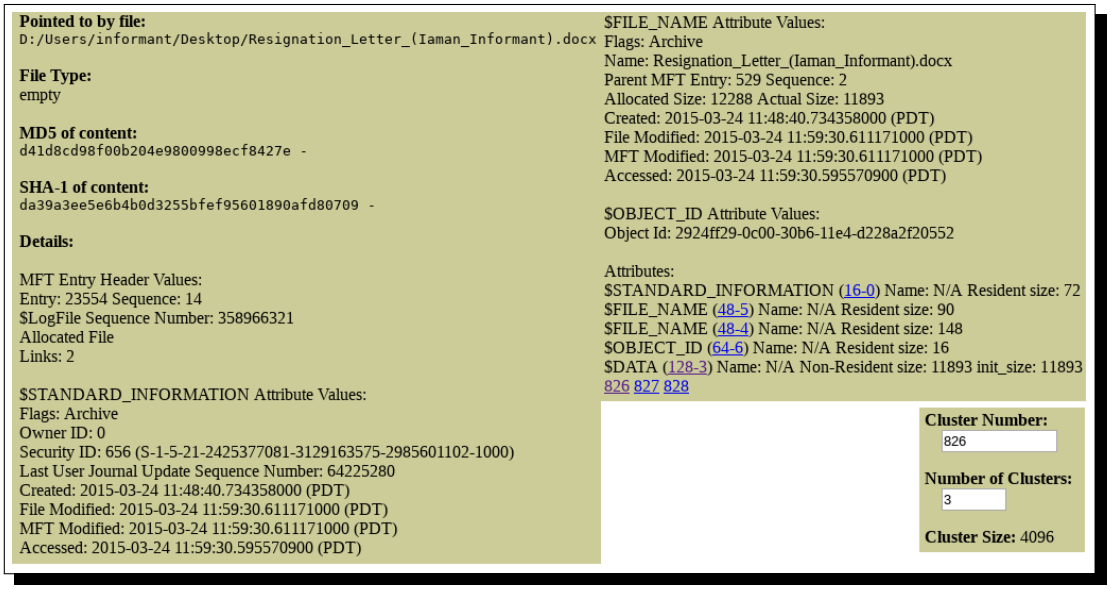


Figure 4.52: Extraction of data via cluster calculation in Autopsy[Kali]

Hence 3 sectors starting from 826 were extracted and the ASCII of the same raw output can be seen in the Figure 4.53 below.

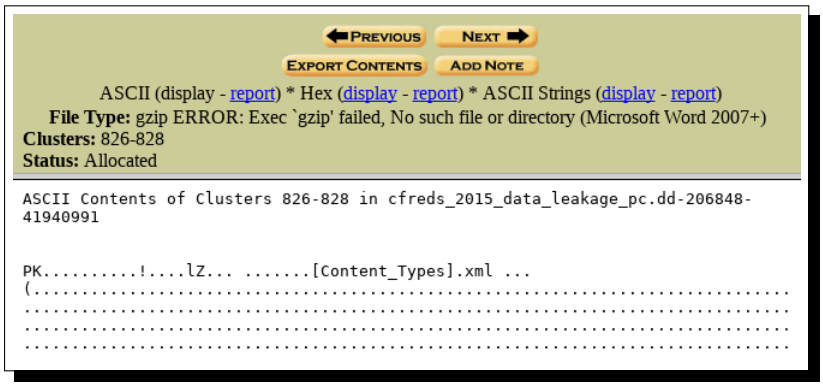


Figure 4.53: Header information extracted via ASCII from a deleted file

It was the same output containing the initials “PK” in the header information, which was found out in the earlier section to be a zipped file. However, the contents of the file were exported. Autopsy extracts all files in raw format which are obtained from certain sectors of a filesystem which are unallocated. Renaming this file directly to a .docx did not work, as found out earlier. Hence the raw file had to be renamed to a .zip file, its contents extracted and then re-zipped as seen in the Figure 4.54 below.

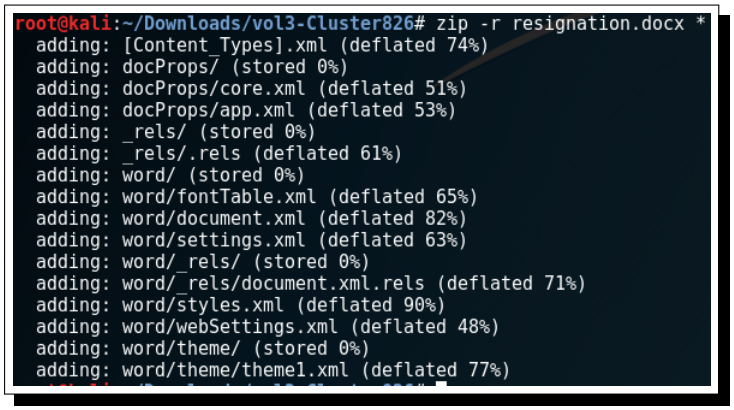


Figure 4.54: Re-zipping of the document to a readable format

Another discovery that was made via Sticky Note analysis of the Windows 7 image was at the location "D:/Users/informant/AppData/Roaming/Microsoft/Sticky Notes/StickyNotes.snt". A screenshot (Figure 4.55) of the acquired ASCII data from the .snt file is listed below which indicates that the suspect imagined that the leak would be undetected the day following the resignation.

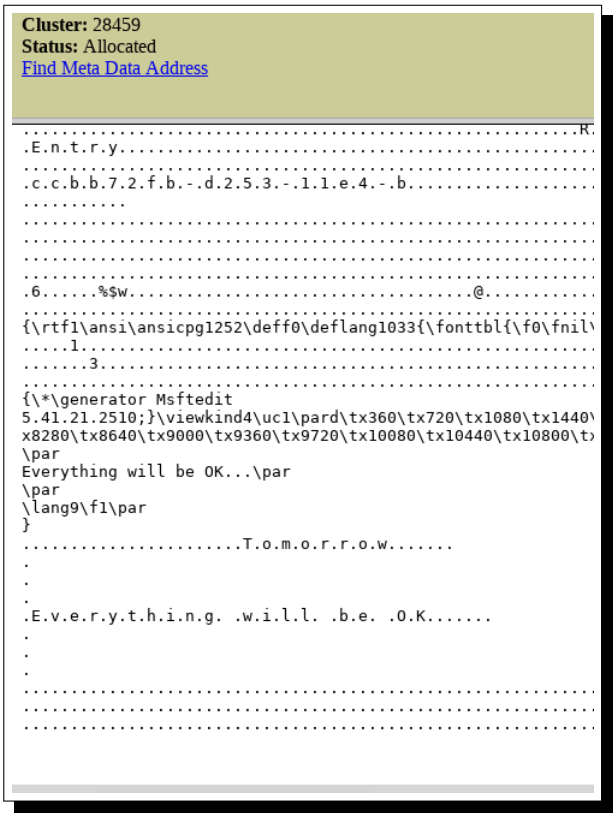


Figure 4.55: Details of Sticky Notes obtained from the image

4.2.7 Cloud and Database Forensics

As cloud integration with modern technology is advancing at a rapid pace, it is increasingly becoming important to delve into the forensics related to it. The reason why database and cloud forensics are integrated within a single section of this report is because they are intertwined. The primary cloud storage provider that will be under microscopic investigation here will be Google Drive and as it creates and holds information related to accounts and activity in a database, their inclusion in the same subsection was justified. Storage location of files related to Google Drive is : Users/username/AppData/Local/Google/Drive/user_default. Once this path is traversed the contents of the folder is depicted in the Figure 4.56 below.

