Practical Digital Forensics

Forensic Lab Setup, Evidence Analysis, and Structured Investigation Across Windows, Mobile, Browser, HDD, and Memory

Dr. Akashdeep Bhardwaj
Keshav Kaushik

bpb

Practical Digital Forensics

Forensic Lab Setup, Evidence Analysis, and Structured Investigation Across Windows, Mobile, Browser, HDD, and Memory

Dr. Akashdeep Bhardwaj

Keshav Kaushik

www.bpbonline.com

First published: 2023

Published by BPB Online

WeWork

119 Marylebone Road

London NW1 5PU

UK | UAE | INDIA | SINGAPORE

ISBN 978-93-5551-150-8

www.bpbonline.com

Dedicated to

Akashdeep Bharadwaj's beloved Parents:

Sh. Kailash Chand Bhardwaj, Smt. Usha Bhardwaj
My wife ***Archana*** *and My Daughter* ***Raavi***

&

Keshav Kaushik's beloved Parents:

Sh. Vijay Kaushik, Smt. Saroj Kaushik
My wife ***Priyanka*** *and My Daughter* ***Kashvi***

About the Authors

Dr. Akashdeep Bhardwaj is currently working as Professor (Cyber Security & Digital Forensics) at University of Petroleum & Energy Studies (UPES), Dehradun, India. He is an eminent industry expert with over 27 years of experience in Cybersecurity, Digital Forensics and IT Management Operations. Dr. Akashdeep mentors national & international graduate, masters and doctoral students and leads several Cybersecurity projects, including Cyber CoE. Dr. Akashdeep holds a post-doctoral in Computer Science along with over 20 IT industry certifications.

Dr. Akashdeep has published over 100 research papers, chapters, books and patent. Dr. Akashdeep has worked as Technology Leader for various multinational organizations and is certified in Cybersecurity, Compliance Audits, Information Security, Microsoft, Cisco and VMware technologies.

Keshav Kaushik is an experienced educator with over eight years of teaching and research experience in Cybersecurity, Digital Forensics, and the Internet of Things. He is working as an Assistant Professor (Senior Scale) in the School of Computer Science at the University of Petroleum and Energy Studies, Dehradun, India. He has published 65+ research papers in International Journals and has presented at reputed International Conferences. He is a Certified Ethical Hacker (CEH) v11, CQI and IRCA Certified ISO/IEC 27001:2013 Lead Auditor, Quick Heal Academy Certified Cyber Security Professional (QCSP), and IBM Cybersecurity Analyst. He acted as a keynote speaker and delivered 50+ professional talks on various national and international platforms. He has edited over ten books with reputed international publishers like Springer, Taylor and Francis, IGI Global, Bentham Science, etc. He has chaired various special sessions at international conferences and also served as a reviewer in peer-reviewed journals and conferences.

About the Reviewer

Dylan Waggy is a Senior Advisor, Incident Response Analyst for an American multinational technology company. He has a Bachelors of Science in Digital Forensics, a Minor in Criminal Justice, and more than 7 years of experience within the Investigative and Incident Response space. He holds the certifications: Certified Forensic Computer Examiner (CFCE), Certified Forensic Analyst (GCFA), and Reverse Engineering Malware (GREM). Dylan has completed over 300 federal crime cases assisting Law Enforcement entities with their Digital Forensic needs. He has helped in building Digital Forensics and Incident Response teams with multiple Fortune 500 companies. He takes pride in knowing he can help, proactively and retroactively, protect a company from threats, both internally and externally.

Acknowledgement

There are a few people I want to thank for the continued and ongoing support they have given me during the writing of this book. First and foremost, I would like to thank my parents for continuously encouraging me for writing the book — I could have never completed this book without their support.

I am grateful to the course and the companies which gave me support throughout the learning process of web scraping and it is very crucial to learn the tools related to web scraping.

Thank you for all the hidden support provided.

My gratitude is towards the team at BPB Publications for being supportive enough to provide me quite a long time to finish the first part of the book and also allow me to publish the book in multiple parts, since image processing, being a vast and very active area of research, it was impossible to deep-dive into different class of problems in a single book, especially by not making it too voluminous.

Preface

This book dives into the basics to advanced technical details of analyzing postmortem forensic images of Windows and Linux systems which have been misused, abused, or the target of malicious attacks. It helps forensic investigators locate and analyze digital evidence found on Linux desktops, servers, and IoT devices. Throughout the book, you learn how to identify digital artifacts which may be of interest to an investigation, draw logical conclusions, and reconstruct past activity from incidents. You'll learn how Linux works from a digital forensics and investigation perspective, and how to interpret evidence from Linux environments. The techniques shown are intended to be independent of the forensic analysis platforms and tools used.

This book covers the full life cycle of conducting a mobile and computer digital forensic examination, including planning and performing an investigation as well as report writing and testifying. Case reviews in corporate, civil, and criminal situations are also described from both prosecution and defense perspectives. This book draws from years of experience in local, state, federal, and international environments and highlights the challenges inherent in deficient cyber security practices.

Topics include the importance of following the scientific method and verification, legal and ethical issues, planning an investigation (including tools and techniques), incident response, case project management and authorization, social media and internet, cloud, anti-forensics, link and visual analysis, and psychological considerations. The book is a valuable resource for the academic environment, law enforcement, those in the legal profession, and those working in the cyber security field. Case reviews include cyber security breaches, anti-forensic challenges, child exploitation, and social media investigations.

This book teaches you how to conduct examinations by explaining what digital forensics is, the methodologies used, key technical concepts and the tools needed to perform examinations. Readers will also learn how to collect evidence, document the scene, and recover deleted data. This is the only resource your students need to get a jump-start into digital forensics investigations. This book guide has been written to provide deep insights into Digital Forensics. This book is organized into **11 chapters**. After an introduction to the basics of digital forensics, the book proceeds with a discussion of key technical concepts, hard disks and file systems. Setting up of a digital forensic lab, acquiring and analyzing digital evidence is explained using concepts as well as hands-on sessions to replicate for the learners. The practical sessions cover digital forensic labs tools, collecting evidence, Windows system, hard disk, network, memory, email, and web browser artefacts, as well as anti-forensics. The book concludes by outlining challenges and concerns associated with digital forensics and writing forensic reports to be used by legal agencies.

Chapter 1, The scope of this chapter is to introduce the history of digital forensics and explain the importance of electronic and digital evidence for solving cybercrime and investigation problems. This chapter explains the digital forensic terminology, goals of forensic analysis, the digital forensics process, and challenges for digital forensics.

Chapter 2, Conducting a digital forensics investigation requires a thorough understanding of some of the main technical concepts of computing. Knowing how data is stored in computers, number theory, how digital files are structured, and the types of storage units and the difference between them are essential

areas to know how to locate and handle digital evidence. This chapter will cover those basic concepts.

Chapter 3, This chapter covers the computer hardware, logical disk structures, booting process, file systems that are often involved during forensic investigations to gather and analyse digital evidence.

Chapter 4, In-house digital forensics analysts usually work closely with law enforcement agencies to solve cases related to their businesses. Having an in-house digital forensics lab in today's digital age is a great investment for any company which values its data assets; however, this comes with a cost.

Chapter 5, The main task of a computer forensics investigator is to acquire and analyze computing devices' memory images. In this chapter, we will cover techniques and tools to create forensics images from both running systems (volatile memory, RAM) and hard drives (HDD, SSD, flash thumb, and any similar digital storage media).

Chapter 6, All analysis work should be conducted on the forensics image only; forensic examiners should not interfere with the original suspect device to avoid damaging original evidence accidentally and thus making the entire investigation useless in a court of law. This chapter analyzes the acquired images for gathering interesting artifacts and critical leads.

Chapter 7, In this chapter, we continue our digital analysis and cover how to build in-depth digital forensics knowledge and analyze different Microsoft Windows operating systems by knowing where forensics artifacts can be found and how we can analyze them to solve the digital crime cases at hand.

Chapter 8, Internet applications already installed on Windows can give important information about user actions performed previously on his/her computer. For instance, a web browser is the only way to access the Internet, and criminals are using it to commit crimes related to the Internet or to target other users online. Internet users use web browsers to socialize, purchase online items, or to send e-mails and browse the web contents, among other things. This fact makes web browsers the preferred target for malicious actors to steal confidential information like account credentials.

Chapter 9, E-mails have become the primary means of communications in today's digital age; for instance, it is rare to see a person who owns a computer, smartphone, or tablet without having an active e-mail account.

Chapter 10, Antiforensics techniques are concerned with making digital forensics investigations very difficult to conduct and time consuming; they focus on frustrating digital forensics examiners through destroying digital evidence, hiding incriminating information so examiners cannot notice it, and manipulating evidence files to mislead the investigation and take it in the wrong direction. Reporting is a key issue in any type of investigation; public investigation that involves courts usually needs more technical details and comprehensive descriptions of the methodology used to acquire and analyze the digital evidence.

Chapter 11, This chapter presents hands-on labs which the learners need to study and replicate in their systems. To implement these labs, Kali Linux (the latest version) or Windows 10/11 operating systems are required to be run in a Virtual Environment. Do not install the Digital Forensics tools and run these on the main physical machine.

Coloured Images

Please follow the link to download the
Coloured Images of the book:

https://rebrand.ly/56ef48

We have code bundles from our rich catalogue of books and videos available at **https://github.com/bpbpublications**. Check them out!

Errata

We take immense pride in our work at BPB Publications and follow best practices to ensure the accuracy of our content to provide with an indulging reading experience to our subscribers. Our readers are our mirrors, and we use their inputs to reflect and improve upon human errors, if any, that may have occurred during the publishing processes involved. To let us maintain the quality and help us reach out to any readers who might be having difficulties due to any unforeseen errors, please write to us at :

errata@bpbonline.com

Your support, suggestions and feedbacks are highly appreciated by the BPB Publications' Family.

Piracy

If you come across any illegal copies of our works in any form on the internet, we would be grateful if you would provide us with the location address or website name. Please contact us at **business@bpbonline.com** with a link to the material.

If you are interested in becoming an author

If there is a topic that you have expertise in, and you are interested in either writing or contributing to a book, please visit **www.bpbonline.com**. We have worked with thousands of developers and tech professionals, just like you, to help them share their insights with the global tech community. You can make a general application, apply for a specific hot topic that we are recruiting an author for, or submit your own idea.

Reviews

Please leave a review. Once you have read and used this book, why not leave a review on the site that you purchased it from? Potential readers can then see and use your unbiased opinion to make purchase decisions. We at BPB can understand what you think about our products, and our authors can see your feedback on their book. Thank you!

For more information about BPB, please visit **www.bpbonline.com**.

Table of Contents

CHAPTER 1

Introduction to Digital Forensics

Introduction

As the world continues to digitize, both the public and private sectors will become dependent upon technology to do business. Organizations use technology to improve productivity, reduce internal and external operating costs, improve data security, and extend business capabilities in today's information age. The key to realizing these benefits is to digitally transform all aspects of work, particularly through data stored digitally instead of using paper documents. Individuals have also become increasingly reliant on technology in their everyday lives; nearly everything they do now involves technology in some capacity. The fast transition to the digital era has been associated with an increase in cybercrime. Cybercrime losses are expected to surpass $6 trillion yearly by 2022, according to cyber security ventures. According to the same report, there may be 6 billion internet users by 2023 (75% of the anticipated world population of 8 billion), resulting in a massive volume of digital data being generated every second.

Structure

In this chapter, we will cover the following topics:

- Defining digital forensics and goals
- Defining cybercrime and cybercrime sources
- Computers in cybercrimes
- Digital forensics categories
- Forensic data analysis
- Digital forensic users
- Investigation types
- Forensics readiness
- Digital evidence types
- Electronic evidence location
- Chain of custody
- Examination process

Objectives

The reader will get to understand digital forensics, goals, cybercrime, and the sources involved as well as computers used in cybercrimes. This chapter discusses the different types of digital forensics

categories such as mobile, network, database, and analysis of Forensic Data and users such as law enforcement, civil litigation, intelligence, and counterintelligence agencies. Further, the various investigations and evidence types, as well as Forensics Readiness, is discussed, including user, machine, and network-created Data, Chain of Custody, and the Forensic examination process.

Defining digital forensics

Digital forensics is a branch of forensic science that uses scientific understanding to acquire, evaluate, record, and present digital evidence related to computer crime in court. The main goal is to figure out what happened, when it happened, and who did it. The term "*digital forensics*" is a catch-all word for computer forensics or, more recently, "*cyber forensics*." These investigations include user laptops, computers, mobile phones, network devices, Webcams, tablets, camcorders, IoT and smart home devices, and storage media such as USB drives, CD/DVD, SD cards, and tapes, among other digital systems and devices that can send, receive, and store digital data.

Data breaches, phishing, ransomware, DoS assaults, and SQL Injunctions are all examples of cyberattacks on digital systems that may be investigated using digital forensics. Cyberespionage or adversarial assaults that compromise accounts and services, unauthorized system and network access, or other associated cyberattacks that cause commercial or reputational harm are all included in this category. Conducting a computer forensic investigation necessitates adhering to certain guidelines that can withstand cross-examination in court. This includes gathering data (both static and volatile) in a forensically sound manner, assessing data using court-approved forensics tools, sifting through the data to locate evidence, and finally, presenting conclusions to the court in an official report. If these procedures are not followed correctly, we risk damaging or erasing digital evidence, rendering it inadmissible in court.

Digital forensics is a relatively new profession in the cybersecurity domain that is becoming increasingly important as the number of crimes and unlawful actions in cyberspace increases. In comparison to conventional forensic science (blood tests, DNA profiling, or fingerprinting), digital forensics is a young science; the fact that it interacts with rapid changes in the computing ecosystem around us and reaches other domains (such as the judicial process, law enforcement, management consulting, information technology, and the borderless scope of the internet), makes it a difficult field that requires constant development of its foes.

Digital forensics goals

The basic goal of digital forensics is to investigate crimes committed with computer systems that store and processes digital data and to extract forensic' digital evidence to present in court. This is achieved in the following ways using digital forensics. Locating and preserving legal evidence on computer devices in a way that is acceptable in a court of law.

- Follow court-approved technological methods to preserve and recover evidence.
- Assigning responsibility for an activity to the person who initiated it.
- Determining data breaches inside a company.
- Identifying the extent of any damage that may occur as a result of a data breach.
- Compiling the findings into a formal report that may be submitted in court.
- Providing expert evidence in court as a guide.

Defining cybercrime

Any illegal activity carried out on a computer or via a computer network, such as the internet, is referred to as cybercrime. According to the US Department of Justice, cybercrime is defined as any unlawful behavior done against or with the use of a computer or computer network. The fundamental motivation for cybercrime is financial gain (*for example*: spreading malware to steal access codes to bank accounts). However, different motives drive a significant portion of cybercrime, including disrupting service (for example, DDoS attacks to shut down a target organization's services), stealing confidential data (for example, consumer data and medical information), cyber espionage (corporate trade and military secrets), or illegally exchanging copyrighted materials.

Sources of cybercrime

Insider threats and external attacks are the two primary sources of cybercrime.

- **Insider threats**: Since they might go unnoticed for a long period, this is the most significant cyber risk threatening enterprises today. Employees—or other persons working within the target company, such as former employees, third-party contractors, or business associates—with authorized access to the target organization's computing systems and/or information about its cybersecurity procedures and defenses—commit insider attacks. This is exemplified by economic espionage.
- **External attacks**: These attempts are typically carried out by skilled hackers who operate from outside the target company. These are the most typical types of cyberattacks against organizations all across the world. A black hat hacker may attempt to enter the target company's networks from another country to get illicit access. To aid their unlawful access, external attackers may gain information about the target corporation's security systems from an insider (disgruntled staff member).

Computers in cybercrimes

Cybercrime may be classified into three types based on how a computer was used to commit a crime.

- The computer is used as a weapon in the commission of a crime. Launching **denial-of-service** (**DoS**) attacks or delivering ransomware are two examples.
- Crime has been committed against a computing device. Obtaining illegal access to a target computer, for example.
- The computer is used to aid in the commission of a crime. Using a computer to keep incriminating data or communicate with other criminals online, for example.

Example of cybercrime: Various types of computer intrusions result in various types of undesirable results. For example, certain cyberattacks may damage or destroy the operating system, compelling you to reinstall it. Another type may try to steal your passwords and login details. Other assaults, on the other hand, may not completely damage your computer, but they will track your online activities and jeopardize your privacy. Criminals are more sophisticated than ever, and harmful software is more complicated than ever. Modern malware may infect a computer and remain undetected for a long time. Rather than inflicting harm on your computer, the majority of intrusions these days are carried out to steal money, acquire

access to personal information, or obtain login credentials. Cybercrime, like traditional crime, may be divided into a variety of categories depending on the motivation of the criminals.

Digital forensics categories

Digital forensics can be classified based on the source of the obtained digital evidence. The collection of digital artifacts contained on the target computer device, which can be used as evidence in court, is referred to as digital evidence, as presented in *figure 1.1*.

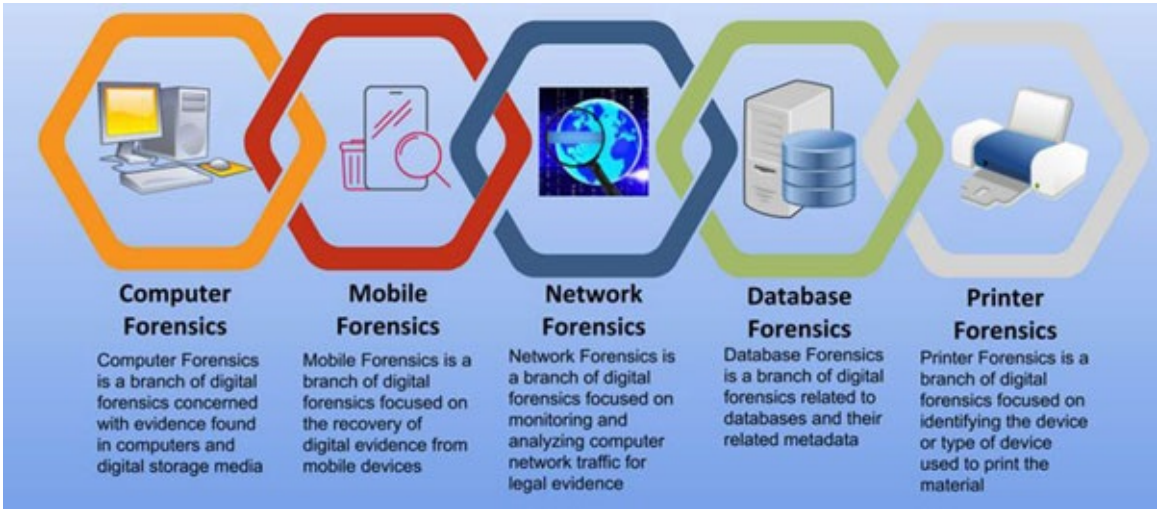


Figure 1.1: *Types of digital forensics*

Computer forensics

This is the most common type of digital forensics; it involves investigating digital evidence on laptops, desktops, and storage devices such as USB drives, SD cards, system memory (RAM), operating systems, and application logs and traces. The primary goal of this type of investigation is to retrieve deleted data from the target device's storage and examine it for incriminating or vindicating evidence.

Mobile forensics

Mobile forensics is a subset of digital forensics that specializes in gathering data from mobile devices. A mobile device is any computing device (such as phones, smartphones, tablets, and wearable devices such as smartwatches) that can make phone calls through traditional communication networks. Such gadgets are usually geolocation-aware, which means they have a GPS or other satellite positioning system built-in. Because of the extensive usage of mobile technology among customers globally, mobile forensics will soon supersede existing methods of digital forensics.

Network forensics

This field of digital forensics entails monitoring and analyzing network traffic to extract evidence, such as the source of a network breach, or to identify intrusions. Data flow via networks can be gathered in bulk in real-time and stored for later analysis. Alternatively, it can be reviewed in real-time with the option of preserving chosen chunks of relevant events for later study (this option requires less storage space). Unlike other types of digital forensics, network forensics focuses solely on volatile live data.

Database forensics

The analysis of data and information held in databases such as Microsoft SQL Server, Oracle, MySQL, and others is known as database forensics. Database forensics looks at who has access to a database and what actions are made to spot malicious behavior.

Forensic data analysis

This analysis is capable of reviewing corporate data to prevent and identify financial criminal fraud. To identify and prevent corporate resource misuse, it searches for relevant patterns, combines data assets, and compares them to past findings. E-mail forensics, cloud storage forensics, forensics for specific applications such as Web browsers, file system forensics (FAT, NTFS, or EXT), hardware forensics, multimedia forensics (text, image, audio, or video), and live volatile or RAM forensics are all small sub-branches of the main types already mentioned.

Digital forensics users

Digital forensics can be used for a variety of scenarios across almost all sectors and businesses. This science has grown more integrated across other domains as a result of the expanding usage of computing technology and internet activities.

Law enforcement

Digital forensics is used by law enforcement authorities to assist them in upholding the law and protecting society and businesses from criminals. Law enforcement agents employ digital forensics in many settings to uncover crimes and connect them to their perpetrators. Most traditional crimes would almost certainly necessitate obtaining digital artifacts from the scene of the crime, such as a USB drive found in a drug dealer’s office, a laptop from a suspect’s home, or a mobile phone confiscated at a murder scene. Law enforcement computer forensics professionals should follow a specified digital forensics procedure while obtaining, evaluating, preserving, and presenting digital evidence.

Civil ligation

Businesses employ digital forensics procedures and methodologies as part of their electronic investigative process in civil litigation to help identify incriminating digital material that may be used as proof in a civil or criminal legal case. Although digital forensics procedures used in civil cases differ from those used in criminal cases in terms of the processes used to gather digital evidence, the scope of the investigation, and the legal ramifications of the case, e-discovery is seen as an essential component of the judicial system. The bulk of business cases is motivated by financial gain. Bribery, tax evasion, thefts of intellectual property or financial assets, fraud, misappropriation of business resources, industrial espionage, and commercial disputes are only a few instances. Other recorded digital crimes include gender, e-mail harassment, age discrimination, and sabotage. Companies employ digital forensics tools as part of their e-discovery process to locate and extract digital evidence to identify the source, entity, or person responsible for such violations. Such investigations may end in the guilty employee being fired, receiving a warning (if the violation was small and insignificant), or being prosecuted if the matter is brought to court. The application of digital forensics in civil litigation is not limited to business cases; it

also extends to personal matters such as family conflicts and divorce.

Intelligence and counterintelligence

Intelligence agencies use digital forensics techniques and tools to combat terrorism, human trafficking, organized crime, and the drug trade, among other severe criminal activities. Digital forensics tool helps investigators uncover important information about crime syndicates by monitoring networks, investigating digital devices, or acquiring information about the person of interest from publicly available sources such as social media sites; this process is known as **Open Source Intelligence** (**OSINT**), which will be covered in another book soon.

Digital forensics investigation types

According to who is in charge of commencing the inquiry, digital forensic investigations may be divided into two categories:

- Public investigation
- Private sector investigations

Criminal cases leveraging investigations are handled according to the legal guidelines set out by the appropriate authorities. Law enforcement agencies participate in public investigations, which are conducted under national or state legislation. The three main phases of these investigations are complaint, investigation, and prosecution. Private investigations are commonly conducted by businesses to investigate policy violations, legal problems, unfair dismissal, or the leak of secret information as industrial espionage. Because it is up to each corporation to determine, there are no fixed regulations for conducting such investigations; nonetheless, many companies are already implementing strict internal standards for investigating digital crimes. These procedures are similar to public investigations into crimes in that some cases may be presented to the court and ultimately transformed into official criminal prosecutions. Businesses can reduce liability associated with computer crime by developing a clear policy that is easy to read and comprehend by their employees. A policy like this can also help digital forensics investigations proceed more easily and with less downtime for the company if they are needed. The most important rule that all firm employees should sign is the computer usage policy. This policy outlines how employees may use business IT networks and computer systems and cautions them that they may face legal consequences if they break the guidelines.

Forensics readiness

Forensics readiness refers to an organization’s ability to acquire, retain, secure, and analyze digital evidence in a forensically sound manner. To keep costs down, the procedure should take place without interfering with existing operations. The usage of digital forensics preparation planning in businesses offers many benefits, which are listed as follows:

- For instances requiring digital evidence, a quick response time is required. When a data breach or information leak occurs, having a clear e-discovery policy in place can allow businesses to respond promptly and get digital evidence in a forensically sound manner.
- The government’s regulations must be adhered to; US Federal Procedures have produced a set of

guidelines for parties in legal disputes on how to obtain and manage digital evidence so that it may be used in court. If the case gets to court, forensic readiness will reduce the cost of gathering digital evidence and almost certainly, result in a faster resolution. Increasing the security defenses of the firm. Monitoring endpoint computer usage may uncover dangerous malware, such as ransomware, before the infection spreads to the entire organization's network, and using forensic readiness will make an organization well-prepared to handle internal and external security incidents and able to identify an attack quickly before it dives deeply into its IT infrastructure (for example, monitoring endpoint computer usage may uncover dangerous malware, such as ransomware before the infection spreads to the entire organization's network).

- Reducing the number of internal attacks, as previously stated, internal threats such as rogue employees are more dangerous than external attacks; the presence of a forensic readiness plan in an organization will make hostile insiders fear being discovered if they participate in illegal behavior.
- Increasing the security posture of an organization as a company's forensic readiness strategy will set it apart as a powerful defender against cyber-attacks. Customers will be more inclined to do business with this organization since their data will be kept private and secure. Investors will also feel secure in the knowledge that their money is protected and that there is a minimal probability that successful attacks on this organization would result in their money being lost.

Type of digital evidence

User-created data and machine-created data are the two most common sorts of digital artifacts.

User-created data

User-generated data is anything created by a person (human) using a digital device. Metadata is data that is included in files created by a computer user; the metadata may be created by the computer user on purpose (for example: author name and e-mail), or it may be generated automatically by the software that created the file, such as captured camera model/type, date and time of clicking the photograph, GPS coordinates of the photograph, and resolution). Metadata should be thoroughly examined throughout any inquiry since it may include important information about the subject at hand. It includes the following, among other things:

- Previous backups (including both cloud storage backups and offline backups such as CDs/DVDs and tapes)
- Account details (username, picture, and password)
- E-mail messages and attachments (both online and client e-mails as Outlook)
- Audio and video files
- Address book and calendar
- Webcam recordings (digital photos and videos)
- Content files (for example, MS Office documents, IM conversations, bookmarks), spreadsheets, databases, and any other digitally stored text
- Hidden and encrypted files (including zipped folders) created by the computer user

Machine and network-created data

Any data that is automatically generated by a digital device is considered machine/network-created data. It includes the following, among other things:

- Configuration files and audit trails, including third-party service providers (for example, **Internet service providers** (**ISPs**) often retain customers' accounts and browser history logs)
- Logs on the computer under Windows OS contain the following logs: Logs for application, security, setup, system, forward events, apps, and services
- GPS tracking information history
- Temporary files
- Information from the browser (browser history, cookies, and download history)
- In addition to the IP addresses associated with a LAN network and the broadcast settings, devices have **Internet protocol** (**IP**) and MAC addresses
- Instant messenger history and buddy list (Skype and WhatsApp) (from devices with GPS capability)
- Application and Windows history (for example, a recently opened file in MS Office)
- Under Windows computers, restore points
- E-mail header information
- Registry files in Windows OS
- Hidden and conventional system files
- Printer spooler files
- Virtual machines
- Surveillance video recordings
- Paging and hibernation files and memory dump files

As a result, digital evidence can be defined as any file or data/metadata that is provided in a digital (binary) format and could be used in a trial.

Locations of electronic evidence

Digital evidence is frequently found on hard drives, but as computer technology advances, digital evidence is increasingly discovered in practically all digitally aware devices. The following is a list of the most common types of devices that must be examined for digital evidence:

- **Systems**: Desktops, Laptops, Tablets, Servers, and RAIDs
- **Network devices**: Hubs, switches, modems, routers, and wireless access points
- **Internet-enabled home automation and IoT devices**: Air conditioners and Smart refrigerators
- DVRs and surveillance systems
- MP3 players
- GPS devices

- Smartphones
- PDA
- Game stations—Xbox, PlayStation
- Digital cameras
- Smart cards
- Pagers
- Digital voice recorders

Chain of custody

A chain of custody is required for any digital forensic investigation approach. A proper chain of custody should detail how digital evidence was discovered, gathered, transported, researched (analyzed), stored, and maintained by the various parties involved in the investigation. The ultimate goal is to protect the integrity of digital evidence by tracking down everyone who had contact with it from the moment it was collected until it was presented in court. If we fail to understand who made contact with the evidence at any time throughout the investigation, the chain of custody will be jeopardized, and the obtained evidence will be rendered useless in a court of law. To maintain a proper chain of custody that is acceptable in court, an audit record for all acquired digital evidence that tracks the movements and possessors of digital evidence at all times must be preserved. If the chain of custody is valid, investigators will be capable of answering questions in a court of law:

- What is the definition of digital evidence? (For example, describe the digital proof that was obtained.)
- Where did you find the digital evidence? (For example, a computer, tablet, or mobile phone; furthermore, the status of the computing device when the digital evidence is acquired—ON or OFF?)
- How did the digital evidence come to be? (For example, tools employed; you should also indicate the procedures done to protect evidence integrity throughout the acquisition phase.)
- What methods were used to transfer, preserve, and handle digital evidence?
- What methods were used to assess the digital evidence? (For example, any tools and procedures used.)
- When, by whom, and for what purpose was digital evidence accessed?
- What was the role of digital evidence in the investigation?

Every movement of digital data must be recorded for an investigator to ensure that the subject data has not been tampered with and that no external evidence has been inserted to fool the investigator during the investigation.

Examination process

Although there is no globally agreed method or procedure for performing digital forensic investigations, various approaches are in place, with varying stages or phases. However, all strategies divide the job into four primary phases.

- Search and seizure
- Acquiring
- Analyze
- Information gathering and reporting

Seizure

During this process, the hard evidence (digital device) will be confiscated and delivered to the forensic lab with care. Any type of computing device can be used as evidence, including a laptop, tablet, phone, external hard drive, USB flash drive, wearable device (for example, a digital watch), or even a desktop PC. You will need permission from the appropriate authorities to confiscate the suspect's machine (for example, a court warrant). When professional experts arrive at the crime scene, the accused's digital device should be examined to ensure that the digital evidence was captured and maintained properly. If the questionable computer was still on, we should try to retrieve as much volatile RAM as possible. It used to be the usual procedure to unplug the computer and place it in an anti-static enclosure. Modern forensic approaches, on the other hand, understand the need to retrieve volatile memory while the machine is still operational. In RAM, you can find chat logs, cryptographic keys, clipboard contents, unencrypted data, and system process information. Because the application used to extract RAM content would create minor changes to the target operating system files, RAM, and hard disc, collecting RAM should be noted in the final inquiry report along with the instrument employed.

Acquisition

If the machine was still operational, this stage dealt with secondary storage devices such as SSDs, USBs, tapes, and volatile RAM. In this step, a computer forensic specialist will duplicate the suspect hard disc to create a complete picture of the confiscated hard drive (also known as a bit-to-bit image). Examiners frequently use hardware duplicators or software imaging tools like Linux's DD command to duplicate CDs. Keep in mind that the suspect's hard disc should be write-protected to prevent tampering with the original while copying the evidence. If the suspect machine was still up and running, RAM should be collected to account for a variety of scenarios.

Analysis

This stage examines the contents of the acquired forensic image file using a range of tools to hunt for useful clues inside the image. Volatility, EnCase, Sleuth Kit, and Forensic Toolkit are examples of specialized tools that can recover deleted, concealed, and encrypted data, as well as messaging conversation logs, Web browser history, and deleted files and e-mails. The forensic tool uses hash signature analysis in this stage to locate notable files or rule out those that are known. The contents of obtained picture files are hashed and compared to pre-compiled lists such as the Reference Data Set of the National Software Reference Library. The library collects software from many sources and converts the software's file profiles into an RDS of data. Forensic tools can search inside the acquired photo file using keyword search terms or phrases. This will greatly expedite the investigation and aid investigators in quickly discovering relevant information.

Reporting

In this phase, the examiner creates a structured report outlining his or her findings. This type of report is usually created for non-technical audiences (such as judges, attorneys, and juries). When preparing the report, consider the writing style, terminology, and presentation of information. Evidence should be included with the report, ideally in digital format. The following should be included in the forensic report's general content:

- A summary of the most important findings.
- A description of the tools (both hardware and software) used throughout the inquiry process, as well as their functions and software versions.
- The method by which the digital evidence was obtained.
- A description of the digital evidence (picture content) as well as the fascinating objects discovered inside it (for example, internet browsing history, e-mail history, USB registry analysis, and deleted files found). When possible, use screenshots to clarify the procedures involved in analyzing digital evidence to the reader.
- Explanation of technical words used in the report, such as "*unallocated disc space*" and "*Host Protected Area*", so that non-technical individuals may comprehend them.
- The original suspect's hard disc, as well as digital copies (images), should be provided to the court together with the report.

Conclusion

This chapter defined digital forensics and differentiated it from other types of cybersecurity domains. The concept of digital evidence was briefly reviewed, as well as the many types of digital evidence and where we may find it in electronic devices. Although there is no universally accepted protocol for conducting digital forensics investigations, we have outlined the essential stages of each digital inquiry as well as the responsibilities that must be fulfilled at each stage. Many attempts have been made to standardize digital forensics standards and procedures through the release of recommendations by recognized government bodies, the most notable of which are those produced in courts. Digital forensics experts are required in practically every industry and sector, ranging from NGOs to government agencies to private firms and corporations. Demand for digital forensics specialists is likely to grow in the coming years as more organizations relocate their work to the digital sphere. In the upcoming chapter, we will go over the key technological concepts that every digital forensics or cybersecurity specialist should know before starting their investigation.

Multiple choice questions/questions

Learning Section

1. **Computer Forensics is also known as.**
    a. Digital Forensic Science
    b. Computer Crime Stream

c. Computer Forensic Science
d. Computer Forensics Investigations

2. **Computer Forensics can also be used in civil proceedings.**

a. True
b. False
c. Can be Yes or No
d. Cannot say

3. **You are supposed to maintain three types of records in Forensics, which of these is not a record?**

a. Chain of Custody
b. Documenting crime scene
c. Searching crime scene
d. Documenting actions

4. **Volatile data resides in.**

a. Registries
b. Cache
c. RAM
d. All the above

5. **Forensic investigators should satisfy ….**

a. Contribute to society and human being
b. Avoid harm to others
c. Honest and trustworthy
d. All Of the Above

6. **Digital evidence is used to establish a credible link between…….**

a. Attacker and victim and the crime scene
b. Attacker And information
c. Either A or B
d. Both A and B

7. **The evidence and proof that can be obtained from the electronic source is called the…….**

a. Digital Evidence
b. Explainable evidence
c. Either A or B
d. Both A and B

8. **Digital Evidence must follow the requirement of the …**
    a. Ideal Evidence Rule
    b. Best Evidence Rule
    c. Exchange Rule
    d. All of the mentioned
9. **A false positive can be defined as …**
    a. An alert that indicates nefarious activity on a system that, upon further inspection, turns out to represent legitimate network traffic or behavior
    b. An alert that indicates nefarious activity on a system that, upon further inspection, turns out to truly be nefarious activity
    c. The lack of an alert for nefarious activity
    d. All of the above
10. **A valid definition of digital evidence is:**
    a. None of the below
    b. Data stored or transmitted using a computer
    c. Digital data of probative value
    d. Any digital evidence on a computer

Answers

1. **c**
2. **a**
3. **c**
4. **d**
5. **d**
6. **a**
7. **a**
8. **d**
9. **a**
10. **c**

CHAPTER 2

Essential Technical Concepts

Introduction

Undertaking a digital forensics investigation demands a deep grasp of some of computing's most fundamental technological concepts. To discover and manage digital evidence, you must understand how information is stored in computers, number theory, how digital files are constructed, and the many types of storage units and their differences. These fundamental topics will be covered in this chapter. Computers store, process, and portray digital data in a certain way, as explained in this chapter.

Structure

In this chapter, we will cover the following:

- Different number system
- Encoding schema
- File carving and structure
- File metadata
- Hash analysis
- System memory
- Storage
- Filesystem
- Cloud computing
- Windows OS
- Networking

Objectives

We will explore how a system represents data in this part, as well as typical numbering systems and the principal encoding strategy used by machines to generate human-readable text. Let us start with the standard numbering scheme.

Decimal (Base-10)

The base-10 system, which employs 10 digits or symbols (0, 1, 2, 3, 4, 5, 6, 7, 8, and 9) to represent its values, is the most extensively used numbering system that we use every day while conducting arithmetic calculations (for example, 17 + 71 = 88). The value a number represents is determined by its position in

decimal, where each digit is multiplied by the power of 10 corresponding with that digit's location. Take, for example, the decimal number 7,564. This number can be interpreted as:

$$7{,}654 = 7{,}000 + 600 + 50 + 4$$

An understanding of the decimal numbering system is essential, as the other numbering systems follow similar rules.

Binary

Data is stored in binary format on computers, which is the base-2 numeric system represented by 1s and 0s. The computer language, binary, follows the same laws as a decimal. Binary, on the other hand, contains two symbols (0 and 1) and multiplies by the power of two, unlike decimal, which has 10 symbols and multiplies by the power of 10. Each 1 OR 0 in a computer is referred to as a bit (or binary digit), and the total of eight bits is referred to as a byte. The most significant bit is the highest-order bit, which is placed in the leftmost bit and has the greatest significant bit value (MSB). On the other hand, the Least Significant Bit is positioned in the rightmost bit and is the **lowest significant bit** value (**LSB**). *Table 2.1* identifies bit values from their position for Binary numbers.

MSB	Binary Digit							LSB
2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
256	128	64	32	16	8	4	2	1

Table 2.1: *Representation of binary numbers*

For example, the binary number 110011001 can be converted into decimal as 409, as calculated in *Table 2.2*.

Binary								
1	1	0	0	1	1	0	0	1
Decimal								
1×2^8	1×2^7	0×2^6	0×2^5	1×2^4	1×2^3	0×2^2	0×2^1	1×2^0
256	128	0	0	16	8	0	0	1
= 256+128+0+0+16+8+0+0+1								
= 409								

Table 2.2: *Binary to decimals*

Data is stored in computer systems in binary format, including Microsoft Word documents, Digital photographs, videos, Excel sheets, social media tweets and posts, e-mails, and anything and everything created and stored on computer systems.

Hexadecimal (Base-16)

The values of the hex numbering system are represented by 16 digits or symbols. It has the following numbers and letters: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, and W (capital letters are used to represent numbers from 10 to 15). When dealing with computers and other digital systems, hex numerals are frequently encountered, particularly when looking at the memory address location. The key objective of this numbering scheme is to express lengthy binary data in a

compact format that people can understand. Hex does this by combining all of the bits (binary digits) into a single group.

Hexadecimal (Base-64)

The only major distinction between Base64 and hex is how bytes are represented. Base16 is also referred to as "*hex*". Hex requires two characters for every byte, whereas Base64 requires four for every three bytes, making it more efficient than hex. A 100K file will require 200K to encode in hex or 133K in Base64, assuming you are using UTF-8 to encode the XML text. Of course, it is possible that you have no interest in space efficiency; in many situations, it will not matter. If it does, Base64 is unquestionably superior in that regard. (Alternatives exist that are even more effective, but they are less prevalent.) Here is a common phrase used in distributed computing "*Many hands help with a task*". The quote is represented when encoded into Base64 as the following byte sequence of 8-bit padded ASCII characters: "*TWFueSBoYW5kcyBtYWtlIGxpZ2h0Ihdvcmsu*" (newlines and white spaces may be included anywhere but are to be ignored on decoding).

Character encoding schema

How can the 1s or 0s end up displayed on the screen as letters like A, B, or X, Y, X, when everything in computers is represented by 0 or 1? To turn binary numbers into meaningful text that a person can read, computers use character encoding schemes like the alphabet content and texts on your screen when the PDF version of this book is opened on the computer screen). There are two major encoding schemas used by computers to represent text:

- **ASCII, or the American Standard Code for Information Interchange** (**https://ascii.cl**), was created many years ago and is still supported by almost all text editors today. Because it only employs seven bits or 128 values, ASCII has a restricted capacity to represent all letters from all languages throughout the world, as well as punctuation and other special symbols from other languages. Enhanced ASCII is another extended form of ASCII that provides 256 characters; however, it still does not support all worldwide languages.

ASCII TABLE

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(	72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29	)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	O	111	6F	o
16	10	[DATA LINK ESCAPE]	48	30	0	80	50	P	112	70	p
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	s
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	v
23	17	[ENG OF TRANS. BLOCK]	55	37	7	87	57	W	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	x
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	;	91	5B	[	123	7B	{
28	1C	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	|
29	1D	[GROUP SEPARATOR]	61	3D	=	93	5D	]	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	~
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]

Figure 2.1: *ASCII table*

- **Unicode consortium** (**https://unicode.org**) invented Unicode encoding, which is a widely used character-encoding strategy that assigns a unique number to each character from every worldwide language. Major operating systems, software platforms, portable devices, and online applications all support Unicode. UTF-8, UTF-16, and UTF-32 are the three types of Unicode.

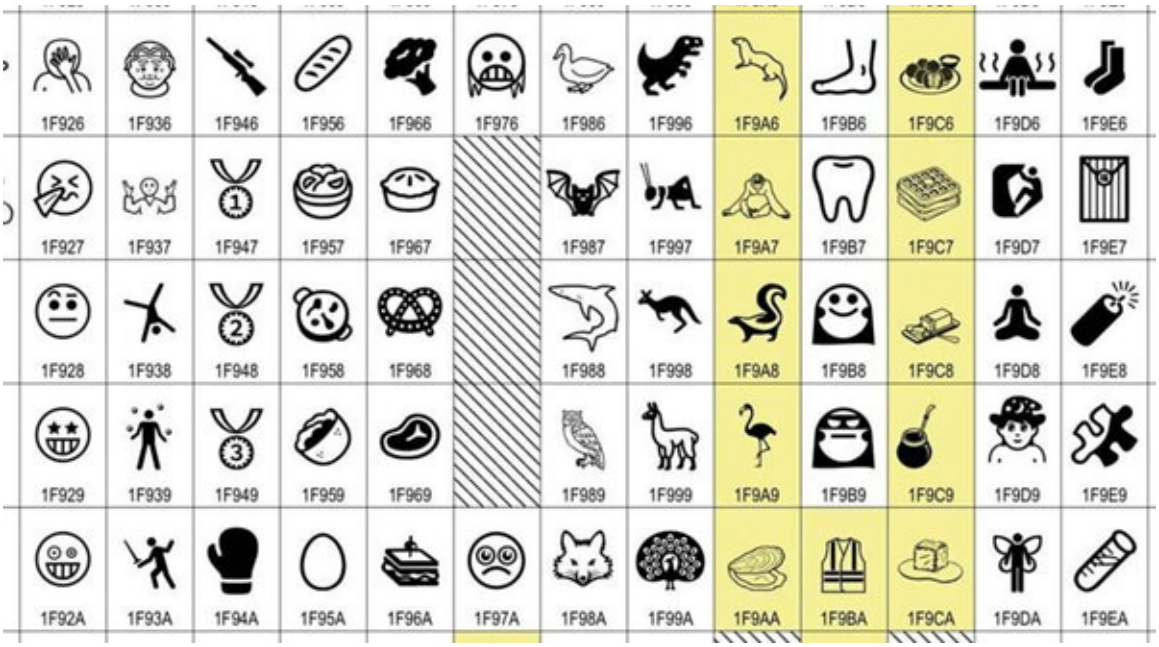


Figure 2.2: *Unicode Ver 12*

File carving

In digital forensics, understanding how computers store and portray data is critical; for example, an analyst may need to recover and open a file from unallocated disc space on the target hard disk drive or from a raw dataset without using the software that produced the file (for example, MS Word). This

method is known as file carving, and it may be used to recover lost files and file fragments from erased or damaged hard drives. To carry out file carving, we must first understand how to separate a file from its signature. File carving is most frequently used to recover files from the unallocated space in a drive since it is a forensics approach that recovers files based solely on file structure and content and without any matching file system meta-data. Unallocated space is the portion of the disc that, according to the file system architecture, like the file table, no longer contains any file information. The entire disc may be affected if the file system structures are broken or absent. In plain English, a lot of filesystems do not completely zero away the data when they erase it. Instead, they only take away the location's information. By scanning the disk's raw bytes and putting them back together, a process known as "*file carving*" reconstructs files. This is often accomplished by looking at a file's header and footer, which are the first few bytes and last few bytes, respectively.

When directory entries are damaged or missing, file carving is a fantastic technique for recovering files and pieces of files. Forensics professionals use this method, in particular, to recover evidence in criminal situations. Law enforcement officers frequently use carving techniques to extract more photos from suspects' hard discs in specific child pornography instances. The hard drives and portable storage devices that American Navy Seals seized from Osama Bin Laden's campus during their raid serve as another illustration. File carving techniques were used by forensic professionals to extract every piece of information from this medium.

File structure

Each file type has its encoding scheme that specifies how information is kept, and a digital file is made up of a series of bits. This schema's name is "*file format*". The file format can be either open source (such as PNG, an ISO/IEC raster image format) or proprietary (such as Adobe Photoshop) (like the **Windows Media Audio** [**WMA**] file format). Many common multimedia file formats may hold multiple content kinds, which is the case with some file formats. For example, the OGG format may hold video, music, text, and metadata in a single container. AVI, WAV, and 3GP files are also included in this category. Users are identified as file types by their extensions, according to the researchers. The DOCX or DOC extension is used for MS Word files, whereas the XLSX or XLS end is used for MS Excel files. As digital forensic investigators, however, we cannot only depend on the file extension to determine the file type because it may be altered to anything (for example, an MS Word file can be changed to a DLL or PNG file to conceal its true identity).

File signature (header) can be checked to establish the type of strategy to detect it. The first 20 bytes of most digital files contain a signature; you may check this signature by opening the subject file in Windows Notepad or another text editor like Notepad++. To analyze a text file or document file, change the extension to, say, JPG and then analyze the JPG file first 20 bytes in a Hex editor (refer to *figure 2.3*). HexEditor would reveal the original file signature as being editable in `Notepad.exe`.

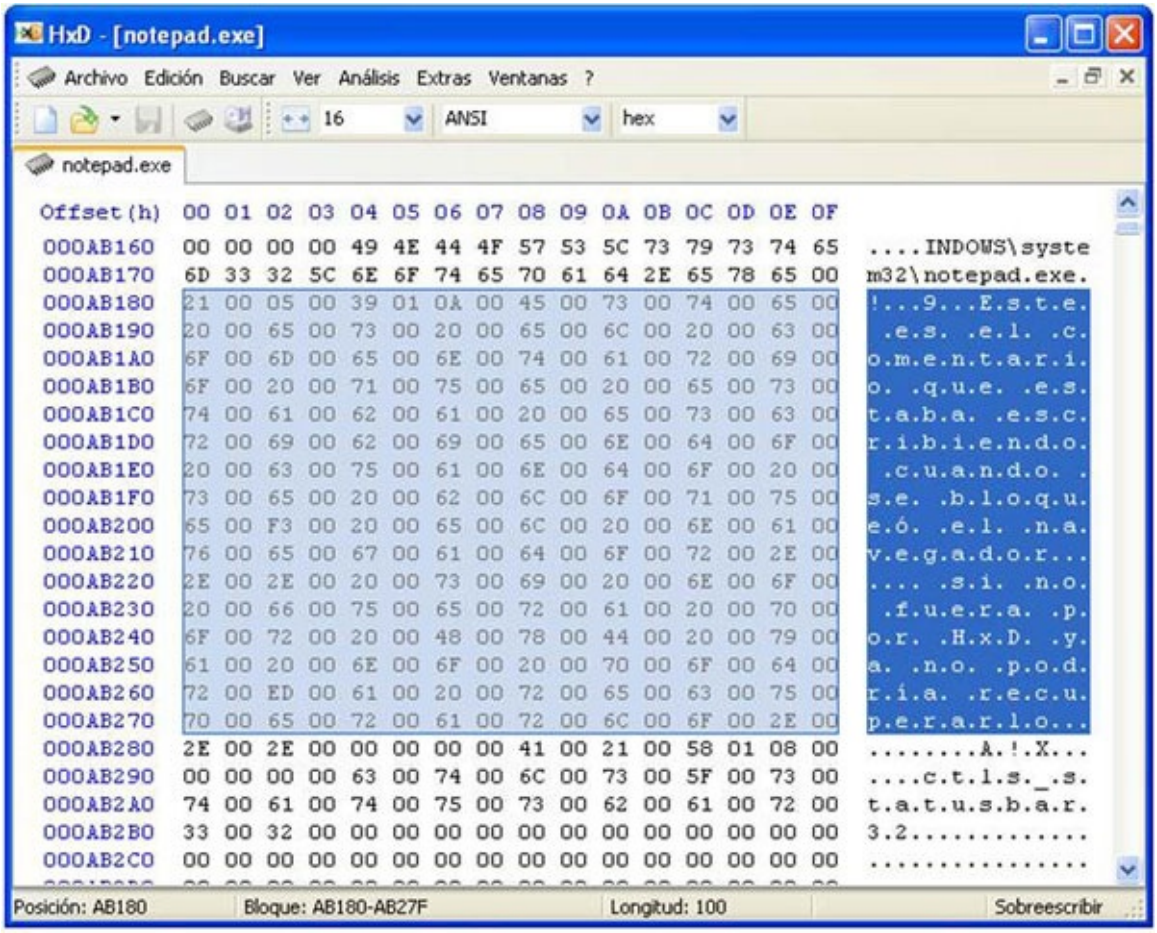


Figure 2.3: *Investigate the first bytes to reveal the true file format*

There are many free Hex editors; some common ones used extensively are as follows:

- wxHexEditor (www.wxhexeditor.org/home.php)
- Free Hex Editor Neo (www.hhdsoftware.com/free-hex-editor)
- PSPad (www.pspad.com/en)

Digital file metadata

Metadata is information about information. Metadata is connected with almost all digital file formats. Although it is frequently included in the same file, certain file formats save their information in a distinct file. Metadata contains information about the file with which it is related. Author name, organization name, computer name, date/time produced, and comments are examples of information found in MS Word files. Metadata may be quite beneficial in many circumstances when it comes to digital forensics. We may, for example, trace file authors using the information linked with them. We may also look for useful information in the file's metadata (most operating systems currently offer searching inside the file metadata information), and most digital forensic suites support searching within the metadata of acquired forensic picture files.

Under Windows OS, we may change the metadata of many different types of digital files without the need for any third-party programs. For example, we may update the metadata information of an MS Office file by right-clicking it and selecting **File** | **Properties** from the menu that appears (see *figure 2.4*) to determine the file author and Statistics as date/time created, modified, accessed, and even printed. The Statistics also reveal the number of pages, paragraphs, lines, words, and total characters with and without spaces.

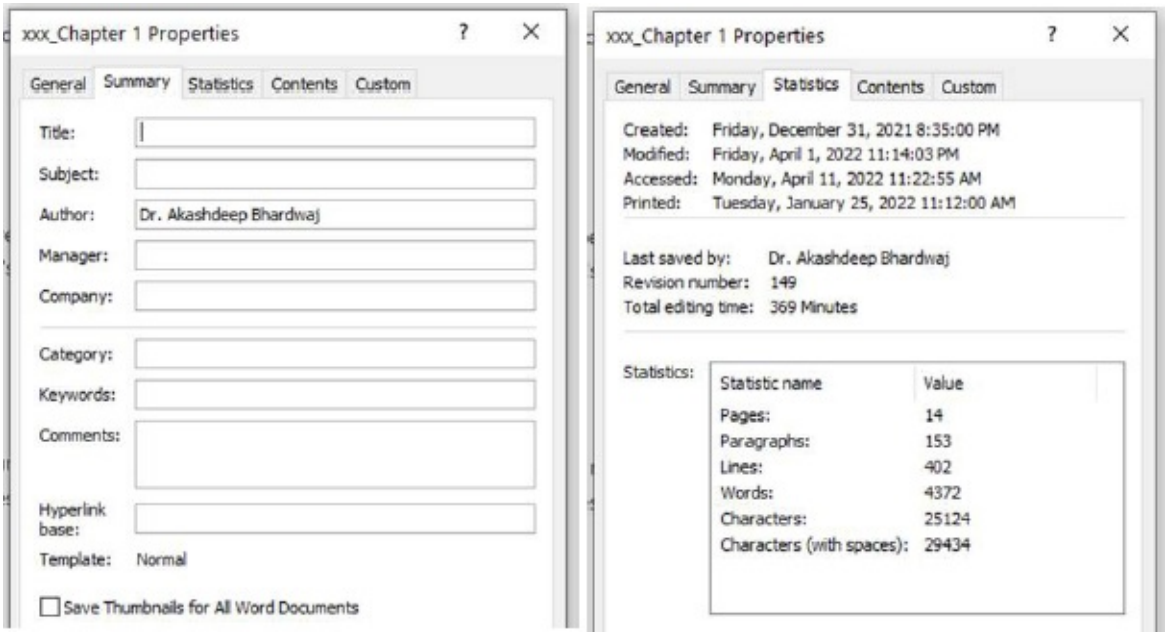


Figure 2.4: *Word file summary and statistics properties*

The metadata of an image file contains critical forensic data such as the timestamp for when the shot was taken and the GPS coordinates for where it was taken (if allowed in the capturing device), as well as camera specifications and settings (see *figure 2.5*). The image metadata information may be seen on Windows in a similar way as MS Office files.

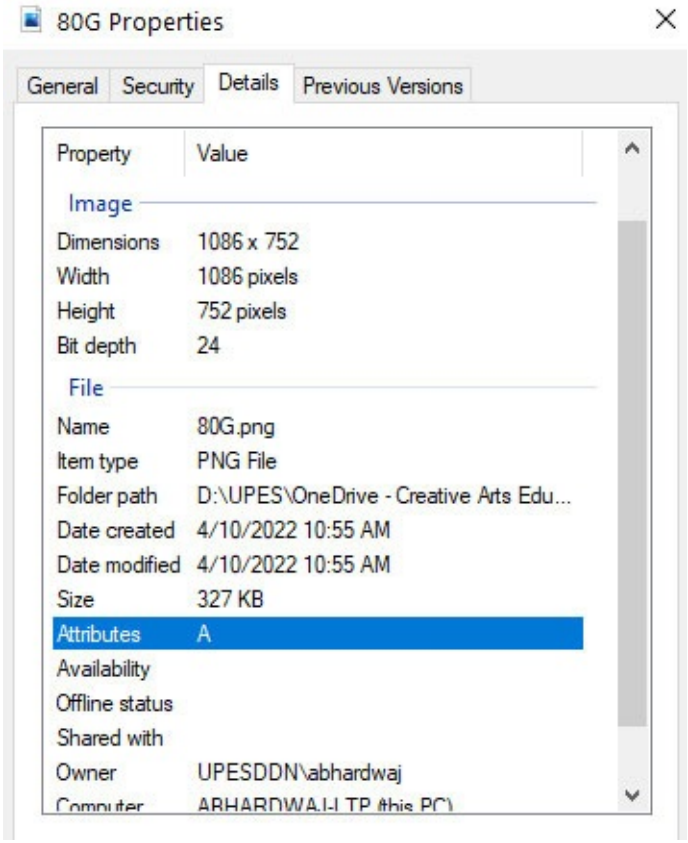


Figure 2.5: *Image metadata info*

Several free tools can also view and edit the metadata information of digital files as follows:

- ExifTool by *Phil Harvey* (*www.sno.phy.queensu.ca/~phil/exiftool*). Read, write, and edit Meta-information for a wide variety of digital files (most image formats).

- Exif Pilot (*www.colorpilot.com/exif.html*). Image metadata editor/viewer.
- GIMP (*www.gimp.org*). Image editor; can manipulate/view image file metadata.
- Pdf Metadata Editor (*http://broken-by.me/pdf-metadataeditor*). For PDF files.
- Mp3tag (*www.mp3tag.de/en*). For audio files.
- XnView (*www.xnview.com/en/*). View/edit image metadata.
- MediaInfo (*https://mediaarea.net/en/MediaInfo*). Metadata viewer/editor for video and audio files.

Timestamps decoder

For Microsoft Office 2010 to 2016, go to the **File | Properties** tab to get the metadata information as illustrated in *figure 2.6*. The **Properties** panel will appear on the right side; by clicking the **Properties** button and choosing **Advanced Properties**, you may examine document metadata. From the standpoint of digital forensics, metadata analysis is critical for any sort of investigation since it may disclose a wealth of information about the case at hand. Some users (for example, criminals) may attempt to modify the metadata of the file to erase the evidence and mislead investigators. Forensic specialists are responsible for detecting such tampering and attempting to reveal it to the court. The majority of computer forensics software allows for mass extraction and search of file metadata. Digital files contain a variety of information, the most essential of which is the timestamp metadata, which is used to indicate various date/time events related to the file of interest, such as the last access date/time, the last updated date, and the creation date. During the investigations, we may come across a date/time that is encoded in a specific fashion that we must decode (for example, date/time data in the Windows registry that are recorded in binary format and must be converted to ASCII) from **www.digital-detective.net/digital-forensic-software/free-tools**.

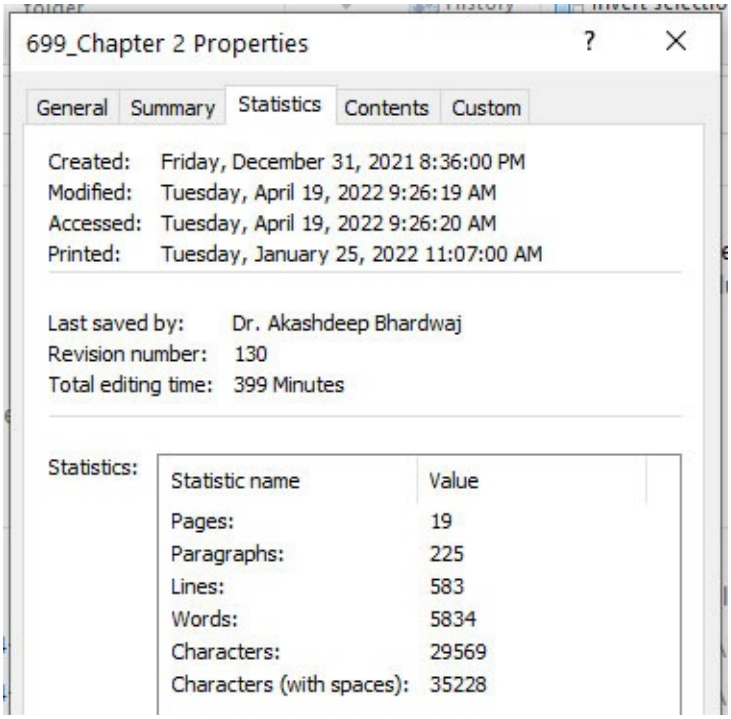


Figure 2.6: *File properties*

Hash analysis

Hashing is an important concept in digital forensics; in fact, you must compute the hash value of every digital evidence you obtain throughout your investigation (whether it be a hard disc image or a single file) to verify that the acquired data (i.e., the digital evidence) has not been tampered with. Hash works by converting a digital file (input) into a fixed string value (output); the resultant hash value is unique and cannot be created using any other file or piece of data. A hash generator tool may be used to determine the hashing value of any digital file or piece of data. MD5 and SHA-256 are two of the most well-known cryptographic hash algorithms. Hashing, referred to as digital fingerprinting, is used in digital forensics investigations the first time to ascertain the acquired forensics image even before analysis begins (to make identical copies of the acquired forensics image) and then the second time after the investigation to verify the integrity of the data and forensics processing as presented in *figure 2.7*.

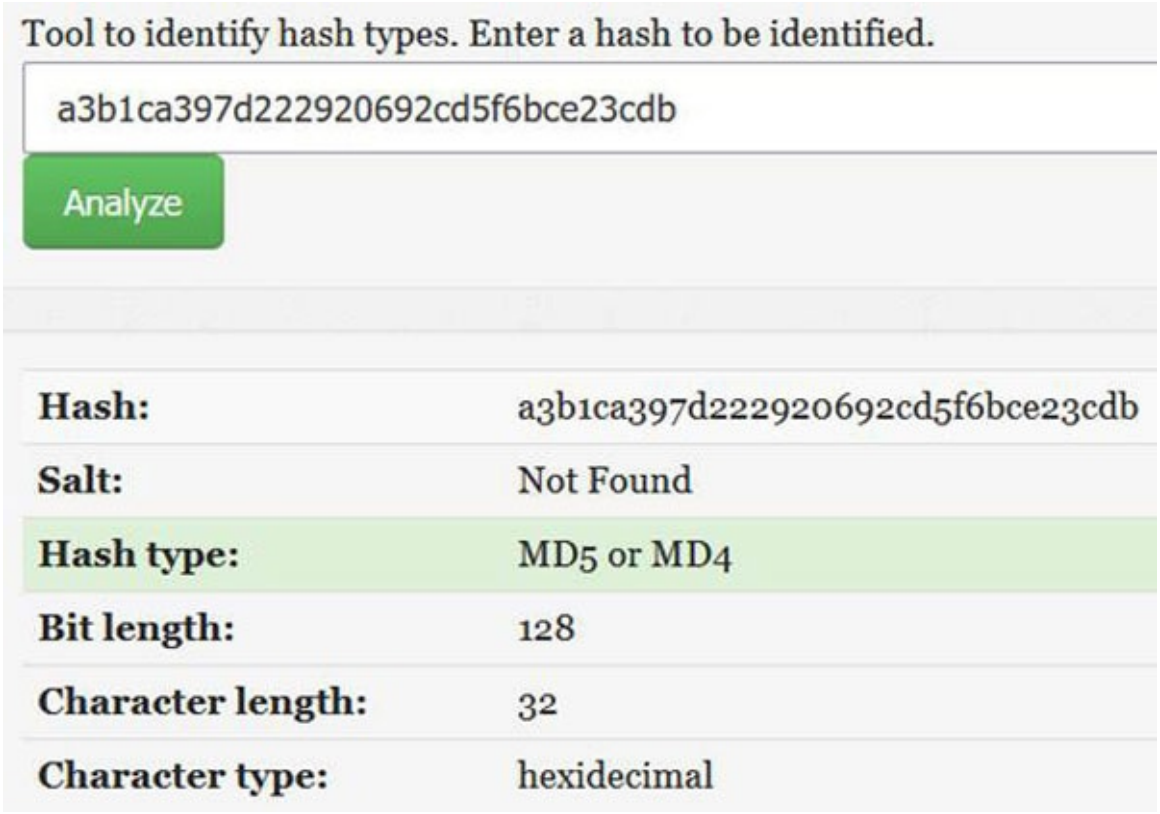


Figure 2.7: *Hash analysis*

Calculate file hash

Hashing capabilities are included in all digital forensics suites; however, you may use a third-party application or the built-in hashing tool in Windows OS.

Method 1: Using a third-party tool

- **Febooti Hash and CRC**: Using a third-party tool (**www.febooti.com**), install this program on your Windows PC, then right-click on the file you wish to calculate the hash for, select **Properties**, and then the **Hash/CRC** tab.
- **HashMyFile**: (**http://www.nirsoft.net/utils/hashmyfiles.html**) is a portable program to display the hash values of chosen directories and files using various hashing techniques (for example, md5, SHA 256).

Method 2: Use built-in windows hashing feature

When using PowerShell to calculate a file hash, Windows defaults to the SHA256 algorithm; however,

you can specify the cryptographic hash function to use by adding the algorithm parameter after the file path followed by one of the following cryptographic hashes (SHA1, SHA256, SHA384, SHA512, and MD5).

System memory

Memory refers to the physical component of a computer that stores data for immediate or later usage. According to how long information is retained on them, we may distinguish between two primary sorts.

- **Volatile memory**: Stores data for a limited period; in fact, it requires electricity to maintain data, but when the power is switched off, it quickly loses its contents. RAM is an example of volatile memory.
- **Non-volatile memory**: Even if the power is switched off, the non-volatile memory may keep data for a long time. This is most commonly used for long-term storage. Computer hard drives, flash memory, and ROM is examples of this type of memory (read-only memory).

Types of computer memory storage

Computer memory storage is divided into two main categories: primary storage and secondary storage, as presented in *figure 2.8*.

Primary storage

This form of storage, often called main storage or system storage, contains a volatile memory that loses data when the power is switched off. Primary storage is used to temporarily store data and programs, and it has a smaller storage capacity, and faster read/write operations than secondary storage. It is also more expensive. RAM and cache are the two types of primary storage memory found in computers (CPU memory).

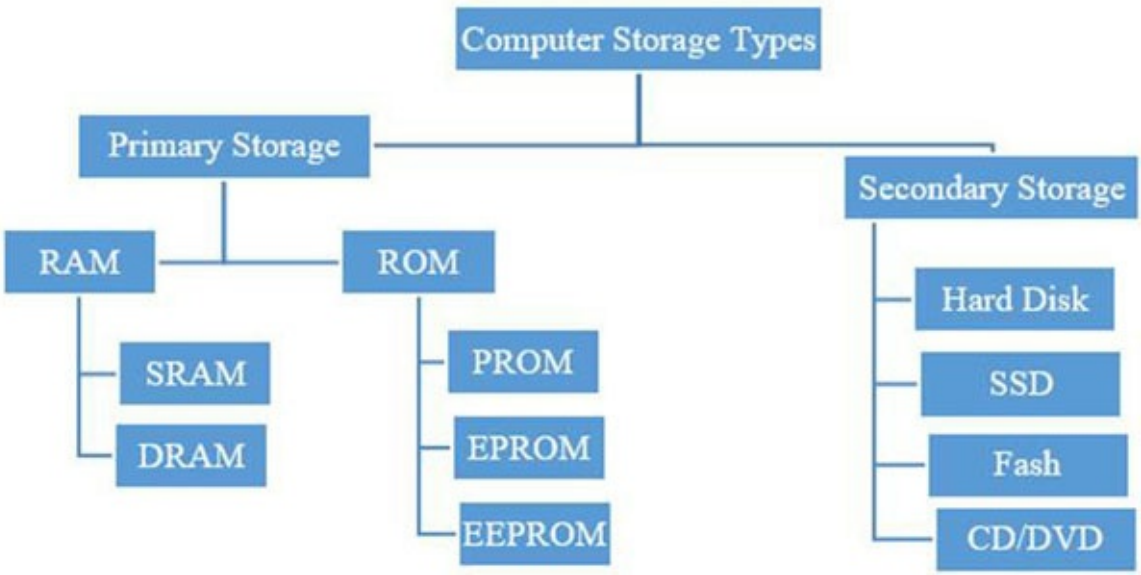


Figure 2.8: *Computer memory types*

RAM

Random Access Memory (RAM) is the most crucial component of any computing device; the volatile memory stores information that the system requires to process right now or in the future. This is high-speed memory as compared to secondary storage media. When you run a Web browser, for example, it will load into RAM. The RAM contains a plethora of data in terms of digital forensics, such as executable applications, network sessions, Web browser history, IM chat, passwords, photographs, decrypted files, and so on. In every digital forensic inquiry involving an operating machine, capturing a RAM image becomes essential. RAM is divided into two types:

- **Dynamic RAM** (**DRAM**) is the first type of memory where "dynamic" refers to the fact that to keep its data, this memory must be updated each second. DRAM is the most common type of primary memory found in PCs, workstations, servers, and smartphones. DRAM (DDR2, DDR3, DDR4, where DDR stands for Double Data Rate) because it is synchronized with the microprocessor's clock speed.
- **Static Random Access Memory** (**SRAM**) is used in CPU memories (CPU Cache) and is extremely fast (faster than DRAM) because it does not require constant refresh (thus, the word "static"). SRAM is more expensive and consumes more power than DRAM. Both forms of RAM are volatile, which means they will lose their contents if the power is switched off.

ROM

Read Only Memory (ROM) is solely used for reading operations; as its name implies, it does not support write operations. This memory is non-volatile because it preserves the information it contains even when the power is turned off. This type of memory is used to store firmware programs (software that is stored on hardware devices such as a computer motherboard and graphics card that provides instructions on how that device should operate) in computers and many other digital devices. Data modification in ROM is challenging and necessitates the use of special applications. There are three different kinds of ROMs:

- ROM that can be programmed (PROM)
- Programmable ROM that can be erased (EPROM)
- Programmable ROM that can be erased electrically (EEPROM)

Secondary storage

Secondary storage is non-volatile, long-term storage. Without secondary storage, all programs and data would be lost the moment the computer is switched off. There are three main types of secondary storage in a computer system: solid-state storage devices, such as USB memory sticks.

Backup storage

External memory or auxiliary memory are other terms for secondary storage. This is a type of non-volatile memory that keeps its contents whether the power is on or off. It is used to keep data for a long time. Secondary storage is slower than primary storage, such as RAM, but it is far less expensive. The following are some examples of secondary storage.

HDD

In a computer, the **hard disc drive** (**HDD**) is the primary permanent storage (non-volatile) destination for data. It stores data using magnetic storage technology for later use. HDDs are a well-established technology that has been in use since 1960 when they became the primary and secondary storage devices for a variety of computing systems such as desktops, servers, and laptops. Any digital forensic investigator has likely dealt with a hard disc drive, and this technology is projected to be around for a long time. There are two types of HDD drives: fixed (internal) and external. The first (fixed) is built into the computer, whereas the external HDD can be linked to it through a USB or eSATA cable to expand storage. Data is stored on platters in **hard disc drives** (**HDDs**). A platter is a round metal disc composed of aluminum, glass, or ceramic that is covered in a magnetic substance and may store data on both sides (top and bottom surfaces). A hard disc can have many platters; however, consumer hard drives with capacities under 500 GB will only have one.

Depending on the physical size, capacity, manufacturer, and model of a large-capacity consumer hard drive, the number of platters can range from one to five. A platter is split into several tracks. On each platter, the tracks create a complete ring. Each of these tracks is subdivided into an equal number of sectors. A partition is a segment of the disc that is separated from the rest of the disc (logical storage unit). A hard drive, as we all know, can have numerous partitions. Disk partitioning is used to treat a single physical disc drive as though it were many drives. This allows us to use various file system types (FAT and NTFS) on each partition, as well as separate operating system and user file partitions.

Partitions are divided into two categories:

- The first partition
- A larger partition

Each HDD can be partitioned into either four primary partitions or three primary partitions plus one extended partition. The extended partition can be subdivided into 24 logical partitions, while the primary partition will hold the operating system booting files; however, newer file systems can exceed the limit of 24 logical partitions.

Hard disk storage

Each platter, as previously stated, contains thousands of tracks, each of which is divided into sectors. The number of sectors on each track on the platter is the same. Millions of sectors can be stored on a hard disc. Each sector has a standard storage capacity of 512 bytes, but newer file systems can hold up to 4 KB. All Windows file systems organize hard discs into clusters based on cluster size (a cluster consists of several sectors). The cluster size is the smallest amount of disc space that a file can occupy.

Cluster size varies from 4 to 64 sectors, depending on the file system used and the partition size. With these default settings, a single cluster can store up to 64 KB of data. At any given time, each cluster can only hold data from one file. As a result, a text file of 11 KB will occupy one cluster (assuming the cluster size is 32 KB); the remaining storage space (21 KB) will remain unused and is referred to as slack space (see *figure 2.9*). Slack space can be used to store potentially incriminating data, or it can simply contain previously recycled leftover files that can be restored for evidence.

Figure 2.9: *Single file stored in a cluster at a time*

Disk Slack Checker (**www.karenware.com/powertools/ptslack**) tool calculates the amount of slack space on a hard disc, as illustrated in *figure 2.10*. HDD platters spin at a high rate to allow other parts to write and read data to and from platters; as a result, this type of disc is also known as a mechanical hard drive. A **solid-state drive** (**SSD**) is a modern type of hard drive that stores data using NAND flash memory (non-volatile), and this is what we will talk about in the next section.

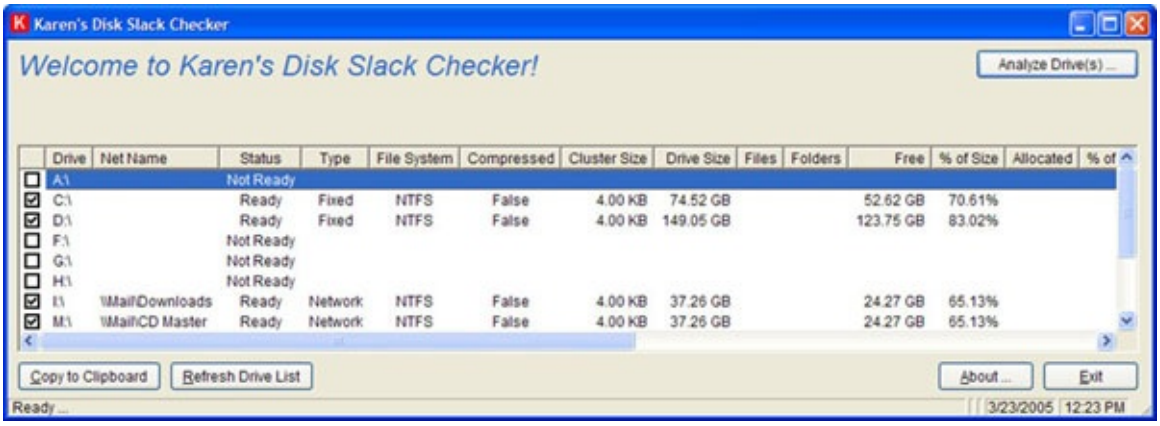


Figure 2.10: *Slack space on disk volumes*

SSD

We can think of SSDs as the modern version of hard disc drives. SSDs have no moving parts (platters) and save data in a series of NAND flash cells or microchips, similar to flash memory (NAND is composed of a set of transistors similar to the one used in RAM; however, this type of transistor does not need to be continually refreshed so it can retain its data, making it a type of non-volatile memory). To determine how to store, retrieve, and cache data, the SSD uses a controller (which is an embedded processor). In comparison to a traditional HDD (10), SSDs consume less power and have faster speeds due to the lack of mechanical moving parts. SSDs had a major disadvantage in the beginning, which was their limited number of write cycles; however, as technology advances, SSD manufacturers are working to solve this problem by developing more efficient algorithms that distribute data evenly across all cells, allowing all SSD cells to live longer without problems. SSDs are becoming increasingly popular in midrange and high-priced notebooks and workstations, and as technology advances daily, we can expect to see SSD prices drop. SSD units still have a low capacity when compared to HDDs, which is a problem. The price and capacity issues will almost certainly be resolved soon, so we can expect to see this type of hard drive in most notebooks and workstations, as well as servers. This will undoubtedly present a challenge for digital forensics investigators, as recovering deleted data from SSD is extremely difficult, if not impossible, in many cases.

Data Measurement	Size
1 Bit	1 or 0
1 Byte	8 Bits
1 Kilobyte (1 KB)	1,024 Bytes

1 Megabyte (1 MB)	1,024 KB
1 Gigabyte (1 GB)	1,024 MB
1 Terabytes (1 TB)	1,024 GB
1 Petabyte (1 PB)	1,024 TB
1 Exabyte (1EB)	1,024 PB
1 Zettabyte (1 ZB)	1,024 TB

Table 2.3: *Data measurement*

DCO and HPA

The HDD manufacturer creates a reserved area that is not accessible by the user, the OS, or the BIOS. This folder usually contains HDD-supporting utilities (such as diagnostic and recovery programs) as well as the installed OS's boot sector files. The **device configuration overlay** (**DCO**) is a reserved area on an HDD that is not supported by all HDD manufacturers; it is found after the HPA partition at the end of the disc. Both HPA and DCO can live on the same hard drive, but DCO should be created first. In terms of digital forensics, both the DCO and HPA areas will survive a full disc format, making them an ideal place for potential offenders to hide incriminating data. Many computer forensics suites can access and image these areas on a hard drive, and most hardware acquisition tools can also image them. Always check the capabilities of the computer forensic tool you want to use.

Considerations for data recovery

Data recovery from SSD is more difficult than data recovery from HDD, and it is sometimes impossible. When you delete a file on an HDD, for example, the data in the file is not immediately deleted; instead, the HDD only deletes the pointer to the file, marking its space on the disc as free. Only when the operating system needs to write new data to its location will the data in the subject file be deleted. When a user deletes a file, for example, the SSD will use the `TRIM` command, which works to delete a subject file instantly, leaving its location free for another file to occupy. The `TRIM` command is implemented differently by each operating system type: some OSs execute it immediately after a user deletes a file, whereas others execute it at regular intervals.

File system

The operating system uses a mechanism called a logical construction map to keep track of files in a partition. Before you can use a storage device to store data or install applications and operating systems, you must first initialize it by writing the file system's data structures to the drive (also known as formatting the drive). Windows OS installs on hard drives using either the FAT or NTFS file systems.

NTFS

Formatting a volume with NTFS results in the creation of several metadata files (see *figure 2.11*), including the master file table ($MFT), $Bitmap, $LogFile, and others, which contain information about all the files and folders on the NTFS volume.

Figure 2.11: *Formatted NTFS volume*

Each file in the NTFS file system is made up of many data streams: the primary stream (which has no name) contains the data that a user sees when they open a file. The alternative data stream is the other stream (ADS). Data streams of all files stored on an NTFS partition should be searched by digital forensics examiners, as they may contain hidden data. To learn how an offender can create an ADS file and hide secret data—or even malware—within it, as well as how an investigator can manually detect such files and use various third-party tools to do so.

- Create, Open, Detect, and Remove ADS: **https://www.minitool.com/partition-disk/alternate-data-streams.html**
- Identifying files hidden with ADS: **https://www.minitool.com/partition-disk/alternate-data-streams.html**
- Extracting ADS with Linux: **https://tmairi.github.io/posts/extracting-alternate-data-streams-with-linux/**

FAT

One of the oldest file systems still in use, **file allocation table** (**FAT**), comes in four flavors: FAT12, FAT16, FAT32, and FATX. All of Microsoft's previous Windows versions, including Windows NT, used FAT as the default file system. FAT is more portable than NTFS because it can be used on a variety of devices. FAT is used in many digital cameras, SD cards, smartphones, USB thumb drives, and embedded devices, for example. Unlike NTFS, which can only be read by Windows OS, storage devices formatted with FAT can be read across multiple platforms. In many ways, NTFS outperforms FAT; for example, NTFS supports large file sizes and includes a file encryption feature. Microsoft uses NTFS to install its modern Windows OS versions, such as Windows 8 and 10, as well as the new Server editions.

Environment for computing

Your choice of how to capture digital evidence will be heavily influenced by your computing environment. In the coming years, we can expect to see a significant shift from centralized computer architectures to non-centralized or distributed computer architectures as technology advances and Internet speeds increase. The most common computing environments are listed as follows:

- **Environment for personal computing**: This is probably the most common these days. All programs are installed locally and run on the same machine in this environment. Data is also saved to the local hard drive of the machine. Personal computing environments include laptops, desktops, printers, tablets, and even smartphones. Because the location of evidence is tied to the subject device only, this environment is the easiest to deal with if a personal device becomes a part of a criminal investigation.
- **Environment for client-server computing**: There are two machines in this environment: a client (for example, a personal computer, laptop, or tablet) and a server. The client uses an HTTP connection to request data from the server, and the server responds with data. The e-mail server you use to get your e-mails is an example of such an environment.

- **Environment for distributed computing**: Applications are installed and run on multiple computers in this environment, allowing one application to split its functions into multiple components, each of which runs on its computer. In this type of environment, data storage is also distributed, and clients and other applications must communicate with remote servers via networks to access data or run programs. In such an environment, capturing digital evidence is difficult because users' private data and logs may be scattered across multiple remote servers, which may be located in different geographical areas and under different jurisdictions. In such environments, the volume of data (and logs) that must be investigated is also an issue, as the volume can be enormous in many cases.

Cloud computing

Cloud computing is a modern technological model that allows a service provider to deliver various computing services to users over the internet as a result of the explosive growth of the internet and online communications. For example, rather than buying an external hard drive to store your backup data, you can store it for a small fee on a cloud provider. The cloud provider will be in charge of user data management in the cloud (for example, making backup copies and protecting this data from malicious software and cyberattacks). Cloud computing is not just for storing user data; it is also being used by businesses to cut IT infrastructure costs. Instead of purchasing a software license for each user individually, a company can use a cloud computing service that provides needed applications (such as the MS Office suite) for its work. When using expensive software such as SQL Server and Windows Server OS, the cost appears to be higher; however, paying for such software on a usage basis while in the cloud is more cost-effective than installing it on-premises. As we will see next, different cloud computing models are used by businesses.

Software as a service (SaaS)

In this model, a user purchases a cloud computing account and then chooses which applications he or she wants to install. Instead of using these applications on a local machine, a user will do his or her work on a cloud (remote) server. Google Apps for Education and Microsoft Office 365 are two examples of such services.

Platform as a service (SaaS)

This model is popular among software/Web development companies, in which a customer—for example, a Web development company—pays for an account with a cloud service provider that provides a customized environment based on the needs of the client (for example, to install needed Web development tools, prepare the development and testing environment, and so on). This enables a customer to begin work quickly and at a low cost.

Infrastructure as a service (SaaS)

In this model, a cloud provider rents out the client's required hardware (physical server and data center hardware) over the internet. The client purchases and installs required applications and operating systems and then configures them to meet business requirements. Web hosting companies and businesses typically use this service for data storage, backup, and recovery outside of their offices. What we are interested in

learning from this discussion is how cloud computing services will make it more difficult for law enforcement to investigate criminal cases. For example, if a UK citizen is a suspect in a criminal case and his data is uploaded to a cloud storage provider in Singapore, can the UK police force the Singaporean provider to hand over a copy of the user data?

Windows versions

As a digital forensics investigator, you must know how to collect current Windows OS information so that you are aware of some differences between versions during the acquisitions and analysis phases. To determine the current Windows version on a computer running Windows 8 or later, follow these steps:

1. Press and hold the Windows key while also pressing and holding the *R* key.
2. In the search box, type "**`winver`**" and press *Enter*.
3. You should see the Windows version and build number (see *figure 2.12*).

Figure 2.12: *Windows version "About"*

In Windows 7, go to Control Panel System and look under Windows edition for current version information to find out about your operating system.

Internet protocol (IP) address

During your investigations, you will almost certainly come across information that requires an understanding of the addressing schema used on the internet and many private networks, so knowing IP protocol is a must for any digital investigator. We will go over the concept of an IP address and how computing devices connect to the internet in this section.

Getting an IP address

When connected to an IP network, an IP address is a unique address that identifies a computing device. Because an IP address is similar to a fingerprint, no two devices on the same IP network can have the same IP address. IP is frequently used in conjunction with the **Transmission Control Protocol** (**TCP**), which allows a computing device to establish a virtual connection between a destination and a source to exchange data. IP version 4 and IP version 6 are the two IP addressing schemas currently in use. The IP v4 protocol is the most widely used on the planet; it is currently used by the majority of online services. IP v4 uses a 32-bit address schema and can hold up to 4.3 billion addresses; however, due to the rapid growth of the internet and the growing number of IoT devices, this number has become limited and may soon run out. As a result, another standard known as IP v6 was developed, which can accommodate more than 7.91028 times as many addresses as IP v4.

There are two types of IP addresses: public and private.

- **Public IP addresses**: These are assigned by your ISP and allow direct Internet access. Each IP address is unique. An e-mail server, for example, requires a globally unique public IP address.
    - Static IP addresses do not change. This address, like your phone number, is fixed and will remain the same as long as the ISP reserves it for you.
    - Dynamic IP Addresses change over time. This one changes each time the user connects to the internet. The ISP uses a protocol called **Dynamic Host Configuration Protocol** (**DHCP**) to assign IP addresses automatically to their subscribers.
- **Private IP addresses** (also known as local IP addresses): This is a non-internet-facing IP address for devices usually sitting behind a routing device. All devices existing in a closed network (for example, Home or school networks) will use private IP addresses. These addresses are usually assigned automatically using the DHCP of the router.

Conclusion

In this chapter, we have covered important technical concepts about computers that must be well understood by any digital forensic examiner. We describe how computers store and represent data digitally, the concept of operating system file structure and its types, and hash algorithms and how we can use them to verify the authenticity of any piece of digital data. In the upcoming chapter, we will discuss how digital forensics works with user devices and investigate system disks, hard drives, and file systems.

CHAPTER 3

Hard Disks and File Systems

Introduction

Undertaking a digital forensics investigation demands a deep grasp of some of computing's most fundamental technological concepts. To discover and manage digital evidence, you must understand how information is stored in computers, number theory, how digital files are constructed, and the many types of storage units and their differences. These fundamental topics will be covered in this chapter. Computers store, process, and portray digital data in a certain way, as explained in this chapter.

Structure

In this chapter, we will cover the following:

- Different number system
- Encoding schema
- File carving and structure
- File metadata
- Hash analysis
- System memory
- Storage
- Filesystem
- Cloud computing
- Windows OS
- Networking

Objectives

We will explore how a system represents data in this part, as well as typical numbering systems and the principal encoding strategy used by machines to generate human-readable text. Let us start with the standard numbering scheme.

Hard disk and file systems

Computer forensics is becoming increasingly important as the number of crimes involving computers and the internet rises. Computer forensic tools have been created to aid computer forensic experts in properly investigating digital crimes. For today's intelligence, police departments, and military groups, digital

forensics is a developing and significant subject of research. As more data is kept digitally, the demand for the capacity to evaluate and filter this data for meaningful evidence has become increasingly complicated. This information is used by digital forensics to analyze and evaluate digital data as evidence. Digital forensics is a relatively recent field. As computers grew increasingly ubiquitous in homes and companies, police saw PCs containing forensic evidence more frequently. The use of computer forensic examination has proven beneficial in a wide range of judicial processes, and the area of digital forensic analysis has seen a fast expansion in recent years. Digital forensics is used to examine not only computerized crimes such as network penetration, data fabrication, and unauthorized material distribution via digital services but also crimes where evidence is saved in any electronic medium on any digital device. The rapidly expanding storage space of media such as hard drives is compounding the greater use and collection of digital evidence. Advanced data compression and duplication technologies are extensively used in key business storage applications, indicating that the fast development in storage capacity is not limited to the domain of forensics.

Hard drives and filesystems are the primary sources of data storage; therefore, understanding them is critical when investigating a computer-based crime. To avoid being tracked, people frequently remove their traces after committing a crime using a computer. When investigating a computer-based crime, retrieving deleted data from hard drives and studying filesystems is critical. A file system, also known as file management or FS, is a method of controlling how and where data is stored on a storage disk. It is a logical storage component that encloses files that are divided into groups called directories. It is conceptual to a specific user and computer-related; as a result, it manages the internal operations of a disk. The directories can contain files and additional folders. Although Windows supports a variety of file systems, NTFS is the most popular in today's world. It would be impossible to have two files with the same name, as well as to remove installed programs and recover specific files without file management. Files would also be unorganized without a file structure. Because files are frequently managed in a hierarchy, the file system allows the user to view a file in the current directory.

Regardless of the type of usage, a disk (for example, a hard disk drive) has a file system. It also includes information on file size, file name, file location fragment information, and where disk data is stored, as well as how a user or application can access the data. The file system is in charge of metadata, file naming, storage management, and directory/folder management. The file system is the technique for storing and accessing file contents, such as data and applications, online. The high-level intricacies of file systems are covered in this article, as well as related subjects, including the disk cache, the file system connected to the kernel, and the user-level APIs that leverage file system capabilities. It will teach you all you need to know about how a file system works in general. The file system is the most important part of the operating system. It creates, manipulates, stores, and retrieves data. A file system is, at its most basic level, a method of managing data on a secondary storage media. Underneath and above the file system, there are several levels.

Documents are saved in sectors on a storage device, and data is stored in blocks, which are groups of sectors. The file system determines the size and location of files, as well as which sectors are ready to be used. FAT and NTFS file systems are found in a variety of operating systems other than Windows. However, Apple's products (such as iOS and macOS) use HFS+ as their operating system, which is compatible with a wide range of file systems.

File systems

When referring to partitions, the term "*file system*" is sometimes used. "*On the hard drive, two file*

systems are available," for example, does not necessarily imply that the drive is partitioned into two file systems, NTFS and FAT. However, it means that there are two separate partitions on the same physical disk.

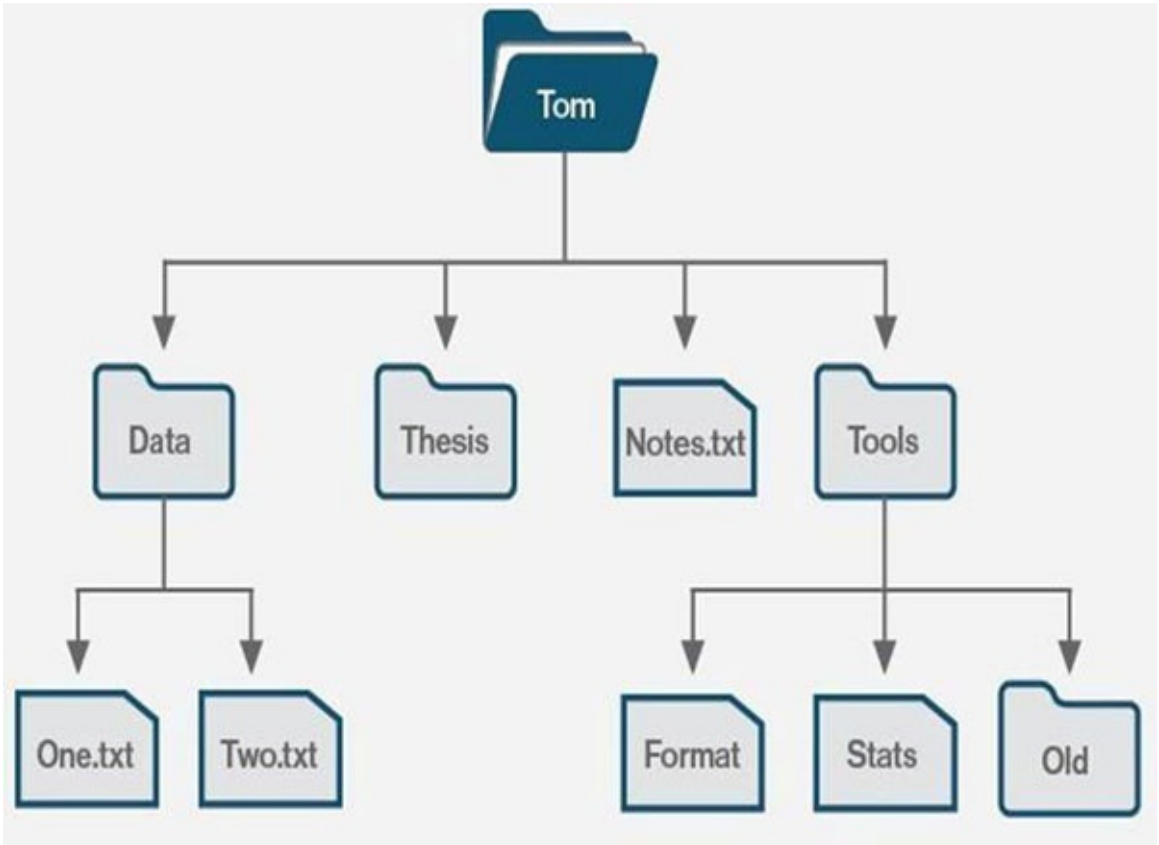


Figure 3.1: *File system*

Most applications you come into contact with require a file system to function; therefore, each partition should have one. Furthermore, because programs are file system-dependent, if a program is built for macOS, you will be unable to use it on Windows.

The following are some examples of file systems:

- **FAT**: **File Allocation Table** (**FAT**) is a file system designed specifically for hard drives. It stands for file allocation table and was first introduced in 1977. It is used for each cluster access into the file allocation table and uses 12 or 16 bits (FAT). It helps Microsoft operating systems manage files on hard drives and other computer systems. It is also commonly found in devices such as digital cameras, flash memory, and other portable devices, where it is used to store file information. It also helps to extend the life of a hard drive by reducing hard drive wear and tear. Later versions of Microsoft Windows, such as Windows XP, Vista, 7, and 10, no longer use FAT in favor of NTFS. The different types of FAT are FAT8, FAT12, FAT32, and FAT16 (for file allocation table).
- **GFS**: **Global File System** (**GFS**) was first developed at the *University of Minnesota* and allows multiple computers to work together as a single machine. However, Red Hat is now responsible for its upkeep. When two or more computers are separated by a large physical distance and are unable to send files directly to each other, a GFS file system allows them to share a group of files directly. With the help of a global file system, a computer can organize its I/O to preserve file systems.
- **Hierarchical file system (HFS)**: It is the file system that a Macintosh computer uses to create a

directory when a hard disk is formatted. Its primary function is to organize and store files on a Macintosh hard disk. Since the release of OS X, Apple has been unable to support writing to or formatting HFS disks. Furthermore, because HFS is a Macintosh format, Windows computers do not recognize HFS-formatted drives. Windows hard drives are formatted using the WIN32 or NTFS file systems.

- **NTFS**: The NT file system stores and retrieves files on the Windows NT operating system as well as other Windows versions such as Windows 2000, Windows XP, Windows 7, and Windows 10. The New Technology File System is another name for it. It offers better methods of file recovery and data protection than the FAT and HPFS file systems, as well as a number of improvements in terms of extendibility, security, and performance.
- **UDF**: A UDF is a file system that stands for Universal Disk Format and was created in 1995 by **Optical Storage Technology Association** (**OSTA**) to ensure data consistency across multiple optical media. It works with both CD-ROMs and DVD-ROMs and is compatible with all major operating systems. It is now used in the packet writing process for CD-Rs and CD-RWs.

FAT32 is an older file system that is less efficient than NTFS and has a smaller feature set, but it is more compatible with other operating systems. Although exFAT is a contemporary alternative for FAT32, and it is supported by more devices and operating systems than NTFS, it is not as widely used as FAT32. By default, Windows uses NTFS, a contemporary file system. The NTFS file system is installed when you install Windows. You will not run into any file or partition size constraints with NTFS because they are supposedly so large. Although it first debuted with Windows NT, NTFS first arrived in consumer windows systems with Windows XP. The FAT32 file system is the oldest of the three file systems supported by Windows. It was first introduced in Windows 95 to replace the FAT16 file system that was used in MS-DOS and Windows 3. The age of the FAT32 file system offers both benefits and drawbacks. FAT32 has a significant benefit in that it is the de-facto standard due to its age. Purchased flash drives are frequently formatted in FAT32 for optimal compatibility with current PCs and just current PCs but also other devices such as gaming consoles and anything with a USB port.

The exFAT file system was first released in 2006 and later brought to previous versions of Windows with Windows XP and Vista upgrades. exFAT is a lightweight file system similar to FAT32 that is geared for flash drives but without the added functionality and overhead of NTFS and the constraints of FAT32. ExFAT, like NTFS, provides very large file and partition size restrictions, allowing you to store data considerably bigger than FAT32's 4 GB limit. Internal hard drives should use NTFS, whereas flash devices should use exFAT. If exFAT is not supported on the device, you need to use it, and you may need to format the external drive using FAT32. Any digital file is saved to a storage media of a certain size. In reality, each storage unit is a linear place for reading or reading books' digital data. Each byte of data on it has an address that refers to its offset from the storage start. A grid of numbered cells can be used to represent storage. Any object saved in the storage is given its own set of cells.

Disk/tape file systems, network system files, and special-purpose system files are the three kinds of file systems.

- **Disk file systems**: A disk file system makes use of disk storage media's ability to solve data randomly in a short period of time. Further factors to consider include the speed with which data is accessed after the original request, as well as the possibility that additional evidence may be sought. This allows multiple users (or applications) to retrieve different data on the disk regardless of the data's sequential placement.
- **Flash file system**: Takes into account flash memory devices' unique capabilities, performance,

and limitations. A disk file system may often be used as the underlying storage medium for a flash memory device, but it is much preferable to use a file system expressly built for a flash device.

- The tape file system is a file system and tape format for storing self-describing files on tape. Magnetic tapes are sequential storage mediums that take substantially longer to retrieve random data than disks, making the construction and maintenance of a general-purpose file system difficult. There is usually a master file directory and a map of used and free data areas in a disk file system. Any modifications, additions, or deletions to files need updating the folder and the used/free maps. This approach works well for disks as random access to data areas is measured in milliseconds. To wind and unwrap potentially extremely long reels of material, the tape needs linear motion. Moving the read/write head from one end of the tape to the other might take anything from a few seconds to many minutes.
- **Database file systems**: The notion of a data system file system is another approach for file management. Files are recognized by their attributes, such as kind of file, subject, author, or other rich information, instead of or in conjunction with hierarchically organized management. IBM DB2 for I (previously known as DB2/400 and DB2 for i5/OS) is a database file system that runs on IBM Power Systems and is part of the object-based IBM I OS (formerly known as OS/400 and i5/OS). It has a single-level store.
- **Transactional file systems**: Some applications need "*all-at-once*" updates to numerous files. A software installation, for example, may create program binaries, modules, and configuration data. The application may become inoperable if the software installation fails. If a crucial system tool, such as the command line, is upgraded during the installation, the whole system may become useless. The isolation promise, which specifies that activities inside a transaction remain concealed from other processes on the computer until the transaction is confirmed and that conflicting operations on the network will be appropriately serialized with the transaction, is introduced by transaction processing. Transactions also offer the atomicity guarantee, assuring that activities inside a transaction are either fully committed or the operation may be canceled, with any incomplete results being discarded by the system.
    - The network file system is a storage device that functions as a client for a remote file access interface, giving access to the files on a server. Programs using local interfaces may seamlessly build, maintain and access hierarchical folder structures in distant network-connected systems.
    - The shared disk file system is one in which a number of connected devices (typically servers) all have connections to the same external disk subsystem (usually a SAN) (usually a SAN). The file system adjudicates access to that subsystem, eliminating write collisions. Examples include GFS2 from Red Hat, GPFS from \sIBM, SFS from DataPlow, CXFS from SGI, and StorNext from Quantum Corporation.
- **Special file systems**: A customized file system displays non-file aspects of an operating system as files so they may be acted on using file system APIs. This is most typically done in Unix-like operating systems, although devices are given file names in several non-Unix-like operating systems as well.
- **Minimal file system**: The late 1970s witnessed the emergence of the microcomputer. Disk and digital tape devices were too pricey for enthusiasts. An affordable basic data storage device was designed that used standard audio cassette tape. When the system required to write data, the user was told to \spress "RECORD" on the cassette recorder, then hit "*RETURN*" on the keyboard to

tell the computer that the cassette recorder was recording. The system wrote a sound to establish temporal synchronization, then modulated noises that encoded a preface, the data, a checksum, and a \ssuffix. When the system required to read data, the user was directed to push "*PLAY*" on the cassette recorder.

- **Flat file system**: There are no subdirectories in a flat file system. Due to the limited amount of data capacity available when floppy disk media was initially introduced, this form of file system was suitable. CP/M computers used a flat file system, in which files were allocated to one of 16 user regions, and general file operations were limited to working on one instead of all. These user areas were just unique properties connected with the files; hence, no explicit quotas for each of these areas were required, and files may be added to groups as long as there was still spare storage space on the disk.

The physical file system, which is the third layer, is responsible for buffering and memory management. It is concerned with the storage device's physical operation and processes physical blocks that are being read or written, as presented in *figure 3.2*. This layer also interacts with the channel and device drivers to drive the storage device.

File Type	Usual extension	Function
Executable	exe, com, bin or none	ready-to-run machine-language program
Object	obj, o	complied, machine language, not linked
Source code	c, p, pas, 177, asm, a	source code in various languages
Batch	bat, sh	commands to the command interpreter
Text	txt, doc	textual data documents
Word processor	wp, tex, rrf, etc.	various word-processor formats
Library	lib, a	libraries of routines
Print or view	ps, dvi, gif, pdf	ASCII or binary file
Archive	arc, zip, tar, gz	related files grouped into one file, sometimes compressed.

Figure 3.2: *File types names and extensions*

Memory capacity and cost are inversely related to speed in the Memory Hierarchy. The devices are grouped from fast to slow or from register to Tertiary memory in this diagram. Inside the CPU, there are registers. They have the quickest access time since they are inside the CPU. Registers are the most costly and have the smallest storage capacity, usually measured in kilobytes. Flip-flops are used to implement them. Cache memory is used to store program parts that the CPU accesses often. It is costly and smaller in size, usually measured in Megabytes, and it is accomplished by using static RAM. Through an I/O

processor, it interfaces directly with the CPU and other memory devices. *Figure 3.3* shows the memory hierarchy of the storage. Main memory is less costly than cache memory and has a bigger storage capacity, usually measured in Gigabytes. Dynamic RAM is used to implement this memory. At Level 3, secondary storage devices such as magnetic disks are present. They are used to store backup data. They are less expensive than main memory and have a bigger capacity of a few TB. At Level 4, tertiary storage systems such as magnetic tape are present. They are the cheapest and biggest in size and are used to store removable data (1–20 TB). The memory hierarchy is the process of organizing several types of storage on a computer based on access speed. CPU registers, which are the quickest to read and write to, are the best-performing memory at the very top. Cache memory is next, supported by traditional DRAM memory, and finally, disk storage with various degrees of performance, such as SSD, optical, and magnetic disk drives. To close or eliminate the performance gap between the CPU and memory, hardware designers are increasingly depending on memory at the top of the memory hierarchy. This is accomplished by creating bigger cache hierarchies (which processors can reach considerably quicker), minimizing the need for slower main memory.

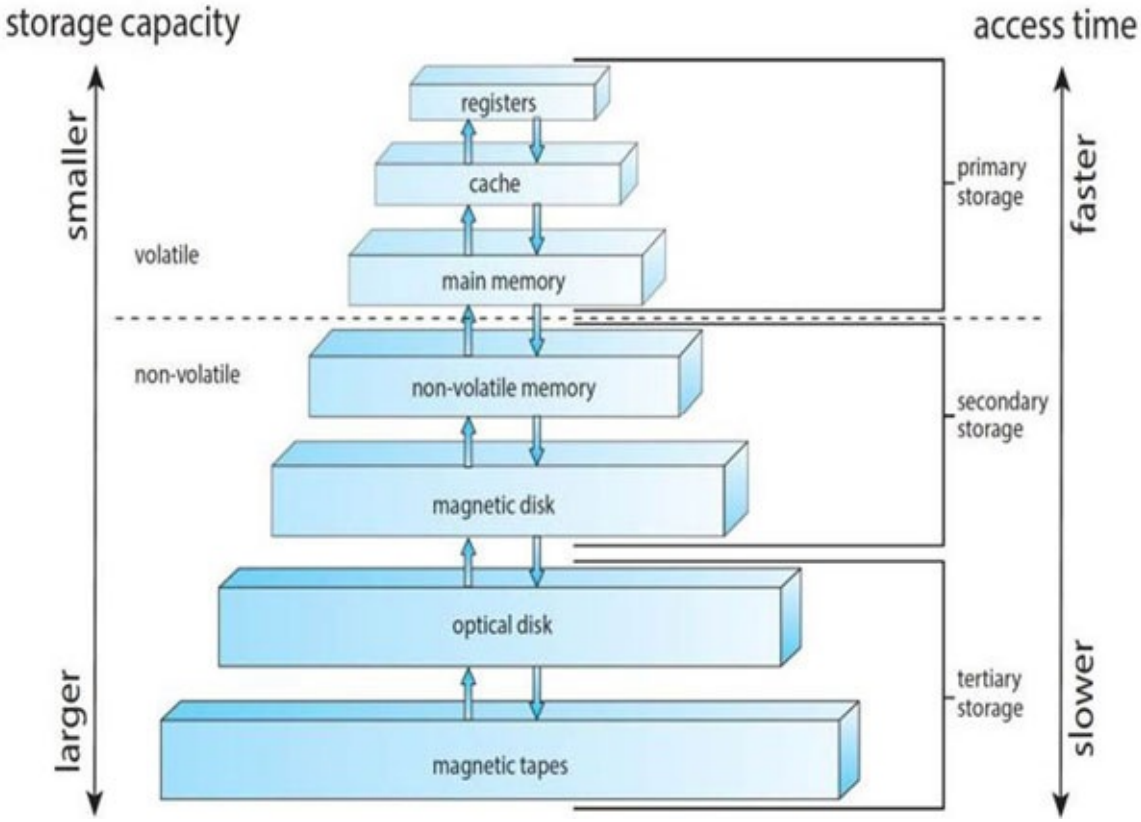


Figure 3.3: *Memory hierarchy of storage*

Hard disk

Despite the explosive growth of devices such as smartphones and PDAs in the latest days, the hard disk drive continues to be a common subject of computer forensic analysis. The core design of the hard disk has remained mostly unchanged since its debut; however, revisions to individual elements have resulted in significant increases in speed, capacity, and reliability. The four key components of the internal structure of a conventional hard disk drive are briefly detailed as follows. The Disk Platter, Head Arm, Chassis, and Head Actuator are the four components. The Chassis is the portion of the hard disk that serves as its foundation and practical support. Platters are circular disks set on a central pole termed as a spindle and stacked one on top of the other. The platters have a coating on either side that allows them to hold information magnetically. Tracks are concentric circles that hold data on both the upper and below

sides of the disk.

When a disk is switched on, and data has to be read or written, the platters spin at a very fast speed, allowing the actuator arm and its accompanying components to read the proper region of the disk. A small device called a read/write head must be situated just above the platter surface in order to read or write data from or to a disk, i.e., to or from a platter. The read/write head is coupled to a "*head slider*," which is itself attached to the head arm, in order to position it accurately. The number of platters controls the number of head arms in a hard disk, with each arm being used to place a read/write head on the opposite side of a platter. An actuator arm, actuator assembly, or head assembly is a structure that connects the head arms. The assembly pivots on an axis that allows the read/write heads to move around the platter surface in order to position them correctly on the platters. This movement, together with the platter's rotation, permits the heads to be properly positioned.

A hard disk, as illustrated in *figure 3.4*, is a sealed device that consists of a stack of platters. Hard drives can be installed either horizontally or vertically. The hard disk is positioned horizontally in this example. Above and below each disk, magnetic read/write heads are positioned. The disk heads move inward toward the center of the platters and outward toward the edges as they rotate. The drive heads will be able to access the whole surface of each platter in this manner. Data is stored on a hard drive in thin, concentric bands. A drive head in one position may read or write a track, which is a circular ring or band. On a 3.5-inch hard disk, there might be over a thousand tracks. Sectors are individual sections of each track. The smallest physical storage unit on a disk is a sector, which is generally always 512 bytes (0.5 kB) in size.