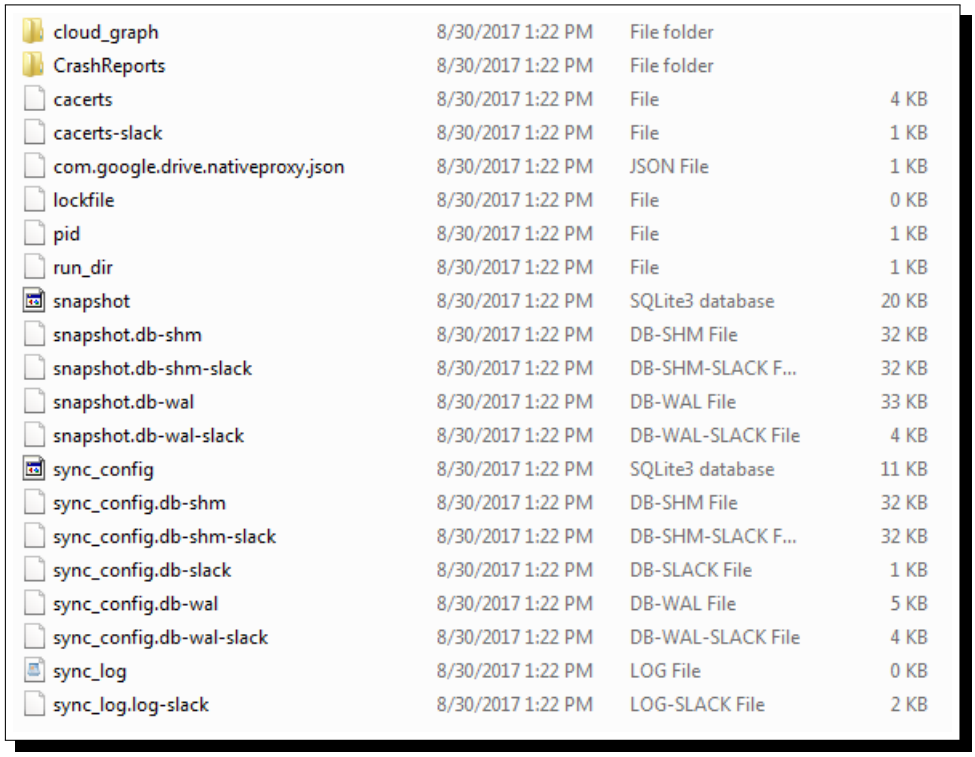
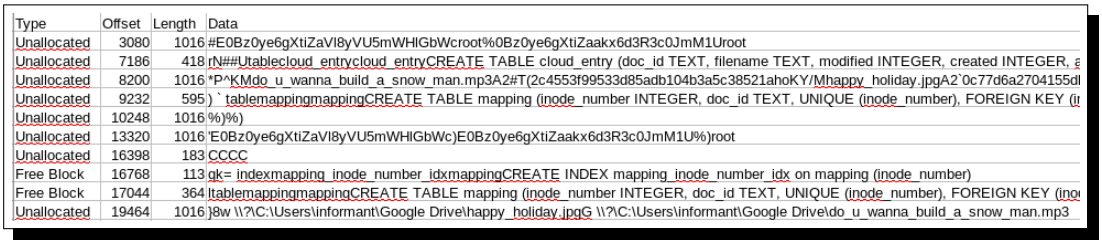


Figure 4.56: Files related to DB and Cloud that can undergo examination

The three major files that will be looked into from this directory is the two database files snapshot.db and sync_config.db, both of which are SQLite3 databases and the sync_log.log file which is a textual container of actions performed by the account owner. The first file that will be looked into snapshot.db file. This db file also had a .wal file which stands for Write Ahead Log(wal). A wal is generated whenever there is a change committed to the db. The wal identifies which page of the db needs

to be updated with the new data and adds this information as a new copy of that page using an identifier, to the wal file. This happens when a certain piece of information is removed from the db too, hence it was mentioned before that this occurs whenever there is a change to a sqlite3 db. A tool called SQLite-Deleted-Records-Parser [33] was used to recover the information that resided in this database. Using the command "python sqlparse_v1.3.py -f snapshot.db -o report.tsv" a tab separated file was generated from within the db and a section of the output of the file is as follows:



Type	Offset	Length	Data
Unallocated	3080	1016	#E0Bz0ye6gXtiZaVl8yVU5mWHlGbWcroot%0Bz0ye6gXtiZaakx6d3R3c0JmM1Uroot
Unallocated	7186	418	rN##Utablecloud_entrycloud_entryCREATE TABLE cloud_entry (doc_id TEXT, filename TEXT, modified INTEGER, created INTEGER, a
Unallocated	8200	1016	*P^KMdo_u_wanna_build_a_snow_man.mp3A2#T(2c4553f99533d85adb104b3a5c38521ahoKY/Mhappy_holiday.jpgA2`0c77d6a2704155dl
Unallocated	9232	595	) ` tablemappingmappingCREATE TABLE mapping (inode_number INTEGER, doc_id TEXT, UNIQUE (inode_number), FOREIGN KEY (i
Unallocated	10248	1016	%)%)
Unallocated	13320	1016	'E0Bz0ye6gXtiZaVl8yVU5mWHlGbWc)E0Bz0ye6gXtiZaakx6d3R3c0JmM1U%)root
Unallocated	16398	183	CCCC
Free Block	16768	113	qk= indexmapping_inode_number_idxmappingCREATE INDEX mapping_inode_number_idx on mapping (inode_number)
Free Block	17044	364	ltablemappingmappingCREATE TABLE mapping (inode_number INTEGER, doc_id TEXT, UNIQUE (inode_number), FOREIGN KEY (ino
Unallocated	19464	1016	}8w \\?\C:\Users\informant\Google Drive\happy_holiday.jpgG \\?\C:\Users\informant\Google Drive\do_u_wanna_build_a_snow_man.mp3

Figure 4.57: Output from SQLParse

An alternative to the aforementioned tool to recover deleted SQLite records is Oxygen Forensics SQLite Viewer [34], but an open source solution to this indicates the prosperous growth of the forensics community. As mentioned earlier about the existence of wal files, in addition it is important to mention recovery of db's without a proper decryption key is nearly impossible without the presence of extraneous(tabs, white spaces, non-printable characters) data. An SQLite db holds data in pages which are like leaf table binary trees, which contains cells. Its architecture drives the storage of new data(addition/removal/modification) towards the very end of the binary tree. Hence, due to the location of the data being at the end, the unallocated space is the area just preceding the fist cells' location and this very unallocated space holds the deleted data which is evident in the Figure 4.57 above.

Next, a data cleansing mechanism had to be applied on the file obtained above. As a sql query was discovered within the file, a new table was created data was extracted via use of additional the WAL file along with snapshot.db to a new spreadsheet, containing the fields that are highly relevant in this scenario. But in order to do that the creation and modified time of the two files uploaded to the cloud platform had to be found out. This information was uncovered from the sync_log.log file as is indicated in the Figure 4.58 below.

The creation and modified times for both the files were found out but they were not in human readable form. Hence they needed to be converted from decimal format

```
2015-03-23 16:32:53,092 -0400 INFO pid=2576 1836:CloudWatcher
common.cloud.cloud_watcher:1582 Built recent change dict:
{'0Bz0ye6gXtiZaVl8yVU5mWHlGbWc': [WorkerCreateCloudEvent
(entry=ImmutableCloudEntry(doc_id=
0Bz0ye6gXtiZaVl8yVU5mWHlGbWc,filename=do_u_wanna_build_a_snow_ma
n.mp3,modified=1422636560,created=
1427142765,acl_role=owner,doc_type=DocType.BLOB,removed=False,pa
rent_doc_ids=frozenset(['root']),child_doc_ids=frozenset
([]),size=6844294,checksum=
2c4553f99533d85adb104b3a5c38521a,change_stamp=0,server_mod_time=
1427142765,is_zombie=False,shared=False,resource_type=file,versi
on=None), doc_id=0Bz0ye6gXtiZaVl8yVU5mWHlGbWc, server_mod_time=
1427142765, version=None, _hash_code=889416526)],
'0Bz0ye6gXtiZaakx6d3R3c0JmM1U': [WorkerCreateCloudEvent
(entry=ImmutableCloudEntry(doc_id=
0Bz0ye6gXtiZaakx6d3R3c0JmM1U,filename=happy_holiday.jpg,modified
=1422563714,created=
1427142762,acl_role=owner,doc_type=DocType.BLOB,removed=False,pa
rent_doc_ids=frozenset(['root']),child_doc_ids=frozenset
([]),size=440517,checksum=
0c77d6a2704155dbfdf29817769b7478,change_stamp=0,server_mod_time=
1427142762,is_zombie=False,shared=False,resource_type=file,versi
on=None), doc_id=0Bz0ye6gXtiZaakx6d3R3c0JmM1U, server_mod_time=
1427142762, version=None, _hash_code=1184066048)]}
2015-03-23 16:32:53,092 -0400 INFO pid=2576 1836:CloudWatcher
common.cloud.cloud_watcher:197 HandleSyncConfigSettingsChange
generated events=set([])
```

Figure 4.58: Creation and modified times from sync_log.log in decimal format

to human readable form and the values are depicted in Figure 4.59.

	A	B	C	D	E	F	G
1	doc_id	0Bz0ye6gXtiZaVl8yVU5mWHlGbWc					
2	filename	do_u_wanna_build_a_snow_man.mp3					
3	created	GMT: Monday, March 23, 2015 8:32:45 PM (converted from decimal 1427142765)					
4	modified	GMT: Friday, January 30, 2015 4:49:20 PM (converted from decimal 1422636560)					
5							
6							
7	doc_id	0Bz0ye6gXtiZaakx6d3R3c0JmM1U					
8	filename	happy_holiday.jpg					
9	created	GMT: Monday, March 23, 2015 8:32:42 PM (converted from decimal 1427142762)					
10	modified	GMT: Thursday, January 29, 2015 8:35:14 PM (converted from decimal 1422563714)					

Figure 4.59: Conversion of times from sync_log.log to human readable format

Additional fields such as the size of the file, checksum and resource type can be further explored as future work on this data set. The data within file sync_config.db contained some additional information which pointed to the account that was used to upload the files and some additional files which have already been discussed about in an earlier section. The indexed information pertaining to the file is depicted in the Figure 4.60 below.

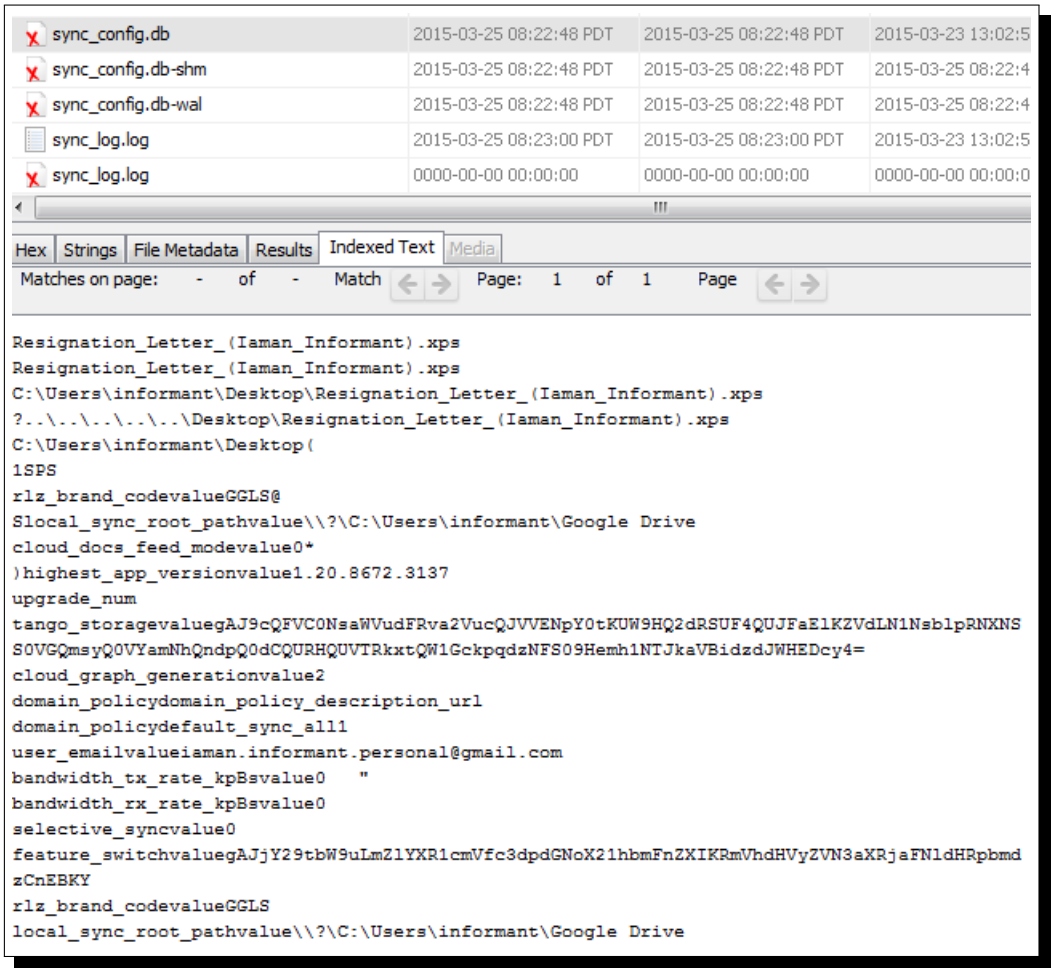


Figure 4.60: Details of account related to Google Drive via sync_config.db

The final file that needed to be analyzed and cleansed was the sync_log.log. This file contained timestamps of every action that related to Google Drive. The account in question was first verified to have logged in and additional actions of creation, modification and deletion of files in question is outlined in the Figure 4.61 below. It also contains the time when the suspect had logged out of Google Drive. It also depicts the login and logout time and all activities in between.