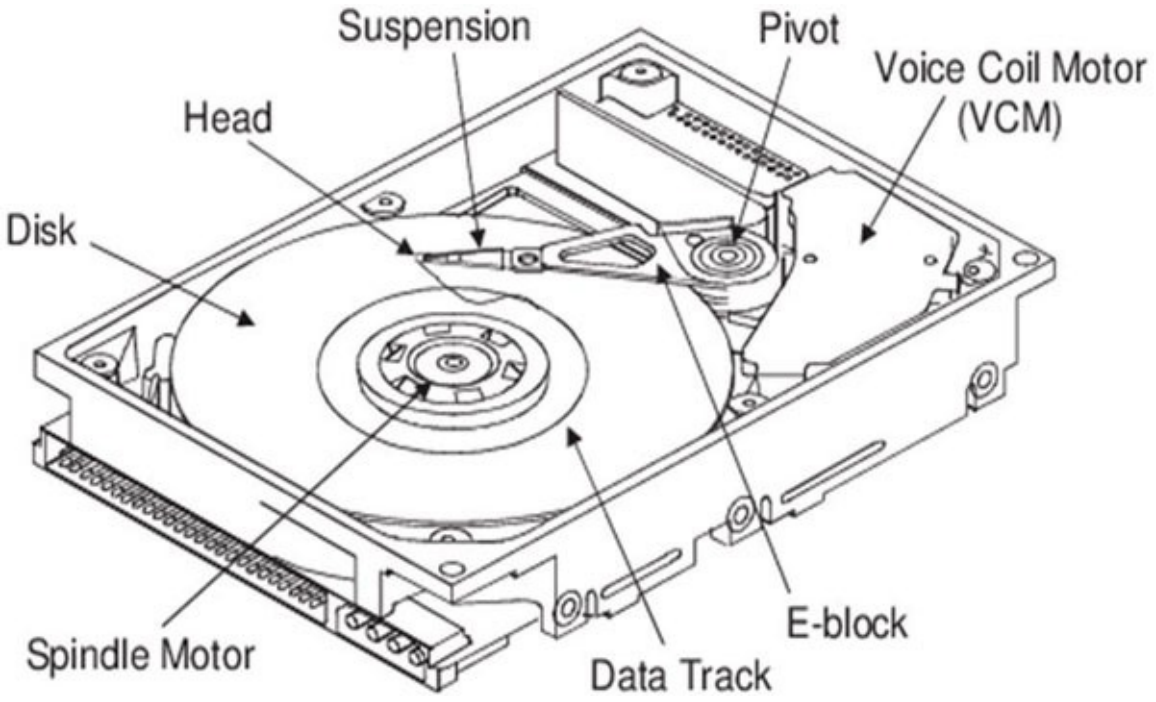


Figure 3.4: *Parts of hard disk*

A cylinder/head/sector nomenclature is used in the construction of earlier hard drives (i.e., prior to Windows 95). When all of the drive heads on the disk are in the same position, a cylinder is produced. The tracks create a cylinder when placed on top of each other. With contemporary hard drives, this approach is gradually being phased out. Because this is how operating systems from Windows 95 onwards like to function, all new drives use a translation factor to make their real hardware layout look continuous. Tracks are conceptual rather than physical structures in the eyes of a computer's operating system, and they are created when the disk is low-level formatted.

The platters in the stack spin at the same rate. While close to the disk's center, the drive head reads from a slower-moving surface than the disk's outer borders. Tracks on the periphery of the disk are less densely packed with data than tracks in the center of the disk to compensate for the physical difference. Because of the varying data density, the same number of information may be read in the same amount of time from any drive head location. A typical plan is used to fill the disk space with data. The operating system does not have access to one side of one platter, which is restricted to hardware track-positioning information. As a result, data may be stored on three sides of a disk assembly with two platters. During factory assembly, recording data is entered into the disk. This data is read by the system disk controller, which then places the drive heads in the correct sector situation.

Since 512 is a power of two, a sector, which is the lowest physical storage facility on the disk, is nearly always 512 bytes. Because the most fundamental of computer languages has just two states—on and off—the Number 2 is used. The factory track-positioning information is used to mark each disk sector. Sector identity writes data to the region just before the sector's contents, and it indicates the sector's beginning address. A continuous series, or all data in a sequence-stored end-to-end in a single phrase, is the best approach for storing a file on a disk. One or more successive sectors can form a cluster. The number of sectors is always divisible by two. A cluster might be as little as one sector or as large as eight sectors. The only odd number of sectors that may make up a cluster is 1. It could not be five sectors or an even number that is not divisible by two. It would not be ten sectors, but rather eight or sixteen.

Clusters are so named because the space is set aside for the data. This technique prevents overwriting of the saved data. If the file is later expanded to 1,600 bytes in size, additional two clusters are allocated, allowing the whole file to be stored in four today's home computers that use disks that revolve at a steady speed. The tracks at the disk's edge are less densely filled with data than those toward the disk's core. As a result, even if the speed of the disk surface is quicker on the tracks positioned further out from the center of the disk, a fixed quantity of data may be read in a consistent length of time. Modern disks set aside one side of one platter for track positioning data, which is recorded to the disk during disk manufacture at the factory clusters. The operating system does not have access to it. When the heads travel to another point on the disk, the disk controller uses this data to fine-tune the head locations. When a site has information about the track location, it cannot be used for data. As a result, a disk assembly with two platters has three sides available for data.

Hard disk forensics

Hard Disk Forensics is the process of extracting actionable data from computer storage in order to use it as evidence in criminal cases. The procedure frequently requires retrieving and recreating material that has been erased or destroyed. Furthermore, crooks' and fraudsters' personal computers may include crucial pieces of evidence that might lead to a quick resolution of a criminal proceeding and punishment.

Texial has a cutting-edge cyber forensics lab and uses some of the industry's finest and sharpest brains. We specialize in recovering forensic data from hard disks that have been physically destroyed by criminals to hide their tracks, no matter how well-hidden or erased it may be. There are several benefits of hard disk forensics. Some important ones are as follows:

- Information from hard drives is searched and extracted.
- Broken data from computer storage is retrieved and rebuilt.
- Let us know what is going on. Retrieval of data from hard drives that are concealed or encrypted. Hackers take advantage of it.

The study of retrieving forensic evidence from digital storage media, such as hard disks, USB devices, Firewire devices, CDs, DVDs, Flash drives, and floppy disks, is known as disk forensics.

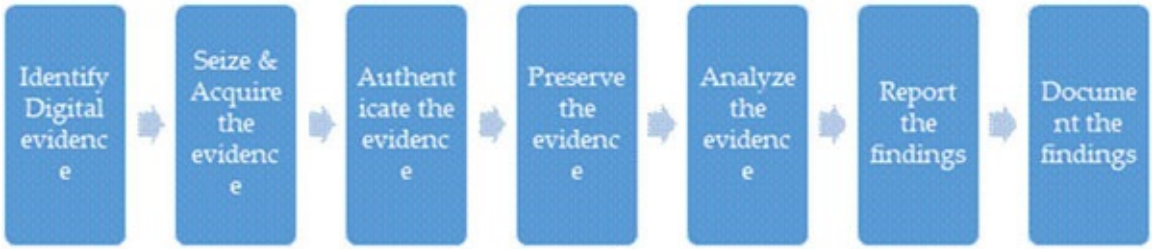


Figure 3.5: *Steps involved in hard disk forensics*

The initial phase is also known as the reconnaissance phase, in which our cyber security experts gather quite much information about the target as possible before initiating an assault. Identification of storage devices at the scene of the crime, such as hard disks with IDE/SATA/SCSI interfaces, CDs, DVDs, Floppy disk, Mobiles, PDAs, flashcards, SIM, USB/Firewire disks, Magnetic Tapes, Zip drives, Jazz drives, and so on, is the first step in disk forensics. These are some of the digital evidence sources. The next step is to seize the storage media in order to acquire digital evidence. This stage is carried out at the crime site. Using a suitable cyber forensics program, the hash value of the storage devices to be seized is computed in this phase. A hash value is a one-of-a-kind signature created by a statistical hashing algorithm that is based on the storage media's content. The storage medium is securely packed and retrieved for further processing once the hash value is computed. It is also known as the reconnaissance phase, in which our cyber security experts gather quite as much information about the target as possible before initiating an assault.

"*Never work on original evidence*," is one of Cyber Forensics' cardinal laws. To guarantee that this criterion is followed, an identical replica of the original evidence must be generated for analysis and gathering of digital evidence. The process of making this precise copy, in which the source storage media is write-guarded and bitstream copying is used to verify that all data is transferred to the destination medium, is known as acquisition. In most cases, source media is acquired in a Cyber Forensics lab.

In the Cyber Forensics laboratory, the evidence is authenticated. Both the source and destination media's hash values will be checked to confirm that they are the same, ensuring that the destination media's content is an identical replica of the source media. Electronic evidence can be tampered with or altered without leaving a trace. The actual evidence should be stored in a secure location away from strong electromagnetic and radiation sources once it has been acquired and authenticated. Another copy of the picture should be made and kept on acceptable media or in a dependable mass storage system. As a mass storage medium, optical media can be used. It is dependable, quick, has a longer life lifetime, and can be reused.

In the Cyber Forensics process, verifying evidence before beginning analysis is critical. Before beginning an analysis, this is done at the Cyber Forensics lab. The evidence's hash value was calculated and contrasted to the hash value obtained during acquisition. There is no difference in the essence of the evidence if both values are equal. There is a change in content if both are different. Verification results should be documented accurately. The type of examination sought by a court or investigating body should be reflected in the case analysis report. It should include the following information: the nature of the case, the examination sought, the physical items and hash values, the outcome of evidence verification, the analysis done and digital evidence obtained, the examiner's observations, and the conclusion. Non-technical people should be able to grasp the substance of the report if it is presented in simple language and in a detailed manner.

Analyzing the registry files

Because most users are unaware of the system's purpose, they leave traces of their activities on the system, particularly in the register. Analyzing such data provides the forensic investigator with preliminary knowledge about the system environment as well as guidance for future investigation. Extracting essential evidence is extremely tough and time consuming due to the registry's complicated structure. This package will automate the work of Windows 7 Registry examination for forensics investigators to solve these challenges. This will enhance standard registry analysis, giving investigators an advantage in forensic analysis by concealing irrelevant information and emphasizing crucial information from the registry, as well as reducing the amount of time required on Windows Registry analysis. RegAlyzer is a registry browsing and editing tool. It was intended to fill up several gaps in the previous regedit tool, such as support for unusual value types, background, pattern matching search, improved bookmarking, and `displaying.reg` files in the familiar layout, as well as a history view. RegAlyzer 2, which is now in beta, will have a multi-tabbed interface, value interpretations, permanent names for variables, and hive snapshots to track changes.

Making several copies of material that will be used later in deceased forensics analysis is critical. This module aids in the procurement procedure by permitting the creation of raw copies/images of hard drives and USB pen drives that may be used for forensics investigation afterward. Because the Alternate Data Streams feature of the NTFS file system enables people to hide data in the file system, forensic investigators must keep this in mind while investigating windows PCs using the NTFS file system. Alternate Data Streams in deleted files are equally important, although they may be ignored because forensic professionals are unfamiliar with them. This module aids in the discovery of data buried in Alternate Data Streams at different locations, such as files, folders, and partitions, as well as data contained in deleted file Alternate Date Streams.

In computer forensics, the data saved in files is the primary source of evidence. Such files on the disk are managed via the file system. A perpetrator can erase evidence from a hard drive by removing evidence-related files. It is critical for forensic investigators to recover evidence that has been erased by the defendant. This component aids forensic investigators in recovering deleted files from NTFS-formatted hard drives and USB pen drives and analyzing the time when the computer system is turned on, and the person who is signed in at that time might reveal important knowledge that can be linked to other proof. Example: On an NTFS file system, the deleted file time may be connected with the person who was logged in at the time to determine who deleted the files. By providing a timeline of user log-on and log-off events, this module assists forensics investigators in analyzing user behavior on the computer. *Figure 3.6* shows the comparison of HDD, SDD, and Flash memory.

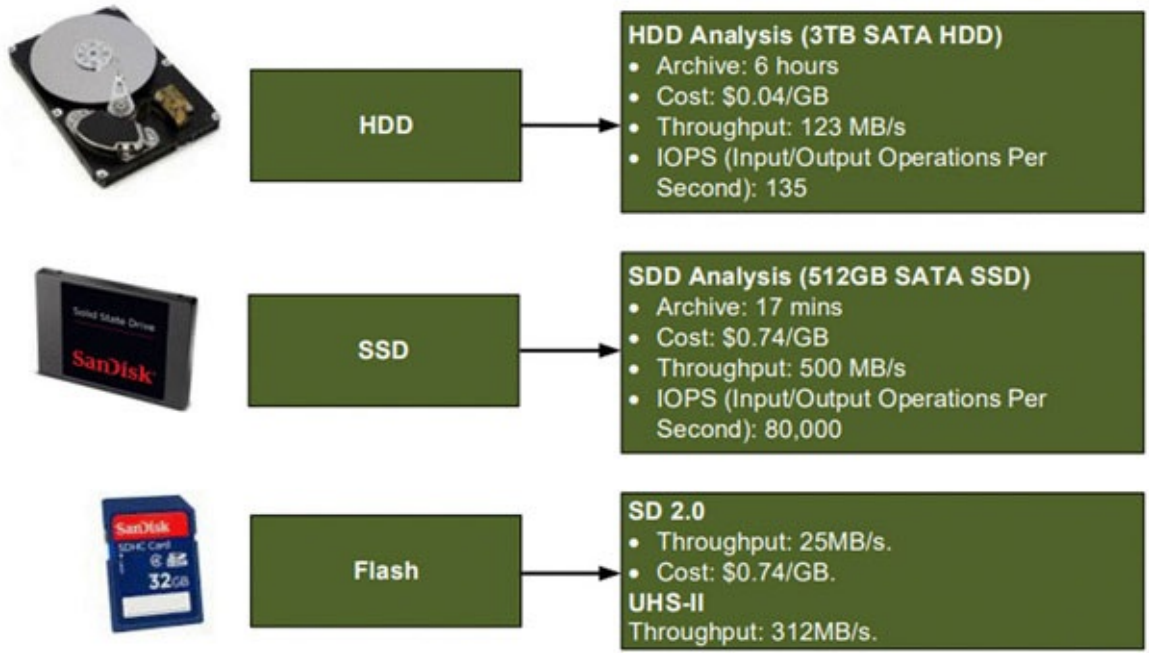


Figure 3.6: *HDD versus SSD versus Flash memory*

Figure 3.7 shows the file allocation table structure.

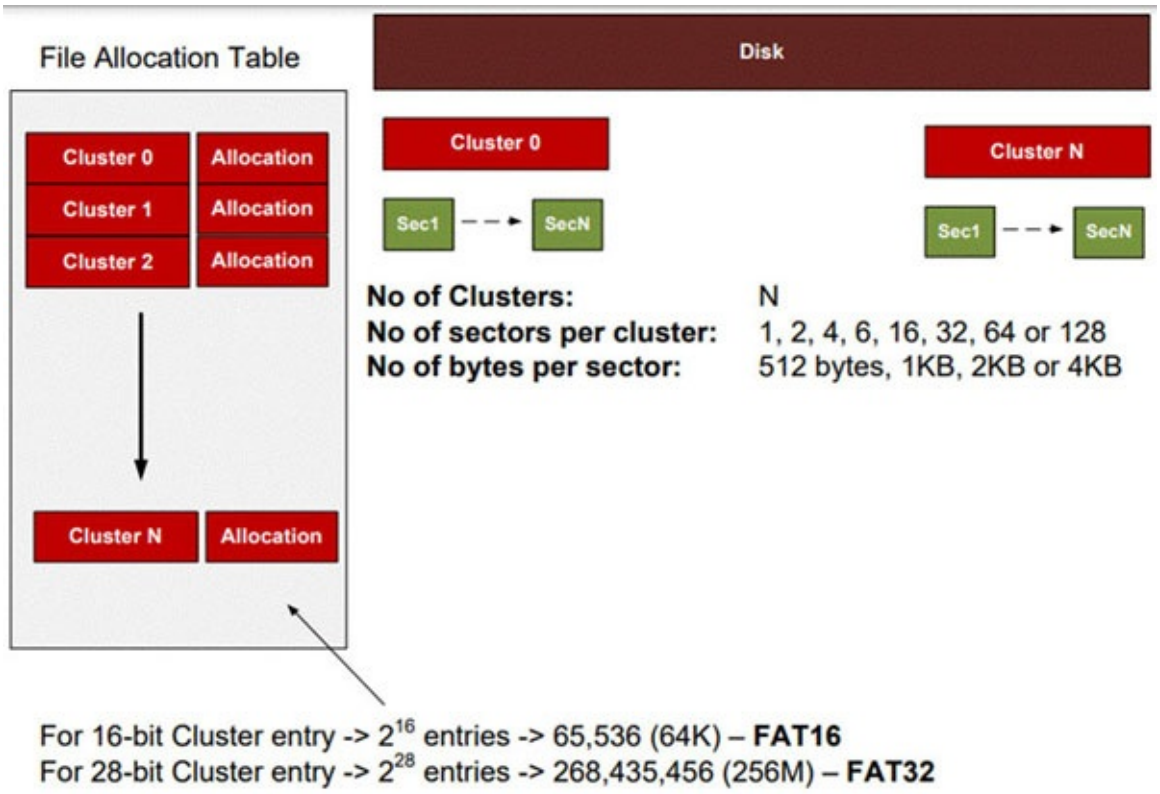


Figure 3.7: *File allocation table structure*

In today's environment, computer forensics is required due to the prevalence of cybercrime. Because Windows 7 is such a widely used operating system, forensics investigations should focus on its artifacts, such as the registry, log files, and NTFS file system. Tools aid in the analysis process; hence, there is a need to develop tools to aid in forensics investigations. As a result, we suggested and created a program for Windows 7 forensics investigation. Our project will aid forensic investigators in undertaking computer forensic investigations of the Windows 7 registry and NTFS system files on Windows 7 machines by saving time and money. Several copies of material that will be used subsequently in deceased forensics analysis are critical. This component aids in the acquisition process by permitting the

creation of raw copies/images of hard drives and USB pen drives that may be used for forensics investigation afterward. Because the Alternate Data Streams feature of the NTFS file system allows users to hide data in the system files, forensic investigators must keep this in mind while investigating windows PCs using the NTFS file system. Alternate Data Streams in deleted files are equally important, although they may be ignored because forensic professionals are unfamiliar with them.

In digital forensics, the content saved in files is the primary piece of evidence. These files on disk are managed via the file system. A perpetrator can erase evidence from a hard drive by removing evidence-related files. It is critical for forensic investigators to recover evidence that has been erased by the perpetrator. This component aids forensic investigators in recovering deleted files from NTFS-formatted hard drives and USB pen drives and analyzing the time whenever the computer is turned on, and the person who is signed in at that time might reveal crucial data that can be linked to other evidence. Instance: On an NTFS file system, the removed file time may be connected with the person who was signed in at the time to determine who deleted the files. By providing a timeline of customer log-on and log-off events, this component assists forensics investigators in analyzing user behavior on the computer. In today's society, when data is the most crucial aspect of human life, it is critical to understand how data may be lost and if it can be recovered. The concept of data recovery is given in the first section, followed by a discussion of why it is necessary. Following that, we will discuss data recovery procedures and obstacles. Companies are increasingly relying on computers to interact with external and internal papers, and digital storage is becoming increasingly important. The majority of the emphasis have been on well-known issues like infections and vulnerabilities.

Whenever data cannot be accessed ordinarily, data recovery is performed. This might be caused by physical or logical damage to the system files, preventing them from being installed by the host system. Logical or physical damage to the file mechanism to prevent it from being installed by the host system may need recovery. Data loss can occur owing to physical and logical errors, as well as data overwriting. And there are several approaches to each of these three requirements. Internal or external problems caused data loss or damage. The suffering was made worse by the increasing haste and pace of life, which resulted in the unintended erasure of vital helpful data. This just reveals one side of the significance of data recovery; the other aspect is the forensic significance of data recovery. The distinction that the forensic needs have is that data may not be mistakenly deleted here, but it creates a distinction in the recovery method as well because recovery will be challenging in this case because the deletion was done with the aim that the data would never be retrieved. Numerous failures may result in physical harm. Hard disk drives might fail for a variety of reasons, such as head stack crashes or tapes breaking. Physical damage always results in some data loss, and in some situations, the file system's logical structures are also shattered. Data recovery from physically damaged hard drives: The majority of physical damage could not be repaired by end users. Viruses, formatting, mispartitioning, miscloning, incorrect operation, network deletion, and power outages during the procedure are all possible software causes. Mis-operation, read errors, cannot find or open the file, reports of no partition, not formatted, password loss, and troublesome characters are the most common symptoms.

Current computer hard drives store a variety of data, comprising operating system applications and user data saved in files. Drives also house the operating system's metadata information, including directories, file characteristics, and allocation tables, as well as a backup store for virtual memory. The following are the most typical ways to harm hard drives:

- Damaging the drive physically, leaving it useless.
- Degaussing the disk to randomize the magnetic domains, which will very certainly leave the drive inoperable.

- Erasing the data on the device so that it cannot be retrieved.
- According to *Anthony Verducci*, there are three ways to delete files:
- Method of the file by file (individual files eliminated, software remains intact).
- The entire-drive approach (the entire drive is permanently erased but still usable).
- The power tool technique (Data is gone, the hard drive is toast).

There are files (used space) and empty space on every electronic storage device (unused space). Each time the computer is used, the information of the files in the used space may be modified, and previously deleted data in the unused space may be overwritten. Individuals with low-level skills remove documents using delete or erase instructions. Expert criminals, on the other hand, follow the destruction by overwriting the data so that it cannot be retrieved. Filling every accessible block with ASCII NUL bytes is one method they use to overwrite a hard drive.

Conclusion

In this chapter, we have covered important technical concepts about computers that must be well understood by any digital forensic examiner. We describe how computers store and represent data digitally, the concept of operating system file structure and its types, and hash algorithms and how we can use them to verify the authenticity of any piece of digital data. In the upcoming chapter, we will discuss about setting up a professional Digital Forensics investigation lab facility, the tools, software and hardware that are required at a bare minimum.

CHAPTER 4

Requirements for a Computer Forensics Lab

Introduction

It is important to have the software and hardware tools you need to start your investigation. With more cybercrime attacks that affect both the public and private sectors, the need for a computer forensics lab that can capture and analyze digital evidence with high accuracy is going to get more and more important. You might think that computer forensics labs are only used by law enforcement. However, this is not true. Many corporations in the United States have digital forensics labs with more advanced investigation tools than many police labs.

Structure

In this chapter, we will cover the following topics:

- Digital Forensic Lab
- Physical requirements
- Environment controls
- Digital Forensic equipment
- Office electrical equipment
- Networked devices

Objectives

In previous chapters, we talked about the basics of Digital Forensics and how investigations can be broken down into two types: public and private. Law enforcement agencies and security services are the first to set up digital forensics' labs. However, with the rise of computing technology and the widespread use of smartphones and wearable devices, most common crimes now have digital evidence. This means that police labs are full of long lists of digital evidence from different legal cases that need to be looked into. It is a good idea for big businesses and even small ones to set up a laboratory inside their businesses so that they can investigate cybercrime problems that affect their work and property.

Digital Forensic Lab

Today, banks, tech companies, retailers (such as Amazon and Walmart), and utility providers are all using their digital forensics labs to speed up the investigation process and cut down on the costs of digital investigation. Compared with police labs, private corporations have more freedom when it comes to buying the most up-to-date software and hardware for their labs. Some **Law Enforcement Agency** (**LEA**) labs may still be using old software versions because they do not have enough money or people

who know how to use it. In most cases, in-house digital forensics analysts work with law enforcement to solve cases that are related to their businesses. People who find evidence of or witness illegal activity (for example, violating rules or industrial sabotage and leaking secrets) will call law enforcement and work with them to get the evidence and move the case to court. The reporting company's digital forensic investigators or the e-discovery team will work with them to get and analyze the evidence.

It is a good idea for any company that cares about its data to have an in-house digital forensics lab. However, this comes at a price. In this example, even the smallest lab would have to spend at least $150,000 a year just to have one forensic analyst and one forensic workstation with most of the tools they need to do their job. This extra cost may not be worth it if the company only has a couple of mishaps a year. People working for small and medium-sized businesses often hire a third-party digital forensics laboratory to do their work for them so that they can cut down on costs.

Accrediting a digital forensics lab is an important thing to think about, whether you want to set up an in-house lab for your company or you want to outsource your digital forensics work to a third-party provider. Accreditation ensures that your lab, or the one you want to use, meets the standards set by the official body in terms of using reliable methods, the right tools (hardware and software), and the right people to do its job. You need to think about how much money you have when planning a digital forensics lab. You also need to think about what you want to do in the lab and what kind of equipment and software you need to do it. Among other things, big companies are spending money to build advanced labs that can handle all types of computers and cases such as malware, outside breaches and network, GPS, and mobile forensics. These labs have well-trained people who use the most up-to-date forensic software and other special hardware tools.

It does not matter how big or small your forensics lab is; it must have the basic tools to capture, store, analyze, and present digital evidence in a forensically sound way. A small digital forensics lab is the most common because it only costs a small amount of money and can start right away. Usually, one to five people work in these labs. They mostly work with a single type of device (for example, mobile forensic or Windows OS forensic). There is not a lot of fancy technology that big labs have, but it still needs to have the right digital forensic software to look at the evidence, as well as the hardware to do that, such as a hardware write blocker, cables, and storage solutions. It also needs a dedicated forensic computer. Before listing the software and hardware needed for the forensic lab, it is important to talk about the physical space needed for this lab. Even though these labs might be a target for cybercriminals who want to stop or sabotage investigations, it is important to keep digital evidence and the lab's equipment safe and sound.

Physical requirements

The following physical needs are very important to have in any digital forensic lab:

- There must be at least one way.
- It is better if there are no windows in the lab.
- The lab must be soundproof, so no one can hear what people are talking about inside the lab. This can be done by putting soundproofing material on the ceiling and walls and carpet on the floor to make the room quieter.
- There must be an alarm system at the entrance of the lab, as well as a biometric system to let people into the lab. People who use the biometric access system must be able to show that they have been to the lab. This log must be kept for a long time for auditing.

- Surveillance cameras should cover the whole lab, especially the main entrance and the room where digital evidence will be kept. Keep the video recorder of the surveillance system in the lab's safest room, which is called the "*evidence storage room*", so it does not get broken into. This can lead to the tempering of evidence and violation of data integrity.
- Fire suppression systems must be in place for this to work.

Environment controls

To keep forensic equipment and seized digital devices from being damaged, the lab environment must be kept very clean. The following controls must be in place for this to work. First, an air cooling system to remove heat from workstations. Because forensic workstations can stay running for a long-time during evidence analysis (like cracking a password), they can get very hot. This is especially important in small spaces, where heat can build up quickly. Second, the lab needs to be a good place to work and clean. It must have a healthy climate in terms of temperature, low humidity, and clean air so that it is safe for people to work there. Third, there should be good lighting in the whole lab and each forensic workstation room. Electricity providers and UPS units to keep the lab running even if there is a sudden loss of power, as well as for forensic workstations, storage servers, and surveillance cameras.

Digital forensic equipment

The following hardware is needed for the forensic lab, broken down into three groups:

Forensic hardware

- Licensing the server (which is required by some digital forensics suites).
- Storage server that is set up to use standard removable hard drives to store digital evidence images and data that has been processed and extracted from those images. This server must not be connected to the internet.
- Forensic workstation.
- A portable forensic computer number is used outside the lab to capture evidence and for doing some analysis.
- Computers that can connect to the internet and the intranet.
- The administrative computer is used to keep track of logs and other things.
- **Hardware Write Blocker**: This is a piece of hardware that connects the media that holds digital evidence (like a hard drive) to a forensic workstation. The goal of this piece of hardware is to keep the data on the evidence drive from being changed during the acquisition process.
- CD/DVD drive.
- USB reader.
- HDD and SSD cases with a USB 3.0 port.
- SD card reader.
- USB 2.0 and USB 3.0 thumb drives of different sizes.

- Tape drives are used to store long-term data.
- The following are the types of data cables and connectors: Ethernet cables (RJ-45), BNC adapters (such as modular adapters), ribbon cables (such as ribbon cables), DIN split cables, VGA split cables (such as VGA split cables), USB cables, audio cables (such as USB extension cables), cable extenders, HDMI and FireWire (IEEE 1394).
- Other tools, such as screwdrivers, a multi-meter, and a flashlight.

Office electrical equipment

- Each workstation, server, and network device needs to have an **uninterruptible power supply** (**UPS**).
- A device for projecting things (in a conference room).
- Scanner
- Photocopier
- A paper shredder is the sixth thing on the list
- Digital cameras, such as video cameras, and their accessories.
- Wireless
- Wi-Fi access point
- Headset
- Symmetrical power source

Networked devices

It is important to note that there should be a separate network inside of the lab that connects forensic workstations to the server that stores digital evidence image copies. The server must be put in the evidence room to keep people from getting to it. This lab-specific network cannot connect to the internet. In the lab, forensic examiners may need to look up more information about their findings or work with other people, so an internet connection should only be available through a direct line to the computer(s) that need it. Router and switch to connect forensic workstations to the storage server in the lab. The lab's internet network should not be connected to the lab's network. The things you need are a firewall, a switch, a router (the three components can be combined in one device), and network cables and wires.

Forensic workstation

There should be forensic workstations that have the most recent version of Windows OS (64-bit version) on them. Windows 10 Pro and Enterprise editions are recommended because they can run on high-end hardware and do a lot of work quickly. Both editions can have up to 6 TB of RAM and four processors. However, compared to modern Windows Server editions, which can have 24 TB of RAM, those two editions are cheaper because they belong to the Windows desktop product line.

Now, let us talk about what kind of hardware is needed for forensic workstations. When working with digital evidence, you need a powerful computer to process and search through image files. Forensic computers need a lot of processing power and a lot of RAM. They also need a lot of storage space and a

lot of expansion slots to connect different types of devices. Building a forensic workstation costs money, but it is still cheaper than buying a ready-made computer forensic workstation, which costs a lot more. For small businesses, this is still a good option.

This is what you should have when you start from scratch and build a forensic workstation.

- At least 24 GB of RAM (DDR4): **https://www.kingston.com/en/memory/ddr4-overview**
- CPU: Each workstation must have at least two physical CPUs (Intel i9 8th generation processor has 10 cores and 20 threads) to run: **https://ark.intel.com/content/www/us/en/ark/products/series/134928/8th-generation-intel-core-i9-processors.html**
- You also need a motherboard that can fit the processors, RAM, and video card.
- **Hard drives**: At least 512GB of SSD and 4TB of HDD.
- **Video card**: NVidia GeForce with at least 8 GB of GDDR5X memory.
- A triple burner is the best (Blu-ray, DVD, or CD).
- External hard drive enclosure with a USB 3.0 port.
- **Write protection**: You can buy this piece on its own or one that can be used with your workstation. The hardware write blocker should be able to read data from SATA, SAS, IDE, USB, FireWire, and PCIe storage devices. Some companies make UltraBlock (**https://digitalintelligence.com/products/ultrablock**) and Tableau Forensic Universal Bridge (**https://security.opentext.com/tableau/hardware/details/t356789iu**).
- A liquid CPU cooling system with at least two fans is the best way to keep your computer cool.
- At least 22 inches for a better display. A full HD IPS display is the best.
- Ports for USB 3.0, Thunderbolt 3, Microphone, and Headphone jack.
- Integrated LAN controller card.

These are the best pieces of hardware to use to build a forensic workstation. Keep in mind that your lab needs at least one digital forensic laptop workstation to get and analyze data outside of the lab. For example, it is better to buy one forensic laptop from a company that specializes in ready-made solutions.

Commercial digital forensic workstations

Companies that specialize in making ready-made forensic workstations make these workstations with a lot of processing power and storage space. They also come with hardware for digital forensic work, like a hardware write blocker and a hard drive duplicator, which are used to make copies of hard drives. Two companies that make ready-made workstations:

- Tri-Tech Forensics: **https://tritechforensics.com/DF-workstations**

  These forensic workstations are high-end computers with fast processing speed, high memory, and disk storage. Such workstations can serve critical processes such as duplication of data, recovering data from deleted files, analyzing data over the network, and retrieving data from the slack, as illustrated in *figure 4.1*.

Figure 4.1: *Momentum T550 digital forensics workstation*

- Digital intelligence: **https://digitalintelligence.com/store**

FRED systems set the standard for forensic acquisition and analysis workstations. The quality, features, performance, and overall capability are second to none. Buying a FRED system means investing in your ability to solve every investigation. FRED systems are designed and built from the ground up as high-performance, forensic acquisition, analysis, and processing platforms. The advanced technology and features available exclusively from digital intelligence set FRED systems apart, and digital intelligence's build quality and service ensure your FRED investment will yield lasting value.

Figure 4.2: *FRED forensic workstation*

Forensic software applications

The type of forensic software you need for your lab will depend on what you are going to be doing. For example, the type of operating system (Windows, Linux, or Mac) and the type of file system you are going to be looking at will help you figure out what tools you need. Most of the popular computer forensics suites are made for Windows. The open-source version is mostly for Linux, but some have been made for Windows. Let us start with the business tools.

Commercial forensics tools

Research well before you buy digital forensic software because it can be pricey to do so. Ask other forensic investigators and try to install a free copy of the software you want to buy. Commercial computer forensics suites like these are the most popular ones. Check each website to see how much each tool costs or how many licenses it comes with.

- **EnCase**: **https://security.opentext.com/encase-forensic**

  OpenText™ EnCase™ Forensic is a court-proven solution for finding, decrypting, collecting, and preserving forensic data from a wide variety of devices while ensuring evidence integrity and seamlessly integrating investigation workflows. With EnCase Forensic, examiners can be confident the integrity of the evidence will not be compromised. All evidence captured with EnCase Forensic is stored in the court-accepted EnCase evidence file formats. EnCase Forensic has been used in thousands of court cases and is known for its ability to uncover evidence that may go unnoticed if analyzed with other solutions. EnCase Forensic supports the latest smartphones and tablets, including more than 35,000+ mobile device profiles, all while empowering the examiner to conduct logical and physical acquisitions. From the new investigator to the seasoned examiner, each level of user can find the evidence they need with mobile acquisitions in EnCase Forensic.

- Balkasoft Evidence Center X: **https://belkasoft.com/x**

  Belkasoft Evidence Center X) is a flagship tool by Belkasoft for computer, mobile, and cloud forensics. It can help you to acquire and analyze a wide range of mobile devices, run various analytical tasks, perform case-wide searches, bookmark artifacts, and create reports. This works out of the box and can be easily integrated into customer workflows. The software interface is so user-friendly that you can start working with your cases right after the Belkasoft X deployment. The tool acquires, examines, analyzes, and presents digital evidence from major sources—computers, mobile devices, RAM, and cloud services—in a forensically sound manner. If you need to share the case details with your colleagues, use a free-of-charge portable Evidence Reader. While performing search tasks for evidence, Belkasoft Evidence Center X uses approaches that enable it to find the most forensically significant artifacts quickly instead of wasting time on redundant operations. Powerful analytical features such as a connection graph, a timeline, and advanced picture and video analysis help you to uncover facts rapidly.

- FTK: **https://www.exterro.com/ftk-imager**

  FTK® Imager is a data preview and imaging tool that lets you quickly assess electronic evidence to determine if further analysis with a forensic tool such as **Forensic Toolkit** (**FTK®**) is warranted. Create forensic images of local hard drives, CDs and DVDs, thumb drives or other USB devices, entire folders, or individual files from various places within the media. FTK®

Imager can create perfect copies or forensic images of computer data without making changes to the original evidence. The forensic image is identical in every way to the original, including file slack and unallocated space or drive-free space. This allows you to store the original media away, safe from harm, whereas the investigation proceeds using the image. Generates hash reports for regular files and disk images to use as a benchmark to prove the integrity of your case evidence. When a full drive is imaged, a hash generated by FTK® Imager can be used to verify that the image hash and the drive hash match after the image is created and that the image has remained unchanged since acquisition. Mounts an image for a read-only view that leverages Windows® Internet Explorer® to see the content of the image exactly as the user saw it on the original drive. See and recover files that have been deleted from the Recycle Bin but have not yet been overwritten on the drive.

- X-ways: **http://www.x-ways.net/forensics/**

  X-ways forensics is an advanced work environment for computer forensic examiners and our flagship product. Runs under Windows XP/2003/Vista/2008/7/8/8.1/2012/10/2016/2019/11*, 32 Bit/64 Bit, standard/PE/FE. (Windows FE is described here, here, and here.) Compared to its competitors, X-Ways Forensics is more efficient to use after a while, by far not as resource-hungry, often runs much faster, finds deleted files and search hits that the competitors will miss, offers many features that the others lack, as a German product is potentially more trustworthy, comes at a fraction of the cost, does not have any ridiculous hardware requirements, does not depend on setting up a complex database, and so on! X-Ways Forensics is fully portable and runs off a USB stick on any given Windows system without installation if you want. Downloads and installs within seconds (just a few MB in size, not GB). X-ways forensics is based on the WinHex hex and disk editor. It is part of an efficient workflow model where computer forensic examiners share data and collaborate with investigators that use X-ways investigator.

Open-source forensic tools

There are a lot of free and open-source code digital forensics tools. Some of them have a lot of features, like commercial suites, but most of them are small tools that do one thing well (for example, retrieve browser history or extract e-mail header information). List of the best free and open-source digital forensic tools:

- Sleuth Kit: **http://www.sleuthkit.org/sleuthkit/**

  Sleuth Kit® (TSK) is a library and collection of command-line tools that allow you to investigate disk images. The core functionality of TSK allows you to analyze volume and file system data. The library can be incorporated into larger digital forensics tools, and the command line tools can be directly used to find evidence, as illustrated in *figure 4.3*.

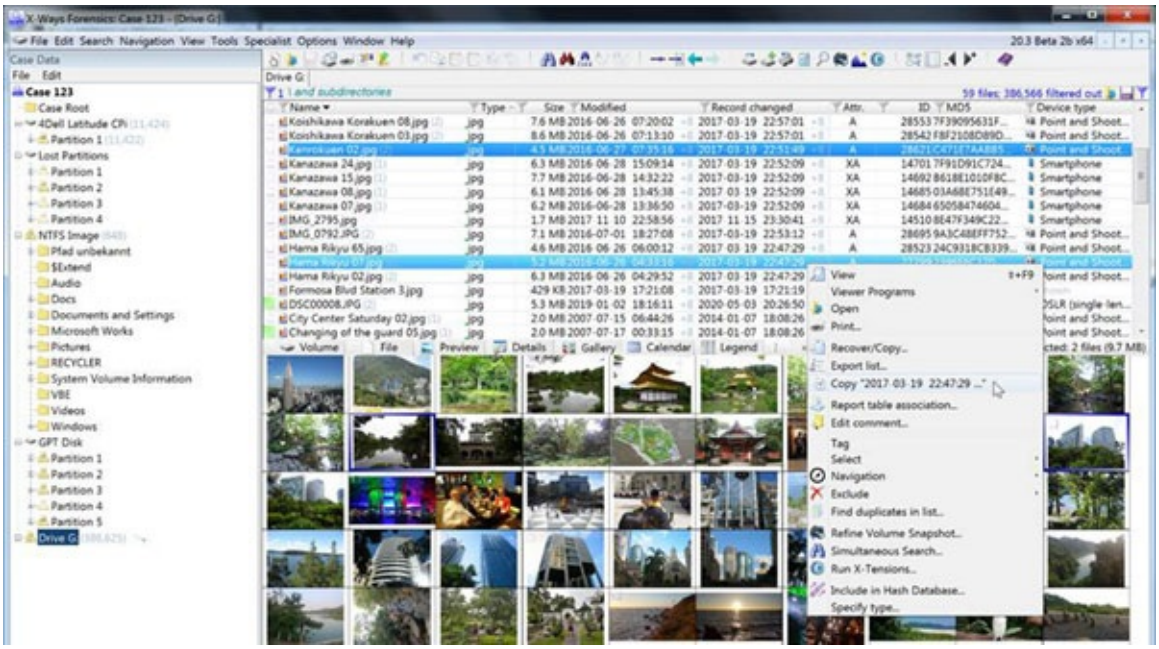


Figure 4.3: *X-Ways Sleuthkit*

- Autopsy: **www.sleuthkit.org/autopsy**

  Autopsy® is a digital forensics platform and graphical interface to The Sleuth Kit® and other digital forensics tools. It is used by law enforcement, military, and corporate examiners to investigate what happened on a computer. You can even use it to recover photos from your camera's memory card. An autopsy was designed to be intuitive out of the box. Installation is easy, and wizards guide you through every step. All results are found in a single tree. An autopsy was designed to be an end-to-end platform with modules that come with it out of the box and others that are available from third parties (refer to *figure 4.4*). Some of the modules provide the following:

  - **Timeline analysis**: Advanced graphical event viewing interface (video tutorial included).
  - **Hash filtering**: Flag knows bad files and ignores known good.
  - **Keyword search**: Indexed keyword search to find files that mention relevant terms.
  - **Web artifacts**: Extract history, bookmarks, and cookies from Firefox, Chrome, and IE.
  - **Data carving**: Recover deleted files from unallocated space using PhotoRec.
  - **Multimedia**: Extract EXIF from pictures and watch videos.
  - **Indicators of compromise**: Scan a computer using STIX.

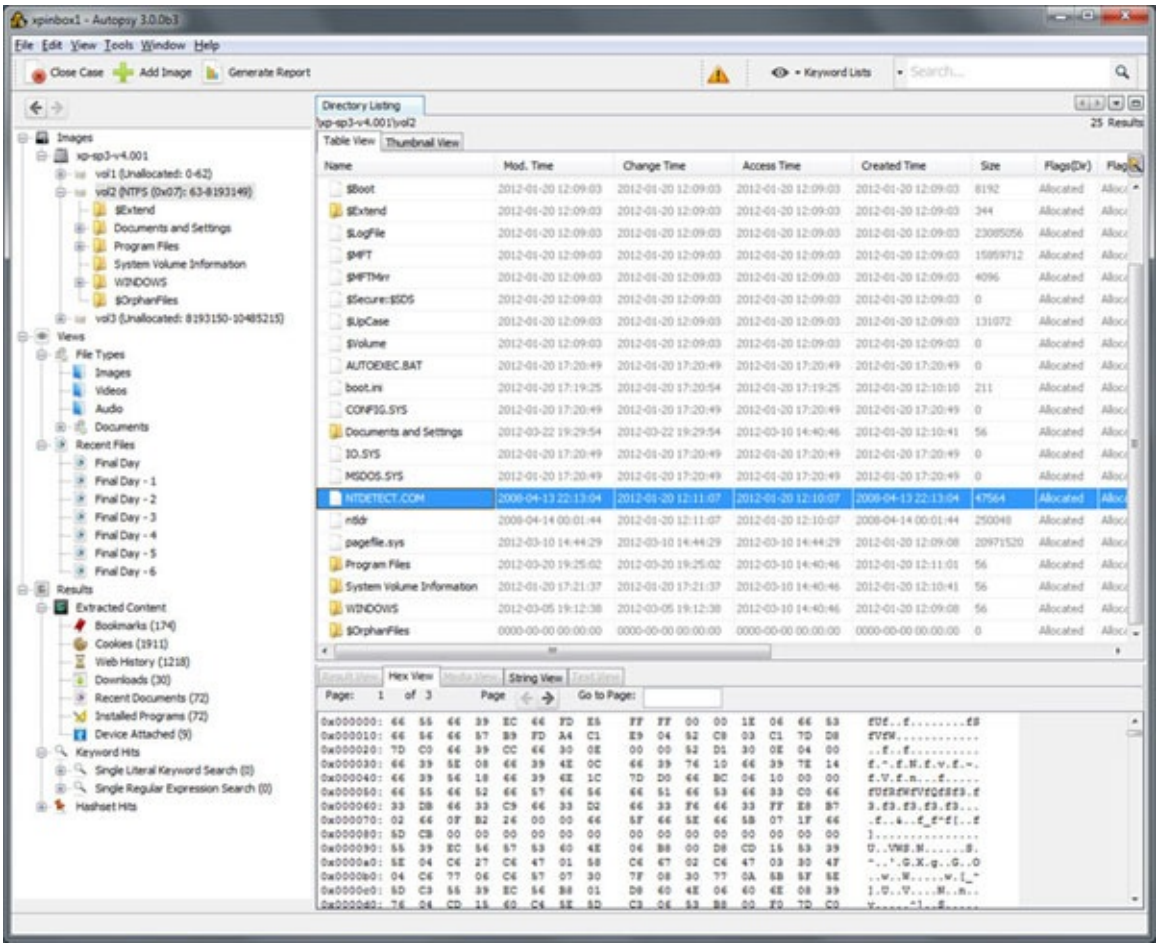


Figure 4.4: *Autopsy*

- DD for Windows: **http://www.chrysocome.net/dd**

  The command-line software application "*dd*" is a "*raw write*" data manipulation program. It reads and writes data more directly than most other methods. This not only makes dd potentially more powerful but also riskier to use because catastrophic, data-destroying mistakes are just a typo away. DD is a command-line utility in the Linux version to convert and copy files. The Windows version does not do any conversion, but it allows the flexible copying of data in a win32 environment.

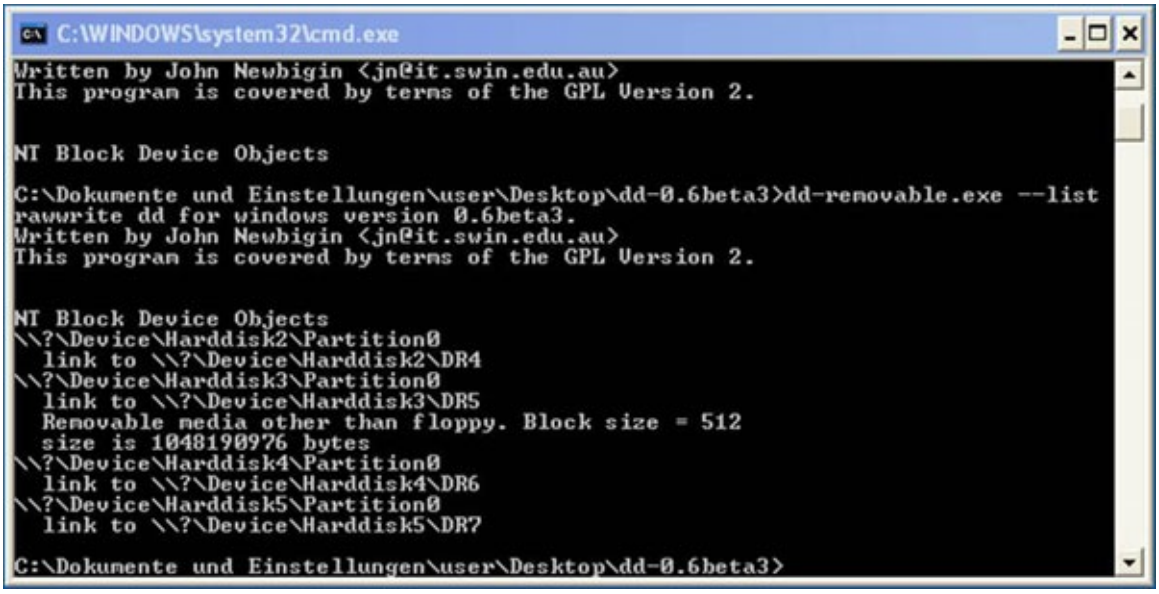


Figure 4.5: *DD for Windows*

- Magnet RAM capture: **https://www.magnetforensics.com/resources/magnet-ram-capture/**

  This tool allows investigators to acquire the memory of a live PC. Memory analysis can reveal a lot of important information about a system and its users. There are often instances where evidence stored in memory is never written to the hard drive and may only be found on the **`pagefile.sys`** or **`hiberfil.sys`**. Memory analysis is essential to many malware and intrusion incidents and can be imperative in recovering valuable evidence for almost any PC investigation. Running processes and programs, active network connections, registry hives, passwords, keys, and decrypted files are just a few examples of evidence that can be found in memory. Many Web apps, like Gmail, or private/incognito browsing modes will only store data in memory, meaning the evidence cannot be recovered from the hard disk. Magnet RAM Capture supports both 32 and 64-bit Windows systems, including XP, Vista, 7, 8, 10, 2003, 2008, and 2012. It will acquire the full physical memory quickly and leave a small footprint on the live system being analyzed. For my system, it took about 3 minutes to image an 8 GB RAM dump.

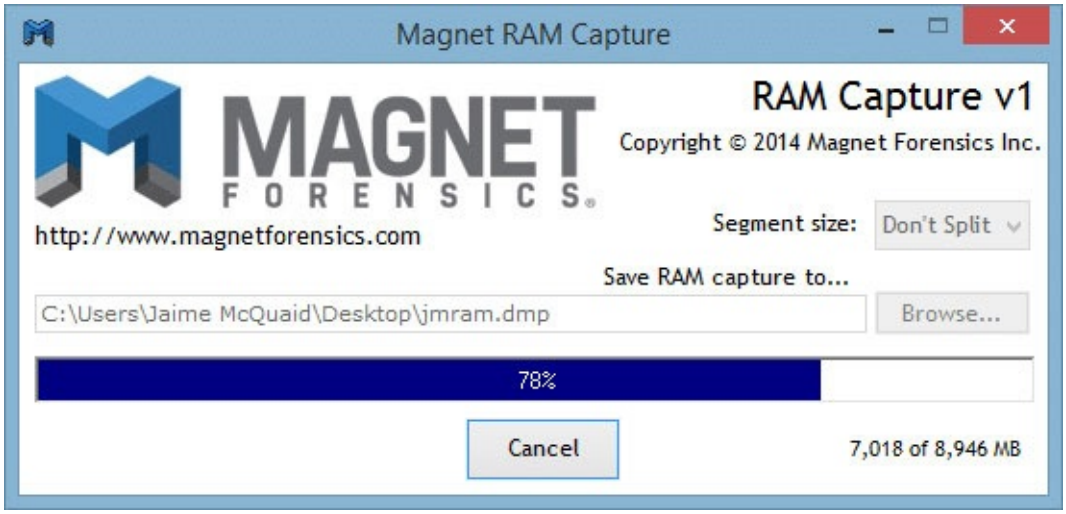


Figure 4.6: *Magnet RAM capture*

- Belkasoft RAM capturer: **https://belkasoft.com/ram-capturer**

  This is a simple-looking application that will extract and create a copy of your volatile memory's contents. In other words, this application is capable of dumping your RAM to a specified output folder. Although the whole mechanism and process sound complicated, the truth is that any user who commands this program will not have to execute any difficult tasks. The application comprises three buttons, a folder output cell, and a progress bar for when the dumping action is executed. When loading the application, the current memory size will be displayed. Depending on your RAM capacity, you might have to deal with longer or shorter wait times. There are not many situations in which you have to want to dump a PC's volatile memory. Still, this could prove crucial in any forensic research, for situations where a file could prove someone's guilt or innocence. It could also work for recovering lost data that would otherwise prove difficult to find, mainly usernames, passwords, or any other type of sensitive data.

  As far as using this particular application goes, one strong reason is the fact that it incorporates anti-dumping and anti-debugging bypass algorithms. Although anti-dumping features could lead to a corrupt dump archive, they could also cause serious damage to your system, which is why it is better to be safe than sorry. Live RAM Capturer is a basic-looking application with complex capabilities that could ultimately prove useful in recovering essential data before it is lost forever. Reading the extracted data will require additional software solutions, so you will only get half of the job done with this particular program.

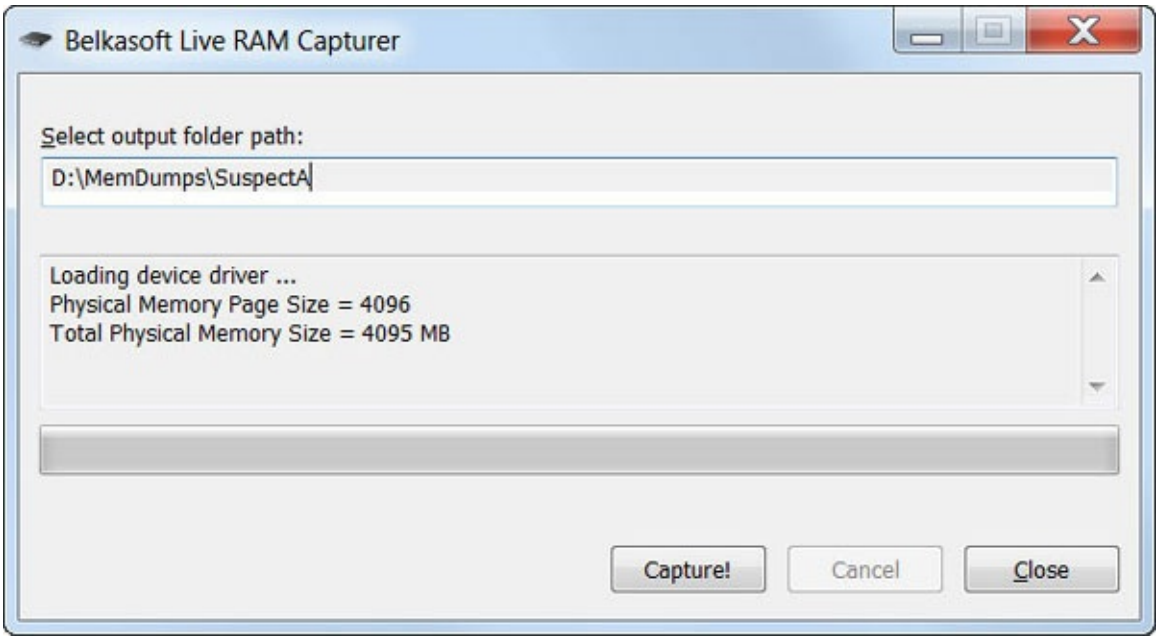


Figure 4.7: *Belkasoft RAM capture*

- Volatility: **https://www.volatilityfoundation.org/releases-vol3**

  In 2020, the volatility foundation publicly released a complete rewrite of the framework, Volatility 3. The project was intended to address many of the technical and performance challenges associated with the original code base that became apparent since its original release in 2007. Another benefit of the rewrite is that Volatility 3 could be released under a custom license that was more aligned with the goals of the Volatility community, the **Volatility Software License** (**VSL**). Details about the rewrite of Volatility 3 can be found in this presentation: Volatility 3 Public Beta: Insider's Preview. Further information about all Volatility 3 releases, including minor releases, can be found in the Volatility 3 project on GitHub, as presented in *figure 4.8*.