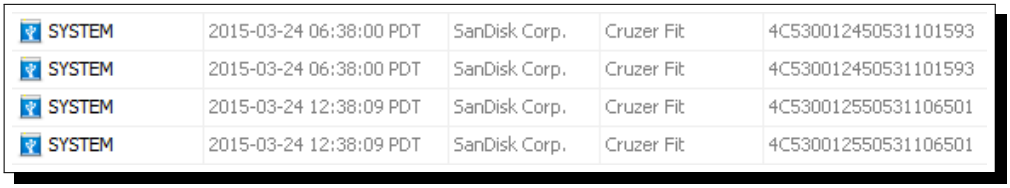
Figure 4.61: Proof of creation, modification and deletion of suspected files from sync_log.log

This marks the end of Chapter 4 which discussed various tools and methodologies used in examination of varied aspects related to the data set in use.

Chapter 5

Evaluation of Removable Devices

A couple of removable storage units were obtained from the suspect and imaged for analysis. These were a USB flash drive and a compact disc. This chapter will focus on these devices and information housed within them. The first thing that needs to be answered here is that whether the flash drive had interacted with the system in question. The first piece of related evidence can be gathered via Autopsy, under the devices tab, which proves that the USB flash drive matching the device id of the confiscated unit was connected four times to the system. Evidence of the same is indicated in the Figure 5.1 below. Hence it is confirmed that this was the device that was connected to this computer.

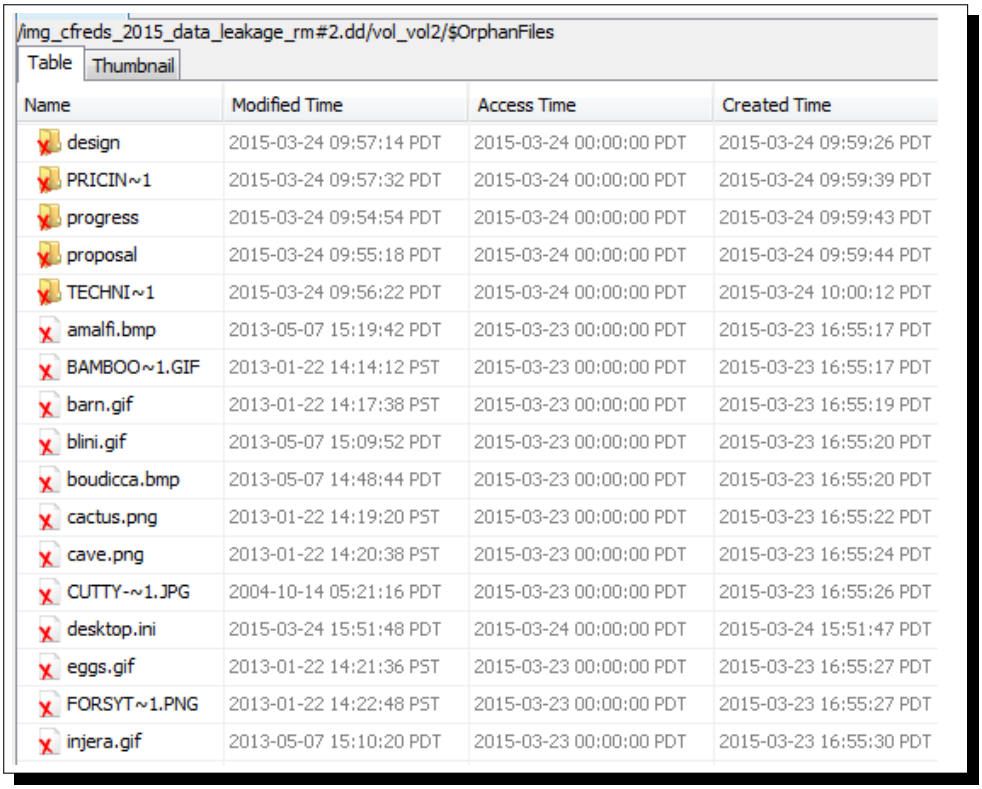


SYSTEM	2015-03-24 06:38:00 PDT	SanDisk Corp.	Cruzer Fit	4C530012450531101593
SYSTEM	2015-03-24 06:38:00 PDT	SanDisk Corp.	Cruzer Fit	4C530012450531101593
SYSTEM	2015-03-24 12:38:09 PDT	SanDisk Corp.	Cruzer Fit	4C530012550531106501
SYSTEM	2015-03-24 12:38:09 PDT	SanDisk Corp.	Cruzer Fit	4C530012550531106501

Figure 5.1: Matching Device IDs(with flash drive) connected to the machine

Additionally, the file structure of the imaged device proves the same. It is identical to the directory traversal paths as was found out in the previous chapter. Alongside that is also the contents of the CD-R, which houses the same files as was discovered during the directory traversal phase of the investigation. The file system of the flash drive is extracted via Autopsy in Windows, although the same can be and was captured via FTK Imager too and the results were identical.

The process to extract data from the CD-R was quite interesting as parsing the image files in neither Autopsy nor FTK Imager or Encase Imager gave an in-depth file



Name	Modified Time	Access Time	Created Time
design	2015-03-24 09:57:14 PDT	2015-03-24 00:00:00 PDT	2015-03-24 09:59:26 PDT
PRICIN~1	2015-03-24 09:57:32 PDT	2015-03-24 00:00:00 PDT	2015-03-24 09:59:39 PDT
progress	2015-03-24 09:54:54 PDT	2015-03-24 00:00:00 PDT	2015-03-24 09:59:43 PDT
proposal	2015-03-24 09:55:18 PDT	2015-03-24 00:00:00 PDT	2015-03-24 09:59:44 PDT
TECHNI~1	2015-03-24 09:56:22 PDT	2015-03-24 00:00:00 PDT	2015-03-24 10:00:12 PDT
amalfi.bmp	2013-05-07 15:19:42 PDT	2015-03-23 00:00:00 PDT	2015-03-23 16:55:17 PDT
BAMBOO~1.GIF	2013-01-22 14:14:12 PST	2015-03-23 00:00:00 PDT	2015-03-23 16:55:17 PDT
barn.gif	2013-01-22 14:17:38 PST	2015-03-23 00:00:00 PDT	2015-03-23 16:55:19 PDT
blini.gif	2013-05-07 15:09:52 PDT	2015-03-23 00:00:00 PDT	2015-03-23 16:55:20 PDT
boudicca.bmp	2013-05-07 14:48:44 PDT	2015-03-23 00:00:00 PDT	2015-03-23 16:55:20 PDT
cactus.png	2013-01-22 14:19:20 PST	2015-03-23 00:00:00 PDT	2015-03-23 16:55:22 PDT
cave.png	2013-01-22 14:20:38 PST	2015-03-23 00:00:00 PDT	2015-03-23 16:55:24 PDT
CUTTY-~1.JPG	2004-10-14 05:21:16 PDT	2015-03-23 00:00:00 PDT	2015-03-23 16:55:26 PDT
desktop.ini	2015-03-24 15:51:48 PDT	2015-03-24 00:00:00 PDT	2015-03-24 15:51:47 PDT
eggs.gif	2013-01-22 14:21:36 PST	2015-03-23 00:00:00 PDT	2015-03-23 16:55:27 PDT
FORSYT~1.PNG	2013-01-22 14:22:48 PST	2015-03-23 00:00:00 PDT	2015-03-23 16:55:27 PDT
injera.gif	2013-05-07 15:10:20 PDT	2015-03-23 00:00:00 PDT	2015-03-23 16:55:30 PDT

Figure 5.2: OrphanFiles recovered from flash drive

system information. An alternative approach was adopted as follows. A tool called PhotoRec [35] was used to recover files residing within the image as is indicated in the Figure 5.3 below.

PhotoRec is an open source data recovery tool used to recover a plethora of file-formats from recovery images varying across multiple file formats including FAT, exFAT, NTFS, HFS+ to name a few popular ones. These file formats save files in data blocks or clusters. The block number stays at a constant number of sectors within the disk after a partition has been formatted. Minimal data fragmentation is the ultimate goal of every storage unit and hence data is stored in an adjoining manner. Upon deletion of a file, its metadata information like creation date, modification time, size, name, location of block number is lost. The names of the deleted files may still reside within the filesystem, but its starting sector information cannot be recovered. This is where PhotoRec steps in and retrieves this information from the either ext2/3/4, or the superblock. This is interchangeably also referred to as the VBR(Volume Boot Record). It traverses through every sector of an image and once the block size is known and matched with the header information of the type of file that is needed to

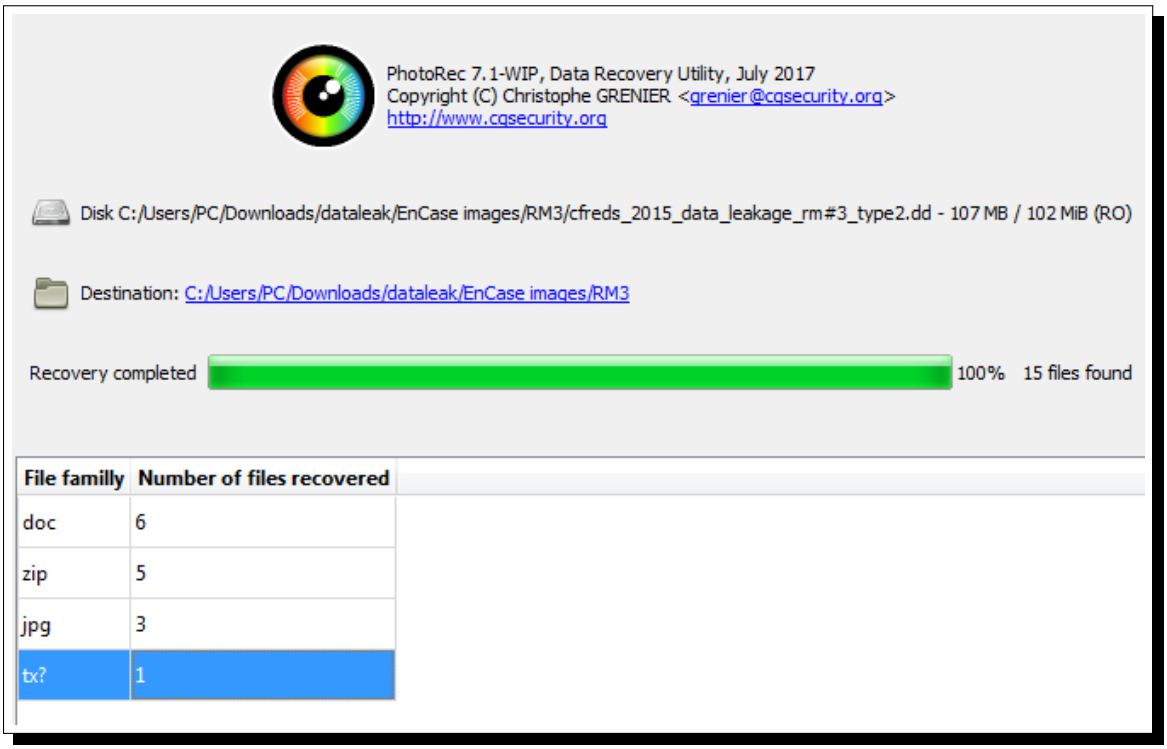


Figure 5.3: Use of PhotoRec to recover certain file types

be recovered and finally recovered. Sometimes data can be fragmented, in which case this tool can identify the original size of the file from its header data and the final extracted file is truncated to the correct size as that of the original. The presence of an unallocated partition in the CD-R image proves that it was wiped by conducting a format. This falls under the broad category of an attempt towards anti-forensics. Anti-forensics was also performed on the flash drive as the recovered files were located in the "OrphanFiles" folder which means that a quick format was conducted on it.

The next step was to take a look at the files that were recovered. The files recovered from the CD-R image are depicted in the Figure 5.4 below. Relevant files that are part of the data leak are highlighted. The information ranges from price analysis of the project, to its technical review, various stages of the progress reports, market shares and analysis, revision points, design details, technical reviews, which are all of extreme importance for an organization's success. A single page of a progress report is depicted in the Figure 5.5 below.

Similarly deleted files were recovered from the flash drive and it contained a lot more data than the CD-R. A Figure (5.6) depicting all the files is shown below. The files were extracted using the method portrayed in section 4.2.6. Files with the "PK"

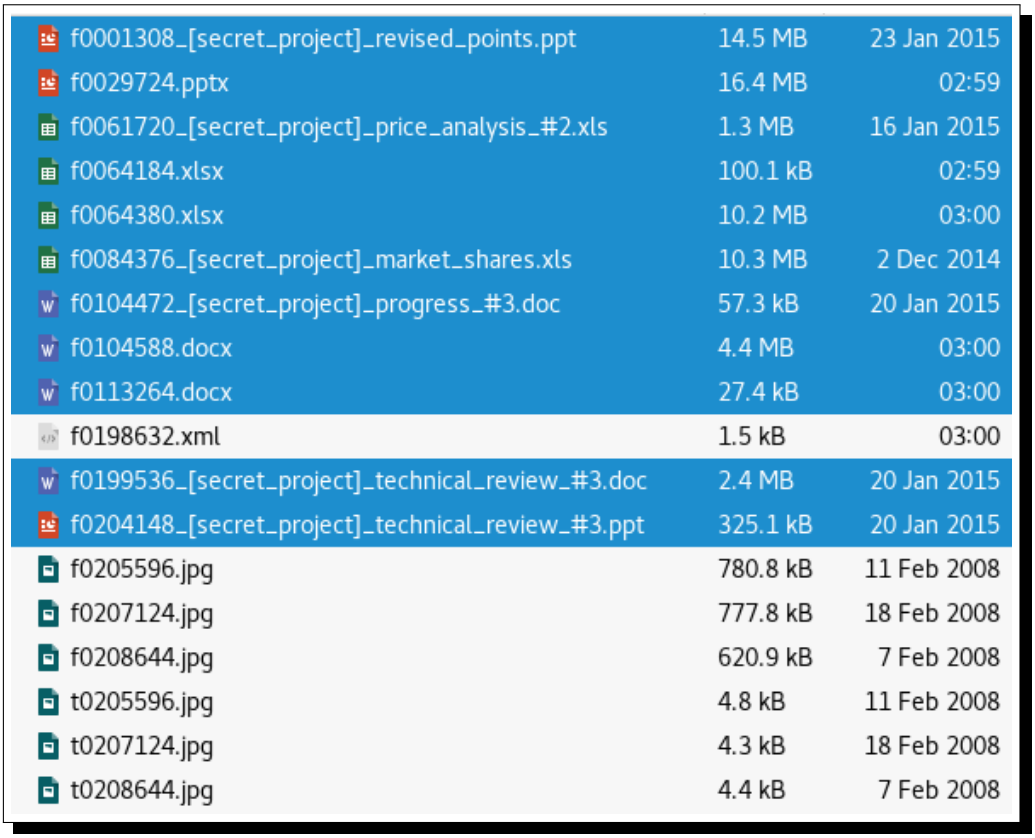
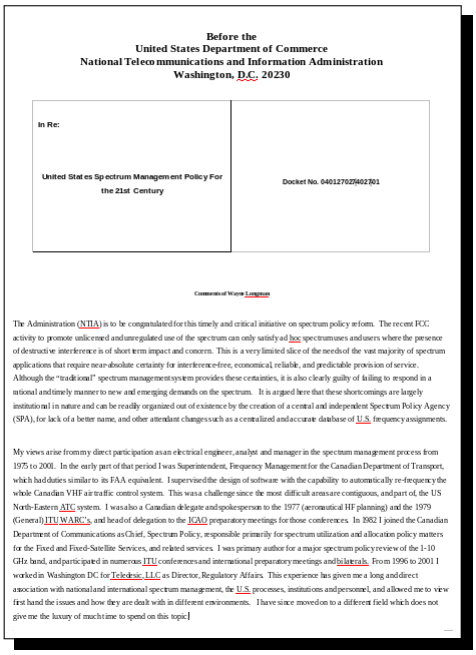