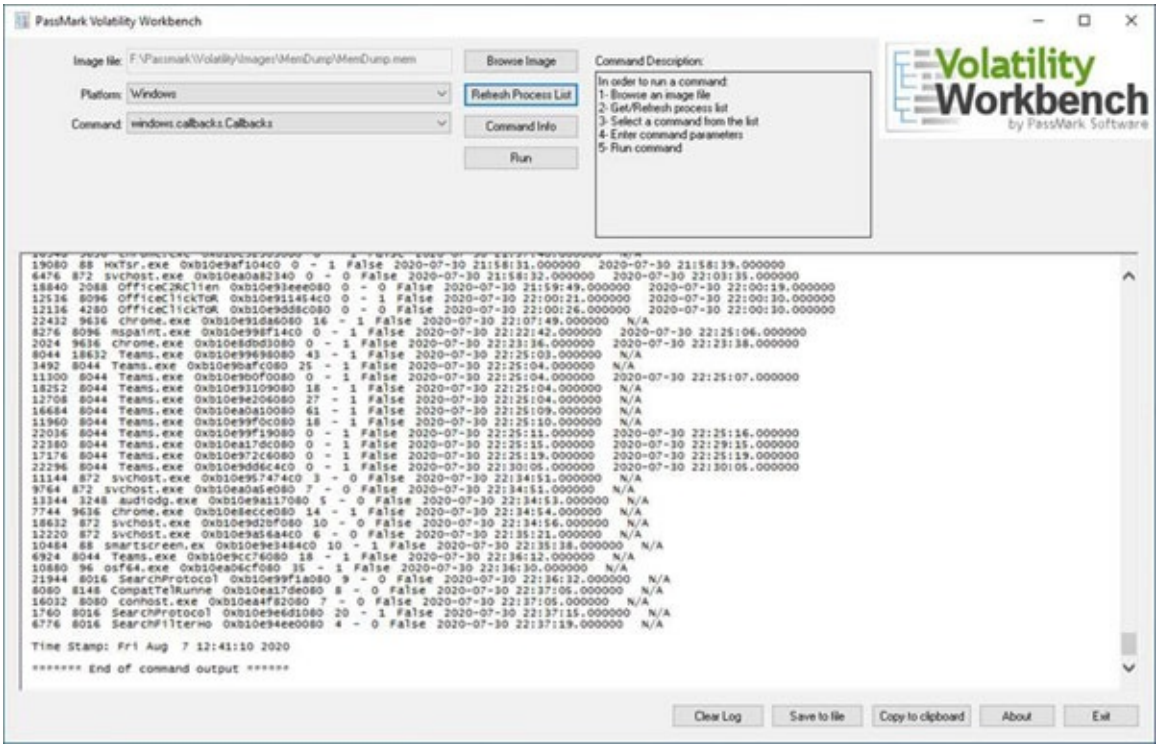


Figure 4.8: *Volatility Ver 3*

Some of the latest release highlights are as follows:

    - Rewrote and redesigned the entire framework (Python 3)
    - Added integrated symbol support (Windows symbols can be automatically downloaded or cached locally for analysis in air-gapped environments)
    - Enhanced API documentation generated from the code
    - Expanded APIs for developers and integrations
    - Significant performance optimizations (object caching and multi-processing)
    - Added native support for 32-bit apps on 64-bit kernels
    - Enhanced support for plugin dependencies and versioning
    - Expanded support for address space layering

- Memoryze: **https://www.fireeye.com/services/freeware/memoryze.html**

  Mandiant's Memoryze™ is free memory forensic software that helps incident responders find evil in live memory. Memoryze can acquire and/or analyze memory images and, on live systems, can include the paging file in its analysis, as presented in *figure 4.9*. Memoryze can:
- Image the full range of system memory (no reliance on API calls).
- Image a process' entire address space to disk, including a process' loaded DLLs, EXEs, heaps, and stacks.
- Image a specified driver or all drivers loaded in memory to disk.
- Enumerate all running processes (including those hidden by rootkits), including:
- Report all open handles in a process (including all files, registry keys, and so on)
- List the virtual address space of a given process, including all loaded DLLs and all allocated portions of the heap and stack.
- List all network sockets that the process has open, including any hidden by rootkits.
- Specify the functions imported and exported by the EXE and DLLs.
- Hash the EXE and DLLs in the process address space (MD5, SHA1, SHA256. This is disk-based).
- Verify the digital signatures of the EXEs and DLLs (disk-based).
- Output all strings in memory on a per-process basis.
- Identify all drivers loaded in memory, including those hidden by rootkits.

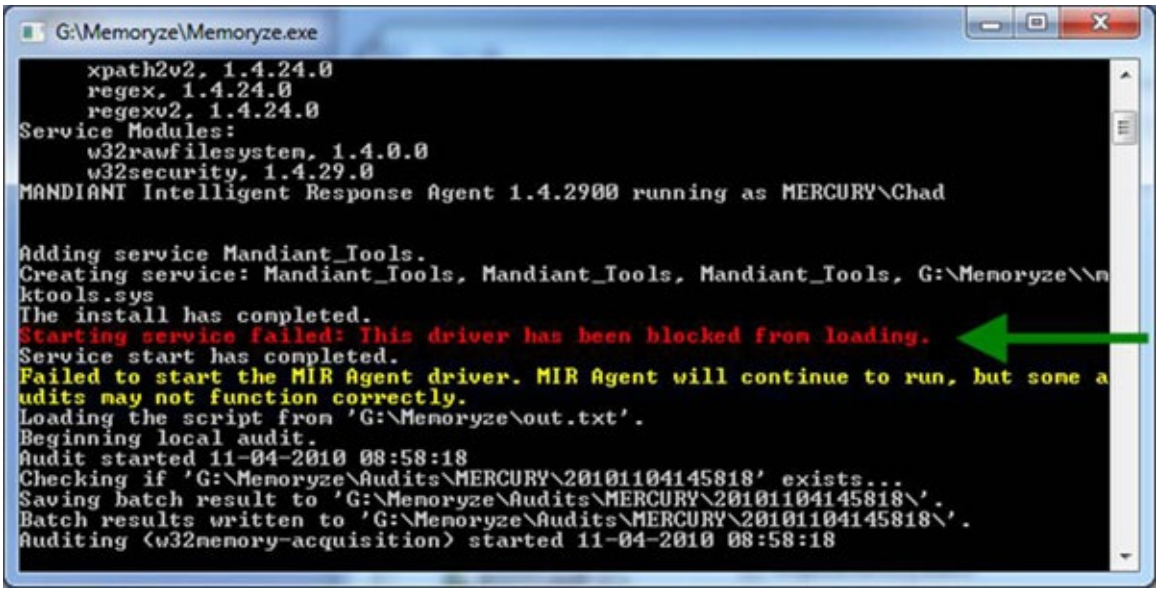


Figure 4.9: *Mandiant Memoryze*

- Mandiant Redline: **https://www.fireeye.com/services/freeware/redline.html**

  Redline®, FireEye's premier free endpoint security tool, provides host investigative capabilities to users to find signs of malicious activity through memory and file analysis and the development of a threat assessment profile, as illustrated in *figure 4.10*. With Redline, you can thoroughly audit and collect all running processes and drivers from memory, file-system metadata, registry data, event logs, network information, services, tasks, and Web history. Analyze and view imported audit data, including the ability to filter results around a given timeframe using Redline's Timeline functionality with the TimeWrinkle™ and TimeCrunch™ features. Streamline memory analysis with a proven workflow for analyzing malware based on relative priority. Perform **Indicators of Compromise** (**IOC**) analysis. Supplied with a set of IOCs, the Redline Portable Agent is automatically configured to gather the data required to perform the IOC analysis, and an IOC hit result review. In addition, users of FireEye's Endpoint Threat Prevention Platform can open triage collections directly in Redline for in-depth analysis allowing the user to establish the timeline and scope of an incident.

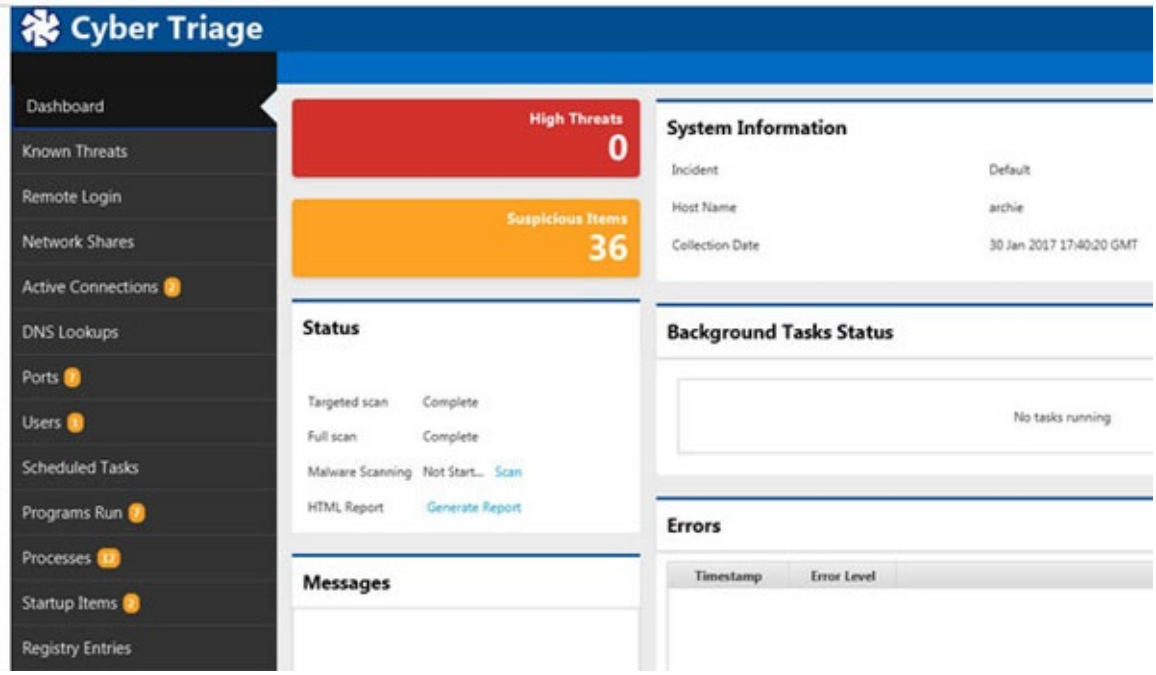


Figure 4.10: *FireEye redline security*

- Encrypted disk detector: **http://www.labsystems.co.in/encryped-disk-detector.html**

  Encrypted Disk Detector (**EDD**) is a command-line tool that checks the local physical drives on a system for TrueCrypt, PGP®, or Bitlocker® encrypted volumes. If no disk encryption signatures are found in the MBR, EDD also displays the OEM ID and, where applicable, the Volume Label for partitions on that drive, checking for Bitlocker® volumes. EDD is useful during incident response to quickly and non-intrusively check for encrypted volumes on a computer system. The decision can then be made to investigate further and determine whether a live acquisition needs to be made to secure and preserve the evidence that would otherwise be lost if the plug was pulled. Currently, EDD detects TrueCrypt, PGP®, Safeboot, and Bitlocker® encrypted volumes, and we are adding to this list with each new release. EDD is available for download now, completely free of charge, as presented in *figure 4.11*.

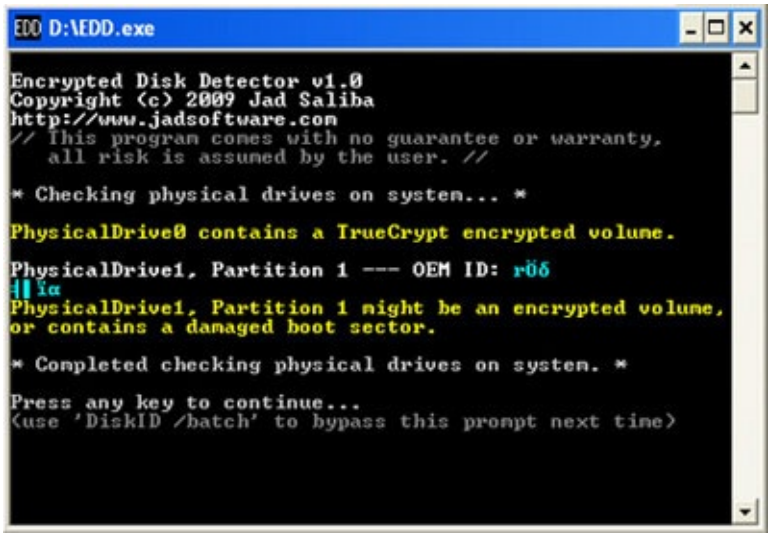


Figure 4.11: *Encrypted disk detector*

Linux distributions

These are special Linux distributions that have been set up to help with digital forensics. They have a live operating system, usually Linux-based, that can be started from a CD, DVD, or USB thumb drive. It has a lot of tools for digital forensics.

- CSI Linux: **https://csilinux.com/download/**

  CSI Linux 2021.2 has many updated tools, features, and additions. To install CSI Linux Tools updates, type "*powerup*" in the terminal window and press *Enter*. When turned on, the CSI_TorVPN encapsulates all traffic through Tor, similar to how Tails works. The CSI_Gateway app is now pointing to a Whonix gateway VM. This gives you two different options when using the Virtual Appliance. If you are using the bootable version, you can only use the CSI_TorVPN. You can also add a VPN or Tor gateway to your network router for an external network layer of security, as presented in *figure 4.12*. CSI Linux SIEM has been separated and is now separate from CSI Linux. MISP, OTX, Malcolm, Moloch, Elasticsearch, Kibana, Logstash, Zeek, and others have been combined into this growing network monitoring and forensic server environment. This will be able to be downloaded onto CSI Linux or used on another system on the network.

Figure 4.12: *CSI Linux*

- CAINE: **https://www.caine-live.net/**

  CAINE Linux stands for Computer-Aided Investigative Environment. It is an Italian Linux live distribution, a digital forensics project that was started in 2008. It uses an old-school desktop environment complemented with top-notch specialty tools, as illustrated in *figure 4.13*. The main objectives that CAINE aims to guarantee are as follows:

  - Its operation environment is designed to provide all the forensic tools that are required to perform digital forensic investigative processes such as preservation, collection, examination, and analysis.
  - It provides a user-friendly graphical user interface with user-friendly forensic tools.
  - It can be booted from removable media like flash drives or from an optical disk and run in memory.
  - It can be easily installed onto a physical or a virtual system.
  - In LIVE mode, CAINE can operate on data storage objects without having to boot up the operating system.

Figure 4.13: *CAINE Linux distro*

CAINE Linux has several software applications, libraries, and scripts that can be used in a command line or graphical environment to perform forensic activities. It can perform data analysis on the data objects created on Microsoft Windows, Linux, and some Unix Systems. One of the interesting features of CAINE Linux version 9.0 is that it sets all the block devices to read-only mode by default. CAINE Linux provides a variety of software tools that can be used for memory, database, network, and forensic analysis. The File Image System analysis of File Systems such as FAT/ExFAT, NTFS, Ext2, Ext3, HFS, and ISO 9660 is possible using command-line mode as well as Graphical user interface mode. CAINE Linux supports disk imaging in raw (dd) and expert witness/advanced file format also. Disk images may be obtained using the tools that are built in the CAINE or using third-party tools such as EnCase or Forensic Tool Kit.

- DEFT: **https://distrowatch.com/table.php?distribution=deft**

  Digital Evidence and Forensic Toolkit (**DEFT**) is a customized distribution of the Ubuntu live Linux CD, as illustrated in *figure 4.14*. It is an easy-to-use system that includes excellent hardware detection and some of the best open-source applications dedicated to incident response and computer forensics.

Figure 4.14: *DEFT Linux*

- Helix: **http://www.e-fense.com/helix3pro.php**

  Helix3 Pro is a unique tool necessary for every computer forensic tool kit with a Live and Bootable side for all investigation needs with one simple-to-use interface. This also provides a multi-platform LIVE side for three environments, Mac OS X, Windows, and Linux, to create forensic images of all internal devices. This also makes the forensic image of physical memory (32- and 64-bit) and determines if disk-level encryption is turned on. The tool also provides a bootable forensically sound environment to boot any ×86 system and search file systems for specific file types (i.e., Graphic files, Document files, and so on) analysis. Helix3 Pro is only available through the e-fense forum. Become a member of the e-fense forum to get support and learn from e-fense experts and other users of the number one computer forensic tool used by law enforcement, government agencies, and computer forensic experts around the world.

- SIFT: **https://linuxhint.com/sans_investigative_forensics_toolkit**

  SIFT is a computer forensics distribution created by the SANS Forensics team for performing digital forensics. This distro includes most tools required for digital forensics analysis and incident response examinations. SIFT is open-source and publicly available for free on the internet. In today's digital world, where crimes are committed every day using digital technology, attackers are becoming more and more stealthy and sophisticated. This can cause companies to lose important data, with millions of users exposed. Protecting your organization from these attacks requires strong forensic techniques and knowledge of your defense strategy. SIFT provides forensic tools for file systems, memory, and network investigations to perform in-depth forensic investigations, as illustrated in *figure 4.15*.

Figure 4.15: *SANS SIFT*

- Santoku Linux: **https://santoku-linux.com/**

  Santoku is a general-purpose kitchen knife that originated from Japan, meaning "*three virtues*" or "*three uses*". This distribution is not from Japan, but the name was suggested by *Thomas Cannon* of viaForensics because the distribution was crafted specifically for Mobile Forensics, Mobile Malware Analysis, and Mobile Security Testing, as presented in *figure 4.17*. The current alpha release is based on a fork of the **Open Web Application Security Project** (**OWASP**) MobiSec Ubuntu distro; thus, making this alpha release OWASP MobiSec Remix (released under GPL) with added tools from viaForensics and some of its contributors or supporters.

Figure 4.16: *Santoku Linux*

  - Mobile forensics

        - Firmware flashing tools for multiple manufacturers
        - Imaging tools for NAND, media cards, and RAM
        - Free versions of some commercial forensics tools
        - Useful scripts and utilities specifically designed for mobile forensics
    - Mobile Malware
        - Tools for examining mobile malware
        - Mobile device emulators
        - Utilities to simulate network services for dynamic analysis
        - Decompilation and disassembly tools
        - Access to malware databases
    - Mobile Security
        - Assessment of mobile apps
        - Decompilation and disassembly tools
        - Scripts to detect common issues in mobile applications
        - Scripts to automate decrypting binaries, deploying apps, and enumerating apps

- Kali Linux: **https://www.kali.org/get-kali/**

  Kali Linux (formerly known as BackTrack Linux) is an open-source, Debian-based Linux distribution aimed at advanced Penetration Testing and Security Auditing. Kali Linux contains several hundred tools targeted toward various information security tasks, such as Penetration Testing, Security Research, Computer Forensics, and Reverse Engineering. Kali Linux is a multi-platform solution, accessible and freely available to information security professionals and hobbyists. Kali Linux was released on March 13, 2013, as a complete, top-to-bottom rebuild of BackTrack Linux, adhering completely to Debian development standards, as presented in *figure 4.17*.

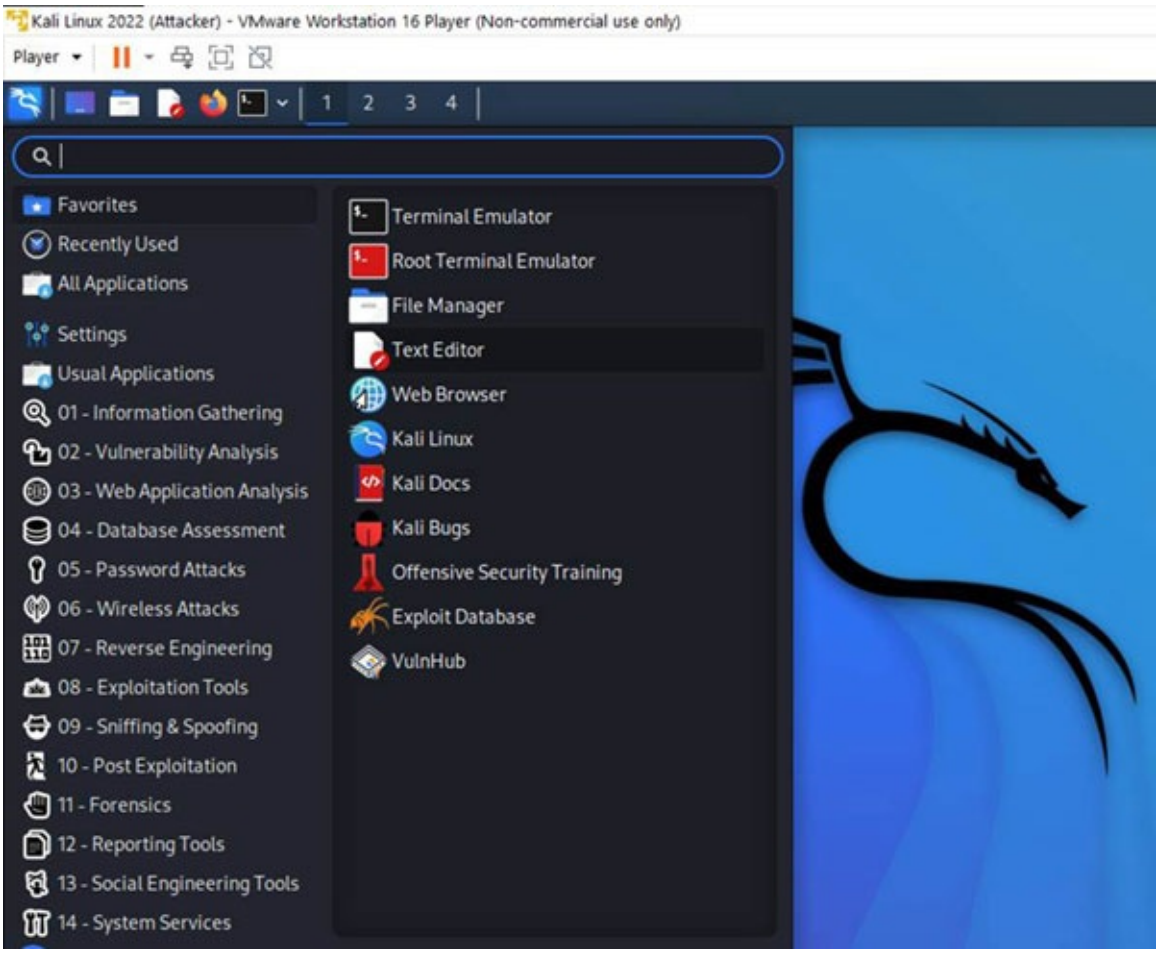


Figure 4.17: *Kali Linux*

Kali Linux features:

- **More than 600 penetration testing tools included**: After reviewing every tool that was included in BackTrack, we eliminated a great number of tools that either simply did not work or which duplicated other tools that provided the same or similar functionality. Details on what is included are on the Kali Tools site.
- **Free (as in beer) and always will be**: Kali Linux, like BackTrack, is completely free of charge and always will be. You will never, ever have to pay for Kali Linux.
- **Open-source Git tree**: We are committed to the open-source development model, and our development tree is available for all to see. All of the source code which goes into Kali Linux is available for anyone who wants to tweak or rebuild packages to suit their specific needs.
- **FHS compliant**: Kali adheres to the Filesystem Hierarchy Standard, allowing Linux users to easily locate binaries, support files, libraries, and so on.
- **Wide-ranging wireless device support**: A regular sticking point with Linux distributions has been supporting wireless interfaces. We have built Kali Linux to support as many wireless devices as we possibly can, allowing it to run properly on a wide variety of hardware and making it compatible with numerous USB and other wireless devices.
- **Custom kernel, patched for injection**: As penetration testers, the development team often needs to do wireless assessments, so our kernel has the latest injection patches included.
- **Developed in a secure environment**: The Kali Linux team is made up of a small group of individuals who are the only ones trusted to commit packages and interact with the repositories, all of which are done using multiple secure protocols.

- **GPG signed packages and repositories**: Every package in Kali Linux is signed by each developer who built and committed it, and the repositories subsequently sign the packages.
- **Multi-language support**: Although penetration tools tend to be written in English, we have ensured that Kali includes true multilingual support, allowing more users to operate in their native language and locate the tools they need for the job.
- **Completely customizable**: We thoroughly understand that not everyone will agree with our design decisions, so we have made it as easy as possible for our more adventurous users to customize Kali Linux to their liking, all the way down to the kernel.
- **ARMEL and ARMHF support**: As ARM-based single-board systems like the Raspberry Pi and BeagleBone Black, among others, are becoming more and more prevalent and inexpensive, we knew that Kali's ARM support would need to be as robust as we could manage, with fully working installations for both ARMEL and ARMHF systems. Kali Linux is available on a wide range of ARM devices and has ARM repositories integrated with the mainline distribution, so tools for ARM are updated in conjunction with the rest of the distribution.

Virtualization

Virtualization technology allows examiners to run more than one operating system on the same workstation. This is useful when doing malware analysis (to keep the forensic workstation from getting infected) or when testing forensic tools before using them in the real world. The virtual machine will run in a sandbox that is completely separate from the operating system on the host machine. VirtualBox (**https://www.virtualbox.org/**) and VMware Workstation Player (**https://www.vmware.com/in/products/workstation-player/workstation-player-evaluation.html**) are two of the most popular virtual machines.

Lab information management system (LIMS)

A content management system is needed in the lab to keep track of how evidence is received, tracked, handled, and returned. If you want to do this job, you can use an open-source content management system such as:

- Drupal: **www.drupal.org/home**
- Moodle: **https://moodle.org**

There is also other software.

Software that can help you look at digital evidence, such as digital file metadata viewers and MS Office or the free alternative OpenOffice, will also be needed. These programs can help you look at different types of files, different operating systems (like Windows XP and 2000), various file types, and different programming languages (for some tools to work) for tool testing and validation.

Lab policies and procedures

Lab policies and procedures are the rules that lab workers must follow when they work in the lab. In the lab policy, there are rules for the following work areas, and more may be added.

- Physical security policy and security measures to be followed when accessing the lab area.
- Evidence handling in the top restricted area.
- Handling evidence like a write blocker attached to the suspect hard drive during acquisition.
- Evidence seized in a criminal case and steps and tools to handle each piece of evidence.
- Chain of custody documenting who has accessed digital evidence since its arrival to the lab, and also when and why.
- Disposition of evidence for how sensitive materials should be disposed of, a paper shredder for paper files, destruction equipment to safely destroy [physically] hard drives and other storage media.
- Writing a digital forensic report as the standard layout for reporting case analysis results.
- Evaluation of expert testimony.
- Backup policy.
- Training rules.
- Standards for quality.

The lab has preprinted forms for each type of work that is done inside or outside of the lab. For example, the evidence acquisition form (which records descriptions of evidence) and the chain of custody form are the two most important forms used in labs. This form will also be used for other stages of work. It will show what happened during this stage and how it was done.

Documentation

People working in a digital forensics lab need to follow the policies and procedures that were mentioned in the previous section. This will make work there easier and more accurate. It is important to have paper or electronic forms for each piece of work done during the investigation process. Examiner notes are also very important and need to be documented in great detail while the investigation is going on. In this way, another examiner can keep working on the same kind of case, and the lab's quality assurance staff can do it all over again to make sure that the same results are always produced every time.

Documentation is an important part of digital forensic investigations. It starts in the field before the computer is taken and goes on in the lab until the testimony is given. If an examiner is called to testify in court, he or she might forget important facts from the case investigation if there is no record of the investigation. This could make the examiner's testimony less credible to the judge and jury.

Lab accreditation

Accreditation shows that your digital forensic lab is following a set of standards that are agreed upon by the authority. People who work for an accredited body will check your lab to see if it uses reliable methods, court-accepted hardware or software, or trained staff, as well as if your lab's physical layout meets established rules. Accreditation is very important for any digital forensics lab, and we are going to talk a little about how to start the accreditation process.

Conclusion

A computer forensic lab is where you investigate, store your digital evidence, and do most of your forensic work. Different kinds of hardware and software are used in labs to get and analyze digital evidence and then write a formal report about what they found. In this chapter, we talked about the things you need to start a digital forensics lab. We talked about how the lab will be set up in the physical space. We also talked about what kind of electrical equipment, lab furniture, and hardware devices are needed for digital investigation work. Finally, we talked about what the forensic workstation needs to be able to do. We talked about how to design and protect the lab network, and then we talked about forensic software. Different kinds: some are for sale, and others can be closed-source or free, or open-source.

It is recommended that the forensic software be approved by a credible body before it is used in an investigation; in-house tools or tools that have not been approved by a third party must be checked internally before they can be used in official investigations.

CHAPTER 5

Acquiring Digital Evidence

Introduction

A computer forensics investigator's main job is to collect and analyze images from computers. In a nutshell, a forensic image is a static snapshot of all or part of the data on a computer's secondary storage (for example, HDD and SSD), attached storage device (for example, USB thumb drive, external hard drive, and magnetic tape), or RAM (when performing live acquisition on running systems). This image can be considered a data container, allowing you to store individual files or the entire drive/live memory files in a single image file. The digital evidence must be retrieved and analyzed to identify indications of security incidents, fraud, and other illegal practices that target information systems will be contained in a forensics image. Remember that forensic images may be used in court, so the tools and techniques used to acquire and analyze them must be legal.

Structure

We will cover how to create forensics images from both running systems (volatile memory, RAM) and hard drives (HDD, SSD, flash thumb, and any other similar digital storage media) in this chapter, but we will save the discussion of how to analyze acquired images of deleted files and other interesting artifacts for the upcoming chapter.

Objectives

We start by discussing the various file formats that forensic images use to store data. Forensic images can be in various file formats, some of which are open source and others that are proprietary to the company that developed the forensic software that was used to create the image. The ones listed as follows are the most commonly used in the business.

Raw format

Raw format is the most commonly used file format; it is a bit-by-bit copy of the drive's raw data and can be used to image the entire drive or a single volume (partition) within it. The ability to ignore minor read errors from the source drive and its fast are two of the raw file format's main advantages. Although the raw format cannot store metadata, some applications do so in a separate file (for example, the image file's hash value, the drive serial number, and so on). Raw format is the default file format for the output generated by the famous Linux/UNIX dd command, and it is supported by most computer forensics software. 001, dd, dmg, raw, and img are just a few of the naming schema (extensions) for raw format. The main disadvantage of the Raw format is that it requires the same amount of storage space as the source drive because data in Raw format cannot be compressed, which may be a problem when purchasing large hard drives.

Advanced forensic format

Advanced forensic format (AFF) is an open-source, extensible file format for forensics images that can be freely integrated into other open-source and proprietary programs. The compression algorithms Zlib and LZMA are both supported by AFF. You can also divide the forensic image file into multiple files after it has been created. AFF (beginning with AFF V2.0) supports drive image encryption, allowing you to password-protect your acquired image. AFF allows for a wide range of metadata information to be stored within the image file itself, reducing the amount of work and allowing for a single file to contain all information related to the acquired forensic image (for example, a metadata file can contain the chain of evidence or audit trail). AFF4 is the most recent version, and AFF3 and AFFLIBv3 are no longer supported and should not be used in new projects. Sleuthkit, Autopsy, OSFMount, X-mount, FTK Imager, and FTK are all computer forensics software that works with newer versions of AFF. For segmented image files, AFF uses the `.afd` extension, whereas AFF metadata is saved as `.afm`.

EnCase: Expert witness transfers

This is a proprietary file format developed by Guidance Software (now OpenText) for their popular "*EnCase Forensic*" product, which is widely used by law enforcement in criminal investigations around the world. This file format can be used to store a variety of digital evidence; it is compressible and searchable, and the image it produces can be split into multiple files. Metadata can be associated with the same image file; however, compared to the AFF file format, the quantity and type of metadata are limited.

EnCase divides the resulting image into 640 MB chunks when capturing hard drives. File extensions will change according to chunk number as a result of this division of forensic image data (for example, first chunk extension ".e01", second chunk extension ".e02", and so on).

Other file formats

Other forensics image file formats are proprietary formats used by some computer forensics suites and are less widely used (such as Safe back by NTI, ILook Imager, and ProDiscover).

Validation of forensic imaging files

Validation ensures that acquired forensics image files are 100% identical to the source and have not been altered during the acquisition process. In the computer forensics industry, hashing is the accepted standard for validating forensic images acquired. The resultant image file's hash value is thought to be an electronic fingerprint. Most computer forensics software will generate a hash value of the captured data when it is finished; however, you can calculate the hash value of any piece of data using third-party tools or the standard utility, which is available in modern Windows versions via PowerShell.

Live memory acquisition

Live acquisition has become an essential component of any digital investigation type, despite receiving little attention until recently. Many types of digital artifacts, for example, exist solely in RAM, with no evidence of their existence written to the hard drive. When a device is turned off or rebooted, the data it contains is considered volatile. Please keep in mind that such information will be overwritten when using a computer (for example, when closing a specific application on a PC, the reserved data space will

disappear from RAM, allowing other applications to use its space for operation) and gets completely lost upon system shutdown. Specialized software (and, in some cases, hardware) tools are required to capture a live memory. Because RAM does not store data in the same way as hard drives, analyzing volatile data forensic image contents also necessitates specialized software.

Because of these two factors, capturing and analyzing volatile memory is more difficult than traditionally acquiring hard drives. For example, networking devices like routers and switches can store volatile data in their logs. Dumping is the process of extracting data from volatile memory, and the method for doing so varies by the operating system. Only computers running Windows OS are discussed in this book.

Information that can be found in RAM includes the following:

- Keys for encryption
- Running procedures
- Console commands that have been executed
- Items in the clipboard
- Data from a network
- Contents of decrypted
- Hives of the registrar
- Image and text files
- Deleted files
- Web Browsing Logs
- Registry keys that are open and active
- Passwords for Web accounts (for example, e-mail, social media, and cloud storage)
- Instant messaging
- Exploit information
- Rootkit and Trojan malwares
- Evidence of activity that is not usually saved on the local hard drive

Getting data out of volatile memory is similar to getting data out of a hard drive, but it requires different tools because data in RAM is not structured like it is on hard drives. Before we get into how to capture RAM, there are a few concepts about volatile memory that any digital forensics examiner should be familiar with.

Virtual memory: Swap space

`Pagefile.sys` (also known as virtual memory) is a Windows file that compensates for RAM's limited capacity. The default location for it is **`C:\> pagefile.sys`**. The initial virtual memory paging file in Windows is normally set to the amount of RAM you have installed; however, a user or system administrator can usually change its size. When your machine's RAM starts to fill up, this feature allows Windows to use the hard disc space as memory. To make room for more data, portions of RAM files are moved into virtual memory. The operating system can no longer directly process any of the files that have been sent to virtual memory. As a result, it will need to send more files to virtual memory to free up space

so that it can retrieve the files it wants to process from virtual memory and put them back into RAM. The user is not aware of this process, which is known as swapping or paging.

Virtual memory acquisition is a critical part of the forensic process because it can contain valuable information such as user passwords, encryption keys, Web browser activity, and other important artifacts that are transferred from the RAM. Virtual memory can be acquired by some RAM capture tools in addition to RAM (for example, FTK imager).

Challenges acquiring RAM

Forensic examiners will face some difficulties when acquiring live memory. When doing a live acquisition, keep the following points in mind.

- **Locked Windows**: It is best to perform a hard shutdown if we come across a computer that is running but has a login screen (locked computer). Some experts argue, however, that by using some tools/techniques to avoid losing RAM contents, we can bypass the Windows login page without having to reboot.
    - To gain access to live the memory and encrypted discs without a password, use CaptureGUARD and Phantom Probe hardware accessories (**www.windowsscope.com**).
    - To log in to the system, use a **direct memory access** (**DMA**) attack to get the password out of RAM.
- Always keep in mind that using such techniques will leave traces in RAM and may not be successful in some cases, so conduct a risk assessment to determine whether the forensic live acquisition is worth the effort and if in doubt, consult a professional examiner.
- DMA is a computer system method that allows some hardware components to interact directly with the computer's physical memory (RAM) and transfer data to and from it without having to go through the computer's CPU first. This method is used to reduce processing time and increase computer throughput by transferring data directly from RAM to the CPU without having to process it first. Computer forensics can use this technique to gain access to sensitive data on a target machine, bypassing all operating system security mechanisms as well as any lock screen or antivirus software.

Imagine a scenario: Digital forensics examiner will connect his or her device (mobile forensic workstation) to the suspect machine and use special cracking software to search the suspect PC's RAM for interesting artifacts such as cryptographic keys, passwords, or decrypted files. The suspect PC must have DMA-capable ports for this method to work. FireWire, Thunderbolt, PCMCIA, PCI, PCI-X, and PCI Express are all examples of these types of ports. The DMA feature is not available on USB ports.

Administration privilege

To work, the majority of RAM capture software tools require administrative privileges. If you come across a running PC with limited user permission (for example, a user account), you can acquire RAM with a hardware acquisition tool (which requires a small driver to be installed on the target machine) or a DMA attack. On the suspect machine, the capturing tool that was used to obtain the RAM will leave traces. Some data may be overwritten as a result of acquiring live memory, despite claims from computer

forensic software vendors that their tools will leave a minimal footprint on the acquired system. To work, hardware acquisition tools will also require the installation of a small driver on the target machine. To avoid making your evidence inadmissible in court, these changes should be well documented in the final report of the investigation. Any Windows machine that is subjected to a live acquisition will usually be changed in the following ways:

- Changes to the registry
- Entries in the memory (overwrite data in RAM)
- It is possible to write a small amount of data to a disc drive

Courts are usually lenient when it comes to small footprints left by RAM capturing tools; however, make sure to document each interaction with the suspect computer while capturing RAM in your final report, and use legally acceptable tools to do so.

Creating RAM dumps can be performed using various imaging tools, such as DumpIt, a small portable tool for acquiring RAM on Windows computers (32- or 64-bit) as follows:

- To access the download section, go to the link **https://my.comae.io/login** and register for a free account.
- Put the tool on a USB drive (if you plan to run it from one); keep in mind that this USB drive will hold the target computer's RAM, so make sure it is big enough to hold the file you will be making. If you want to record an 8-GB RAM, for example, your USB drive should have 9-GB of free space.
- To use DumpIt, double-click it and type "y" to confirm that you want to copy the target computer's (Windows) RAM (*figure 5.1*). The RAM file that was captured will be stored in the same location as DumpIt.
- It is worth noting that the captured image is larger than the RAM (in this case, we are capturing a PC with 8 GB of RAM; the image size is approximately 8.269 GB).

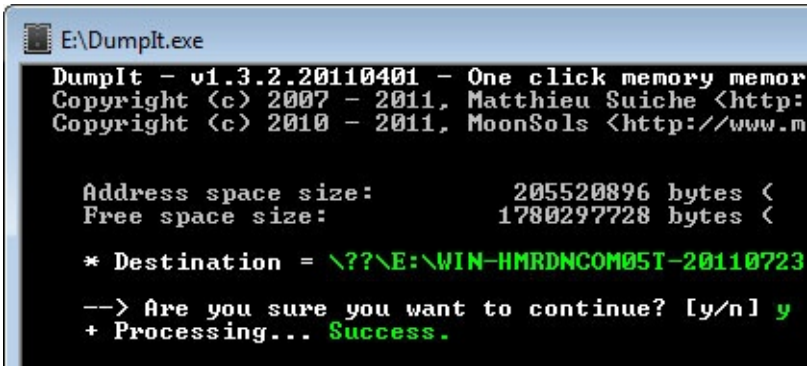


Figure 5.1: *RAM with the DumpIt tool*

After completing the acquisition, DumpIt creates two files: a DMP file containing the RAM image and a JSON file containing important technical information about the captured machine, such as Machine Info, Architecture Type, Machine Name, Physical Memory, Username, OS Version, and Service Info.

Live RAM capturer

Belkasoft is the second tool that we will use to capture RAM. This is a small free tool that runs from a

USB thumb drive and can capture the entire contents of RAM, even if the system is protected by an active anti-debugging or anti-dumping system. To keep the tool's footprint as small as possible, separate 32-bit and 64-bit versions are available. All versions and editions of Windows, including XP, Vista, Windows 7, 8, 10, 2003, and Server 2008, are supported by Belkasoft Live RAM Capturer. Follow these steps to use this tool:

- Visit **https://belkasoft.com/ram-capturer** to get the tool (you will need to fill out a simple registration form first to proceed to the download section).
- Place the tool on a USB drive with more storage than the target computer's RAM.
- On the PC where you want to capture the RAM, run the program and click the "**`Capture`**" button (*figure 5.2*).

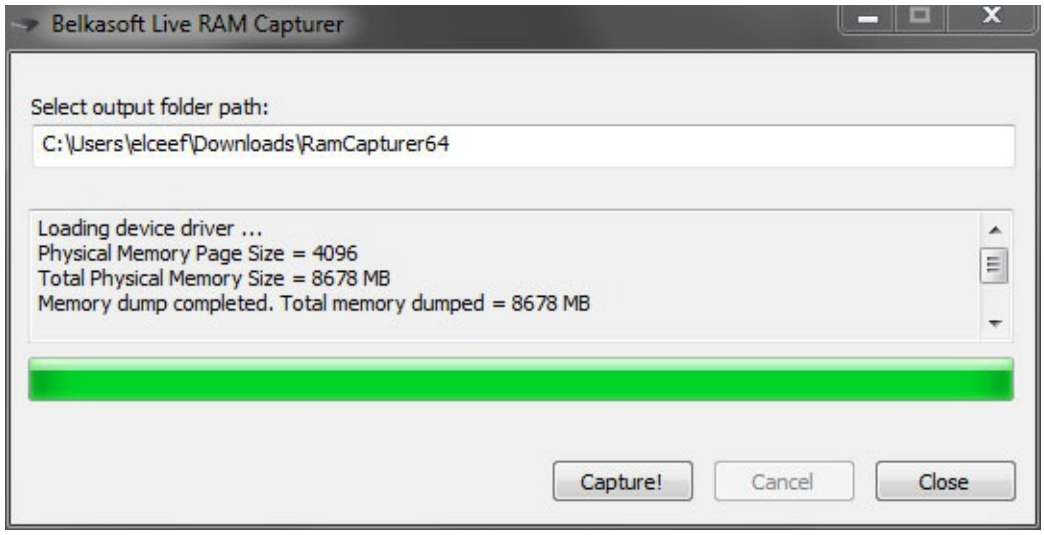


Figure 5.2: *Belkasoft to capture RAM*

Magnet RAM capture

Magnet is a portable RAM capture tool that claims to have a small footprint on the target machine and supports almost all Windows OS versions, including Windows XP, Vista, 7, 8, 10, 2003, 2008, and 2012 (32- and 64-bit). It is easy to use this tool; go to **www.magnetforensics.com/free-tool-magnet-ram-capture/** and fill out a short form to get the download link. Place the tool on your USB thumb drive and connect it to the target machine; then run the tool and choose where you want the resulting RAM image to be saved. Finally, press the **`Start`** button to start the recording (*figure 5.3*).

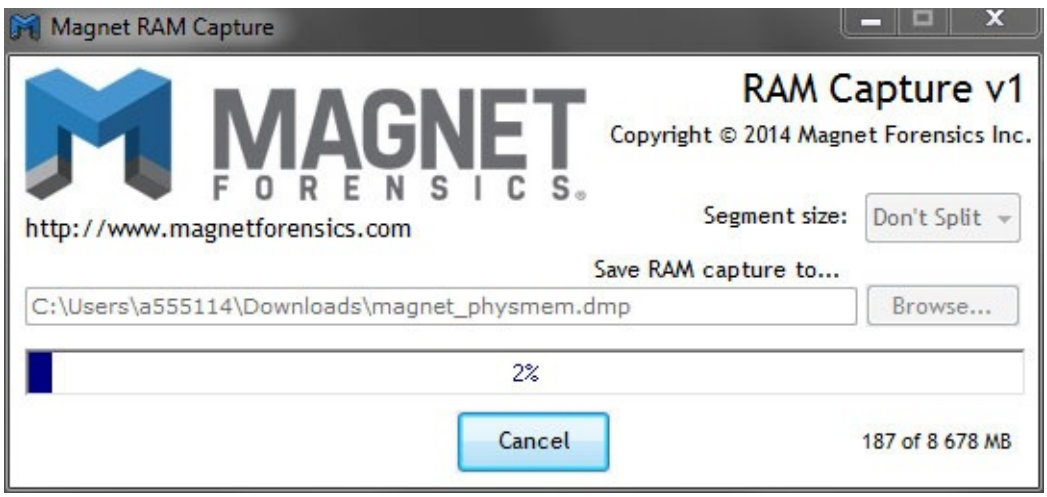


Figure 5.3: *Magnet RAM capture*

FTK imager

FTK imager is a data preview and imaging tool for creating forensic images of target computer data without altering the original evidence. You can make forensic images of local hard drives, floppy diskettes, zip disks, CDs, DVDs, entire folders, or individual files from various locations within the media with this tool, as presented in *figure 5.4*.

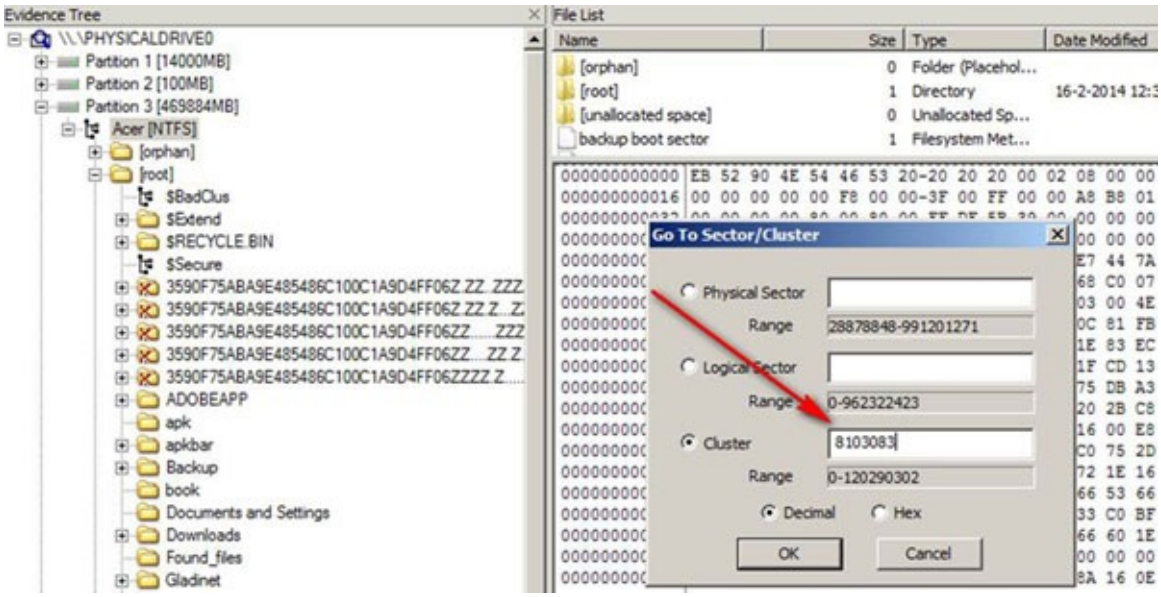


Figure 5.4: *FTK Imager for RAM capturing*

FTK imager can also be used for tasks other than image acquisition, such as the following:

- Installing a read-only image.
- Examining the content of forensic images.
- Files and folders can be exported from forensics images.
- Obtaining the Windows registry.
- Recovering files that have been deleted.

The tool can be installed locally on the machine where it will be used, or it can be run from a USB thumb drive connected to a field machine (the latter is preferred when conducting live forensics on running systems). We will show you how to install this tool on a portable media device (such as a USB thumb drive) before we start using it:

- Select which version of "FTK Imager" you want to download from **https://accessdata.com/product-download**. You must first complete a registration form (inputting your full name, e-mail address, and job title, among other information); after that, a download link will be sent to the e-mail address you provided.
- Run the installation program. You can now install it on your USB thumb drive in one of two ways:
    - Run the installation on a local computer, then copy the FTK Imager folder to the USB thumb drive from [Drive Letter]: Program FilesAccessDataFTK Imager.
    - Avoid installing the FTK Imager files on a local computer by installing them directly to the

thumb drive. The installer will unzip the downloaded files to the portable drive, which can then be connected to any computer with a Windows operating system, and the program file (FTK **`Imager.exe`**) will be executed from the portable USB device.

Follow these steps to use this tool for RAM capture:

1. FTK Imager is launched from the USB thumb drive (if you select to install it on a USB, as we already demonstrated). Select File Capture Memory from the File menu. A new window appears with options for capturing the current machine's RAM (*figure 5.5*).

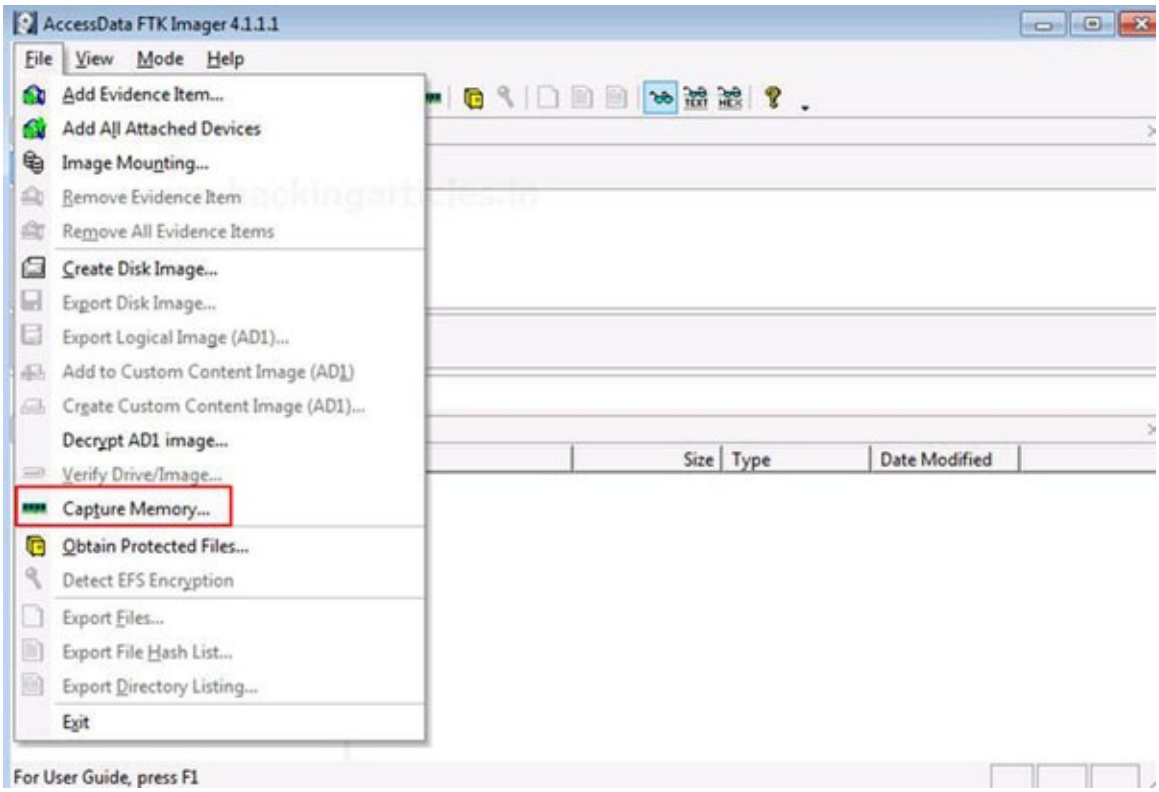


Figure 5.5: *FTK imager: capture memory*

2. The user will be prompted to choose a location for the generated RAM picture, as well as whether or not to include the pagefile in the newly opened window (**`pagefile.sys`**). Click the **`Capture Memory`** button when everything is in place, as shown in *figure 5.6*:

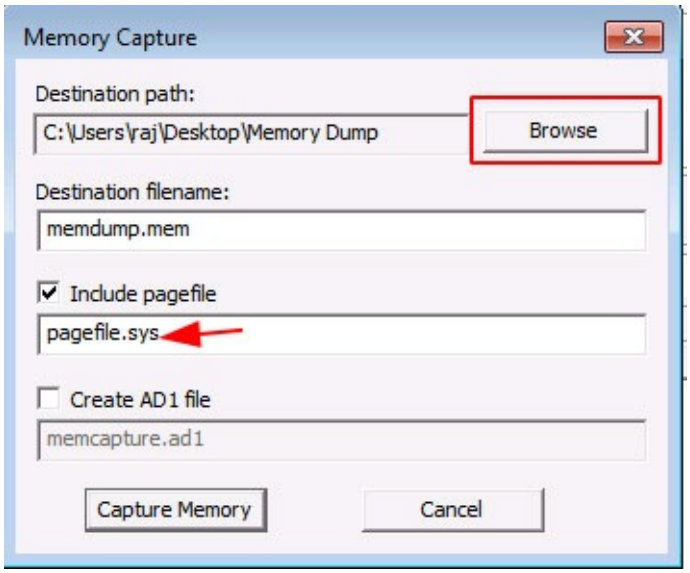


Figure 5.6: *Browse the destination path*

3. The capture will now commence, with a progress meter displaying how far the capture has progressed. The open window will indicate whether or not the memory dump was successful once the capture is finished, as illustrated in *figure 5.7*.

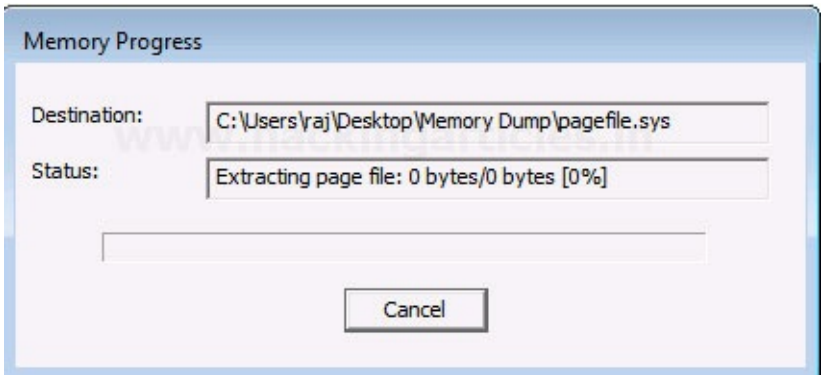


Figure 5.7: *Memory capture process*

4. Navigate to the memory dump's saved location. If you choose to acquire it, you should discover two files: mem dump.mem (or whatever name you gave it) (*figure 5.8*) and the **`pagefile.sys`** (refer to *figure 5.9*). When the dump was processed, these two files had the whole contents of RAM.

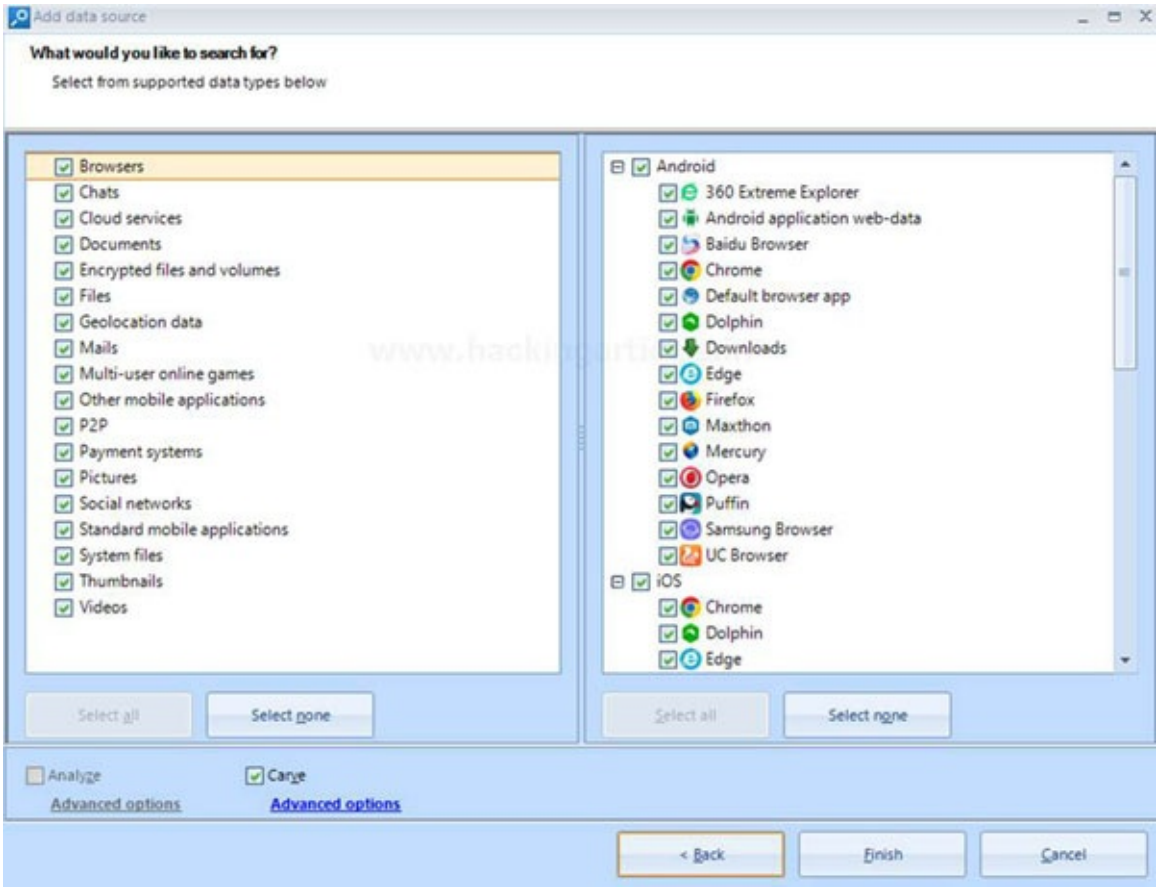


Figure 5.8: *FTK imager RAM*

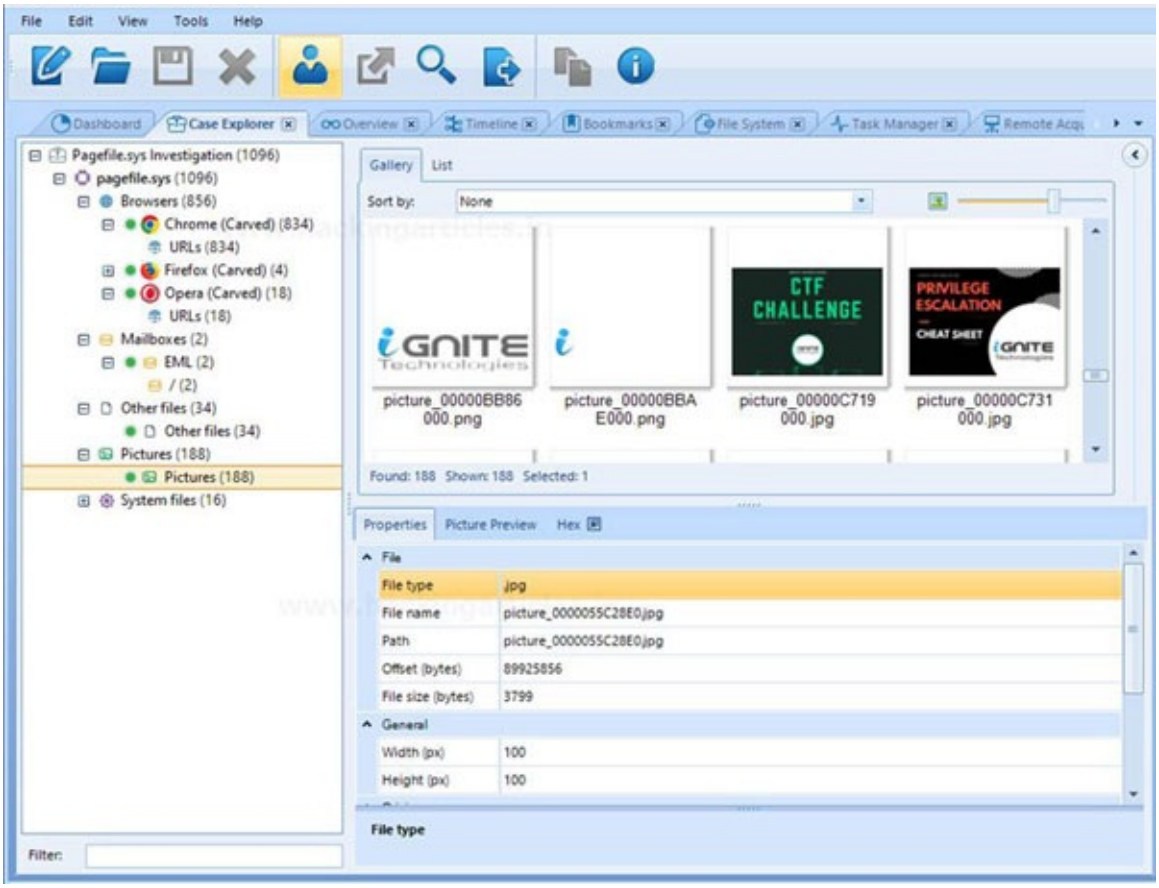


Figure 5.9: *Pagefile.sys dump*

Acquiring nonvolatile memory

Nonvolatile memory refers to any storage medium that can keep data even after the power has been turned off for an extended period. Hard drives and flash memory are two of the most common types (flash drives). Hard drive images are the most important part of any computer forensic investigation because they contain the majority of data that may contain incriminating or exculpatory evidence. FTK Imager, Pro Discover, EnCase, and X-Ways Forensics are just a few of the tools that can be used to acquire hard drive images in the Windows OS. Before you attach the suspect hard drive to your forensic workstation, make sure it is write-protected. Hardware tools or software programs can be used to protect data from unauthorized access. Many investigators prefer to boot from a CD/DVD using a Linux forensic distribution like CAINE (**www.caine-live.net**) or DEFT (**www.deftlinux.net**), which is preconfigured to block automatic disc mounting, and then attach a suspect drive without the risk of data from external sources being manipulated.

Hard disk acquisition

During investigations, you can use a variety of static acquisition methods. We should consider the following criteria before determining which one to go with:

- The size of the (suspicious) source drive. (Obtaining large-capacity hard drives necessitates huge storage units to hold the forensic image created, which can take longer during processing.)
- Time frame in which the acquisition will be completed (if time is limited, you cannot spend hours on acquiring the entire hard drives of suspect computer[s]).
- Is it possible to bring the questionable digital media (for example, a hard drive) to the lab with

you, or should the acquisition be done on the spot?

- Is it possible to shut down the target machine to obtain its drive data, or is this impossible due to a variety of factors (for example, shutting down an e-mail server could result in a significant business loss)?

You can choose the acquisition strategy that best suits the scenario after examining these and other variables. The three most common methods for obtaining forensic photographs are listed as follows.

Acquiring physical resources

We make a bit-by-bit/sector-by-sector clone of a hard disc using this method, also known as a bit-stream image. This approach will also capture metadata from the file system, deleted files, deleted file fragments, and unallocated space. Unless compression is used during the acquisition process, the resulting image will be a complete reproduction of the source (an exact copy). For example, if we make a forensics image of a 500 GB hard drive, the resulting image will be exactly 500 GB. Any computer forensics program can read bit-stream images, and as we previously stated, you must connect the suspect's a hard drive to a hardware write blocker so that the forensic workstation used to acquire the image does not write data to the suspect's hard drive during the acquisition process.

According to where the acquired data is stored, we can distinguish between two types of physical acquisition:

- **Bit-stream disk-to-image file conversion**: Captured data is saved as an image file. This is the approach of investigation that is most frequently used. It allows you to make an identical bit-for-bit copy of the source drive and save it as an image file. The key benefit of this procedure is that it allows you to make many copies of a questionable disc while keeping the original media intact.
- **Bit-stream disk-to-disk**: In this method, data is copied (bit by bit) from the source drive to a newer drive with the same or slightly greater storage capacity. Although this method is not generally used, it is nevertheless necessary for specific situations, such as when purchasing an old hard disc drive. Some computer forensics software (for example, EnCase and X-Ways forensics) can alter the shape of the new hard disc (destination) drive such that the collected data is in the same location as the source (suspect) disc drive.

Logical acquisition

We are just capturing a subset of active data using this approach. When we talk about "*active data*", we are talking about the information that is in front of us when we are using a computer. This method will not capture unallocated space, file system data, deleted or partially erased files, hidden data, or all unused space. If you logically capture a 500-GB drive with just 100 GB of active data, you will only get the 100 GB image. When the target (suspect) drive is too big (for example, RAID storage) and the first responder does not have time to execute a full volume (physical) acquisition on-site, logical acquisition is a viable option. It is also possible if we wish to obtain a specific file(s) in a targeted manner (for example, acquire e-mail files only from the target machine or when we want to capture all photo files existing on a suspect drive). When dealing with certain sorts of civil action, a logical acquisition may be the only feasible option (e-discovery). You may also use search terms to find a specific keyword or combination of keywords among large datasets and then only get the results.

Sparse acquisition

This method is similar to logical acquisition in that it captures only specific files that are related to the investigated case; however, deleted data and fragments thereof are also captured during the capturing process in sparse acquisition. When executing static acquisition on RAID systems or when the suspect is not tech skilled enough to deploy advanced anti-forensic measures, this method is frequently used. It is time to start gathering hard disc images now that we understand the various ways of hard drive acquisition. Many different types of software can conduct hard drive acquisition; however, we will not be able to cover them all in this book, so we will use FTK Imager, which is a free and dependable program.

Capturing hard drives using FTK imager

We have already used this tool to capture RAM; the process for capturing a hard disc is similar.

- FTK Imager can be downloaded and installed.
- Go to **`File`** Create Disk Image in the AccessData FTK Imager application.
- A new window will open (*figure 5.10*), where you must select the type of source evidence.
- Choose **`Physical Disk`**, and then press **`Next`** to go to the window where you can choose which physical drive to an image, as illustrated in *figure 5.10*. There are five choices available to you:
    - **Physical drive**: It is the most popular approach. Physical drives enable you to capture all data on a hard drive (bit by bit), even unallocated space and deleted files.
    - **Logical disk**: Capture only a certain partition within a drive, such as the D: drive.
    - **Image file**: Choose an image file to use as a source here.
    - **Folder contents**: Assign a source to a folder.
    - **Fernico device**: Recover forensics pictures from a variety of sources (including multiple CDs/DVDs).

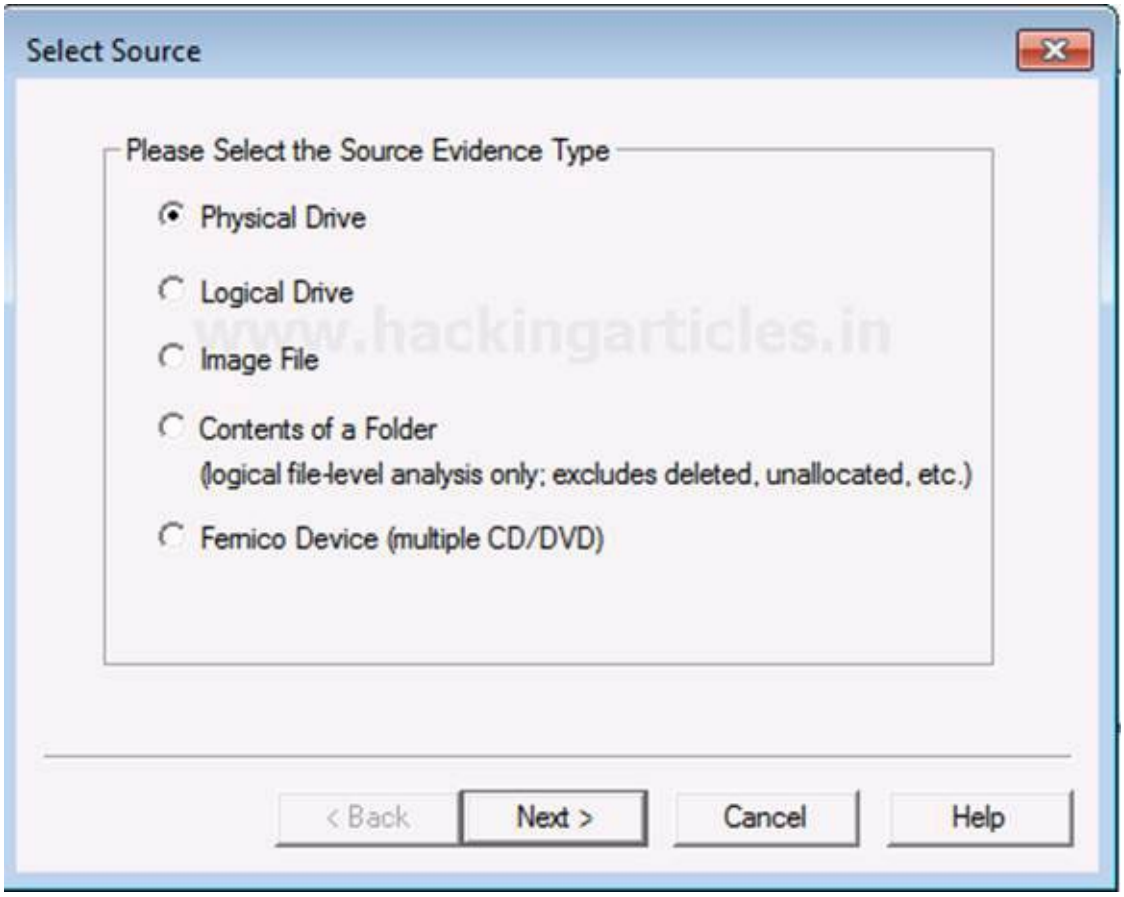


Figure 5.10: *Select physical drives in FTK imager*

- Select the physical drive you wish to image (refer to *figure 5.11*) and proceed to the next window; click the **`Finish`** button; you will be asked to save the drive image to a specific location. When you click the **`Add`** button, a new dialogue box will display, prompting you to select one of the options formats.

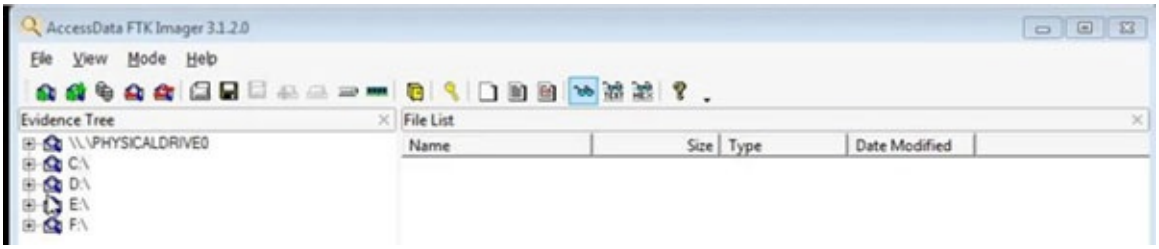


Figure 5.11: *FTK imager for hard disk analysis*

- In this example, we will choose “Raw (dd)”; click the “**`Next`**” button, and a new window will emerge, prompting you to add evidence information such as the case number, evidence number, unique description, examiner, and notes. There are no required fields (*figure 5.12*). To move on, click on **`Next`**.

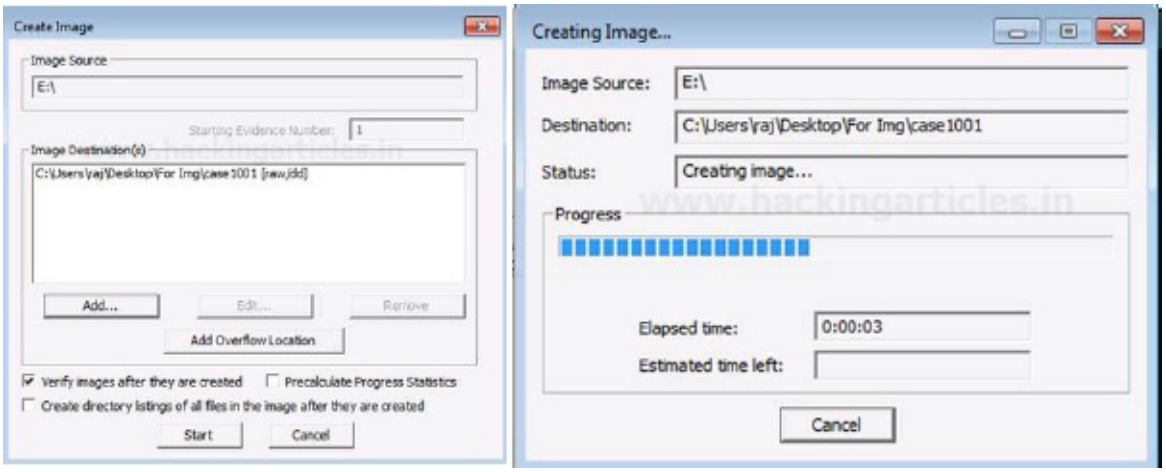


Figure 5.12: *Raw image creation*

- The investigator can choose where to save this image, as well as the file name and whether or not to break it into several fragments (fragment size is measured in megabytes). A forensic image in the Raw file format cannot be divided if you want to protect the image with a password, select **`Use AD Encryption`** and type a password twice after clicking **`Finish`**. To protect the image, the FTK imager employs the AES-256 algorithm (*figure 5.13*).

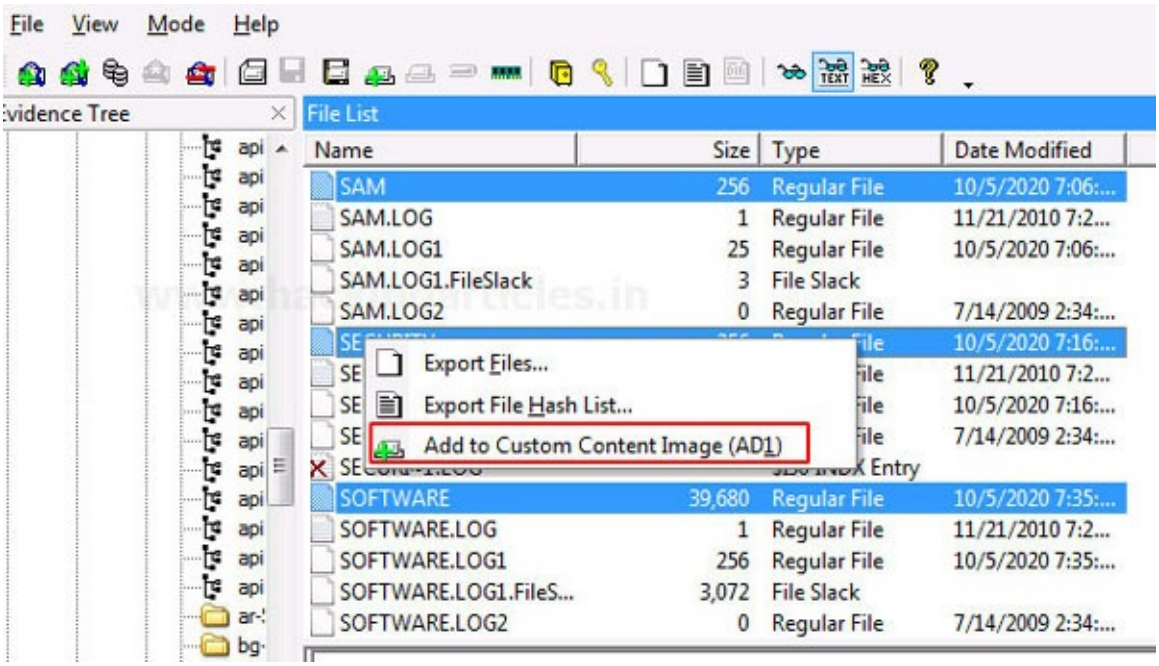


Figure 5.13: *Content image AD encryption*

- We do not need to protect the image with a password in this situation, so we will just click **`Finish`**. This will return us to the original **`Create Image`** screen, where the **`Start`** button will be enabled (*figure 5.14*).

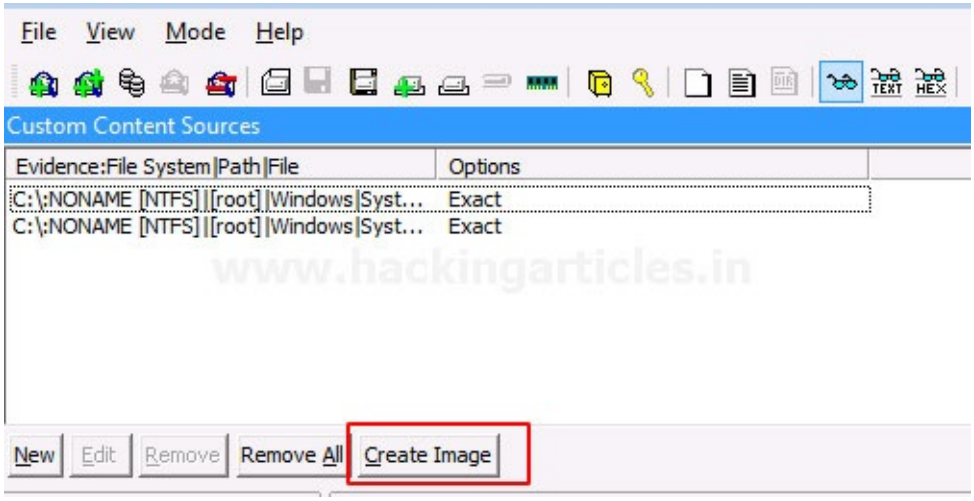


Figure 5.14: *Create FTK imager image*

- Check the box next to "Verify images once they are made" to guarantee that the source drive and the final image are the same. After you have finalized everything, hit the **`Start`** button to start the acquisition process.

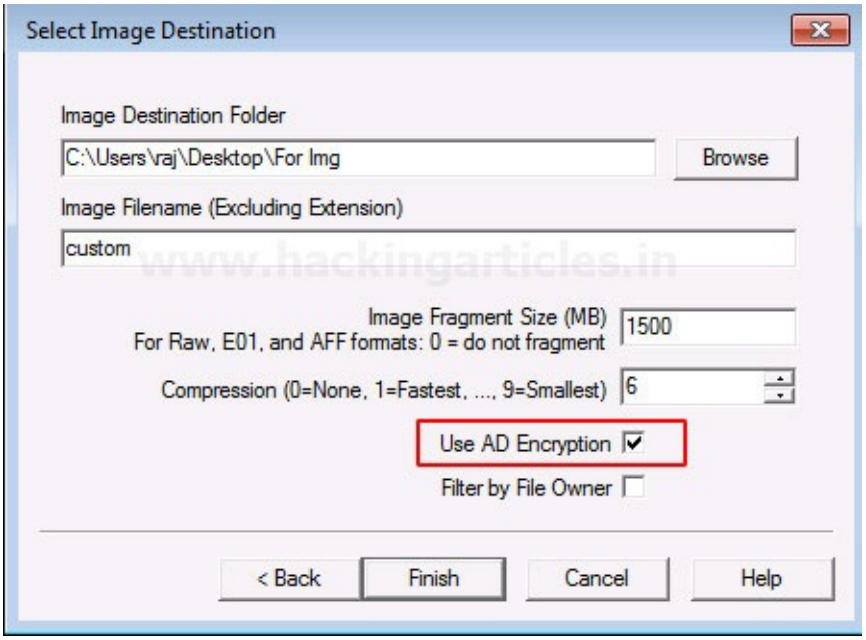


Figure 5.15: *Add the destination of the image*

- The status of the acquisition will be displayed in a window with a progress bar. The verification procedure starts once the image has been created (*figure 5.16*).

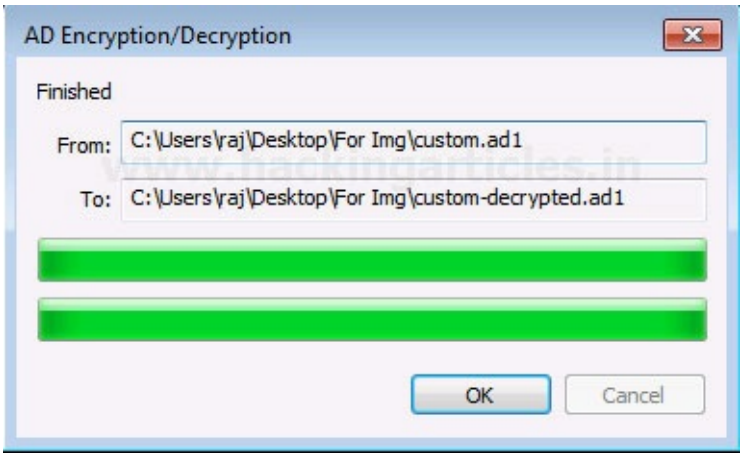


Figure 5.16: *Decrypted image creation*

- Now, you will be able to view the encrypted files using the pass.

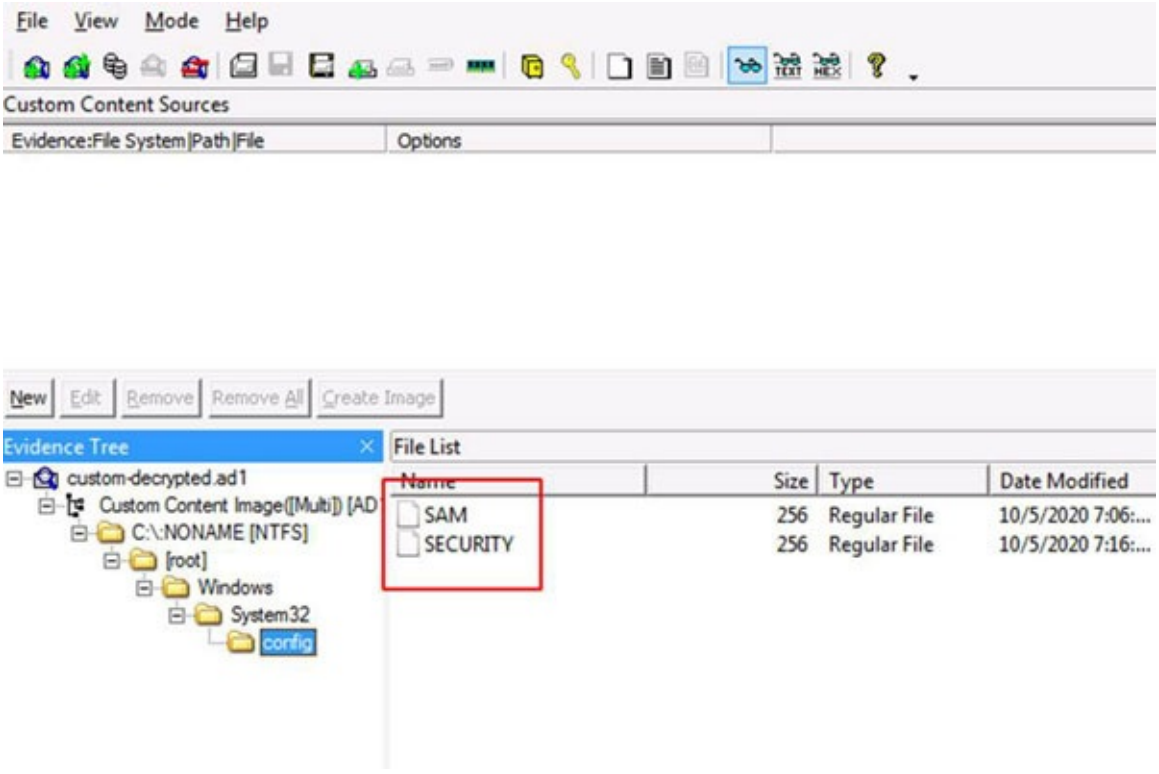


Figure 5.17: *View encrypted files*

- Navigate to the location where you want the acquired image to be saved. There should be two files there (*figure 5.18*), the image file and the metadata file connected with it (text file). Because we choose to store the image in Raw format, FTK Imager has created a separate metadata file for the image. If we use a different file format (for example, .e01), we will just have one file, the picture file, with the information attached to it.

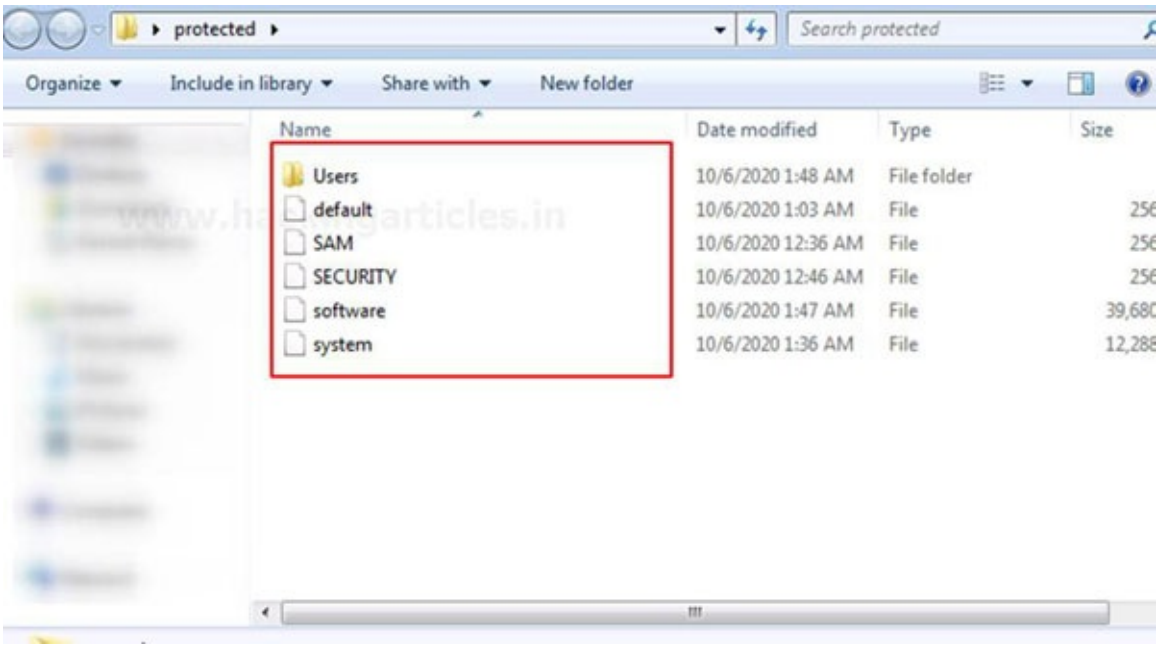


Figure 5.18: *View protected files in one place*

Network acquisition

Network forensics and cloud forensics face similar issues; for example, e-crime, which involves the use of networked computers, is on the rise. Working on criminal cases involving the usage of computer networks as a forensic examiner, you can anticipate confronting the following challenges.

- In most cases, you will have to collect and analyze a big amount of data (for example, acquisitions of redundant array of independent disks [RAID], which involves two or more hard drives).
- Because the evidence may be scattered over several device kinds in the target network, technological expertise will be required.
- Corporations will impose organizational challenges that will necessitate an investigation; for example, you cannot terminate a specific service because it is critical to the company's success.
- Multiple jurisdictions might create difficulties; for example, there have been instances where a storage server was located in Europe, but the investigation—or the breach—was conducted in New York. What legal steps will you take? Other legal concerns occur when multiple jurisdictions impose different privacy standards; for example, private information (for example, regarding customers, partners, or workers) may be exposed to the examiner in a network breach, and such data could be protected by different privacy regulations.

Limitations of a forensic tool

Some forensics collection software is unable to copy or access data in HPA and DCO, yet, these two locations may contain damning information that must be obtained for examination. Always check the documentation of the acquisition tool to see if it has this feature; if it does not, it is best to use a hardware acquisition tool.

Conclusion

Capturing a computer memory image is the primary goal of any digital forensics inquiry. The most popular approach for obtaining digital images is a bit-stream image, which involves copying all data from the suspect drive, including deleted files, fragments of deleted files, and unallocated space, into a forensic image file that can be evaluated later for digital evidence. Two types of image orensic acquisitions can be distinguished:

- Capture RAM and other volatile data, such as network information of the live acquisition.
- Static acquisition of nonvolatile memory (HDD, SSD, and digital storage media).

To maintain the integrity of collected forensics images, a hash value is used to ensure that they are 100% similar to the source and have not been altered throughout the collection process. It is always a good practice to create multiple copies of the image file; this way, you will keep the original media undisturbed while having many images to work with in case something goes wrong (for example, the image is mistakenly edited) during the analysis process. In this chapter, we have discussed the use of different tools to capture RAM and the various considerations and challenges that come with it. We also reviewed various methods for acquiring hard drives and using FTK Imager to capture an HDD. In the upcoming chapter, we analyze the process now that we have obtained a questionable hard drive and RAM image.

CHAPTER 6

Analysis of Digital Evidence

Introduction

In the previous chapter, we explored how to capture and create a forensic picture of both RAM and hard disks. Now, we are ready to go on to the next step of forensic investigation, that is, evaluating obtained images for relevant leads. All analytical work should be completed on the forensics image only; forensic examiners should not meddle with the original suspect equipment to prevent harming original evidence accidentally and so rendering the entire inquiry useless in a court of law. In this chapter, we will learn how to mount and analyze collected hard disk forensics images using several tools, concentrating on open-source and free tools.

Structure

This chapter presents and discusses the following in detail:

- Arsenal Image Mounter
- OSFMount
- Autopsy
- Fireeye tools
- Volatility

Objectives

The hard drive is the main data storage used in a computer system: the bulk of computer users' and corporations' data is kept on it. When a hard drive becomes a part of a legal inquiry, it should be lawfully acquired, as we saw in the previous chapter, and evaluated for information that can aid in solving the subject occurrence. Hard drives today are huge and frequently contain a tremendous number of data; it is the responsibility of the forensic examiner to analyze this data and connect the dots to solve a crime or to explain what happened during an occurrence.

Arsenal Image Mounter

Arsenal Image Mounter is a free, open-source application. It can mount a forensic image as entire drives in Windows (actual SCSI disks), allowing investigators to examine image contents as if they were exploring any directory of files. Although the free version may mount any forensic picture, the commercial one includes more sophisticated capabilities. This utility supports forensic pictures in Raw and EnCase file format, and it also supports all file systems used by the Windows OS, including NTFS and FAT32, as presented in *figure 6.1*.

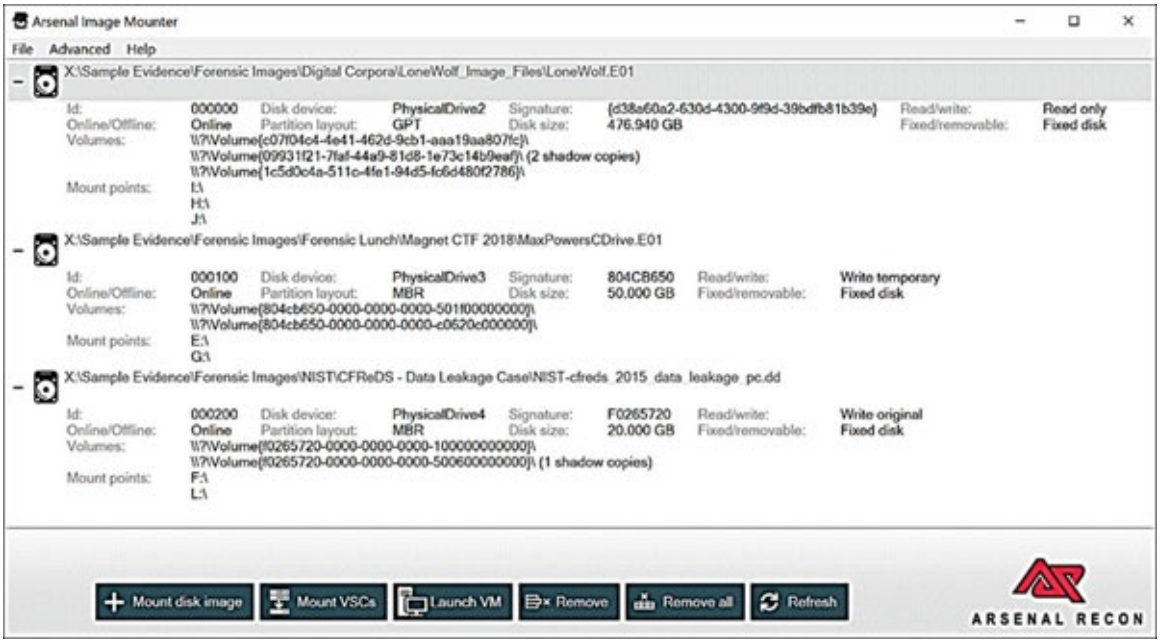


Figure 6.1: *Arsenal Image Mounter*

Using this utility is quite simple: browse to **https://arsenalrecon.com/** and download the software to your PC. Arsenal Picture Mounter is a portable software, so just launch the program and then click the **Mount Image** button in the main program window to choose the image file; now, the tool will ask you for mount choices (see *figure 6.2*). Check the first option, "Read-only," and then hit the **OK** button.

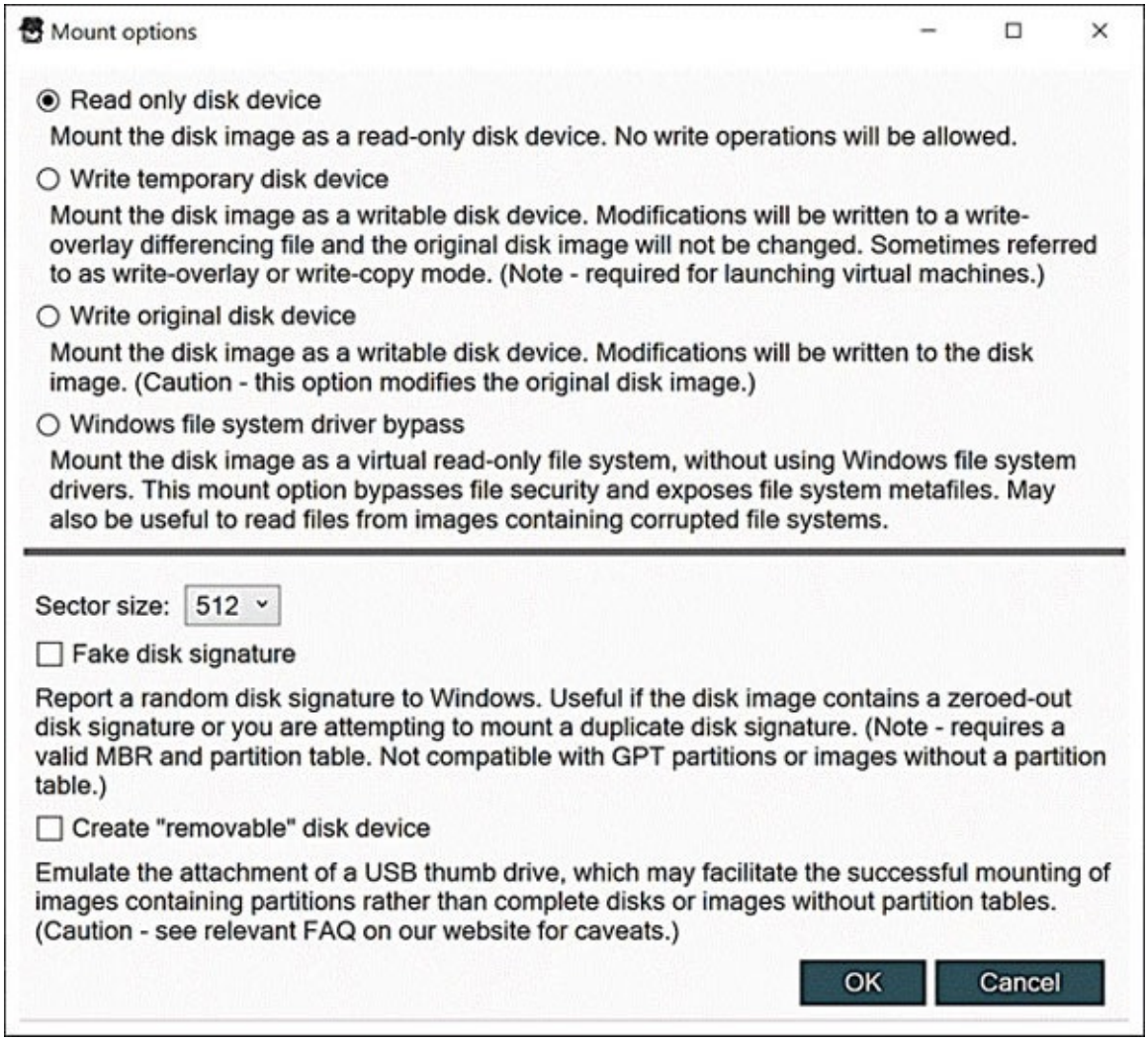


Figure 6.2: *Image Mount options*

Now, the forensic image will get mounted (see *figure 6.3*) as a virtual drive in Windows (you may view it via Windows file explorer) as if it were a local disk.

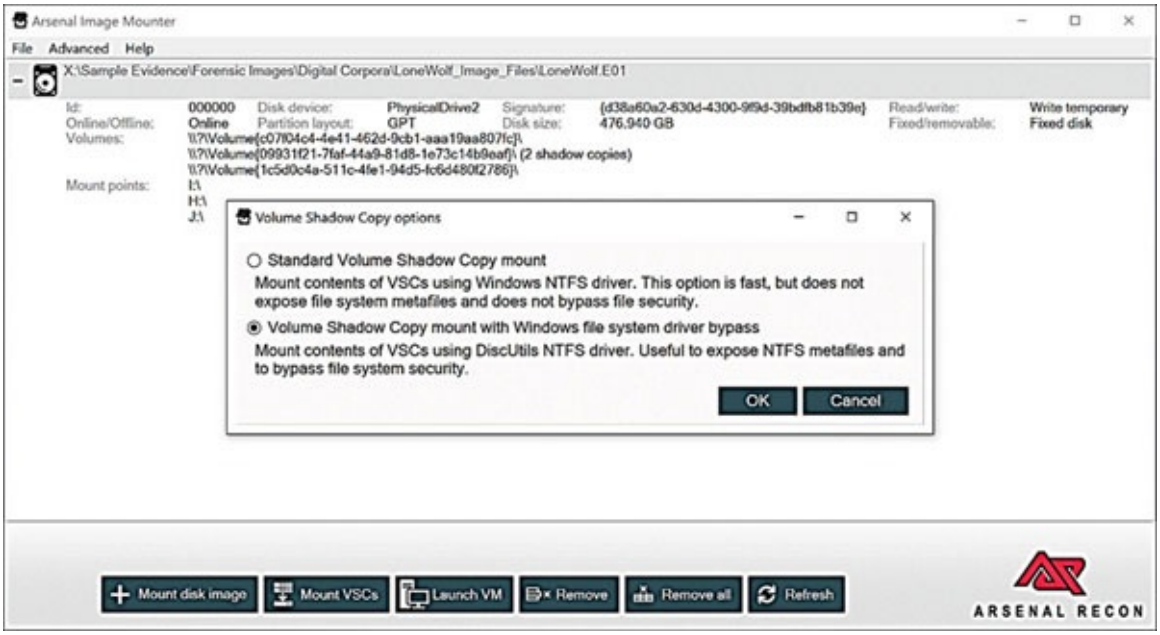


Figure 6.3: *Arsenal volume shadow copy mount*

OSFMount

This is another tool for mounting the forensic disc image as local Windows drives; OSFMount enables mounting images of CDs in ISO format. It also enables the construction of RAM discs (a disk mounted into RAM). Supported image file types include, among others, AFF, Raw, split Raw, and EnCase. To use this software, follow these steps:

- Go to **www.osforensics.com/tools/mount-disk-images.html** and download the utility that matches your current Windows version (32 or 64 bits).
- Install the software as you do with any Windows program.
- When the software successfully opens, click the **Mount new** button; a new box will then appear where you may pick the picture file and configure additional mount parameters (see *figure 6.4*). OSFMount will mount the image by default as read-only, and you can view it from Windows File Explorer like any other local disk.

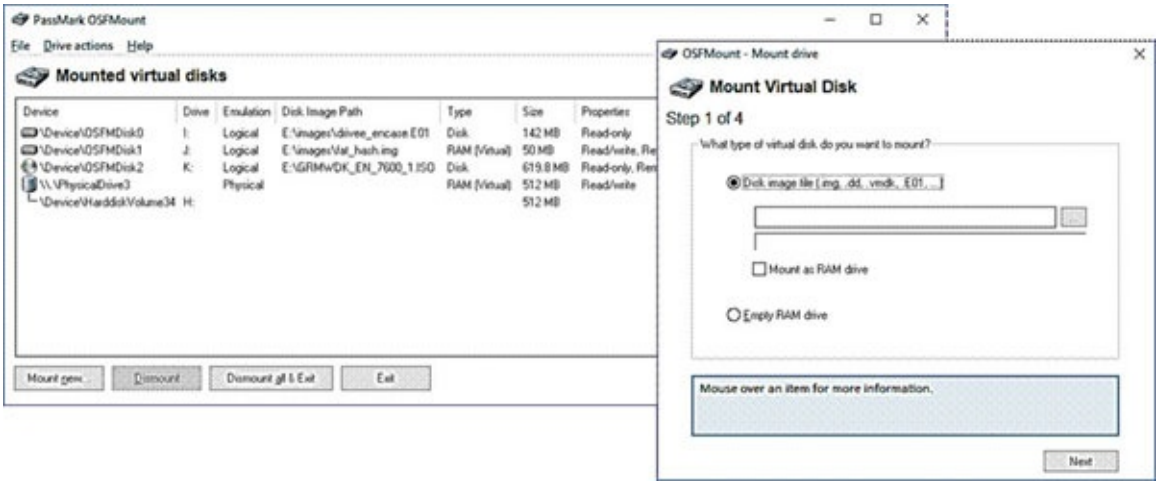


Figure 6.4: *OSFMount mount drives*

Autopsy

An autopsy is a **graphical user interface** (**GUI**) software that offers simple access to the command-line tools and the C library contained in The Sleuth Kit and other digital forensics tools. The tools included in the Sleuth Kit—and other digital forensics tools—will allow Autopsy to automate much of the forensics analysis tasks required in most investigations, such as recovering deleted files, analyzing Windows registries, and investigating e-mail messages, investigating unallocated disc space, and many more. Autopsy contains extra tools that enable examiners to be more effective throughout their analytical job. The key features of Autopsy are presented as a case by adding a forensic image to perform basic forensic analysis. An autopsy is a comprehensive forensic platform used by thousands of users throughout the world; it has active support from a volunteer community in addition to commercial assistance for premium customers. Autopsy features may be enhanced with customized modules, also called ingest modules, that can be built using Python (Jython) or Java computer language. Covering all Autopsy features in-depth will need more than a chapter; nonetheless, this section presents the process to start using this powerful tool properly to investigate your cases. Before we open the Autopsy wizard, we need to download and install it first on our forensic workstation.

- Go to **https://sleuthkit.org/autopsy/download.php**. Then download the version that matches the forensic workstation OS (Autopsy is supported on Windows, Linux, and OS X).
- Install Autopsy as you do with any Windows application (assuming you are installing it on Windows OS).
- Now, it is time to build our first case in Autopsy. Follow these steps:
- The first time you run Autopsy, the wizard displays (see *figure 6.5*); select the **New Case** option to begin the wizard. A case is a container that stores information about one inquiry. Each case must have at least one forensic image connected with it; however, you can add extra images to the case as needed. Always make sure that each case has information about one investigation only, and all captured images relating to this investigation should be associated with this case alone.

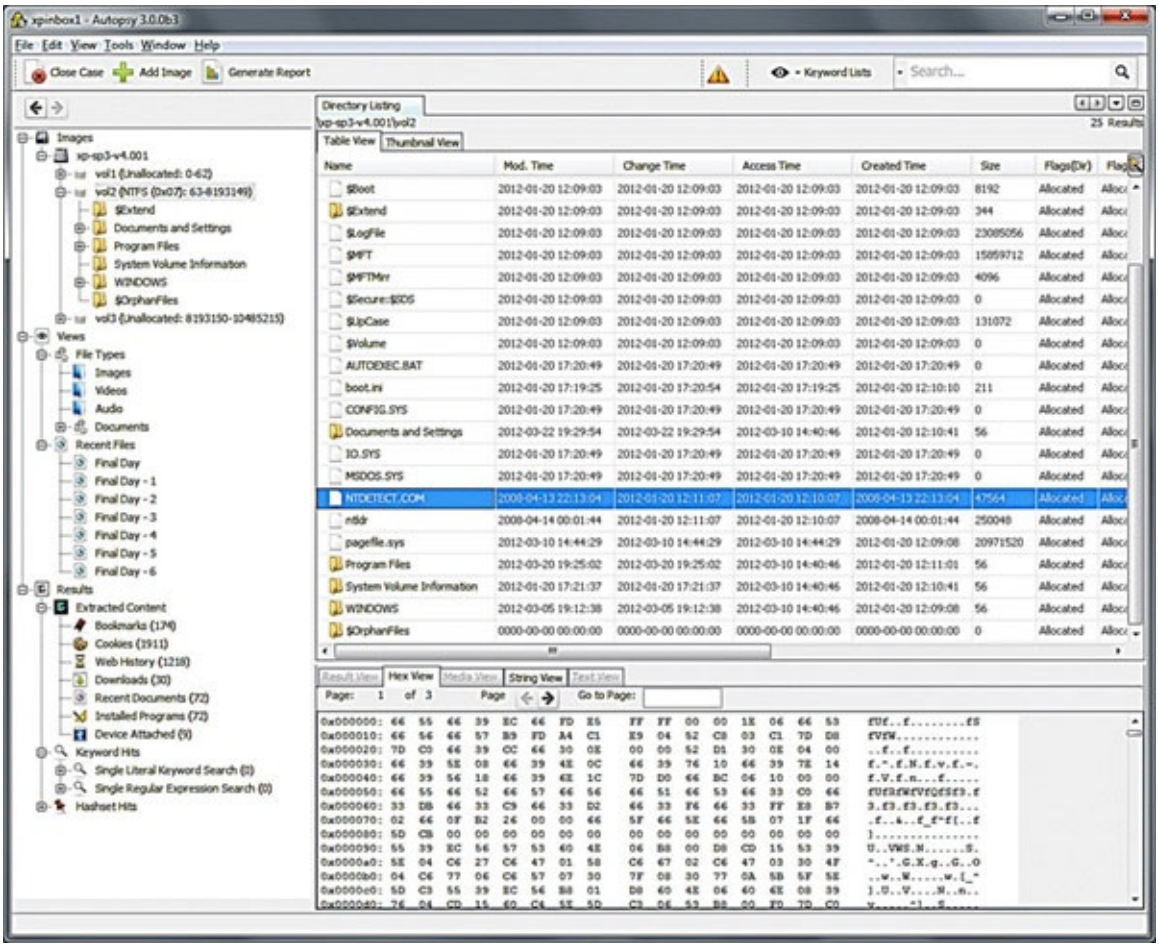


Figure 6.5: *Autopsy display*

- The following box, named “New Case Information,” opens; here, you need to input “Case Name” and the location (directory) where you wish to save case database files (see *figure 6.6*). After entering this info, click **Next** to continue.