Figure 5.4: List of files recovered via PhotoRec

header were renamed to a .zip file and extracted. After extraction they were zipped back again according to the relevant file type(ppt, doc, xls, pptx, docx, xlsx).



Before the
United States Department of Commerce
National Telecommunications and Information Administration
Washington, D.C. 20230

In Re:

United States Spectrum Management Policy For the 21st Century

Docket No. 040127023402T01

Comments of Wayne Longman

The Administration (NTIA) is to be congratulated for this timely and critical initiative on spectrum policy reform. The recent FCC activity to promote unlicensed and unregulated use of the spectrum can only satisfy ad hoc spectrum uses and users where the presence of destructive interference is of short term impact and concern. This is a very limited slice of the needs of the vast majority of spectrum applications that require near-absolute certainty for interference-free, economical, reliable, and predictable provision of service. Although the "traditional" spectrum management system provides these certainties, it is also clearly guilty of failing to respond in a rational and timely manner to new and emerging demands on the spectrum. It is argued here that these shortcomings are largely institutional in nature and can be readily organized out of existence by the creation of a central and independent Spectrum Policy Agency (SPA), for lack of a better name, and other attendant changes such as a centralized and accurate database of U.S. frequency assignments.

My views arise from my direct participation as an electrical engineer, analyst and manager in the spectrum management process from 1975 to 2001. In the early part of that period I was Superintendent, Frequency Management for the Canadian Department of Transport, which had duties similar to its FAA equivalent. I supervised the design of software with the capability to automatically re-frequency the whole Canadian VHF air traffic control system. This was a challenge since the most difficult areas are contiguous, and part of, the US North-Eastern ATC system. I was also a Canadian delegate and spokesperson to the 1977 (aeronautical HF planning) and the 1979 (General) ITU WARC's, and head of delegation to the IGAO preparatory meetings for those conferences. In 1982 I joined the Canadian Department of Communications as Chief, Spectrum Policy, responsible primarily for spectrum utilization and allocation policy matters for the Fixed and Fixed-Satellite Services, and related services. I was primary author for a major spectrum policy review of the 1-10 GHz band, and participated in numerous ITU conferences and international preparatory meetings and bilaterals. From 1996 to 2001 I worked in Washington DC for Telesesic, LLC as Director, Regulatory Affairs. This experience has given me a long and direct association with national and international spectrum management, the U.S. processes, institutions and personnel, and allowed me to view first hand the issues and how they are dealt with in different environments. I have since moved on to a different field which does not give me the luxury of much time to spend on this topic.

Figure 5.5: Contents of a confidential file recovered via PhotoRec

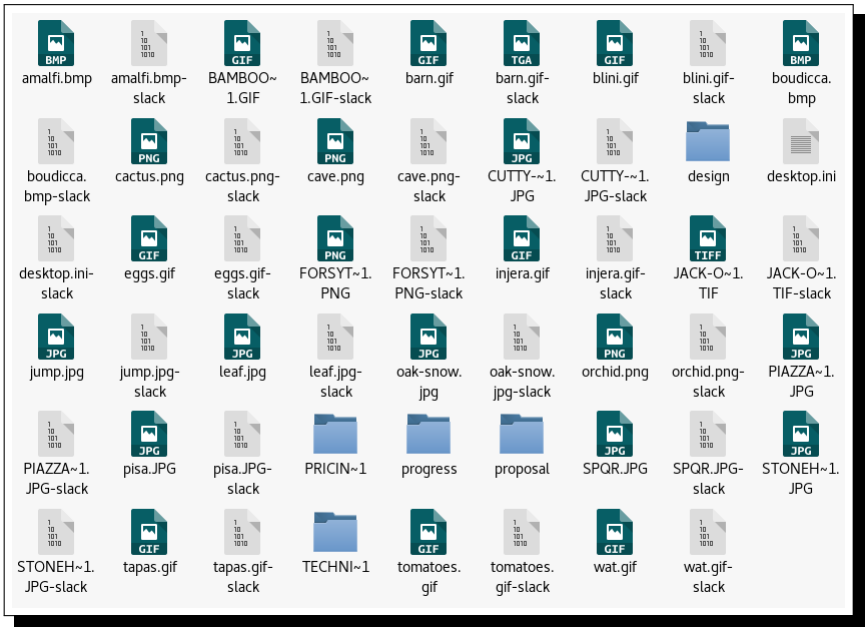


Figure 5.6: List of files/folders recovered from unallocated partition of compact disc

Proper re-packaging of an Open Office document related to the case can be seen in the Figure 5.7 below. Along with this, all other files were re-packaged and were found out to be related to the data leakage case, which further proves that the suspect tried

to smuggle data outside of the organization, not just via the use of a cloud storage provider but also by use of physical media such as a compact disk and an USB flash drive.