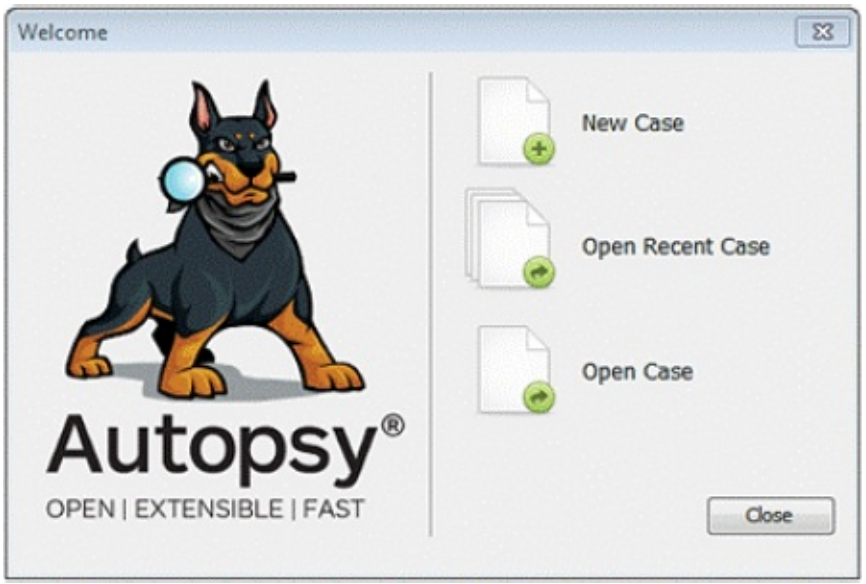


Figure 6.6: *Opening new autopsy case and case properties*

- The next window allows you to input more information regarding the case, like number and examiner details (name, phone, e-mail, and remarks). This is optional information; reject or supply the info and then hit the **Finish** button (see *figure 6.7*).

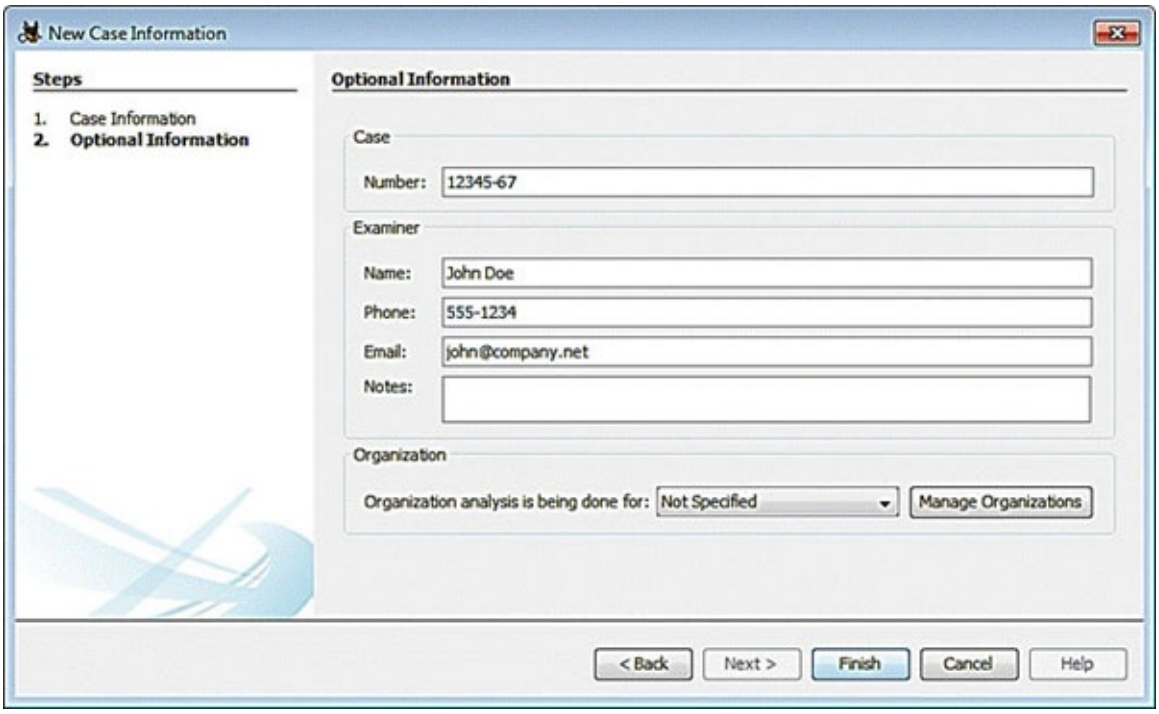


Figure 6.7: *Entering extra optional case information*

Autopsy establishes a case database and saves it in your preferred place. Then the wizard will display the **`Upload Data Source`** box, where you may add a collected forensic picture that you are going to investigate/analyze. Select **Disk Image or VM File** and then click **Next** to proceed (see *figure 6.8*). Autopsy supports the following data source types:

- Disk image (the most common option)—an obtained disc or hard drive image.
- Local disk—local hard drive and storage attached to the system like a USB flash drive.
- Logical files—like single files or directories.
- Unallocated space image file—Autopsy will search within the unallocated space of an image file for deleted files.

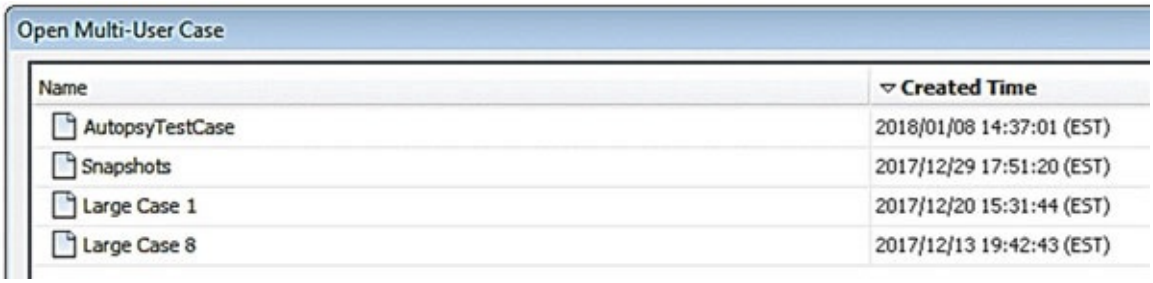


Figure 6.8: *Select "Disk Image" as a data source*

- In the following box, you need to upload the forensic picture file, click the **Browse** button, go to the directory on your computer/network drive where the image is saved, and pick it (see *figure 6.9*). You can modify the time zone of the image if the examiner—the one who acquires the image—resides in a different time zone. Leave other selections as they are and click **Next** to proceed.

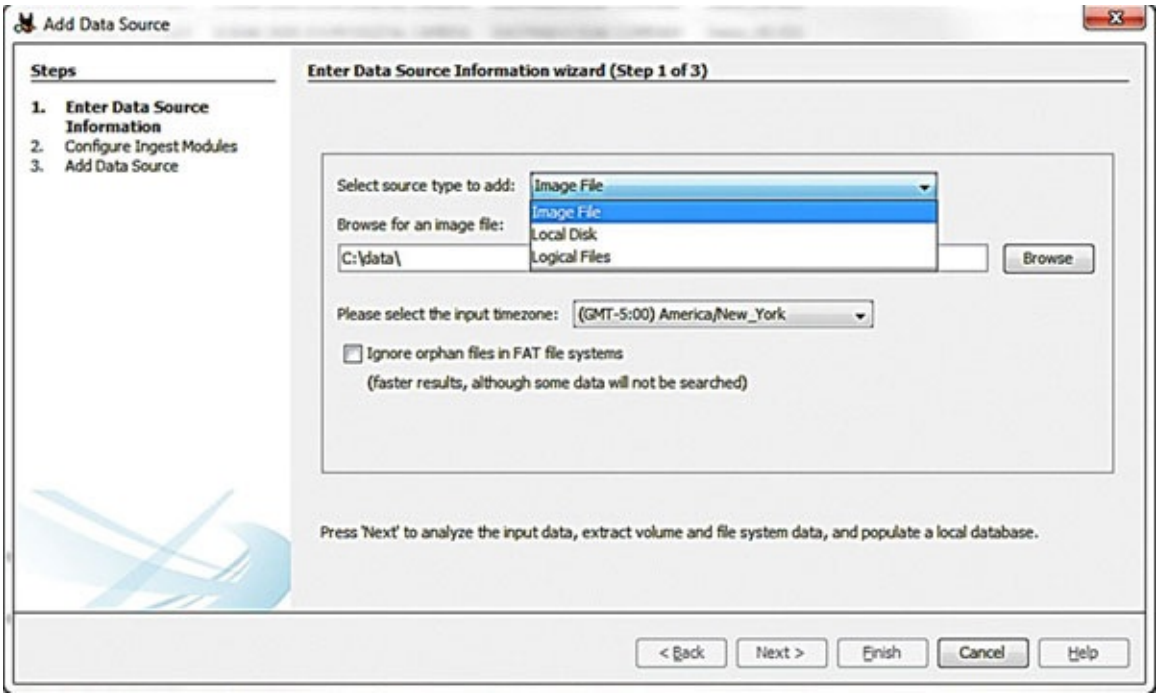


Figure 6.9: *Add a forensic image to a case*

- Now, Autopsy sends you to the **Configure Ingest Modules**, where you may specify the choices for forensic picture analysis. Each module is responsible for one forensic analysis job (see *figure 6.10*).

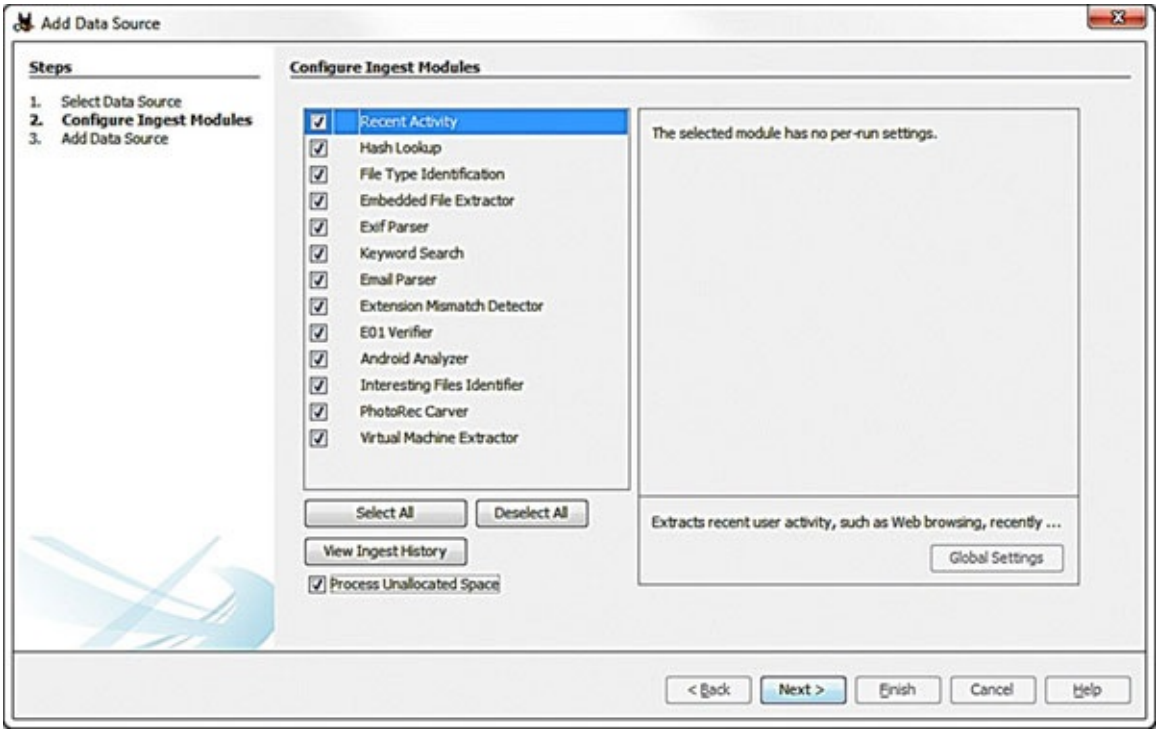


Figure 6.10: *Autopsy configure ingest modules option*

An autopsy is an automated forensic tool; consequently, when uploading a forensic image to it, Autopsy will automatically extract the most frequent information needed in digital forensic analysis from this image without needing to do this manually. Autopsy provides default ingest modules for analyzing a supplied data source (for example, forensic picture); it is up to you to select/deselect any module during the case creation phase. As we have already said, each ingests module is specialized in examining one sort of data in the specified data source. The size of the captured forensic picture under investigation and the number/type of ingest module you pick during

the initial case formation will impact how long it will take to finish analyzing image contents. Sometimes, evaluating image contents may take too long, especially if you are using a computer with low hardware specs.

- For now, let us return to “Configure Ingest Module.”
- Check the modules you wish to employ to analyze picture contents. For instance, if you are unclear about the needed modules, you may choose all, but this may take too long time if the forensic picture is enormous. Click **Next** to continue.
- An autopsy will now include the selected forensic picture into the case (see *figure 6.11*) and click **Finish** to close the wizard.

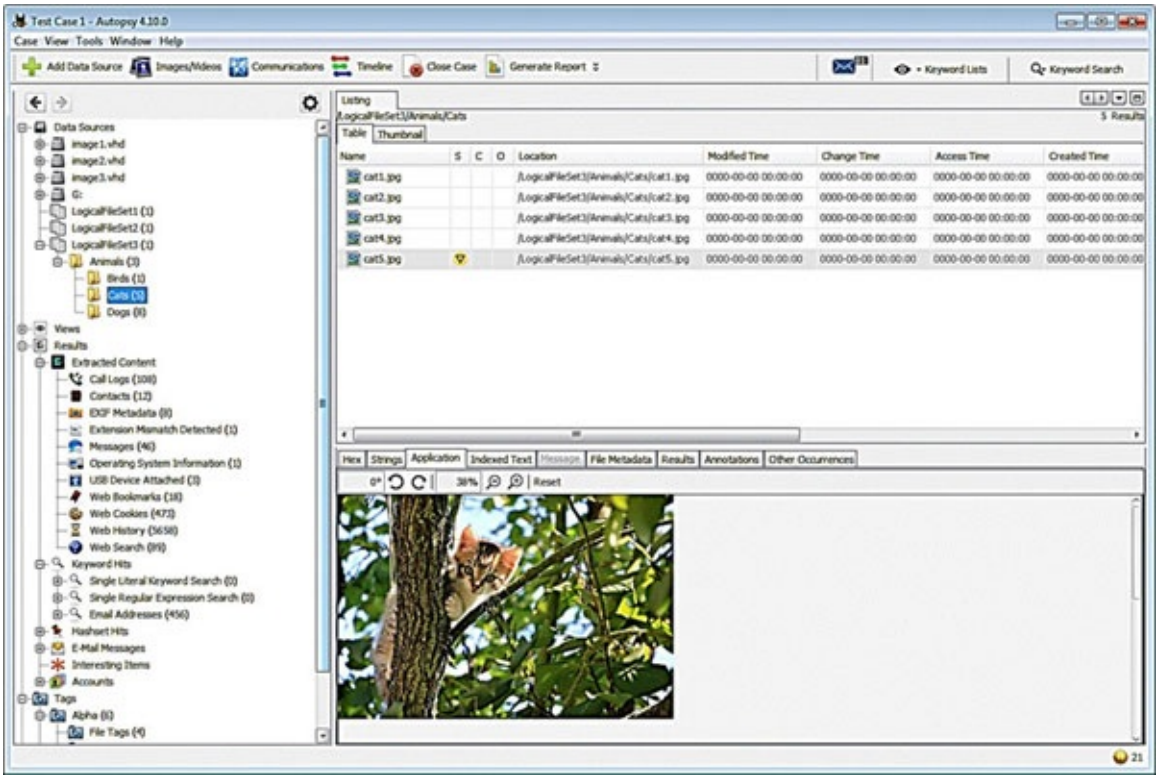


Figure 6.11: *Images inserted into the case*

- Now, Autopsy will begin analyzing the contents of the forensic image; we can see the status of image analysis in the lower right corner of the Autopsy main window.
- While working, you may watch the findings arrive gradually on the left side of the window (Data Explorer pane). If you wish to cancel the analysis process at any moment before completion, right-click the blue progress bar in the main window and then choose **Cancel Process**.
- An autopsy will display a popup box asking you to confirm. Click **Yes**. A second prompt will follow asking if you want to cancel simply the currently ongoing ingest module activity or the entire module; in our instance, we want to cancel all modules and restart the procedure again.
- The cancellation can be advantageous for a first-time user of Autopsy for several reasons; for example, a novice user of Autopsy may choose to pick all defaults to ingest modules while generating his/her case for the first time.
- As we already indicated, if we choose all ingest modules and the forensic picture file is enormous, Autopsy will take a substantial amount of time to finish analyzing image contents. Here comes the benefit of terminating the procedure and reinitiating the analysis, as we shall see in the following steps:

- After successfully terminating the analysis process of the current case, go to Autopsy Data Explorer, right-click over the picture you wish to process, and then choose **Run Ingest Modules** (see *figure 6.12*).

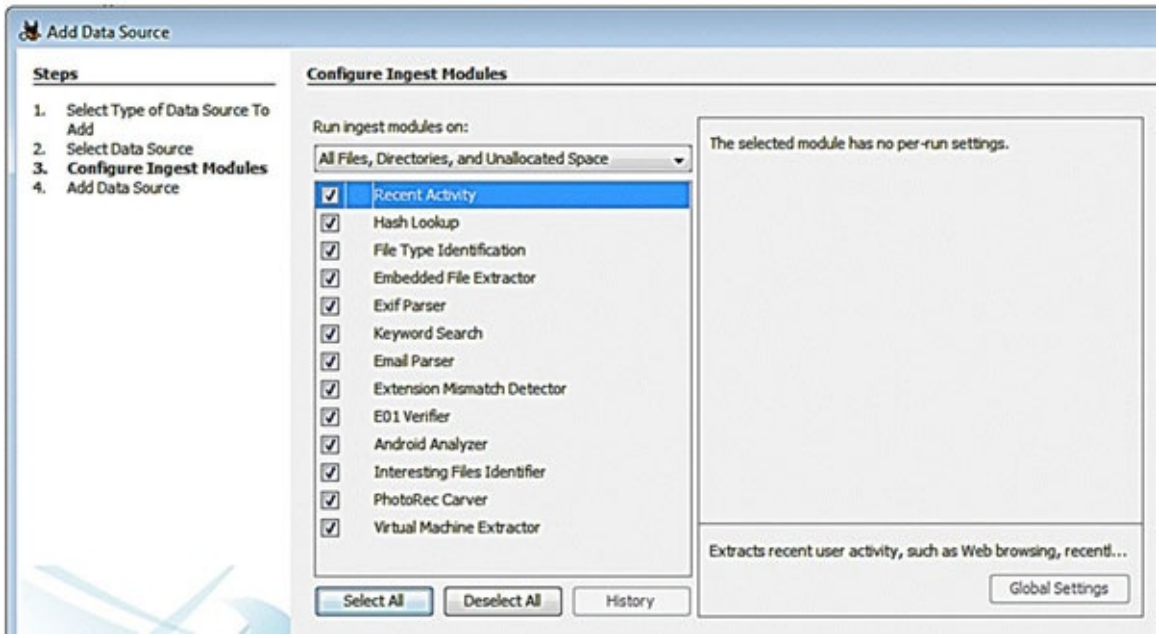


Figure 6.12: *Custom filters for an existing case*

- The "Configure Ingest Modules" window will open again; pick only the modules needed for your inquiry, then click the **Finish** button.
- Now, Autopsy will return to the analysis of selected picture contents again using only the specified ingest modules.
- An autopsy will begin analyzing picture contents immediately after executing "Ingest Modules" again; analysis findings will display in the directory tree window on the left side (also known as the **Data Explorer** pane) of the main window.

Now that we have a decent idea of how the core Autopsy functions operate for analyzing forensic images let us try exploring some additional capabilities that can considerably shorten the forensic analysis time. To recover deleted files from the specified data source (for example, forensic image), navigate to the Data Explorer window on the left. Select **Views** ➤ Deleted Files. To recover a file(s), right-click over it, select Export File(s), and pick a place where you wish to store it (see *figure 6.13*).

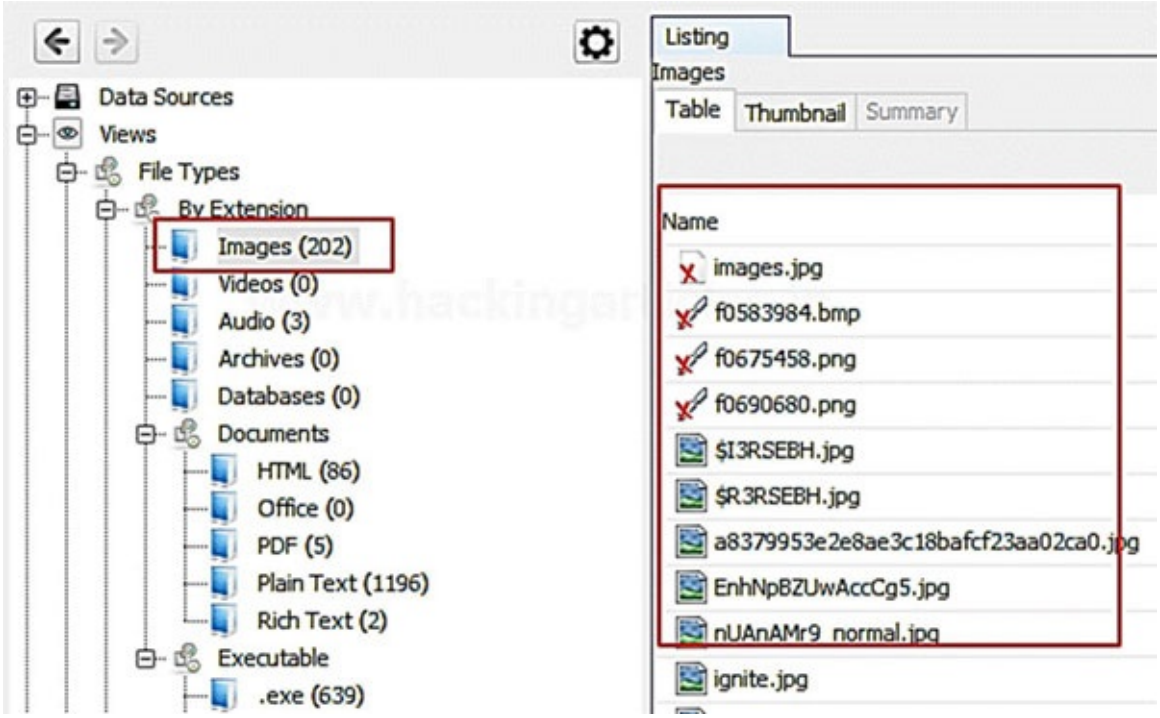


Figure 6.13: *Recover lost files*

The hash database ingests module allows an investigator to compare forensic image files' hash values to a precompiled hash value (Autopsy uses MD5 hashing) of known (good) or bad files. The recognized files are generally those belonging to the operating system itself and popular programs like MS Office files. By disregarding these files, Autopsy can cut the time needed for processing picture contents greatly. The bad file kinds include malware files and the like, and this type demands extra attention from the examiner. To use the hash database functionality in Autopsy, follow these steps (you must enable the hash to ingest module first before proceeding):

1. Go to the Autopsy Tools menu --> **Options**; the options window should display.
2. From the options box, pick Hash Sets and then click **Import Hash Set** (see *figure 6.14*).

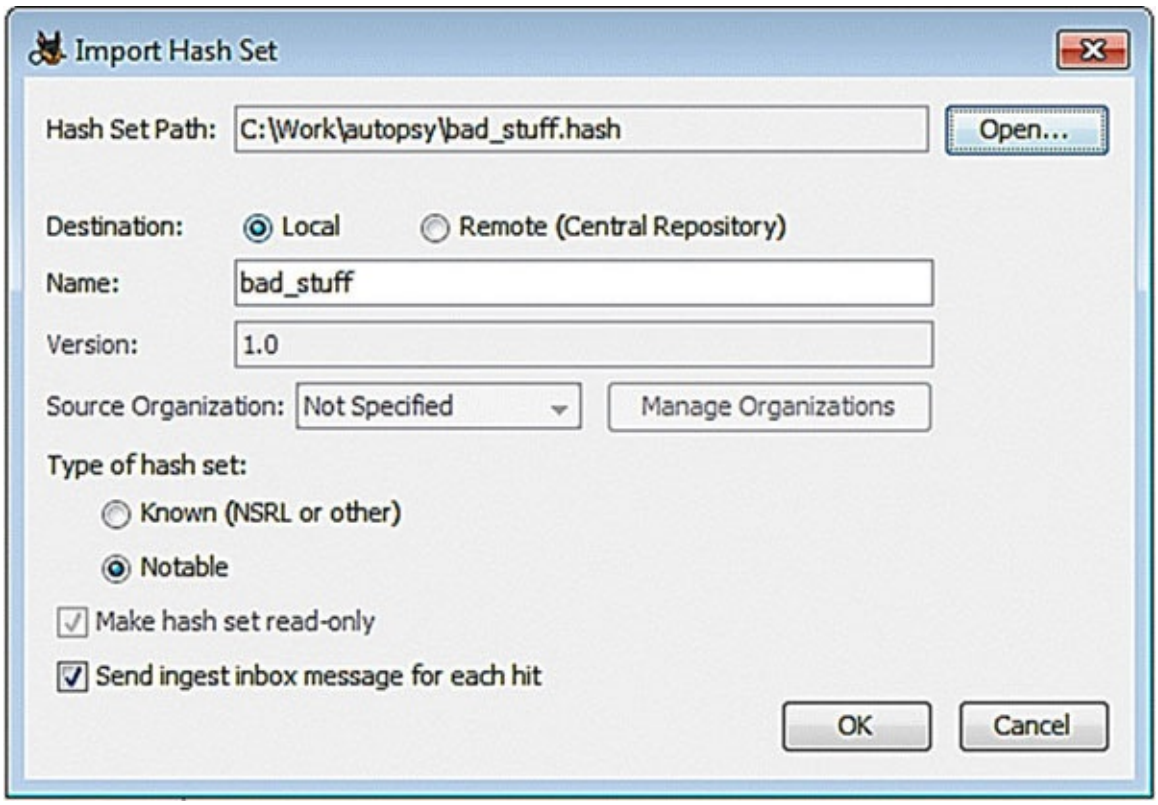


Figure 6.14: *Access hash data set window*

3. The **Import Hash Set** window opens; click the **Open** button to go to where the hash data set is located on your computer, pick it, and Autopsy will update the hash data set box automatically to indicate the name of the hash data set you have selected (see *figure 6.15*).

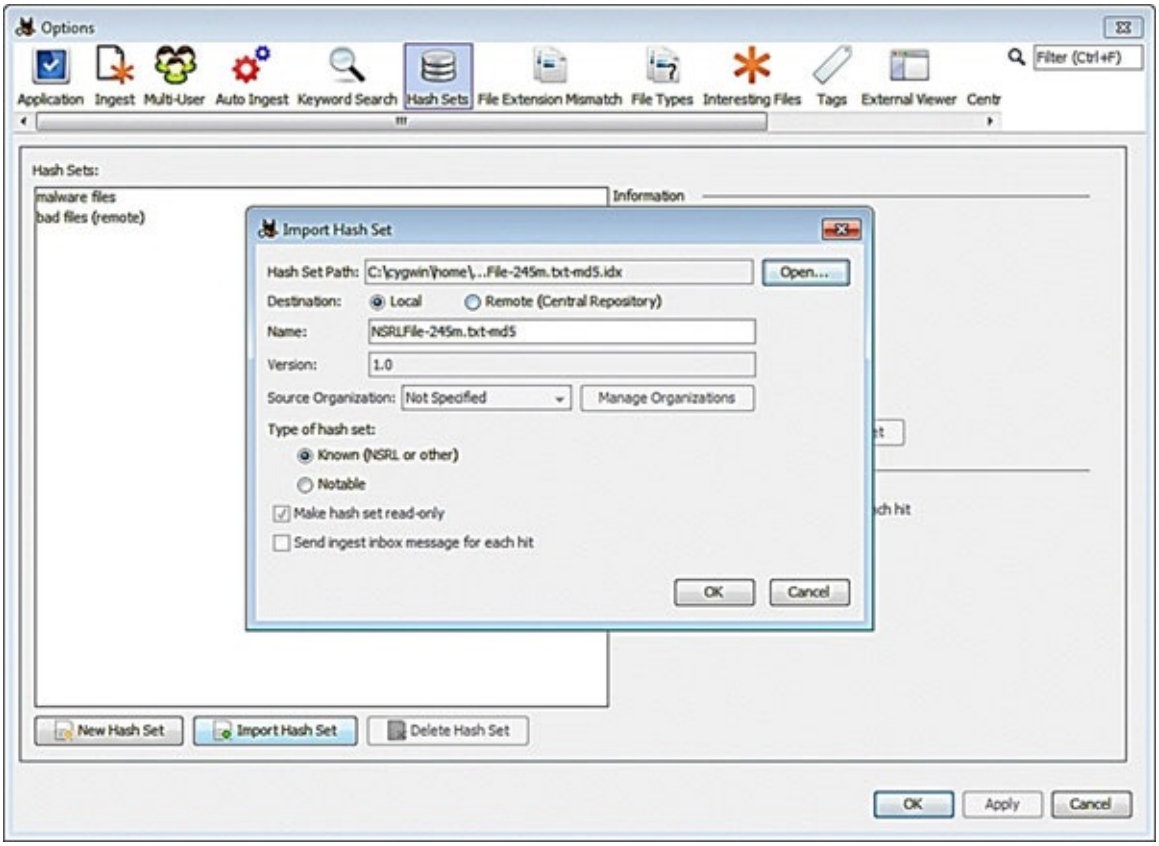


Figure 6.15: *Import hash set dialogue*

4. In *figure 6.16*, you may designate the imported hash set as the known file (which can be safely disregarded from Autopsy analysis) or noteworthy (hazardous or bad files that trigger suspicions like malware); for instance, we are utilizing an NSRL hash database; therefore, we will pick **Known**. Click the **OK** button after configuring all choices in the **Import Hash Set** box.
5. Now, the **Options** window opens again; if the imported hash set has an index-linked with it, the **Index Status** will indicate this (see *figure 6.16*); otherwise, the **Index** button will be active so that you may click it to allow Autopsy construct an index for you.

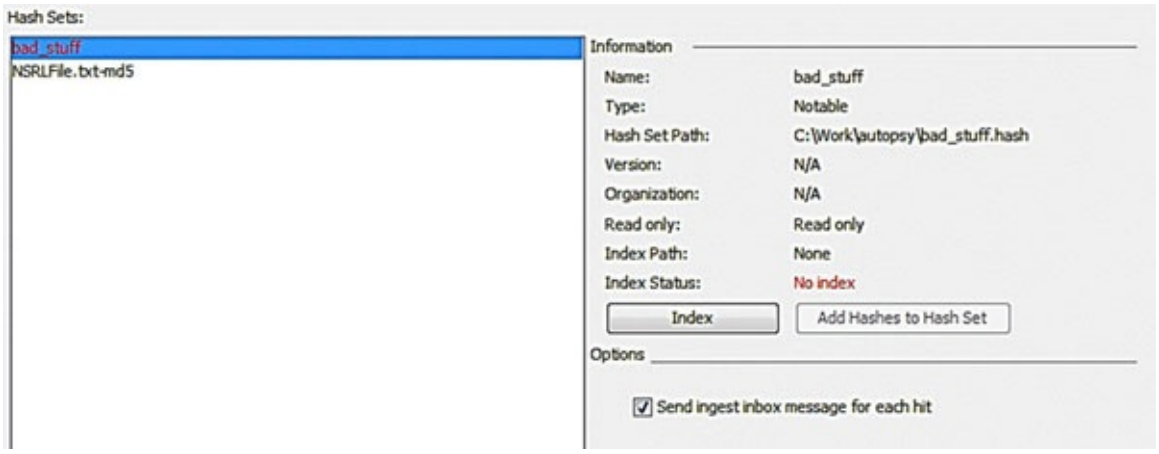


Figure 6.16: *Indexing hash data set*

6. Click the **OK** button to end the Options dialogue, and you are done!

Now you have successfully introduced a hash set to Autopsy, you need to run the ingest modules again to use the newly added HashSet. To achieve this, take the following steps:

- Cancel the picture analysis process.
- Right-click over the picture you want to process and then choose “Restart Ingest Modules”, as we did previously. Now, make sure to pick “Hash Lookup” and the hash database that you want to use throughout the analysis (on the right pane of the window) (see *figure 6.17*).

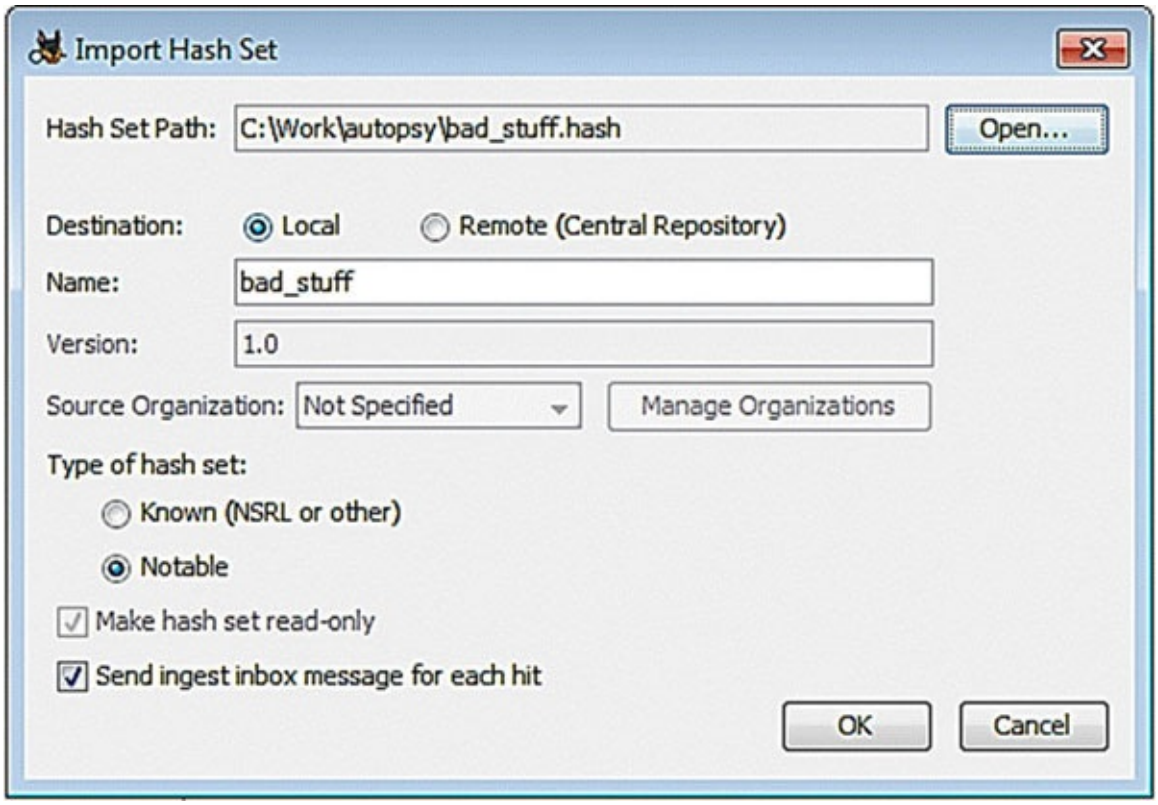


Figure 6.17: *Hash lookup module*

Analyzing RAM forensic image

We have explored how to acquire RAM using four tools: the acquired forensic picture may be examined using any of the main computer forensic suites, including EnCase, Belkasoft Evidence Center, and X-Ways Forensics. However, as we previously indicated at the beginning of this chapter, we will focus on using free and open-source technologies to complete the work. To examine RAM forensic images, we have two reliable and popular free tools: Memoryze and Redline from FireEye and Volatility from the volatility foundation. FireEye provides the forensics community with two popular free forensic tools to undertake digital forensics investigations:

Memoryze

This is a physical memory imaging and analysis command-line tool. In addition to taking RAM snapshots, it can undertake extensive analysis of live memory while the machine is still running. Memoryze can also analyze memory picture files, whether they were obtained using it or any other forensic program (DD-format) (DD-format). However, the analysis will yield more thorough findings when the forensic picture is captured using the Memoryze tool itself.

- Capture memory image snapshot running processes, opened files, and registry data.
- Filter (narrow) results according to numerous specified criteria (such as a given timeline of compromise events [when it started, which files were touched, and how long the compromise continued]) and/or filter known legitimate data based on precompiled MD5 hash values of well-known files.

Redline

This is a Windows tool for doing a memory study of harmful artifacts in Windows' physical memory. Before we can use Redline to record memory, we need to download it first.

- Go to **www.fireeye.com/services/freeware/redline.html**; you need to complete a short registration form, and the download link will send to the provided e-mail address.
- Install the software on your Windows system as you do with any other Windows application. Redline is compatible with practically all versions of Windows: Windows XP, Windows Vista, Windows 7, Windows 8 (32- and 64-bit), and Windows 10.
- Launch the software, and the main window will display (see *figure 6.18*). From the **Collect Data** tab, click **Create a Comprehensive Collector**.

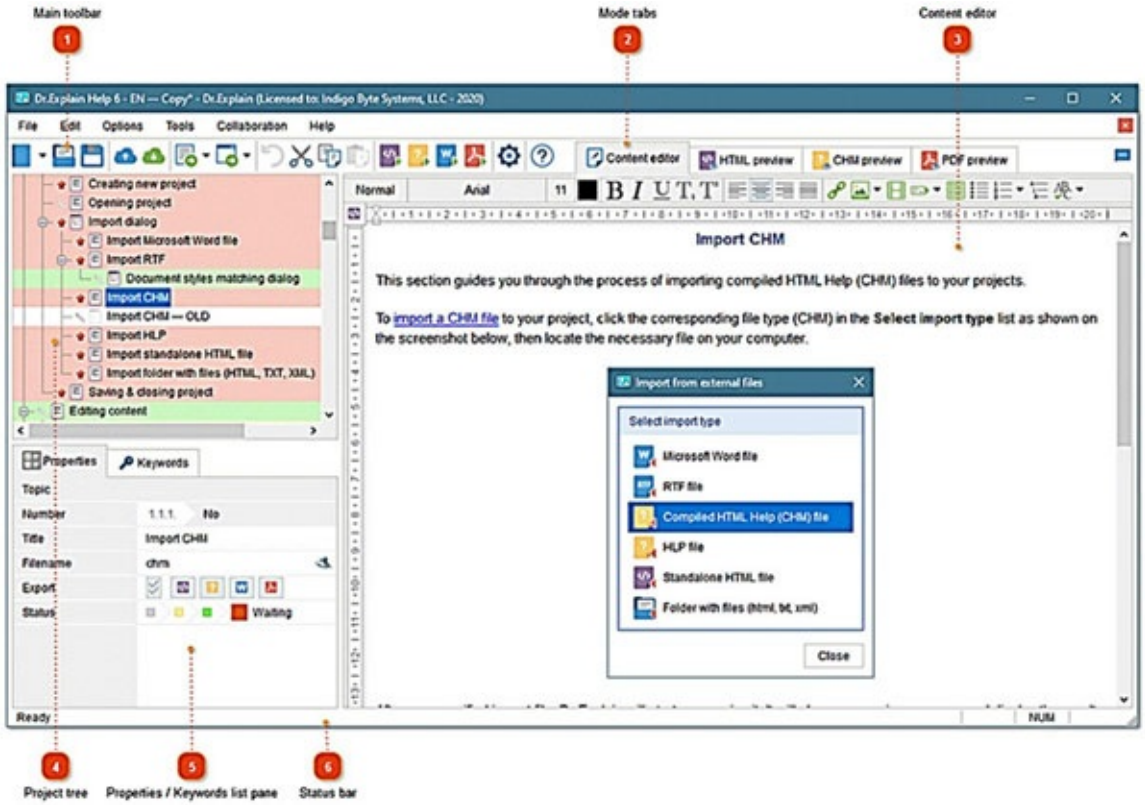


Figure 6.18: *Redline application main window*

Before we continue, let us offer a quick summary of various choices when using redline to capture data from a questionable Windows PC. Redline has three categories of collectors:

- **Standard Collector**: This kind gathers the least amount of data (primarily process and loaded driver information) (mainly process and loaded driver information).
- **Comprehensive Collector**: This kind captures most of the data that redline requires throughout its analysis process. This form of collection is extremely suggested, and this is what we are going to

use during this investigation.

- **IOC Search Collector**: This kind collects just the data that matches chosen **Indicators of Compromise** (**IOCs**).

In this example, we will pick “Comprehensive Collector”; click it, and the collector settings box appears. From this box, you can define what you intend to capture by selecting **Edit your script**; you should also tick the option **Acquire Memory Image** to acquire the target memory image (see *figure 6.19*).

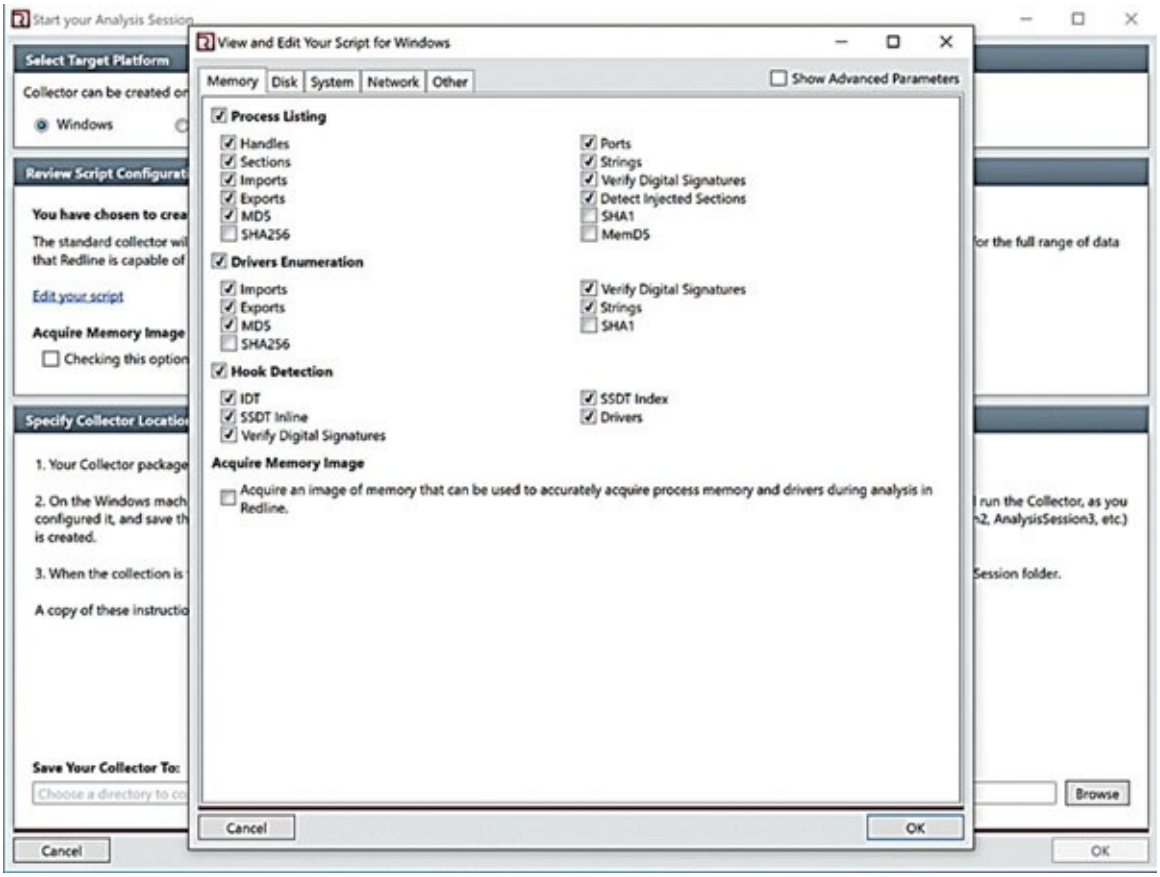


Figure 6.19: *Configure standard and comprehensive collector in redline*

Redline collector script (you may access it from “Edit Your Script”) has memory, disc, system, network, and other parameters preselected. You can adjust these settings inside any collection type. For instance, the comprehensive collector type (which we have chosen to use for this example) has most choices already checked by default, so you may continue comfortably with the default settings (see *figure 6.20*).

Figure 6.20: *View/edit collector settings in redline*

Now, from the collector configuration box, click **Browse** under **Put Your Collector To** and pick an empty directory where you wish to save this collector. For instance, we will put it on a USB flash drive so that we may use it later to capture a memory image from a suspicious system. Click **OK** to begin writing to the Redline collection (see *figure 6.21*).

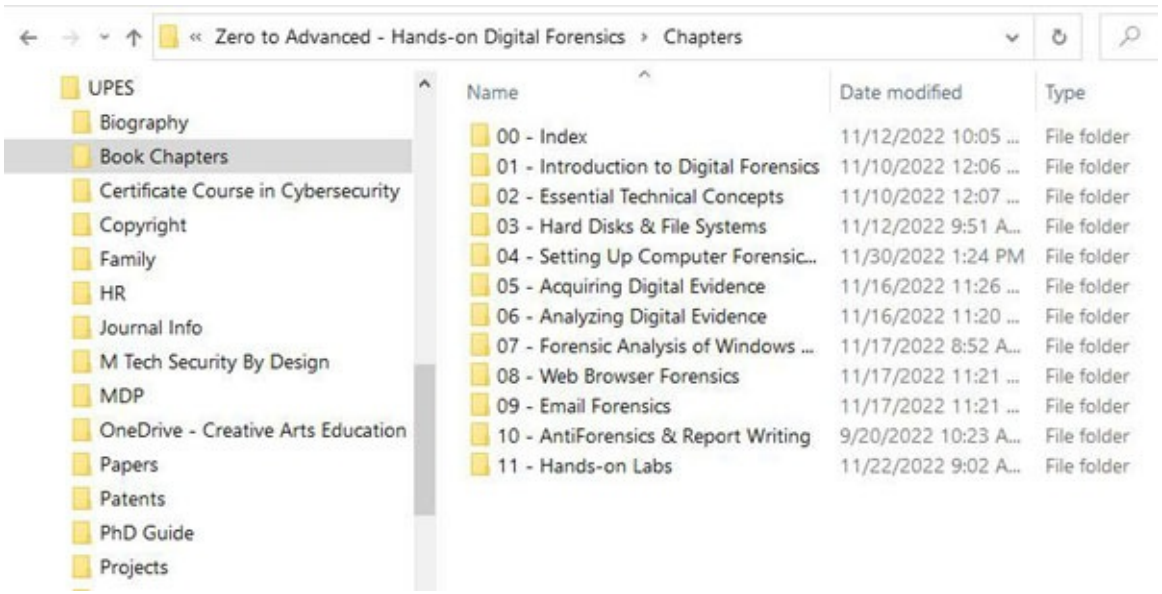


Figure 6.21: *Select the location to save collection files*

When redline finishes constructing the collector, a success popup will open up, offering you crucial information on how to use this collector to gather memory data from possibly infected systems.

Now that we have established our collector, let us see how we can use it to acquire memory images:

- Go to the directory where you have saved your freshly generated collector, and move the complete collector folder into a USB flash drive.
- Attach the USB thumb drive to the target machine.
- Execute the script named **`RunRedlineAudit.bat`** in the collection folder to execute the collector.

The collector should begin its collection operation by presenting a CMD window, and it will save obtained data to a folder named **`Sessions\AnalysisSession1`** in the same location. Every time you

execute the script, a new **`AnalysisSession`** folder (**`AnalysisSession2`**, **`AnalysisSession3`**, and so on) is generated. After the collector completes the collection, the CMD window should vanish without showing any notice. Now, go inside the **`AnalysisSession`** folder, and you will notice an **`Audits`** folder and an **`AnalysisSession1.mans`** file.

Volatility framework

Volatility is another notable tool for analyzing RAM forensic pictures; it is a research project that has arisen from published academic research articles in the field of advanced memory analysis and forensics. It is a free, open-source, and cross-platform tool written in Python; its development is presently financed by a nonprofit organization known as the Volatility Foundation. Volatility comes already loaded with many Linux security editions like Kali; nevertheless, this utility is also supported on Windows workstations (a standalone portable application) (a standalone portable application) from **https://www.volatilityfoundation.org/releases-vol3**. Volatility employs a series of plug-ins to conduct its tasks; these plug-ins are similar to Autopsy modules in that each one will do one analysis function in the given forensics image file. Let us now experiment using this program to evaluate a RAM picture using a Windows machine:

- Download Volatility (Windows version) and unpack the zipped file.
- Launch the command-line prompt; modify the command prompt to where Volatility lives (use the CD command switch) (use the CD command switch).
- Type the name of the Volatility file into the prompt, followed by the –h switch to display the available options.
- Before you can start analyzing any memory image, you need to know the operating system type that the specified image belongs to; this allows Volatility to know which commands to use during the analysis process. Imageinfo is the plug-in used by Volatility to perform this task (see *figure 6.22*). When a Memory dump is taken, it is extremely important to know the information about the operating system that was in use. Volatility will try to read the image and suggest the related profiles for the given memory dump. The image info plug-in displays the date and time of the sample that was collected, the number of CPUs present, and so on.

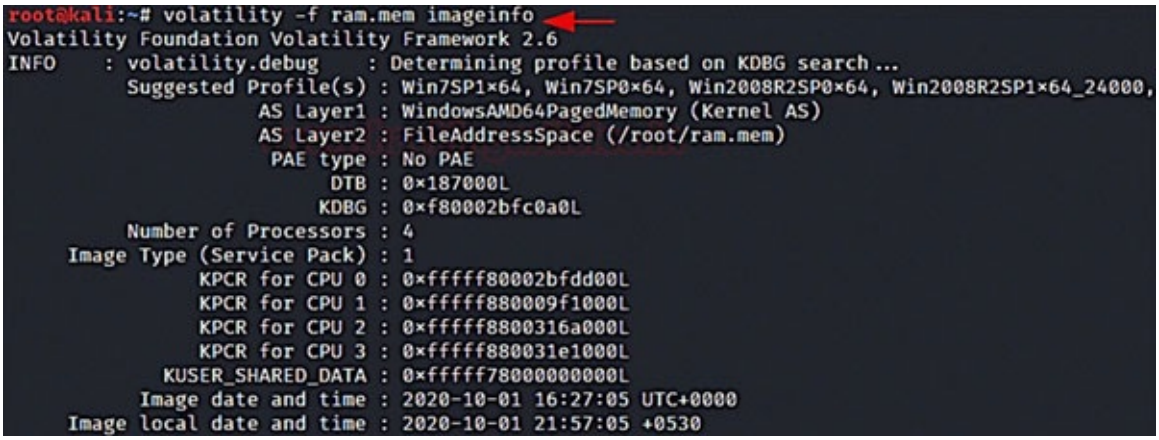


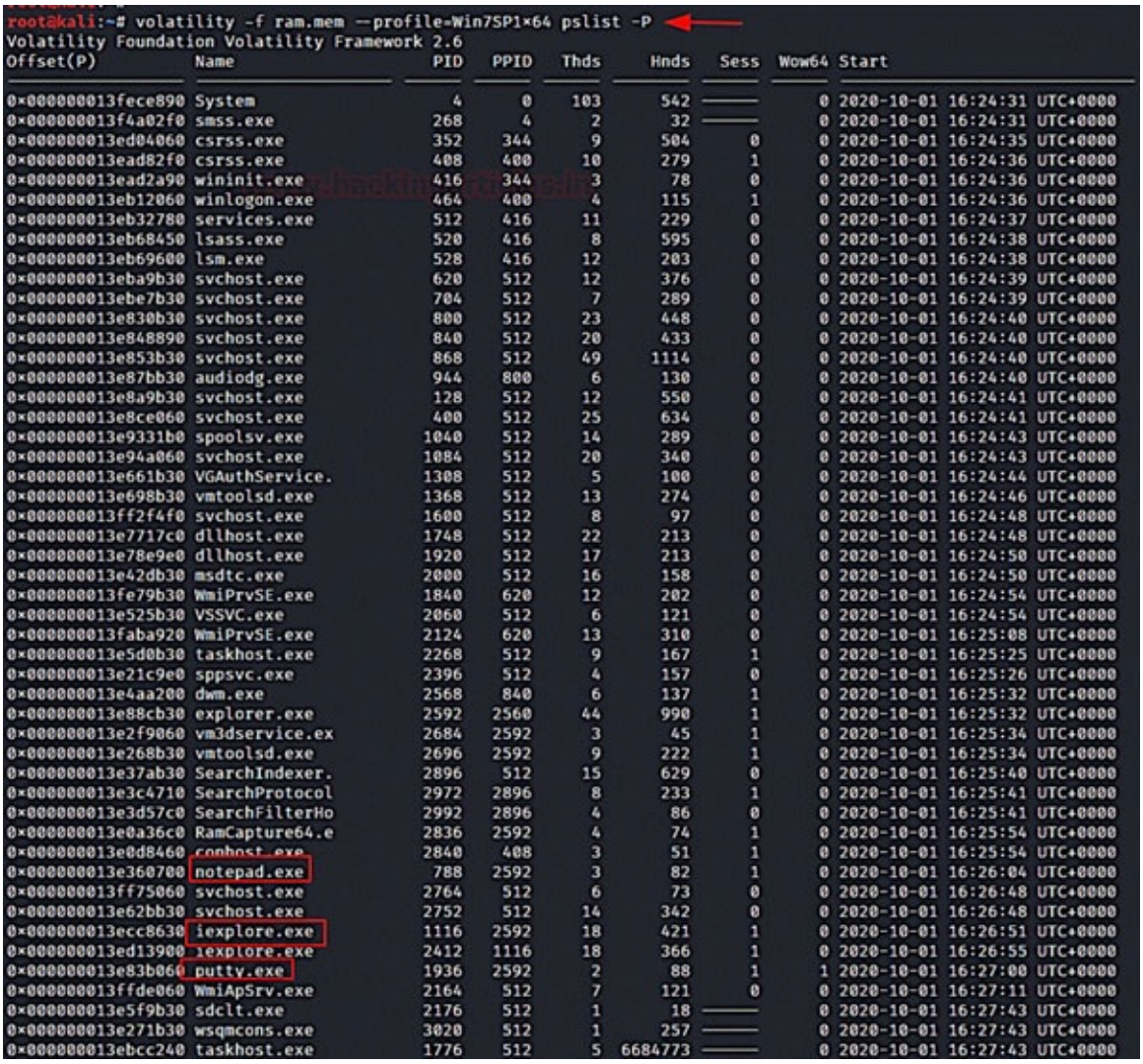
Figure 6.22: *Launching Volatility*

Kdbgscan plug-in finds and analyses the profiles based on the Kernel debugger data block. This provides the correct profile related to the raw image, as illustrated in *figure 6.23*.