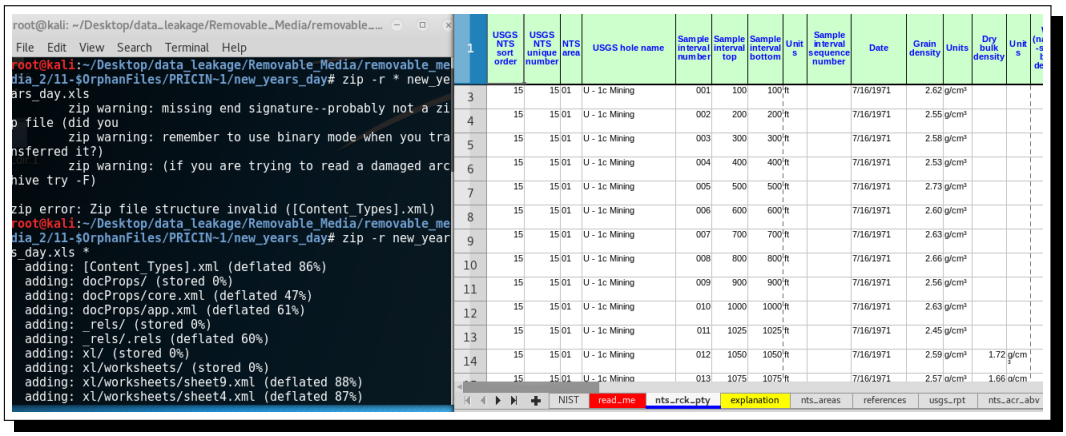


Figure 5.7: Re-zipping of a spreadsheet into readable format

Another example:

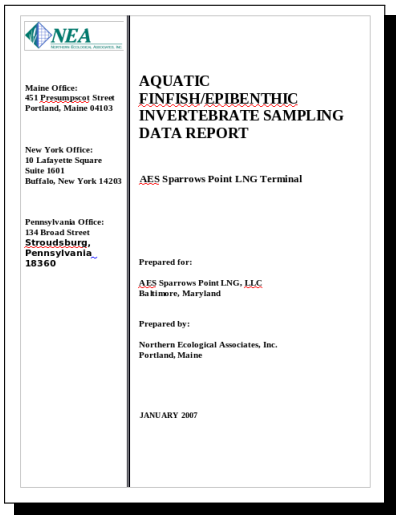


Figure 5.8: Contents of a confidential file recovered from the compact disc

A couple of considerations here could be the use of open source tools like Foremost [36] and Scalpel [37] to recover files of a certain format via use of carving techniques. Foremost has a built-in mechanism to extract popular file formats such as doc, pdf, xls, ppt etc. Scalpel was developed based upon Foremost, but it does not have support for built in file formats, so a configuration file needs to be created to extract files of a certain type.

Use of Scalpel related to this case demonstrated via screenshots:

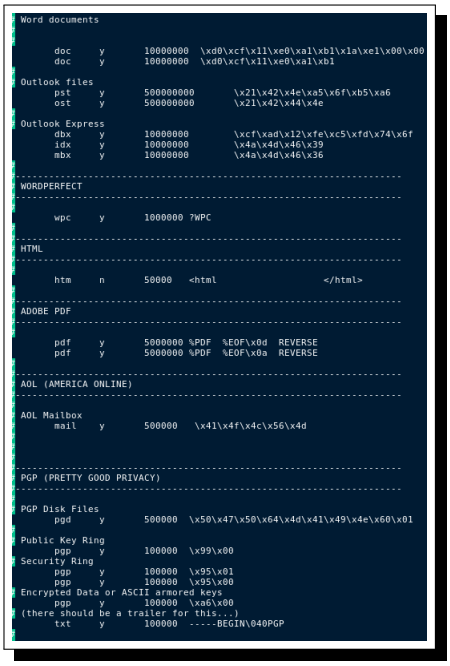


Figure 5.9: Making required changes to Scalpel configuration file

Figure 5.10: Running Scalpel

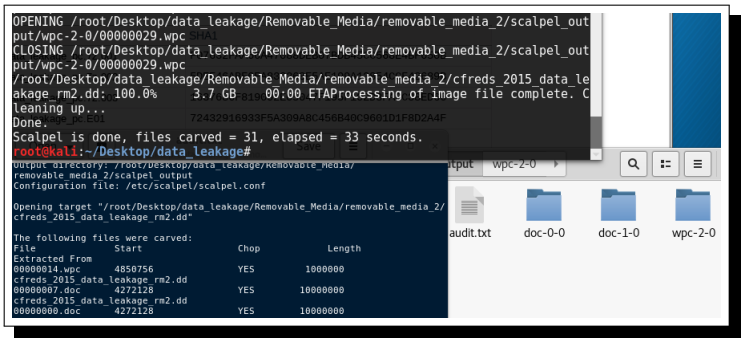


Figure 5.11: Scalpel Output

Use of Foremost related to this case demonstrated via screenshots:

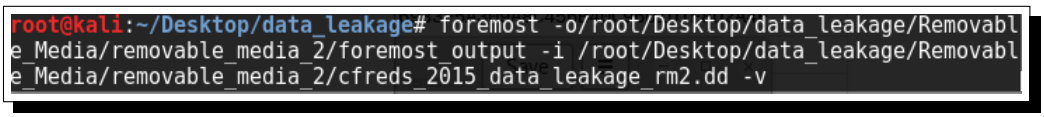


Figure 5.12: Running Foremost

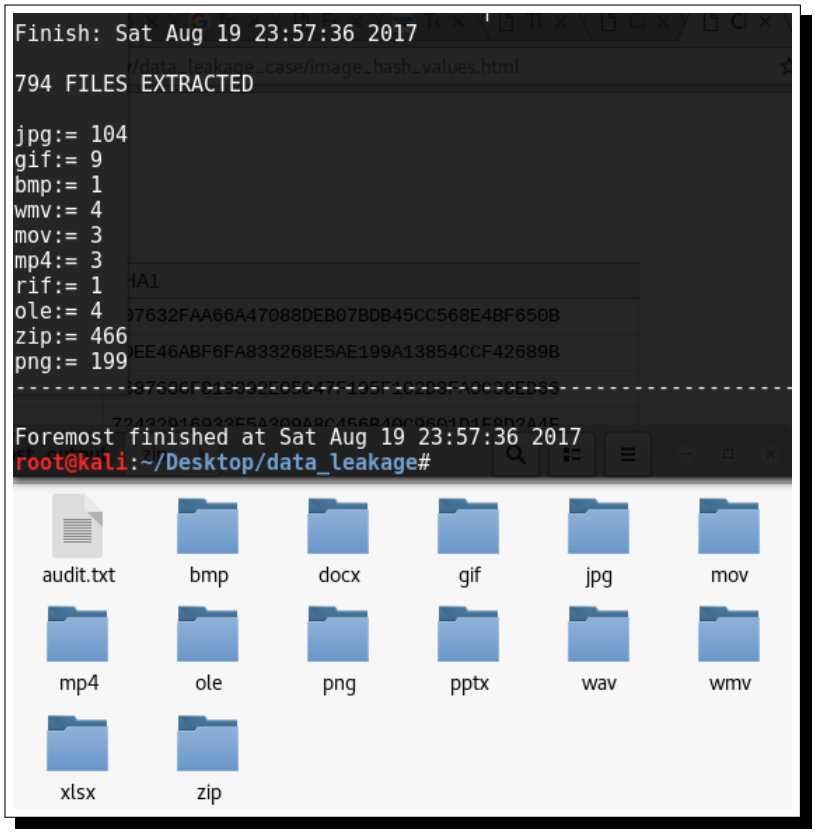


Figure 5.13: Output from Foremost

The final consideration in this section is to convert recovered files to plain text format before opening them. Sometimes attackers can plant a payload within a .pdf, .xls, .ppt or .doc file which can trigger a series of events that can cause harm to the system or even to the network which it is connected to. Payloads can also be hidden within image files, hence they should be opened with extreme caution. Their header information should be checked first before opening it via an image editor, or just a preview of the same is a better approach to take rather than opening the file itself. But in case of any text based document types mentioned before, the best approach is to convert the files into a plain text format before looking into their contents. To

depict this, a test scenario is demonstrated. Let the case be such that two pdf files were extracted from an image pertaining to a certain case and one of those files had important information in it but also housed a payload that was injected via Metasploit [38]. A simple pdf to text parser can be used to achieve this. The XPDF package [39] houses a pdftotext.cc file written in C++ which converts a pdf file to plain text and in cases like this where there are a large number of files being dealt with, parsing each file individually can become quite repetitive and consume additional time. The following script was used to automate this process and by just running this shell script once every .pdf file in a folder was converted to individual .txt file.

```
#BASH
for file in *.pdf;
do pdftotext "$file" "$file.txt";
done
```

After running the shell script all the pdf files in the folder are converted to text as depicted in the Figure 5.14 below.

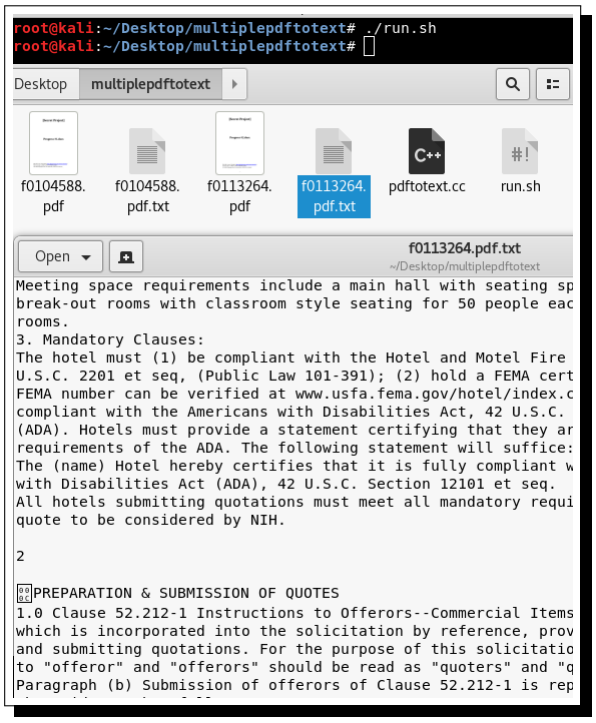


Figure 5.14: Batch conversion mechanism of .pdf to .txt

This marks the end of Chapter 5 which discussed the investigation of removable devices that were confiscated from the suspect and also provided an insight into data

retrieval techniques from external devices that underwent data deletion as a result of a quick format.

Chapter 6

Conclusion

The primary motive behind this project was to demonstrate the use of tools and mechanisms used in the real world in the field of digital forensics. Not only has the use of each tool/software been depicted, but its working methodologies have also been explained which helps the reader to gather a firm understanding of the topic area. Chapter 1 outlined the purpose of the project with a quick introduction about the data set in use. This helped the reader to develop an idea about the scope of this particular data set. Chapter 2 answered the question behind the selection of this particular data set and also detailed information about the various imaging techniques that were used to obtain it. Chapter 3 outlined the approach and the model that was adopted in tackling a forensic investigation of this nature. It also discussed the major steps taken in each phase during the analysis. Chapter 4 and 5 are where the majority of the experiments are conducted along with validation and integrity testing of data, Windows Registry analysis and forensics related to web browsers, email clients, cloud providers and databases. These 2 chapters depicted how a suspect's actions can be tracked in a forensic investigation and how to uncover information which could potentially be used to prove that the data leak was indeed conducted by the suspected internal employee of the organization. Detailed steps were included to provide an in-depth insight into the on-goings of a realistic life-like digital forensic investigation.

Bibliography

[1] Hacks and Data Breaches. URL : http://www.wired.co.uk/article/hacks-data-breaches-2017

[2] Reported versus Unreported Attacks.
URL : https://www.helpnetsecurity.com/2010/01/27/many-cybercrimes-go-unreported/

[3] National Institute of Standards and Technology. URL : https://www.nist.gov/

[4] Computer Forensic Reference Data Sets (CFReDS).
URL : https://www.cfreds.nist.gov/

[5] Computer Forensics Tool Testing (CFTT) project.
URL : https://www.cftt.nist.gov/

[6] DD Imaging Format. URL : http://www.forensicswiki.org/wiki/Dd

[7] E01 Forensic Imaging Format.
URL : https://whereismydata.wordpress.com/2008/08/10/e01-files/

[8] Tableau USB Bridge T8-R2.
URL : https://www.guidancesoftware.com/tableau/hardware//t8u

[9] Guidance Software. URL : https://www.guidancesoftware.com/

[10] bchunk. URL : http://he.fi/bchunk/

[11] The Systematic Digital Forensic Investigation Model (SRDFIM).
URL : https://en.wikipedia.org/wiki/Digital_forensic_process

[12] MD5 Checksum. URL : https://en.wikipedia.org/wiki/MD5

[13] SHA-1(Secure Hash Algorithm 1) checksum.
URL : https://en.wikipedia.org/wiki/SHA-1

[14] Kali Linux - A Digital Forensic Distribution.
URL : https://en.wikipedia.org/wiki/Kali_Linux

[15] Digital Forensics toolset options in Kali.
URL : https://tools.kali.org/tools-listing

[16] The Sleuth Kit and Autopsy. URL : https://www.sleuthkit.org/

[17] Security Account Manager in Windows 7.
URL : https://en.wikipedia.org/wiki/Security_Account_Manager

[18] John the Ripper. URL : https://en.wikipedia.org/wiki/John_the_Ripper

[19] Rainbow Tables to decrypt password hashes via chaining mechanism.
URL : http://kestas.kuliukas.com/RainbowTables/

[20] RegRipper. URL : https://tools.kali.org/forensics/regripper

[21] samparse extension used for analysis of SAM in Windows environment.
URL : https://github.com/appliedsec/forensicscanner/blob/master/plugins/samparse.pl

[22] Google Drive. URL : https://www.google.com/drive/

[23] Apple iCloud. URL : https://www.icloud.com/

[24] Eraser. URL : https://eraser.heidi.ie/

[25] CCleaner. URL : https://www.piriform.com/ccleaner

[26] Chrome Forensics Analysis via use of open source tool called Hindsight. URL :
https://github.com/obsidianforensics/hindsight

[27] BrowsingHistoryView. URL : http://www.nirsoft.net/utils/

[28] Data transfer via transmission of heat.
URL : https://www.wired.com/2015/03/stealing-data-computers-using-heat/

[29] PST Viewer. URL : http://www.pstviewer.com/

[30] PKZIP. URL : https://en.wikipedia.org/wiki/PKZIP

[31] ShellBags Explorer. URL : https://ericzimmerman.github.io/

[32] shellbags. URL : https://github.com/williballenthin/shellbags

[33] SQLite-Deleted-Records-Parser. URL : https://github.com/mdegrazia/SQLite-Deleted-Records-Parser

[34] Oxygen Forensics SQLite Viewer.
URL : http://www.oxygen-forensic.com/en/features/analyst/data-viewers/sqlite-viewer

[35] PhotoRec. URL : http://www.cgsecurity.org/wiki/PhotoRec

[36] Foremost. URL : http://www.forensicswiki.org/wiki/Foremost

[37] Scalpel. URL : http://www.forensicswiki.org/wiki/Scalpel

[38] Metasploit. URL : https://www.metasploit.com/

[39] XPDF. URL : http://gnuwin32.sourceforge.net/packages/xpdf.htm