Figure 6.23: *Kdbgscan plugin*

To identify the presence of any rogue processes and to view any high-level running processes, one can use **`volatility -f ram.mem --profile=Win7SP1x64 pslist –p`**.

On executing this command, the list of processes running is displayed, their respective process ID assigned to them, and the parent process ID is also displayed along. The details about the threads, sessions, and handles are also mentioned. The timestamp according to the start of the process is also displayed. This helps to identify whether an unknown process is running or was running at an unusual time, as illustrated in *figure 6.24*.



```
root@kali:~# volatility -f ram.mem --profile=Win7SP1x64 pslist -P
Volatility Foundation Volatility Framework 2.6
Offset(P)          Name                    PID   PPID   Thds     Hnds   Sess  Wow64 Start
------------------ -------------------- ------ ------ ------ -------- ------ ------ ------------------------------
0x000000013fece890 System                    4      0    103      542 ------      0 2020-10-01 16:24:31 UTC+0000
0x000000013f4a02f0 smss.exe                268      4      2       32 ------      0 2020-10-01 16:24:31 UTC+0000
0x000000013ed04060 csrss.exe               352    344      9      504      0      0 2020-10-01 16:24:35 UTC+0000
0x000000013ead82f0 csrss.exe               408    400     10      279      1      0 2020-10-01 16:24:36 UTC+0000
0x000000013ead2a90 wininit.exe             416    344      3       78      0      0 2020-10-01 16:24:36 UTC+0000
0x000000013eb12060 winlogon.exe            464    400      4      115      1      0 2020-10-01 16:24:36 UTC+0000
0x000000013eb32780 services.exe            512    416     11      229      0      0 2020-10-01 16:24:37 UTC+0000
0x000000013eb68450 lsass.exe               520    416      8      595      0      0 2020-10-01 16:24:38 UTC+0000
0x000000013eb69600 lsm.exe                 528    416     12      203      0      0 2020-10-01 16:24:38 UTC+0000
0x000000013eba9b30 svchost.exe             620    512     12      376      0      0 2020-10-01 16:24:39 UTC+0000
0x000000013ebe7b30 svchost.exe             704    512      7      289      0      0 2020-10-01 16:24:39 UTC+0000
0x000000013e830b30 svchost.exe             800    512     23      448      0      0 2020-10-01 16:24:40 UTC+0000
0x000000013e848890 svchost.exe             840    512     20      433      0      0 2020-10-01 16:24:40 UTC+0000
0x000000013e853b30 svchost.exe             868    512     49     1114      0      0 2020-10-01 16:24:40 UTC+0000
0x000000013e87bb30 audiodg.exe             944    800      6      130      0      0 2020-10-01 16:24:40 UTC+0000
0x000000013e8a9b30 svchost.exe             128    512     12      550      0      0 2020-10-01 16:24:41 UTC+0000
0x000000013e8ce060 svchost.exe             400    512     25      634      0      0 2020-10-01 16:24:41 UTC+0000
0x000000013e9331b0 spoolsv.exe            1040    512     14      289      0      0 2020-10-01 16:24:43 UTC+0000
0x000000013e94a060 svchost.exe            1084    512     20      340      0      0 2020-10-01 16:24:43 UTC+0000
0x000000013e661b30 VGAuthService.         1308    512      5      100      0      0 2020-10-01 16:24:44 UTC+0000
0x000000013e698b30 vmtoolsd.exe           1368    512     13      274      0      0 2020-10-01 16:24:46 UTC+0000
0x000000013ff2f4f0 svchost.exe            1600    512      8       97      0      0 2020-10-01 16:24:48 UTC+0000
0x000000013e7717c0 dllhost.exe            1748    512     22      213      0      0 2020-10-01 16:24:48 UTC+0000
0x000000013e78e9e0 dllhost.exe            1920    512     17      213      0      0 2020-10-01 16:24:50 UTC+0000
0x000000013e42db30 msdtc.exe              2000    512     16      158      0      0 2020-10-01 16:24:50 UTC+0000
0x000000013fe79b30 WmiPrvSE.exe           1840    620     12      202      0      0 2020-10-01 16:24:54 UTC+0000
0x000000013e525b30 VSSVC.exe              2060    512      6      121      0      0 2020-10-01 16:24:54 UTC+0000
0x000000013faba920 WmiPrvSE.exe           2124    620     13      310      0      0 2020-10-01 16:25:08 UTC+0000
0x000000013e5d0b30 taskhost.exe           2268    512      9      167      1      0 2020-10-01 16:25:25 UTC+0000
0x000000013e21c9e0 sppsvc.exe             2396    512      4      157      0      0 2020-10-01 16:25:26 UTC+0000
0x000000013e4aa200 dwm.exe                2568    840      6      137      1      0 2020-10-01 16:25:32 UTC+0000
0x000000013e88cb30 explorer.exe           2592   2560     44      990      1      0 2020-10-01 16:25:32 UTC+0000
0x000000013e2f9060 vm3dservice.ex         2684   2592      3       45      1      0 2020-10-01 16:25:34 UTC+0000
0x000000013e268b30 vmtoolsd.exe           2696   2592      9      222      1      0 2020-10-01 16:25:34 UTC+0000
0x000000013e37ab30 SearchIndexer.         2896    512     15      629      0      0 2020-10-01 16:25:40 UTC+0000
0x000000013e3c4710 SearchProtocol         2972   2896      8      233      1      0 2020-10-01 16:25:41 UTC+0000
0x000000013e3d57c0 SearchFilterHo         2992   2896      4       86      0      0 2020-10-01 16:25:41 UTC+0000
0x000000013e0a36c0 RamCapture64.e         2836   2592      4       74      1      0 2020-10-01 16:25:54 UTC+0000
0x000000013e0d8460 conhost.exe            2840    408      3       51      1      0 2020-10-01 16:25:54 UTC+0000
0x000000013e360700 notepad.exe             788   2592      3       82      1      0 2020-10-01 16:26:04 UTC+0000
0x000000013ff75060 svchost.exe            2764    512      6       73      0      0 2020-10-01 16:26:48 UTC+0000
0x000000013e62bb30 svchost.exe            2752    512     14      342      0      0 2020-10-01 16:26:48 UTC+0000
0x000000013ecc8630 iexplore.exe           1116   2592     18      421      1      0 2020-10-01 16:26:51 UTC+0000
0x000000013ed13900 iexplore.exe           2412   1116     18      366      1      0 2020-10-01 16:26:55 UTC+0000
0x000000013e83b060 putty.exe              1936   2592      2       88      1      1 2020-10-01 16:27:00 UTC+0000
0x000000013ffde060 WmiApSrv.exe           2164    512      7      121      0      0 2020-10-01 16:27:11 UTC+0000
0x000000013e5f9b30 sdclt.exe              2176    512      1       18 ------      0 2020-10-01 16:27:43 UTC+0000
0x000000013e271b30 wsqmcons.exe           3020    512      1      257 ------      0 2020-10-01 16:27:43 UTC+0000
0x000000013ebcc240 taskhost.exe           1776    512      5  6684773 ------      0 2020-10-01 16:27:43 UTC+0000
```

Figure 6.24: *PSList: the presence of any rogue processes*

Psscan plug-in can be used to give a detailed list of processes found in the memory dump. It cannot detect hidden or unlinked processes as **`volatility -f ram.mem --profile=Win7SP1x64 psscan`**, as presented in *figure 6.25*.

```
root@kali:~# volatility -f ram.mem --profile=Win7SP1x64 psscan
Volatility Foundation Volatility Framework 2.6
Offset(P)          Name                PID   PPID PDB                Time created
------------------ ---------------- ------ ------ ------------------ ------------------------------
0x000000013e0a36c0 RamCapture64.e     2836   2592 0x0000000071ec4000 2020-10-01 16:25:54 UTC+0000
0x000000013e0d8460 conhost.exe        2840    408 0x0000000071a49000 2020-10-01 16:25:54 UTC+0000
0x000000013e21c9e0 sppsvc.exe         2396    512 0x00000000893d4000 2020-10-01 16:25:26 UTC+0000
0x000000013e268b30 vmtoolsd.exe       2696   2592 0x000000007ffab000 2020-10-01 16:25:34 UTC+0000
0x000000013e271b30 wsqmcons.exe       3020    512 0x00000001297c4000 2020-10-01 16:27:43 UTC+0000
0x000000013e2f9060 vm3dservice.ex     2684   2592 0x0000000804a6000 2020-10-01 16:25:34 UTC+0000
0x000000013e360700 notepad.exe         788   2592 0x0000000072e8e000 2020-10-01 16:26:04 UTC+0000
0x000000013e37ab30 SearchIndexer.     2896    512 0x000000007d35d000 2020-10-01 16:25:40 UTC+0000
0x000000013e3c4710 SearchProtocol     2972   2896 0x0000000086f7b000 2020-10-01 16:25:41 UTC+0000
0x000000013e3d57c0 SearchFilterHo     2992   2896 0x000000007be61000 2020-10-01 16:25:41 UTC+0000
0x000000013e42db30 msdtc.exe          2000    512 0x000000008fd96000 2020-10-01 16:24:50 UTC+0000
0x000000013e4aa200 dwm.exe            2568    840 0x000000008a6bb000 2020-10-01 16:25:32 UTC+0000
0x000000013e525b30 VSSVC.exe          2060    512 0x000000008ccbf000 2020-10-01 16:24:54 UTC+0000
0x000000013e5d0b30 taskhost.exe       2268    512 0x000000008a364000 2020-10-01 16:25:25 UTC+0000
0x000000013e5f9b30 sdclt.exe          2176    512 0x00000001290b6000 2020-10-01 16:27:43 UTC+0000
0x000000013e62bb30 svchost.exe        2752    512 0x000000006b96e000 2020-10-01 16:26:48 UTC+0000
0x000000013e661b30 VGAuthService.     1308    512 0x0000000095d5f000 2020-10-01 16:24:44 UTC+0000
0x000000013e698b30 vmtoolsd.exe       1368    512 0x0000000094a93000 2020-10-01 16:24:46 UTC+0000
0x000000013e7717c0 dllhost.exe        1748    512 0x0000000092967000 2020-10-01 16:24:48 UTC+0000
0x000000013e78e9e0 dllhost.exe        1920    512 0x00000000916ef000 2020-10-01 16:24:50 UTC+0000
0x000000013e830b30 svchost.exe         800    512 0x000000009cc2c000 2020-10-01 16:24:40 UTC+0000
0x000000013e83b060 putty.exe          1936   2592 0x0000000061d5b000 2020-10-01 16:27:00 UTC+0000
0x000000013e848890 svchost.exe         840    512 0x000000009cd3f000 2020-10-01 16:24:40 UTC+0000
0x000000013e853b30 svchost.exe         868    512 0x000000009ca47000 2020-10-01 16:24:40 UTC+0000
0x000000013e87bb30 audiodg.exe         944    800 0x000000009c231000 2020-10-01 16:24:40 UTC+0000
0x000000013e88cb30 explorer.exe       2592   2560 0x00000000885d6000 2020-10-01 16:25:32 UTC+0000
0x000000013e8a9b30 svchost.exe         128    512 0x000000009b4d3000 2020-10-01 16:24:41 UTC+0000
0x000000013e8ce060 svchost.exe         400    512 0x000000009aed9000 2020-10-01 16:24:41 UTC+0000
```

Figure 6.25: *Volatility PSScan*

The Pstree plug-in represents the child-parent relationship and shows any unknown or abnormal processes. The child process is represented by indention and periods as **`volatility -f ram.mem --profile=Win7SP1x64 pstree`**, as presented in *figure 6.26*.

```
root@kali:~# volatility -f ram.mem --profile=Win7SP1x64 pstree
Volatility Foundation Volatility Framework 2.6

Name                                                  Pid   PPid   Thds   Hnds Time
-------------------------------------------------- ------ ------ ------ ------ ----
 0xfffffa80322d2a90:wininit.exe                        416    344      3     78 2020-10
. 0xfffffa8032332780:services.exe                      512    416     11    229 2020-10
.. 0xfffffa80324a9b30:svchost.exe                      128    512     12    550 2020-10
.. 0xfffffa80325331b0:spoolsv.exe                     1040    512     14    289 2020-10
.. 0xfffffa80323e7b30:svchost.exe                      704    512      7    289 2020-10
.. 0xfffffa803282db30:msdtc.exe                       2000    512     16    158 2020-10
.. 0xfffffa8032661b30:VGAuthService.                  1308    512      5    100 2020-10
.. 0xfffffa803254a060:svchost.exe                     1084    512     20    340 2020-10
.. 0xfffffa8030f2f4f0:svchost.exe                     1600    512      8     97 2020-10
.. 0xfffffa8032430b30:svchost.exe                      800    512     23    448 2020-10
... 0xfffffa803247bb30:audiodg.exe                     944    800      6    130 2020-10
.. 0xfffffa803278e9e0:dllhost.exe                     1920    512     17    213 2020-10
.. 0xfffffa8032b7ab30:SearchIndexer.                  2896    512     15    629 2020-10
... 0xfffffa8032bd57c0:SearchFilterHo                 2992   2896      4     86 2020-10
... 0xfffffa8032bc4710:SearchProtocol                 2972   2896      8    233 2020-10
.. 0xfffffa8030f75060:svchost.exe                     2764    512      6     73 2020-10
.. 0xfffffa8032448890:svchost.exe                      840    512     20    433 2020-10
... 0xfffffa80328aa200:dwm.exe                        2568    840      6    137 2020-10
.. 0xfffffa8032925b30:VSSVC.exe                       2060    512      6    121 2020-10
.. 0xfffffa803262bb30:svchost.exe                     2752    512     14    342 2020-10
.. 0xfffffa80327717c0:dllhost.exe                     1748    512     22    213 2020-10
.. 0xfffffa8032698b30:vmtoolsd.exe                    1368    512     13    274 2020-10
.. 0xfffffa80329d0b30:taskhost.exe                    2268    512      9    167 2020-10
.. 0xfffffa80323cc240:taskhost.exe                    1776    512      5 66 ... 3 2020-10
.. 0xfffffa8032453b30:svchost.exe                      868    512     49   1114 2020-10
.. 0xfffffa80323a9b30:svchost.exe                      620    512     12    376 2020-10
```

Figure 6.26: *Volatility PSTree*

DLLs stand for dynamic-link library automatically that is added to this list when a process calls Load Library, and they are not removed until. To display the DLLs for any particular process instead of all processes as **`volatility -f ram.mem --profile=Win7SP1x64 dlllist -p 1116,788`**, as presented

in *figure 6.27*.

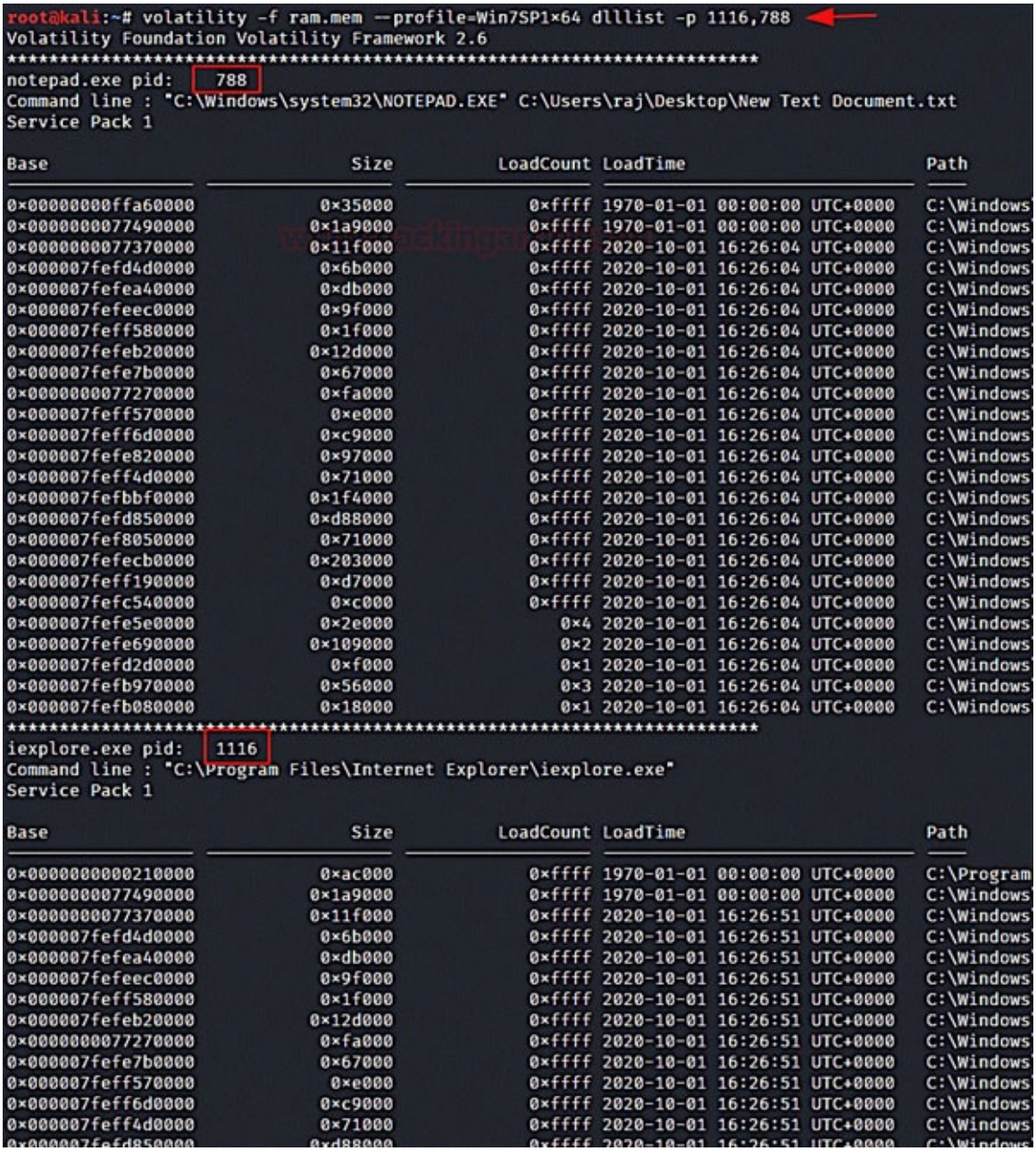


Figure 6.27: *Volatility PIDs*

DLL Dump plug-in is used to dump the DLLs from the memory space of the processes into another location to analyze them. To take a dump of the DLLs, use **`volatility -f ram.mem --profile=Win7SP1x64 dlldump --dump-dir /root/ramdump/`**, as illustrated in *figure 6.28*.

```
root@kali:~# volatility -f ram.mem --profile=Win7SP1x64 dlldump --dump-dir /root/ramdump/
Volatility Foundation Volatility Framework 2.6
Process(V)         Name                 Module Base        Module Name          Result
------------------ -------------------- ------------------ -------------------- ------
0xfffffa80318a02f0 smss.exe             0x0000000047850000 smss.exe             OK: module.268.13f4a
0xfffffa80318a02f0 smss.exe             0x0000000077490000 ntdll.dll            OK: module.268.13f4a
0xfffffa8032104060 csrss.exe            0x000000004a520000 csrss.exe            OK: module.352.13ed0
0xfffffa8032104060 csrss.exe            0x0000000077490000 ntdll.dll            OK: module.352.13ed0
0xfffffa8032104060 csrss.exe            0x000007fefd440000 basesrv.DLL          OK: module.352.13ed0
0xfffffa8032104060 csrss.exe            0x000007feff6d0000 USP10.dll            OK: module.352.13ed0
0xfffffa8032104060 csrss.exe            0x0000000077270000 USER32.dll           OK: module.352.13ed0
0xfffffa8032104060 csrss.exe            0x000007fefd460000 CSRSRV.dll           OK: module.352.13ed0
0xfffffa8032104060 csrss.exe            0x0000000077370000 kernel32.dll         OK: module.352.13ed0
0xfffffa8032104060 csrss.exe            0x000007fefeb20000 RPCRT4.dll           OK: module.352.13ed0
0xfffffa8032104060 csrss.exe            0x000007fefd2d0000 CRYPTBASE.dll        OK: module.352.13ed0
0xfffffa8032104060 csrss.exe            0x000007fefe7b0000 GDI32.dll            OK: module.352.13ed0
0xfffffa8032104060 csrss.exe            0x000007fefeec0000 msvcrt.dll           OK: module.352.13ed0
0xfffffa8032104060 csrss.exe            0x000007feff570000 LPK.dll              OK: module.352.13ed0
0xfffffa8032104060 csrss.exe            0x000007fefd4d0000 KERNELBASE.dll       OK: module.352.13ed0
0xfffffa8032104060 csrss.exe            0x000007fefd2e0000 sxs.dll              OK: module.352.13ed0
0xfffffa8032104060 csrss.exe            0x000007fefd3f0000 sxssrv.DLL           OK: module.352.13ed0
0xfffffa8032104060 csrss.exe            0x000007fefd400000 winsrv.DLL           OK: module.352.13ed0
0xfffffa80322d82f0 csrss.exe            0x000000004a520000 csrss.exe            OK: module.408.13ead
```

Figure 6.28: *Volatility DLLDump*

The handle plug-in is used to display the open handles that are present in a process. This plug-in applies to files, registry keys, events, desktops, threads, and all other types of objects. To view the handles present in the dump, use **`volatility -f ram.mem --profile=Win7SP1x64 handles`**, as presented in *figure 6.29*:

```
root@kali:~# volatility -f ram.mem --profile=Win7SP1x64 handles
Volatility Foundation Volatility Framework 2.6
Offset(V)          Pid    Handle     Access Type            Details
------------------ ------ ---------- ------ --------------- -------
0xfffffa8030ece890      4          0x4 0x1fffff Process         System(4)
0xfffff8a0000711f0      4          0x8 0x2001f Key             MACHINE\CONTROLSET
0xfffff8a000008060      4          0xc 0xf000f Directory       GLOBAL??
0xfffff8a00001aca0      4         0x10    0x0 Key
0xfffff8a00008ed30      4         0x14 0x2001f Key             MACHINE\CONTROLSET
0xfffff8a000072fa0      4         0x18 0xf003f Key             MACHINE\CONTROLSET
0xfffff8a00008ee20      4         0x1c 0x2001f Key             MACHINE\SETUP
0xfffffa8030efea40      4         0x20 0x1f0001 ALPC Port      PowerMonitorPort
0xfffffa8030f0a070      4         0x24 0x1f0001 ALPC Port      PowerPort
0xfffff8a000072ba0      4         0x28 0x20019 Key             MACHINE\DESCRIPTI
0xfffffa8030ff57e0      4         0x2c 0x1fffff Thread         TID 172 PID 4
0xfffff8a00008fa90      4         0x30 0xf003f Key             MACHINE\CONTROLSET
0xfffff8a00008be80      4         0x34 0xf003f Key             MACHINE\CONTROLSET
0xfffff8a000057fa0      4         0x38 0xf003f Key             MACHINE\CONTROLSET
```

Figure 6.29: *Volatility handles*

The Getsids plug-in is used to view the SIDs stands for Security Identifiers that are associated with a process. This plug-in can help in identifying processes that have maliciously escalated privileges and which processes belong to specific users. To get detail on a particular process id use **`volatility -f ram.mem --profile=Win7SP1x64 gets its -p 464`**, as presented in *figure 6.30*.

```
root@kali:~# volatility -f ram.mem --profile=Win7SP1x64 getsids -p 464
Volatility Foundation Volatility Framework 2.6
winlogon.exe (464): S-1-5-18 (Local System)
winlogon.exe (464): S-1-5-32-544 (Administrators)
winlogon.exe (464): S-1-1-0 (Everyone)
winlogon.exe (464): S-1-5-11 (Authenticated Users)
winlogon.exe (464): S-1-16-16384 (System Mandatory Level)
root@kali:~#
```

Figure 6.30: *Volatility GetSIDs*

The Netscan plug-in helps in finding network-related artifacts present in the memory dump. It makes use of pool tag scanning. This plug-in finds all the TCP endpoints, TCP listeners, UDP endpoints, and UDP listeners. It provides details about the local and remote IP and also about the local and remote port using **`volatility -f ram.mem --profile=Win7SP1x64 netscan`** (refer to *figure 6.31*).

```
root@kali:~# volatility -f ram.mem --profile=Win7SP1x64 netscan
Volatility Foundation Volatility Framework 2.6
Offset(P)          Proto    Local Address                  Foreign Address      State
0x13e0de9e0        UDPv4    127.0.0.1:65024                *:*
0x13e8dcce0        UDPv4    0.0.0.0:0                      *:*
0x13e8dcce0        UDPv6    :::0                           *:*
0x13e8e4ad0        UDPv4    0.0.0.0:5355                   *:*
0x13e9c2d60        UDPv4    0.0.0.0:4500                   *:*
0x13e9c2d60        UDPv6    :::4500                        *:*
0x13e9d9270        UDPv4    0.0.0.0:4500                   *:*
0x13e9d9930        UDPv4    0.0.0.0:500                    *:*
0x13e9de010        UDPv4    0.0.0.0:500                    *:*
0x13e9de010        UDPv6    :::500                         *:*
0x13e9de500        UDPv4    0.0.0.0:0                      *:*
0x13e9de500        UDPv6    :::0                           *:*
0x13e9deb10        UDPv4    0.0.0.0:0                      *:*
0x13eaed860        UDPv4    192.168.2.11:138               *:*
0x13eb35920        UDPv4    192.168.2.11:137               *:*
0x13e6fb790        TCPv4    0.0.0.0:49155                  0.0.0.0:0            LISTENING
0x13e6fbef0        TCPv4    0.0.0.0:445                    0.0.0.0:0            LISTENING
0x13e6fbef0        TCPv6    :::445                         :::0                 LISTENING
0x13e6feef0        TCPv4    0.0.0.0:49155                  0.0.0.0:0            LISTENING
0x13e6feef0        TCPv6    :::49155                       :::0                 LISTENING
0x13e70f670        TCPv4    0.0.0.0:3389                   0.0.0.0:0            LISTENING
0x13e7728f0        TCPv4    0.0.0.0:49156                  0.0.0.0:0            LISTENING
0x13e7c3a60        TCPv4    0.0.0.0:49156                  0.0.0.0:0            LISTENING
0x13e7c3a60        TCPv6    :::49156                       :::0                 LISTENING
0x13e7e8320        TCPv4    0.0.0.0:3389                   0.0.0.0:0            LISTENING
0x13e7e8320        TCPv6    :::3389                        :::0                 LISTENING
0x13e805430        TCPv4    0.0.0.0:49152                  0.0.0.0:0            LISTENING
0x13e805430        TCPv6    :::49152                       :::0                 LISTENING
0x13e805ef0        TCPv4    0.0.0.0:49152                  0.0.0.0:0            LISTENING
0x13e844880        TCPv4    0.0.0.0:49153                  0.0.0.0:0            LISTENING
```

Figure 6.31: *Volatility Netscan*

Hivelist plug-in can be used to locate the virtual addresses present in the registry hives in memory and their entire paths to hive on the disk as **`volatility -f ram.mem --profile=Win7SP1x64 hivelist`**, as shown in *figure 6.32*.

```
root@kali:~# volatility -f ram.mem --profile=Win7SP1x64 hivelist
Volatility Foundation Volatility Framework 2.6
Virtual            Physical           Name
------------------ ------------------ ----
0xfffff8a00000f010 0x00000000a97f2010 [no name]
0xfffff8a000024010 0x00000000a987d010 \REGISTRY\MACHINE\SYSTEM
0xfffff8a000057010 0x00000000a95b0010 \REGISTRY\MACHINE\HARDWARE
0xfffff8a00058a010 0x00000000a8270010 \SystemRoot\System32\Config\SECURITY
0xfffff8a00058c010 0x00000000a83f2010 \SystemRoot\System32\Config\SOFTWARE
0xfffff8a00058f010 0x000000009d700010 \SystemRoot\System32\Config\DEFAULT
0xfffff8a0005ff010 0x00000000a8182010 \SystemRoot\System32\Config\SAM
0xfffff8a000e4d010 0x000000009d4e5010 \??\C:\Windows\ServiceProfiles\NetworkService\N
0xfffff8a000eef010 0x000000009d536010 \??\C:\Windows\ServiceProfiles\LocalService\NTU
0xfffff8a0015d7010 0x000000008a545010 \??\C:\Users\raj\ntuser.dat
0xfffff8a0015e5010 0x000000008aa5c010 \??\C:\Users\raj\AppData\Local\Microsoft\Window
0xfffff8a0021c8010 0x00000000610c4010 \??\C:\System Volume Information\Syscache.hve
0xfffff8a00307c010 0x00000000a58f7010 \Device\HarddiskVolume1\Boot\BCD
```

Figure 6.32: *Volatility hivelist*

The Timeliner plug-in usually creates a timeline from the various artifacts found in the memory dump. To locate the artifacts according to the timeline, use **`volatility -f ram.mem --profile=Win7SP1x64 timeliner`**, as presented in *figure 6.33*.

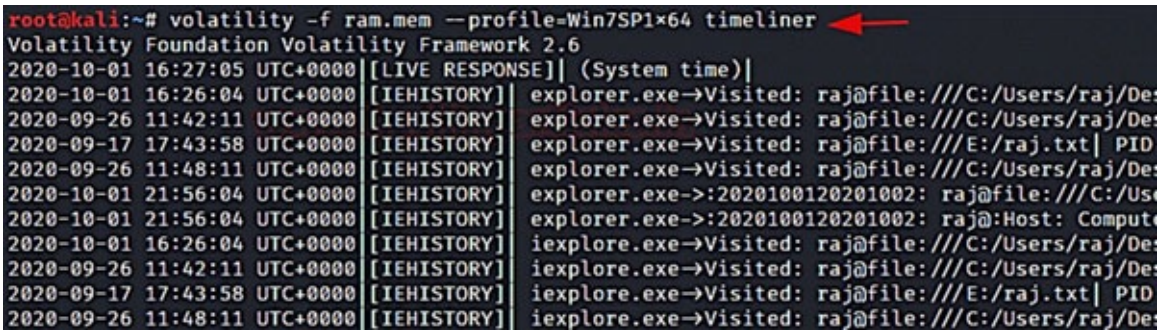


Figure 6.33: *Volatility Timeliner*

Hashdump plug-in can be used to extract and decrypt cached domain credentials stored in the registry, which can be availed from the memory dump. The hashes that are availed from the memory dump can be cracked using John the Ripper, Hashcat, and so on. To gather the hashdump, use **`volatility -f ram.mem --profile=Win7SP1x64 hashdump`**, as presented in *figure 6.34*.

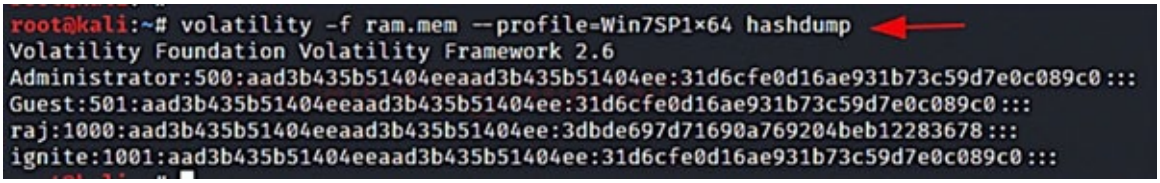


Figure 6.34: *Volatility Hashdump*

The Lsadump plug-in is used to dump LSA secrets from the registry in the memory dump. This plug-in gives out information like the default password, the RDP public key, and so on. To perform a lsadump, use **`volatility -f ram.mem --profile=Win7SP1x64 lsadump`**, as illustrated in *figure 6.35*.

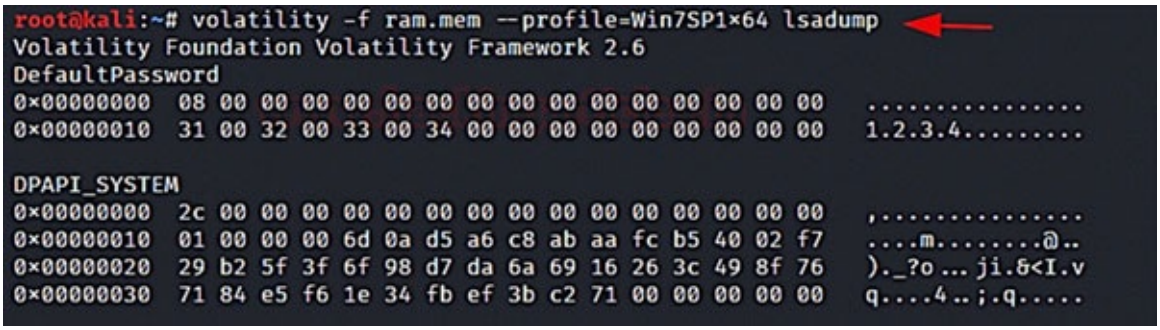


Figure 6.35: *Volatility LSADump*

The Modscan plug-in is used to locate kernel memory and its related objects. It can pick up all the previously unloaded drivers and also those drivers that have been hidden or have been unlinked by rootkits in the system using **`volatility -f ram.mem --profile=Win7SP1x64 modscan`**", as presented in *figure 6.36*.

```
root@kali:~# volatility -f ram.mem --profile=Win7SP1x64 modscan
Volatility Foundation Volatility Framework 2.6
Offset(P)          Name                    Base                  Size File
------------------ ---------------------- ------------------ ------ ----
0x0000000002fa45e1                        0x894c304b8b48ffad 0x8435e800
0x000000005bdeb5e1                        0x894c304b8b48ffad 0x8435e800
0x000000013e230c00 spsys.sys              0xfffff88005a00000    0x71000 \SystemRoot
0x000000013e2be010 RamCaptur...er64.SYS   0xfffff88005a71000     0x7000 \??\C:\User
0x000000013e611350 secdrv.SYS             0xfffff88005927000     0xb000 \SystemRoot
0x000000013e6171b0 srvnet.sys             0xfffff88005932000    0x31000 \SystemRoot
0x000000013e629520 rdpdr.sys              0xfffff88005b7d000     0x2e000 \SystemRoot
0x000000013e634480 srv2.sys               0xfffff88005975000     0x6b000 \SystemRoot
0x000000013e6385a0 tdtcp.sys              0xfffff88005bab000     0xb000 \SystemRoot
0x000000013e70e770 vmhgfs.sys             0xfffff88005b51000     0x2c000 \SystemRoot
0x000000013e7d33d0 tssecsrv.sys           0xfffff88005bb6000     0xf000 \SystemRoot
0x000000013e8c5510 rspndr.sys             0xfffff8800545d000    0x18000 \SystemRoot
0x000000013e8c71b0 lltdio.sys             0xfffff88001823000    0x15000 \SystemRoot
0x000000013e91e770 HTTP.sys               0xfffff88005475000     0xc9000 \SystemRoot
0x000000013e975e10 bowser.sys             0xfffff8800553e000     0x1e000 \SystemRoot
0x000000013e9783c0 mpsdrv.sys             0xfffff8800555c000    0x18000 \SystemRoot
0x000000013e97b510 mrxsmb.sys             0xfffff88005574000     0x2d000 \SystemRoot
0x000000013e981300 mrxsmb10.sys           0xfffff880055a1000     0x4d000 \SystemRoot
0x000000013e9a88e0 mrxsmb20.sys           0xfffff88005400000    0x24000 \SystemRoot
0x000000013e9afcf0 srv.sys                0xfffff88005ab8000    0x99000 \SystemRoot
0x000000013e9c24c0 vmmemctl.sys           0xfffff88005424000     0xa000 \SystemRoot
```

Figure 6.36: *Volatility ModScan*

The Filescan plug-in is used to find FILE_OBJECTs present in the physical memory by using pool tag scanning. It can find open files even if there is a hidden rootkit present in the files using **`volatility -f ram.mem --profile=Win7SP1x64 filescan`**, as presented in *figure 6.37*.

```
root@kali:~# volatility -f ram.mem --profile=Win7SP1x64 filescan
Volatility Foundation Volatility Framework 2.6
Offset(P)          #Ptr   #Hnd Access Name
------------------ ------ ------ ------ ----
0x000000013e000910      1      1 RW-rw- \Device\HarddiskVolume1\Users\raj\AppData\Local\Microsof
0x000000013e00c4a0      2      1 ------ \Device\NamedPipe\MsFteWds
0x000000013e00c740      2      0 R--r-d \Device\HarddiskVolume1\Windows\System32\rasdlg.dll
0x000000013e00c9f0      1      0 RW-rwd \Device\HarddiskVolume1\$PrepareToShrinkFileSize
0x000000013e01baf0     12      0 R--r-d \Device\HarddiskVolume1\Windows\System32\wlanutil.dll
0x000000013e01d6e0      1      1 R--rw- \Device\HarddiskVolume1\Windows\winsxs\amd64_microsoft.
0x000000013e020560     14      0 R--r-- \Device\HarddiskVolume1\wkssvc
0x000000013e020920      4      0 R--r-d \Device\HarddiskVolume1\Windows\System32\WWanAPI.dll
0x000000013e021a70     18      1 RW-r-- \Device\HarddiskVolume1\Windows\System32\winevt\Logs\Mi
0x000000013e021dd0     12      0 R--r-d \Device\HarddiskVolume1\Windows\System32\wwapi.dll
0x000000013e021f20      5      0 R--r-d \Device\HarddiskVolume1\Windows\System32\bthprops.cpl
0x000000013e022070      4      0 R--r-d \Device\HarddiskVolume1\Windows\System32\QAGENT.DLL
0x000000013e023070     17      1 R--r-d \Device\HarddiskVolume1\Windows\System32\en-US\bthprops.
0x000000013e023f20     11      0 R--r-d \Device\HarddiskVolume1\Windows\System32\FXSRESM.dll
0x000000013e025900      1      1 R--rw- \Device\HarddiskVolume1\Windows\winsxs\amd64_microsoft.
0x000000013e025c70      7      0 R--r-d \Device\HarddiskVolume1\Windows\System32\FXSST.dll
0x000000013e025dc0      7      0 R--r-d \Device\HarddiskVolume1\Windows\System32\FXSAPI.dll
0x000000013e027070     16      0 R--r-d \Device\HarddiskVolume1\Windows\System32\en-US\gameux.dl
0x000000013e02add0      1      1 R--rw- \Device\HarddiskVolume1\Windows\winsxs\amd64_microsoft.
0x000000013e02af20      1      1 R--rw- \Device\HarddiskVolume1\Windows\winsxs\amd64_microsoft.
0x000000013e02bc20      1      1 -W-rw- \Device\HarddiskVolume1\Users\raj\AppData\Local\Temp\FXS
0x000000013e0344a0      1      1 R--r-d \Device\HarddiskVolume1\Windows\System32\en-US\gameux.dl
0x000000013e035e20      4      0 RW-rwd \Device\HarddiskVolume1\$Directory
0x000000013e037430      1      1 RW-rw- \Device\HarddiskVolume1\Users\raj\AppData\Local\Microsof
```

Figure 6.37: *Volatility Filescan*

The Svcscan plug-in is used to see if the services are registered on your memory image; use the svcscan command. The output shows the process ID of each service, the service name, service name, display name, service type, and service state, and also shows the binary path for the registered service—which will be a **`.exe`** for user-mode services and a driver name for services that run from kernel mode using **`volatility -f ram.mem --profile=Win7SP1x64 svcscan`** in *figure 6.38*.

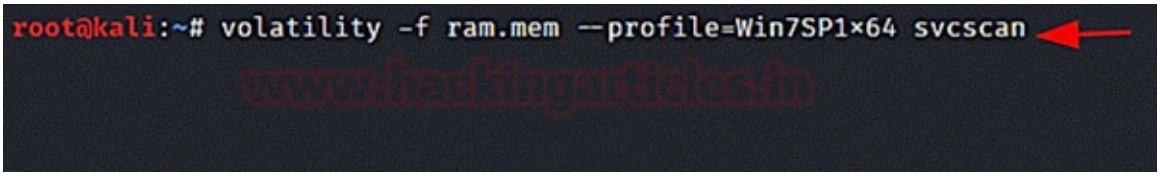


Figure 6.38: *Volatility SVCScan*

Conclusion

We discussed how to analyze acquired forensics images from both hard drives and RAM in this chapter. The emphasis was on performing the analysis using free and open-source software. Thousands of investigators around the world use open-source forensics software, and it has a proven track record in official investigations; however, keep in mind that commercial software will usually have more rich features when investigating certain types of digital devices, and some of these tools are officially enforced by law enforcement agencies in some countries. As a forensic examiner, you should practice using a variety of forensics tools, both free and paid, because there is no single tool that can perform all of the tasks required in computer forensics. Some tools are always superior to others when it comes to performing certain tasks.

We will continue our analysis work in the upcoming chapters, looking at how to develop in-depth digital forensics knowledge of operating systems by understanding where forensics artifacts can be found and how to analyze them to solve the case at hand.

CHAPTER 7

Windows Forensic Analysis

Introduction

The process of conducting or executing forensic investigations of computers running Windows operating systems is known as Windows forensics. This chapter examines incident response, recovery, and auditing of Windows-based technology used to carry out criminal activities. Investigators must thoroughly understand Microsoft Windows operating systems to complete such complex forensic investigations. Most systems temporarily store data about the current session in the registry, cache, and RAM. When the user turns off the computer, this data is readily lost, as is the session information. As a result, the investigators must retrieve it as soon as possible. This chapter will explain what volatile data is, why it is important, and how to extract it.

Structure

This chapter presents and discusses the following in detail:

- Timeline analysis tools
- Data carving
- Windows registry analysis
- USB device forensics
- Printer registry information
- File format identification
- Windows thumbnail forensics
- Windows 10 forensics
- Notification area database
- Cortana forensics

Timeline analysis tools

Most digital forensics investigations include timeline analysis because it provides a comprehensive perspective of the sequence of events that occurred on the system in order and is used to answer a key question in any investigation: when did a given action occur? Investigators can save time by limiting the amount of data that has to be studied in a specified timeframe, such as following an occurrence, using timeline analysis. When analyzing malware occurrences, timeline analysis is critical for determining when a system state has changed due to a malware attack. The Autopsy timeline interface aggregates distinct anomalies discovered inside the given forensic images according to their timestamps.

For each file in the forensic image, there is a timestamp. The time properties of a file include when it was created, edited, and accessed. Please keep in mind that the "*Changed*" and "*Created*" time properties of files are handled differently by each operating system. For example, the file creation and change times in Windows indicate the content changes, whereas in UNIX, the creation time attributes are not stored, and a file is considered modified when its metadata is altered, regardless of whether the file's content has changed.

Timeline analysis is useful for a variety of investigation types, to generate a timeline of events, and is often used to provide answers to computer usage and events that occurred before or after a given event. Autopsy contains an advanced timeline interface, as illustrated in *figure 7.1*.

1. Launch Autopsy to generate and build a new case or launch an existing case.
2. Click on the **`Tools`** menu | **`Timeline`**.
3. Depending on the size of the forensic image, Autopsy can take some time to populate the timeline data.

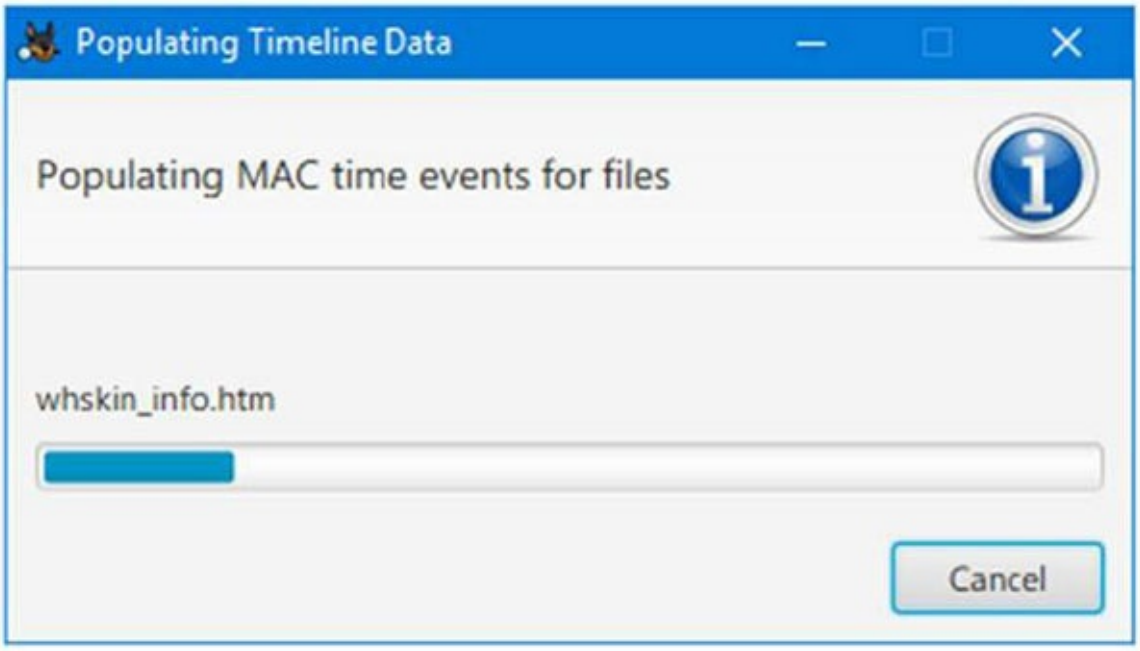


Figure 7.1: *Autopsy timeline data in progress*

After the finalization of the timeline data populating process, Autopsy can provide data in three view modes:

- **Mode for bar charts** (**counts**): As depicted in *figure 7.2*, this mode provides less information and is meant to answer inquiries regarding the amount of data modification that occurred over a specified period. Autopsy permits Forensics Investigators to inspect the contents of a forensic picture file using a variety of viewing applications.
- **Detail mode**: This mode provides information about occurrences and presents them using a unique clustering method by grouping the same event files into one folder and displaying the URLs belonging to one domain as a single event.
- **List mode**: Similar to detail mode, but displays the results in a chronologically ordered list.

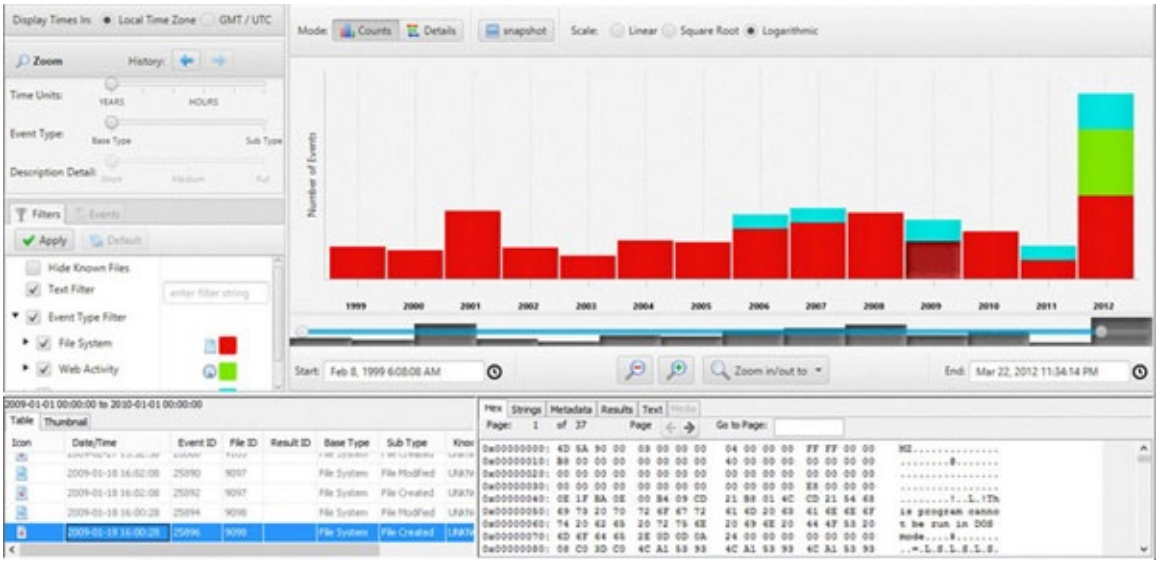


Figure 7.2: *Autopsy data occurrence (bar chart)*

Autopsy lets you create a report in Text, Excel, HTML, and other file formats, including the time stamps for each file in the forensic picture you have provided. This capability offers up the possibility of using such data in applications apart from the Autopsy tool. To generate the timeline report, simply click on the Tools menu → select Generate Report wizard. This allows selection from various report format modules, as illustrated in *figure 7.3*.

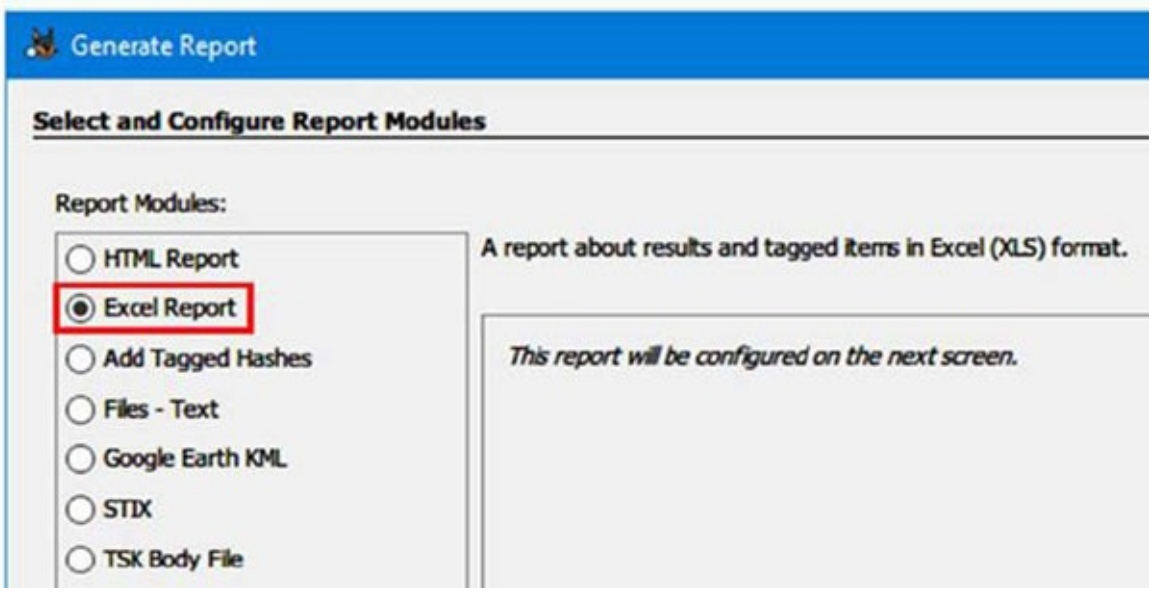


Figure 7.3: *Select report format*

Figure 7.3 displays the selected **`Excel Report`** so we can verify the data using Microsoft Excel or other alternatives such as Apache OpenOffice. Click on **`Next`**, and we can configure two options for the results Tagged or All Results. Here, once we select all results, Autopsy starts generating the report, as presented in *figure 7.4*.

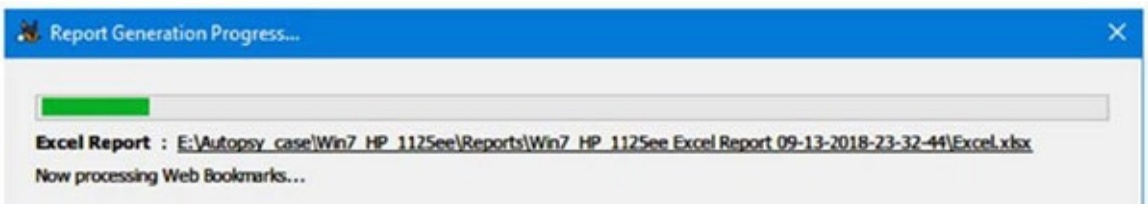


Figure 7.4: *Report generation progress window*

Once the report generation process finishes, Autopsy displays a link for accessing the generated report, as displayed in *figure 7.5*.

Figure 7.5: *Autopsy-generated report*

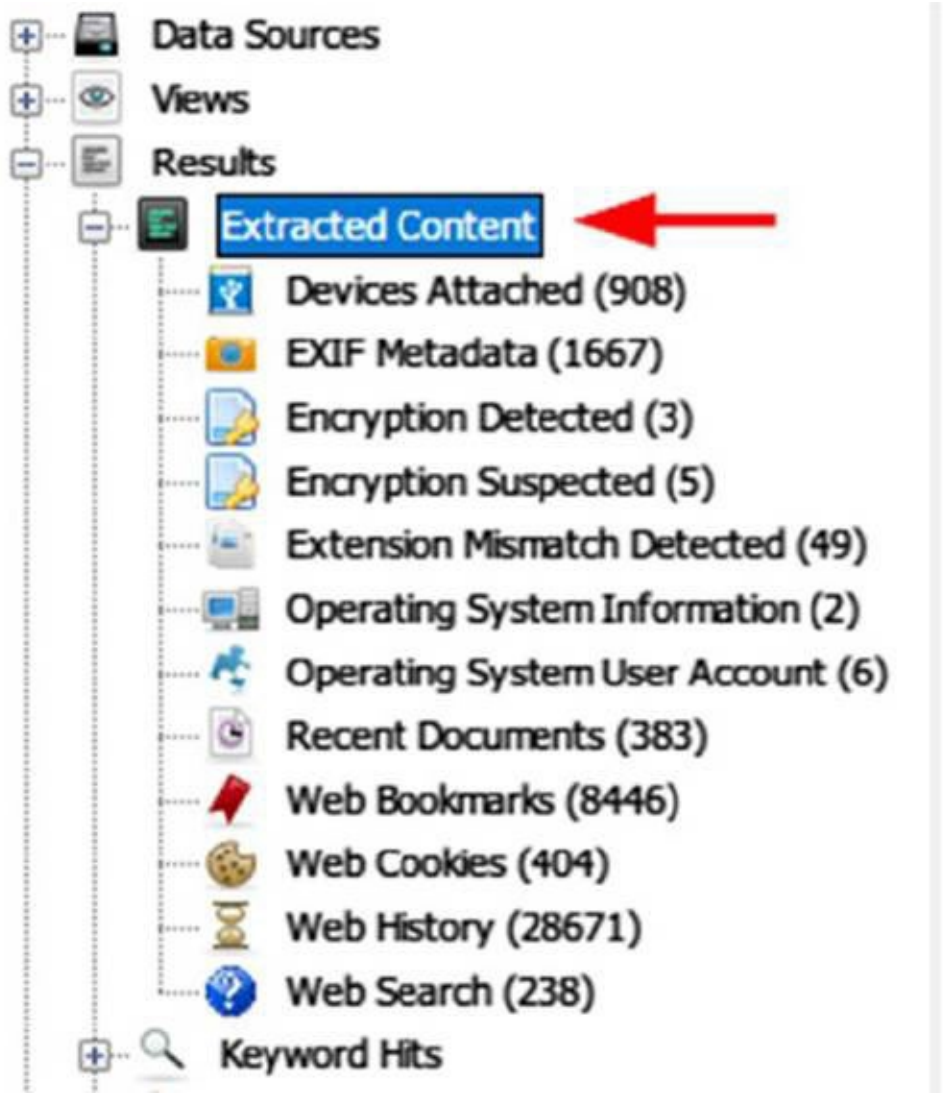


Figure 7.6: *View results of the "Recent Activity" module*

File recovery

In any form of digital forensic investigation, analyzing deleted files is a critical duty. Good digital forensic investigators should understand file deletions and locations in Windows, even after they have been removed, and analyze the files (for example, obtaining deleted files' metadata to aid a criminal investigation). In this part, we will go over several tools and procedures for recovering essential documents and file fragments that can aid in the investigation of the issue at hand.

Undeleting files

The forensic examiner is not required to intervene while using Autopsy to retrieve deleted files. You only need to build the case by picking the "PhotoRec Carver module" from the intake modules, and you are good to go. An Autopsy will recover data from the provided data source's unallocated space, which is displayed under the Views Deleted Files in the Data Explorer pane. PhotoRec is a free, standalone, and open-source utility for recovering data and files from a variety of digital media devices, including USB, HDD, and external cards, and SD cards from Smartphones, CD-ROMs, or digital cameras. PhotoRec can be used in conjunction with TestDisk, another free source application that specializes in recovering missing partitions and/or repairing non-booting discs so they can be booted again. A detailed walkthrough on how to use TestDisk.

Recycle bin forensics

Recycle bin in Windows OS was initially introduced in Windows 95; this holds files that have been removed by users but remain on the system. When a user deletes a file (by pressing the conventional delete key after selecting the target file OR by selecting a file, right-clicking it, and selecting the "Delete" of the pop-up menu, Windows sends the subject file to the recycling bin rather than permanently erasing it. This is Windows' default behavior; however, a user can change the recycle bin settings to delete files permanently rather than moving them to the recycle bin. Few users hold and press the *Shift* key when deleting files—this option permanently deletes the data and files instead of moving them to the recycle bin.

In practice, few individuals delete recycled files permanently (or even know about it); as a result, the recycle bin might store vital recycled artifacts, which are regarded as valuable sources of digital evidence. When a user deletes a file, Windows will automatically send it to the recycle bin. The names and locations of the recycle bin files vary between Windows versions. Files deleted under the FAT file system of Windows XP are saved in the **`C:\Recycler`** folder in C: drive. This folder also contains another significant file entitled INFO2 as a hidden file. To view them, enable the show hidden files, including C:\ OS files. There are a few folders inside the **`Recycler`** folder that are named as per user **security identification** (**SID**); if a system has many users, each of them will have its folder that contains the deleted items belonging to that user account. Inside each user's recycling bin folder is another crucial file named **`INFO2`**, which holds an index of all the files that the user has previously removed. It also includes details about each deleted file, such as the file's original location, size, and deletion date/time. Vista and later versions of Windows (7 to 10) modified the recycle bin, deleted items, and the main folder.

Deleted files, for example, are saved in a folder called **`$Recycle.Bin`**, which has a subdirectory for each user on the system named after that user's SID. The **`$Recycle.Bin`** file is kept on the C: disc (assuming Windows is installed there). When a file is deleted in the recent Windows OS versions, it is moved to the recycle bin as the first file, including the recycled file's contents beginning with "$R", and the other file containing the deleted file's metadata beginning with "$I". This eliminates the need for prior Windows versions of **`INFO2`** file, which was used to recycle a file's metadata. When it comes to the amount of deleted data that Windows recycle bin can hold, it has a restricted storage capacity. The recycle bin in Windows XP is set to contain 10% of the hard disk by default; if it fills up to its full capacity, it will trash old data to create a way for newly deleted ones. The default size in recent Windows versions, such as Vista and beyond, is 10% of the first 40 GB of storage space and 5% of the remaining storage space over 40 GB. Let us try removing a file and then analyzing it with Windows 10, and a free program called $I

Parse. Open a command prompt and change the working directory to **`$Recycle.Bin`** folder on the C: disk with the **`CD`** command. Using the **`DIR`** command and the /a switch, you can view the contents of a folder (to display hidden system files). *Figure 7.7* illustrates these instructions.

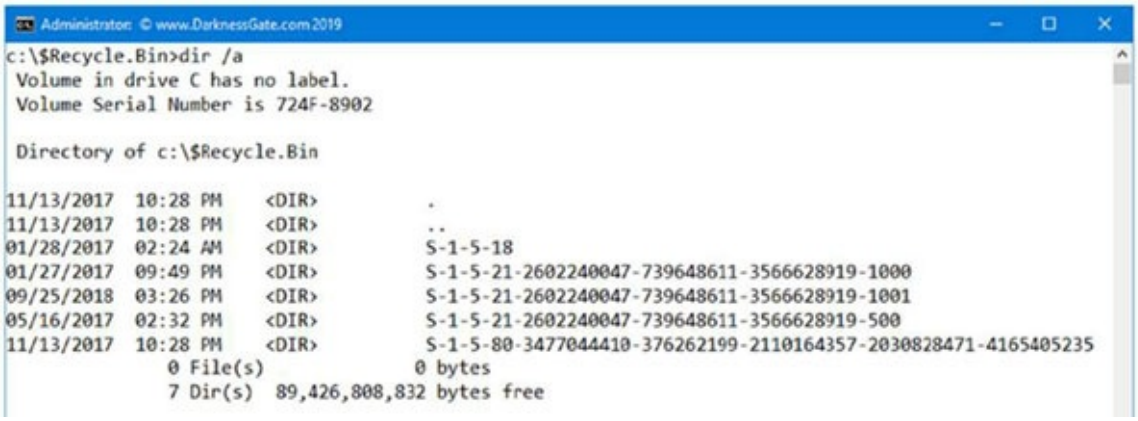


Figure 7.7: *View $Recycle.Bin contents*

`$Recycle.Bin` is divided into four subfolders, each corresponding to the SID of the person who removed the file. When the file is deleted, it is sent to the recycle bin for the first time, and then a subfolder is also created. The command "wmic useraccount get name, sid" is used to find out the name of the user account that owns a given SID subdirectory. This will reveal all of the target machine's user accounts, allowing us to determine which SID subdirectory in the **`Recycle.Bin`** corresponds with the user, as presented in *figure 7.8*.

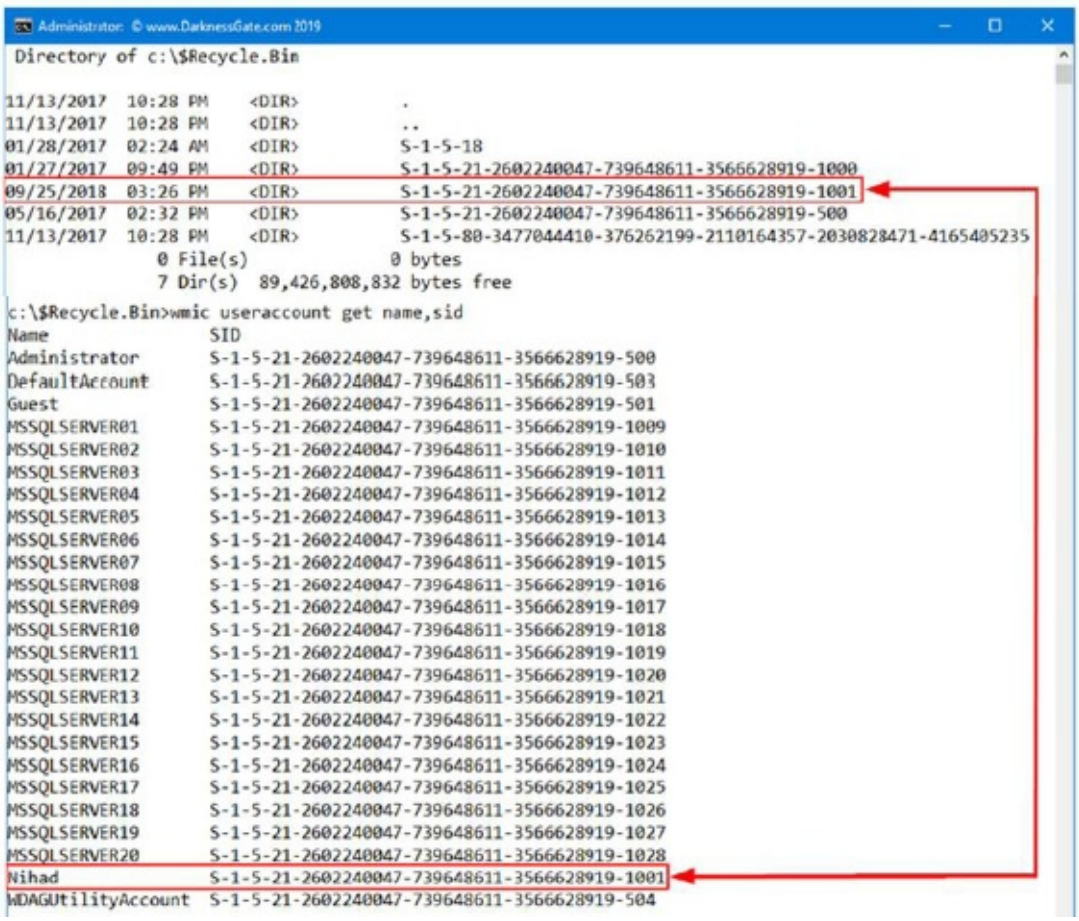


Figure 7.8: *Determine owner of specific SID subfolder inside $Recycle.Bin*

We can use the **change directory** (**CD**)command to access the target account's recycling bin once we know which one it belongs to. To display the contents of a directory, use the **`DIR`** command with the /a

switch, as illustrated in *figure 7.9*.

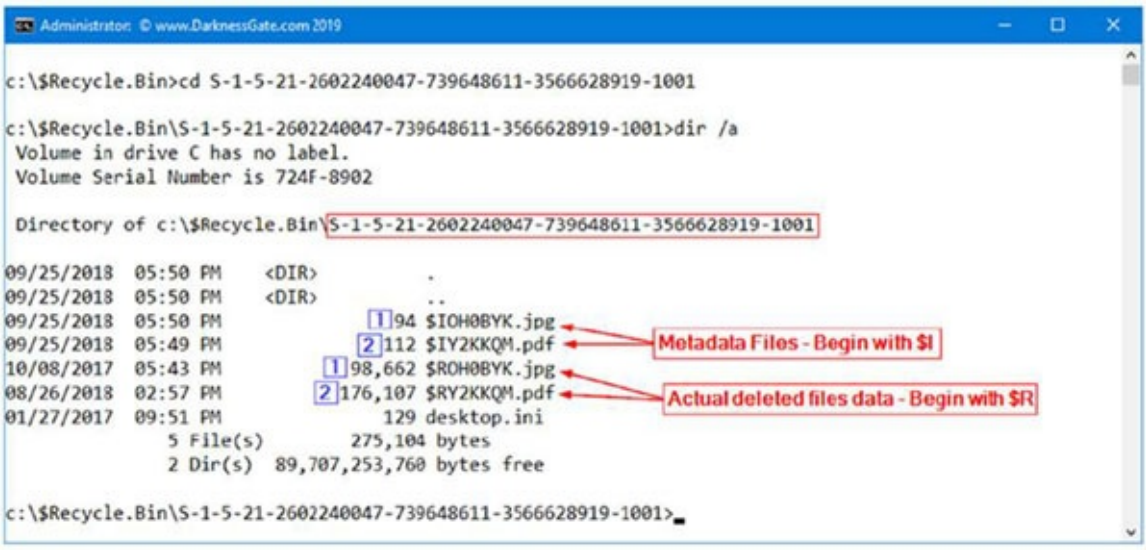


Figure 7.9: *View the target recycle bin*

The preceding figure displays the target recycle bin which contains four files—these are from the deleted files as presented in *figure 7.10*. As previously stated, each deleted file contains two different files in the recycle bin—the Metadata file along with the actual contents of the deleted file, which are recoverable.

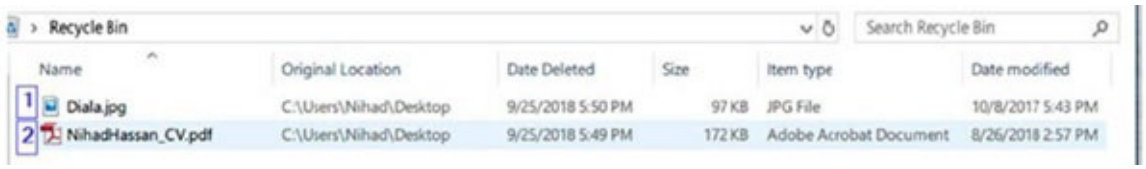


Figure 7.10: *Deleted files in recycle bin*

Let us look at the metadata of the deleted file, also known as index files (start with $I), in Windows Vista's recycle bin and then extract its contents with a free program called $I Parse (if it is zipped). To use this utility, you must first extract the metadata file from the recycled file. To do so, open a command prompt and type the following (see *figure 7.11*): $I* \users\ desktop\recover is copied, and the $I Parse tool is run from the **`File`** menu. Select the metadata files folder from the list of options.

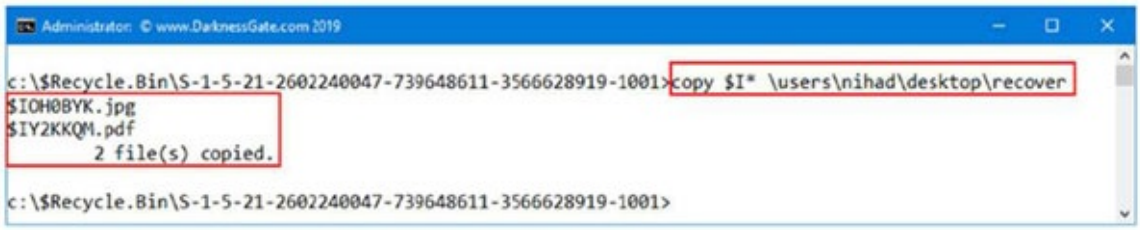


Figure 7.11: *Copy recycled file metadata from Recycle to another folder*

Select where to store the output CSV file, which would hold the parsed results of the program code menu by clicking the **`Choose`** button (see *figure 7.12*).

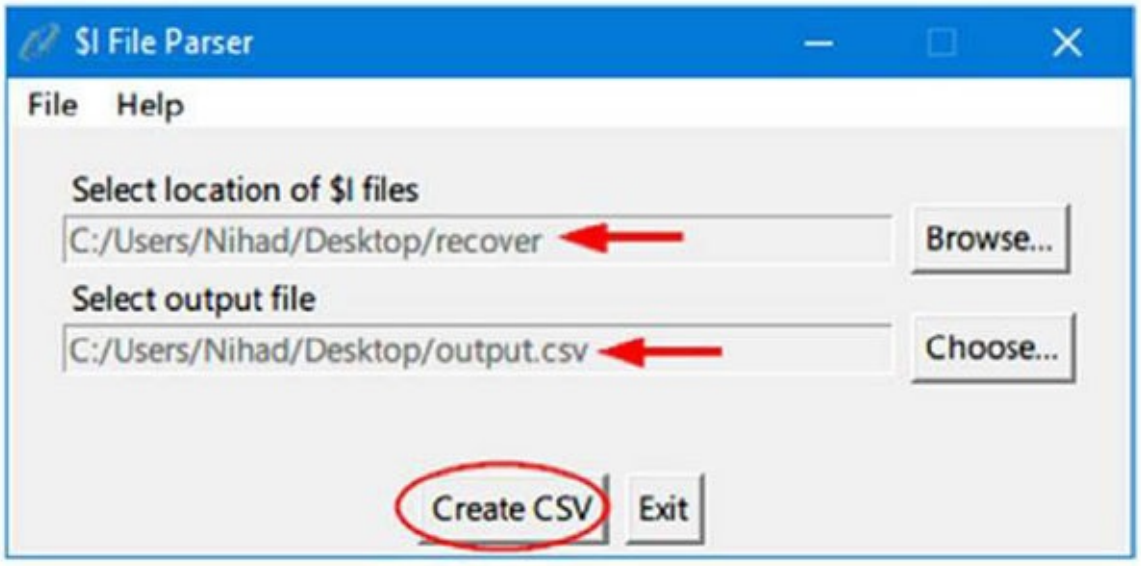


Figure 7.12: *Parse metadata files in the target directory*

Finally, click **`Create CSV`** when all of the files have been parsed, a success window will appear, and you are done. Now open the output file (**`Output.csv`**) to see a list of all recycled file names in the target recycle bin, together with the metadata information for deletion date/time, original path, and file size, as presented in *figure 7.13*.

Figure 7.13: *Output.csv displays recycled files' metadata files*

"Rifiuti2" can be used for extracting information from the recycled metadata INFO2 files in Windows XP (and higher versions of Windows OS).

Data carving

Data carving is used in digital forensic investigations as a part of sophisticated data recovery when extracting specific files using the file footer and header information. This is gathered from unallocated space (raw data) without using any MFT or file system structure. When the file system that was originally responsible for arranging these files on the hard drive is absent or corrupted, data carving can be the only method for retrieving vital artifacts and evidence files from file fragments in a criminal investigation. When extracting a file(s) from a recorded network traffic stream, data carving is also required. Data carving is a more advanced digital forensics method that is outside the scope of this book. Expert forensic investigators can, however, use data carving techniques to extract (recover) organized data, and therefore, a file, like a document or a photo, from non-structured or raw data. Although file carving can be done using just a Hex editor, certain tools can help investigators, such as Foremost, Scalpel, Jpegcarver, and Forensics wiki.

Associated user account action

A suspicious Windows PC can contain many accounts, such as one for Nihad, another for Rita, and yet another for Susan. The SID is a one-of-a-kind number that differentiates each account on a Windows PC.

A digital forensic examiner can use this SID to determine which user account performed which activity or when a certain user account caused a specific event.

Windows registry analysis

The registry is the core of the Windows operating system; it stores vital information that the operating system and installed applications require to run. Almost every action taken by a Windows user is recorded in the registry in some fashion, making the registry a rich source of evidence that can be incredibly useful in any digital forensic investigation.

Windows registry architecture

The registry is a hierarchical database that contains the user's preferences and the computers and programs' usage history, along with Windows configuration settings for software applications, operating systems, and the hardware. The data in the registry is organized in a tree fashion, with each node in the tree being referred to as a key. In addition to data values, a key might include other keys (subkeys), as presented in *figure 7.14*.

Registry hive	Supporting files
HKEY_LOCAL_MACHINE\SAM	Sam, Sam.log, Sam.sav
HKEY_LOCAL_MACHINE\Security	Security, Security.log, Security.sav
HKEY_LOCAL_MACHINE\Software	Software, Software.log, Software.sav
HKEY_LOCAL_MACHINE\System	System, System.alt, System.log, System.sav
HKEY_CURRENT_CONFIG	System, System.alt, System.log, System.sav, Ntuser.dat, Ntuser.dat.log
HKEY_USERS\DEFAULT	Default, Default.log, Default.sav

Figure 7.14: *Windows registry structure*

The Windows registry has five root folders or hives. When you first use the registry editor, the first folders appear on the left side of the screen, and all other entries are minimized. When it comes to data durability, root hives are separated into two categories: volatile and nonvolatile. The nonvolatile HKEY LOCAL MACHINE and HKEY USERS keys are saved on the hard disk; however, the remaining hives are volatile and need to be collected while the system is running to obtain relevant information. The Windows registry can be examined by digital forensic investigators using the registry contained within a forensic picture. Consequently, the computer forensics tool will be used to explore registry data in the same manner as Windows File Explorer. Second, by using the live analysis to access the registry using the Windows registry editor as you would on any other machine.

It is vital to know where the registry files are located if we are studying the Windows registry using a recorded forensic image Registry hives are stored in the **`Windows`** | **`System32`** | **`Config`** folder, so if your operating system is installed on the C: disk, you will find your registry files in the **`C:\Windows`** |

`System32` | `Config` folder. You will find numerous files in this folder (one for each root hive and a few supporting files for each, except the HKEY CURRENT USER hive, which is placed in your profile folder). A registry editor is included with Windows, allowing any user with administrator privileges to view, modify, and back up the registry.

Acquiring windows registry

When obtaining the target machine's system drive or doing a complete hard disk acquisition, computer forensics tools will acquire Windows registry data. You can also remove only the registry files from a running system and save them separately for subsequent study (this is known as a *Registry Image*). This is demonstrated using AccessData FTK Imager in the following example. To acquire the target Windows machine registry using FTK Imager, you must first download the AccessData FTK Imager onto a USB flash drive. Attach the USB device containing FTK Imager to the compromised computer, launch FTK Image, and then navigate to the File menu to acquire Protected Files.

Figure 7.15 depicts the new dialogue box; select the location where you wish to store the retrieved files, and check the option *Password recovery and all registry files* before clicking `OK`. The export progress of the registry files is displayed in a progress window, which disappears upon completion without displaying a success message.

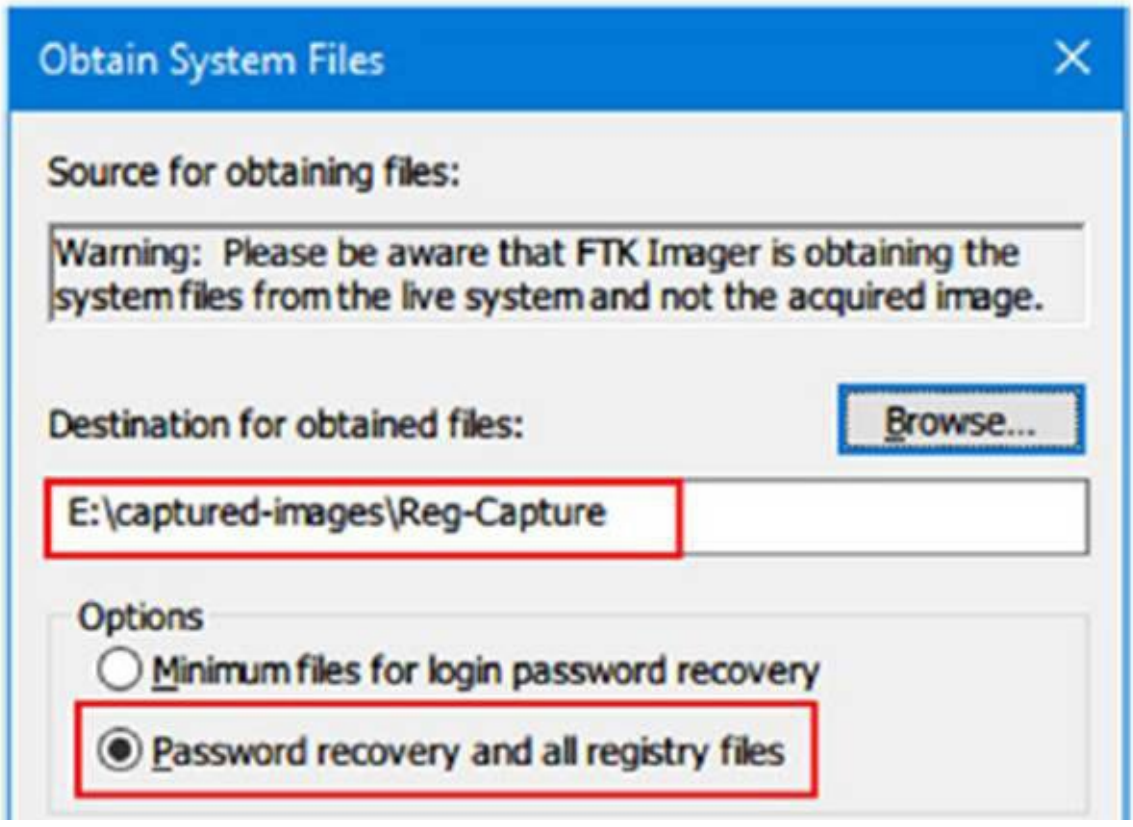


Figure 7.15: *FTK Imager to acquire target Windows registry database*

To view the generated files, navigate to the directory where you stored your registry files; you should see five files and one folder (see *figure 7.16*).

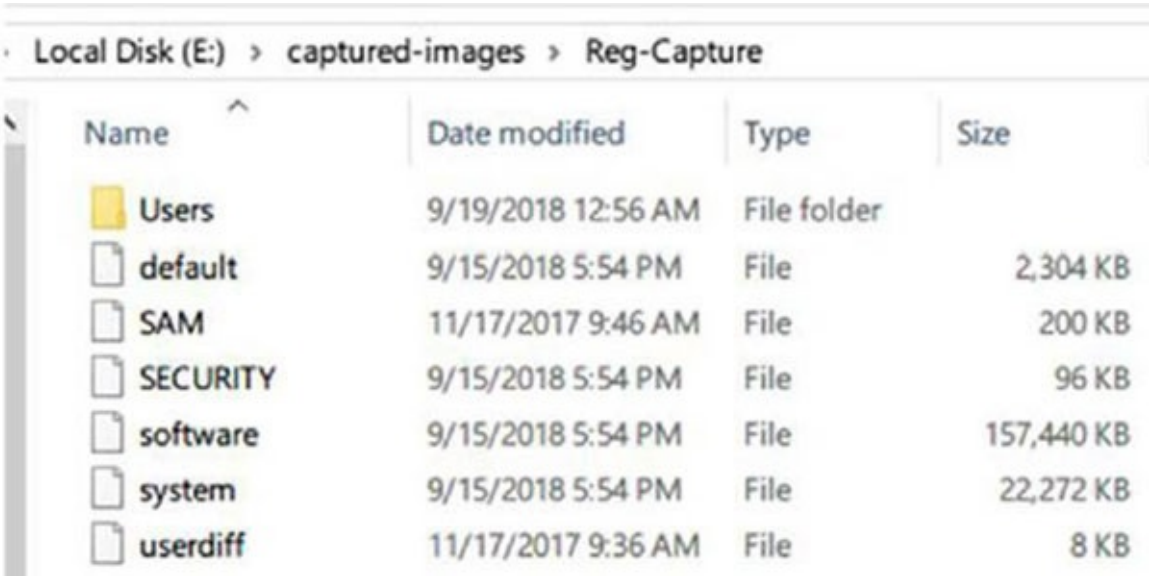


Figure 7.16: *Registry forensic image captured with AccessData FTK Imager*

Since the target machine's registry has been exported, we can now evaluate further using various forensics tools.

Registry examination

From the collected forensic picture, most computer forensics apps can probe the Windows registry. Other tools focus solely on studying Windows registry data. We will presume in this part that we have booted up with a suspect forensic image—to execute different forensic analyses on it. Where relevant, several specialized tools for researching specific sections within the register will be provided as well. Windows contains a capability that allows the program to start automatically as the operating system starts; this functionality is important for anti-virus software, which must run first to block any dangerous software before Windows can fully boot. Malicious software, such as keyloggers and botnets, can create entries in the Windows registry to run automatically when the operating system starts up, as illustrated in *figure 7.17*.

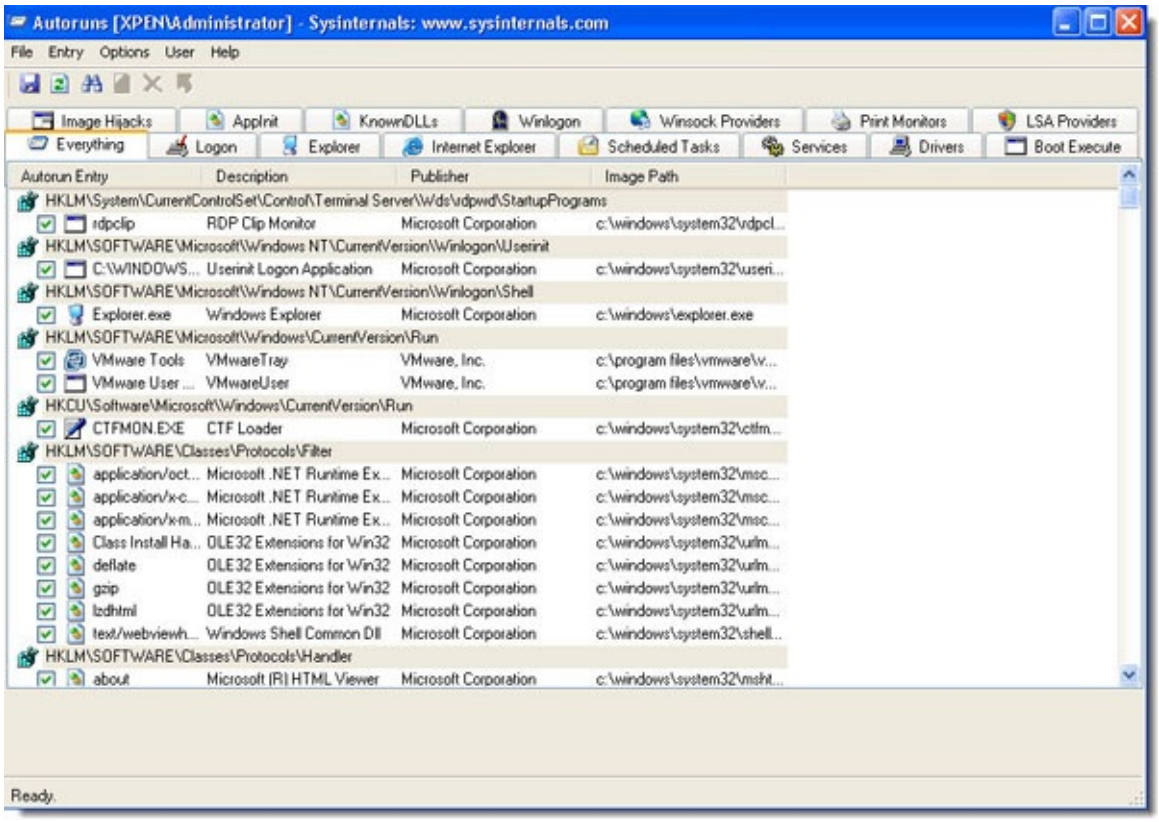


Figure 7.17: *Autorun from Sysinternals*

In many circumstances, investigating starting programs can help with forensics; for example, malware can take control of a suspicious system and use it to execute denial-of-service attacks without the owner's awareness. Even if his or her computer was used to commit a crime, a suspect could become highly open to authorities when something like this is probed.

Windows registry program keys

Forensic investigators can benefit greatly from knowing what programs are presently or were previously installed on the suspect system. For instance, the presence of steganography and encryption applications or the remnants of such a tool indicates that the suspicious system can hold concealed data or be merely used to run such a program. The following registry locations in Windows maintain track of all installed apps. We can use automated tools to look for a program installed inside the Windows registry or search for lost information like pieces of installed programs, apps that have been left behind, or any other data items that can be concealed in the Windows registry. RegScanner is a simple program offered by Nirsoft that searches the Windows registry based on specified search criteria supplied by the user. The retrieved results are shown in a list, and a user can select any item in the list to open RegEdit and view the related value. We can also save the registry values contained in the **`a.reg`** file. Following the execution of this program, a search options window will display, where you can input your search criteria and specify certain search settings (see *figure 7.18*).

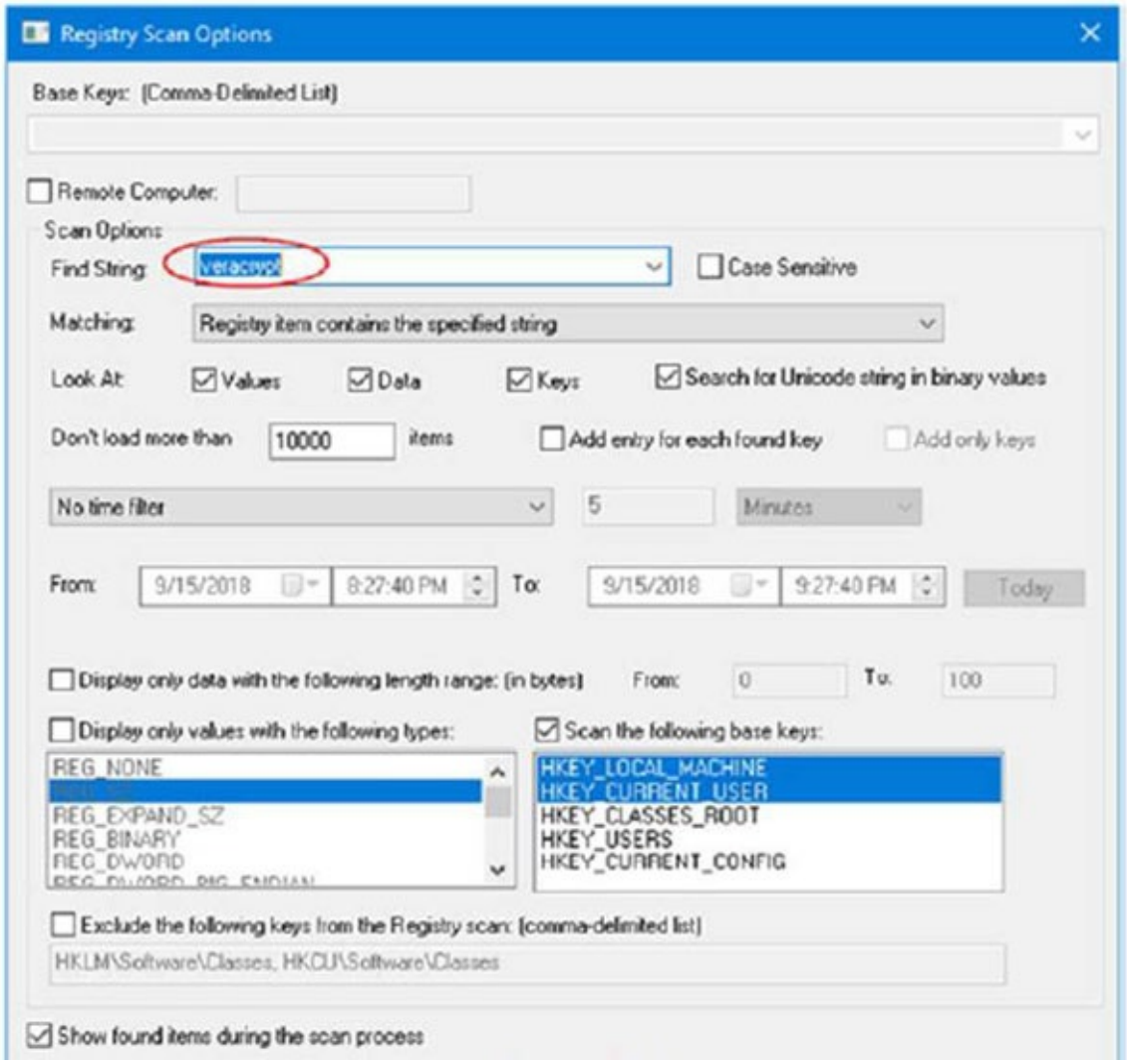


Figure 7.18: *Registry scan options by RegScanner*

While not all programs require the installation of a registry entry before use; for example, portable apps do not require installation on Windows to execute, such as apps run from USB drives. The registry can be examined for previously connected USB devices for assessing any possibilities of running portable apps from a questionable PC. Another way to find out if portable apps are being executed is to look in the Windows Prefetch folder. These are the *temporary files stored in the System folder name as a prefetch*. Prefetch is a memory management feature. The log about the frequently running application on your machine is stored in the prefetch folder. The log is encrypted in Hash Format so that no one can easily decrypt the data of the application.

USB device forensics

All previously connected USB devices, their connection timings, and the user account that installed them are saved in Windows' history record. Important technical information for each connected USB device is also stored in the Windows registry, including the vendor ID, product ID, revision, and serial number. Windows uses five registry keys to hold USB history information, each containing a distinct piece of information about the connected device. The Windows registry is a hierarchical database where information is presented on a number of levels (up to six). Hive keys are on the first level. There are five hive keys, each of which begins with "HKEY_" and the name of the keys as HKEY_CLASSES_ROOT, HKEY_CURRENT_USER, HKEY_LOCAL_MACHINE, HKEY_USERS, and HKEY_CURRENT_CONFIG. Investigators will be able to see how an offender used detachable devices,

such as a USB, to conduct/facilitate his/her acts after combining this information. You can download a free utility called USBDeview from Nirsoft to automate the process of discovering information about current and prior USB-connected devices. This tool can execute all of the things we just completed manually. Extensive information (for example, device name/description, device type, serial number, and much more) about each connected USB device will show after running this utility on the target machine. The Last Plug/Unplug Date reflects the first time the device was connected to the system. When the same device is reinserted many times, the date does not update. The "Created Date" indicates when the same device was last connected to the system.

Unfortunately, not all USB device types, such as USB devices that employ the **media transfer protocol** (**MTP**) to connect to PCs, will leave traces in the Windows registry, as we have shown. The MTP protocol is used by devices running newer Android OS versions, as well as Windows phones and Blackberry; this protocol does not leave traces in the Windows registry when a USB device is connected to a Windows PC. This needs the use of a specialist instrument to investigate such objects. USB Detective (**https://usbdetective.com**) can detect USB devices that connect to Windows through the MTP protocol. It also has advanced tools for extensively analyzing linked USB devices, such as constructing timelines of every unique connection/disconnection and deletion timestamps for each device; however, to access all of these functions, you must upgrade to the premium professional edition. To sum up, this part, obtaining traces from a USB device connected through an MTP connection requires specific treatment; check your computer forensic software manual for the existence of such a function.

Most recently used list

When you open a file using Windows File Explorer, a standard open/save dialogue box, or an MS-DOS command prompt on the registry, Windows keeps track of the most recently accessed files. Numerous Windows applications, including recently opened MS Office files and recently visited websites, have **most recently used** (**MRU**) lists; these applications list the most recently accessed files.

Network analysis

When a Windows user connects his or her computer to the Internet or an intranet, Windows records the connection in the registry. Knowing the network connection is crucial for forensic purposes; for example, the register details all network cards that have been used on the suspect system, whether built-in or external (for example, via a USB port). The registry will also show the wireless connection profile (name, IP address, subnet mask, and DHCP), as well as the date the connection was formed and the last time it was used.

Windows shutdown time

Under the ShutdownTime value in the registry entry HKEY LOCAL MACHINE\SYSTEM\Current\ControlSet\Control\Windows, Windows records when the machine was last shut down. The shutdown value is saved as a binary value written in Little Endian format; use Digital Detective's DCode tool to decode it into a readable format. To use this tool, first, extract the binary value from the target key, then enter it into the DCode application using the choices, as shown in *figure 7.19*.

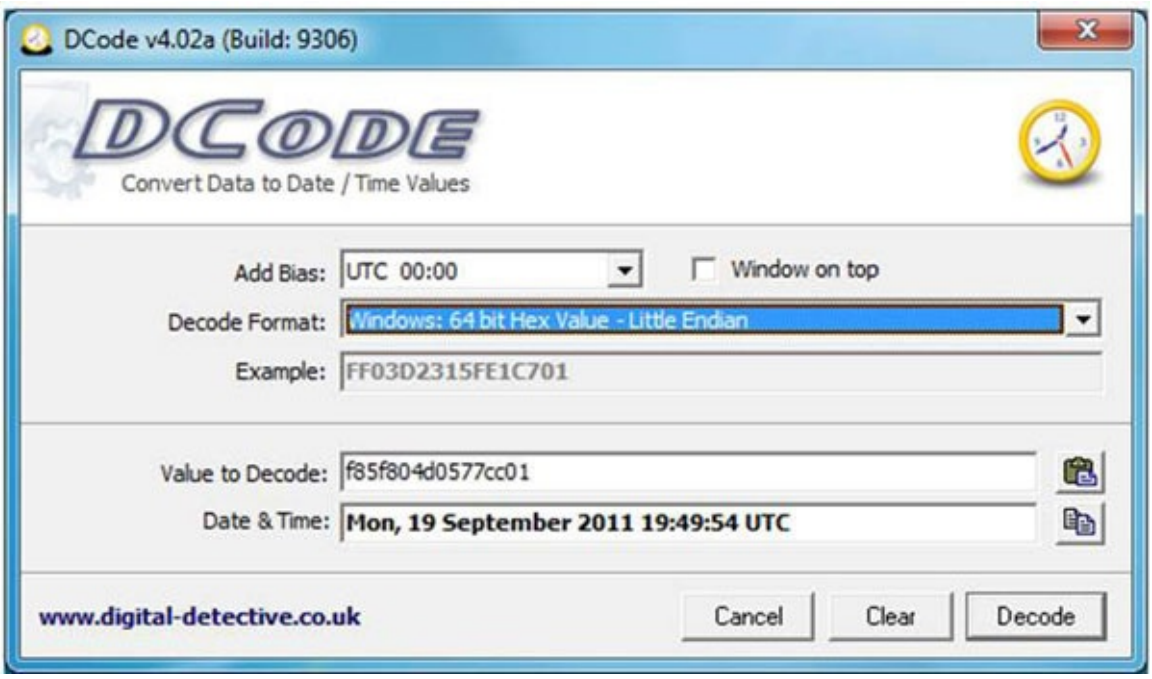


Figure 7.19: *Decode Windows shutdown time*

UserAssist forensics

UserAssist maintains track of all recently opened executable applications, as well as the frequency of use (number of executions) for each one. The registry key UserAssistinformation can be located in the following location: HKEY CURRENT USER\Software\Microsoft\Windows\CurrentVersion\Explorer\ UserAssist. The ROT-13 encoding scheme is used to encode the information stored in UserAssist keys. Use UserAssist-View, a Nirsoft program that can show stored data in an understandable format, to decode this data.

Printer registry information

For example, to check the properties of presently installed printers on the target system, navigate to **`HKEY LOCAL MACHINESYSTEM\Current\ControlSet\Control\PrintPrinters\`** Printer-name in the Windows registry (see *figure 7.20*).

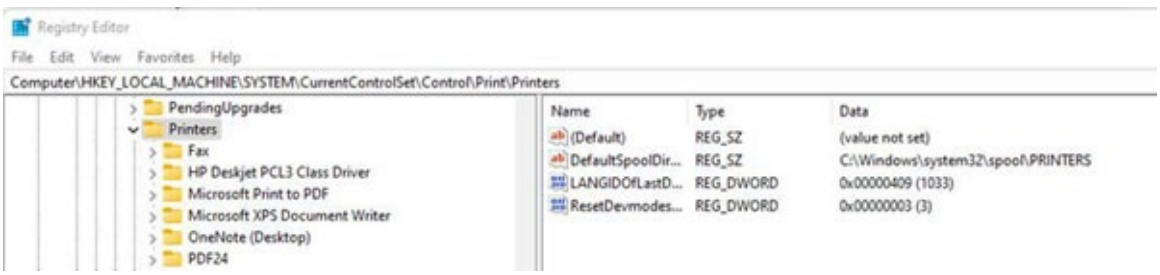


Figure 7.20: *View installed printer properties*

File format identification

Signature analysis is a procedure in which file headers and extensions are compared with a known database of file headers and extensions to see if an attempt to disguise the original file type (by changing the file extension to something else to hide it from investigators' eyes) has been made. As we all know,

each file in Windows has its signature, which is normally contained in the first 20 bytes. By inspecting a file using Notepad or a Hex editor, we can determine its original file signature. By manually studying a file's signature, we can establish its kind. We can use HexBrowser, a free program, to automate this procedure. HexBrowser is a Windows program that can detect and display extensive information about over 1,000 different file types, as presented in *figure 7.21*.

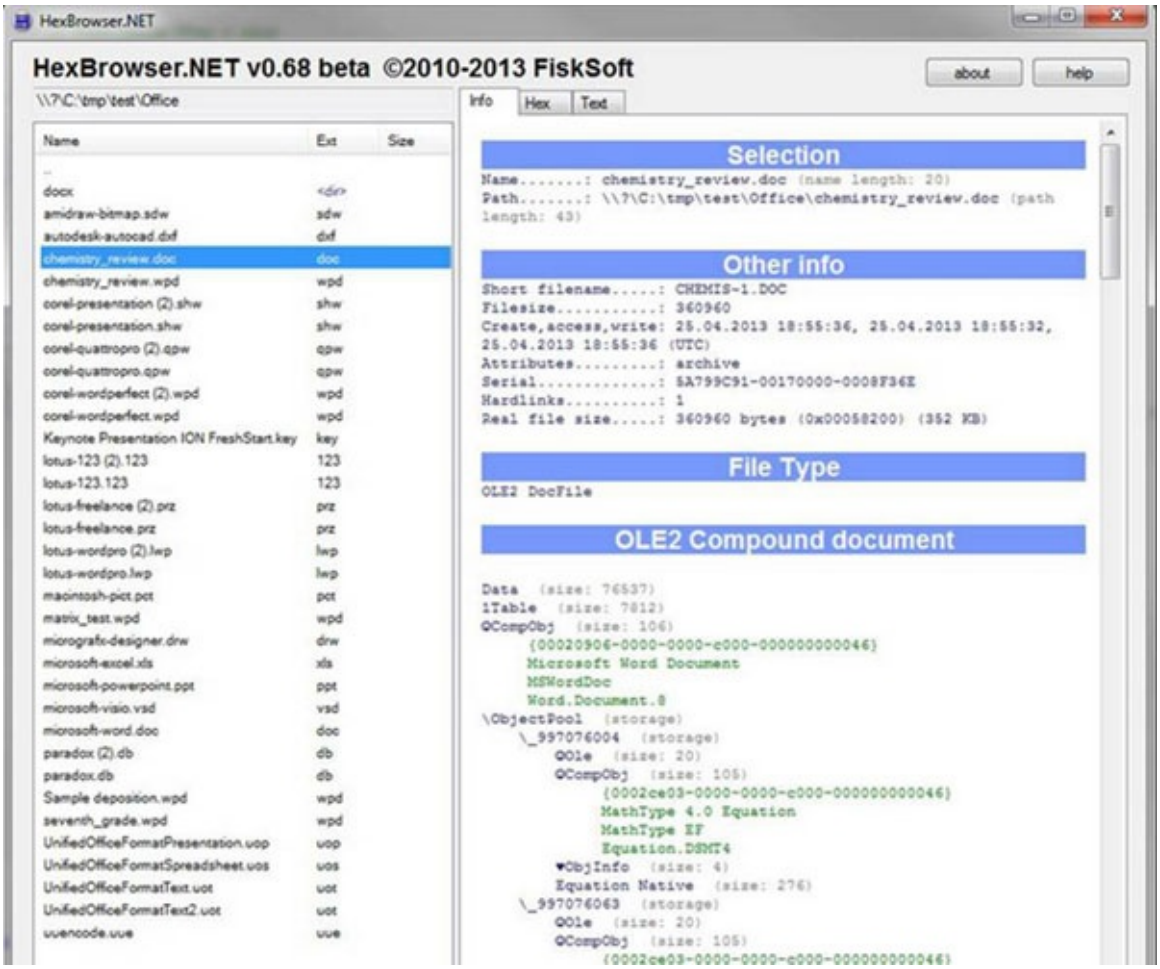


Figure 7.21: *HexBrowser discovers original formats*

In this case, HexBrowser was used to explore a file with a DLL extension, and it was revealed that the original file type was MS Word 2016. The Autopsy can detect file extension mismatches; to use this capability, you must first enable the "Extension Mismatch Detector" module. By navigating to the **`Tools`** menu and selecting Choices File Extension Mismatch, you can further customize your file mismatch search options. You can add or delete extensions from here based on your case requirements, and the results are displayed in the Results tree under **`Extension Mismatch Detected`**.

Windows thumbnail forensics

When a user chooses to view files as thumbnails, Windows saves thumbnails of graphical files (JPEG, BMP, GIF, PNG, and TIFF), some document types (DOCX, PPTX, and PDF), and video files in the **`thumbs.db`** thumbnail cache file for subsequent viewing. Examining this feature can reveal prior files (for example, photos) that were there on a system even after the user erased them, as image thumbnails can persist at **`thumbs.db`**. Thumbnail previews are stored in a central location in modern Windows versions (Vista+). The cache is kept as a succession of files with the usual name thumbcache **`xxx.db`** (XXX refers to its size), as well as an index file to find thumbnails in each database, in **`%userprofile%\AppData\Local\Microsoft\Windows\Explorer`**.

Thumbs Viewer is a portable utility that extracts thumbnail images from the database files **`Thumbs.db`**, **`ehthumbs.db`**, **`ehthumbs vista.db`**, **`Image.db`**, **`Video.db`**, **`TVThumb.db`**, and **`musicThumb.db`** are available on all versions of Windows OS. It is available for download at **https://thumbsviewer.github.io**. If you need to access thumbcache **`*.db`** files, try Thumbcache Viewer (**https://thumbcacheviewer.github.io**), a tool from the same creator that lets you extract thumbnail pictures from the thumbcache **`*.db`** and icon cache **`*.db`** database files found in Windows Vista, 7, 8, 8.1, and 10. In Windows Vista and later, thumbcaches are normally found in **`Users\<USERNAME>\AppData\Local\Microsoft\Windows\Explorer`**.

Windows 10 forensics

Edge browser, Windows 10 apps, Cortana (Microsoft's voice-controlled digital assistant), and many more new features and applications are available to Windows 10 users. The most significant was the introduction of the **Universal Program Platform** (**UAP**), which allows the same application to operate on various platforms, including laptops, desktops, IoT devices, tablets, smartphones, and more. Windows 10 comes with a slew of new features. In this section, we will investigate two of them:

- Notification area database
- Cortana forensics

Notification area database

This new feature debuted with Windows 8 and is now available in Windows 10 and 11 versions. Any program that can generate a systray notification will save it in a centralized database. Under the name **`wpndatabase.db`**, the notification area database is located in the **`C:\Users\UserName\AppData\Local\Microsoft\Windows\Notifications`** folder. The notification database stores a variety of notification types that Windows users see in the bottom right corner of the screen, including pop-up messages from various parts of the OS (for example, backup and restore), e-mail alerts, and messages related to specific apps such as Torrent downloads, among other things. Windows notification has forensic relevance as they can show earlier user behaviors on the target machine.

Cortana forensics

Cortana is a personal assistant that responds to voice commands, much like Apple's Siri. Cortana was initially launched in Windows Phone 8.1 and then migrated to the Windows desktop with the release of Windows 10. Its major objective is to provide Windows 10 users with a personalized experience by delivering search suggestions, remembering events, sending e-mails on the user's behalf (when correctly configured), browsing the Internet, and monitoring the weather, among other useful functions. Cortana operates by accumulating knowledge. If a user converses through the PC microphone or typing, the system will have a better understanding of their habits and attitudes, resulting in more accurate outcomes in future encounters.

In terms of digital forensics, in addition to longitude and latitude geolocation data of location-based reminders) and online searches, Cortana can reveal a plethora of information about a user's prior activity on the target system. Although Cortana provides a lot of useful information, it is not always enabled on Windows devices, as this tool has a reputation for being a privacy invader, and many Windows users have already disabled it owing to privacy concerns. Cortana saves information related to its work on an

extensible storage engine (**ESE**) database at:

\Users\<UserName>\AppData\Local\Packages\Microsoft.Windows.Cortana_xxxx\AppData\Indexed D\IndexedDB.edb

\Users\<UserName>\AppData\Local\Packages\Microsoft.Windows.Cortana_xxxx\LocalState\ESEDatab

The **CortanaCoreDb.dat** file contains forensically significant information on user geolocation data, as well as reminders issued by the user and where and when they have occurred. Please keep in mind that Cortana has access to a lot of personal information about its users; nevertheless, it appears that Microsoft has moved a lot of Cortana interactions with the Microsoft cloud. Cortana-related artifacts are also found inside local machine folders at **\Users\<User>\AppData\Local\Packages\Microsoft.Windows.Cortana_1234\LocalState\LocalRecorde**

This folder contains recordings of voice commands (WAV audio files) sent to Cortana by a user to execute a job. Not all computer forensic suites can decode the Cortana database; always read the instructions or check the tool's functionality before purchasing. EnCase, for example, offers a script that decodes Cortana search words from **IndexedDB.edb** files given by the user.

Conclusion

Windows stores a large quantity of data about its users, which are referred to as artifacts in the computer forensics field. This data might be dispersed around the system in many areas. Few people are aware that artifacts from the program that has been run, as well as any USB flash drives that have been attached to a Windows PC since its installation, are saved in several locations in Windows. The same is true for deleted files: the recycle bin will save information about each deleted file and the account that deleted it. Even whether the data was deleted on purpose or merely overwritten, copies of the deleted, formatted, edited, damaged, or lost files and folders can still be found on the target system in numerous locations.

The upcoming chapter discusses Web browser investigation to solve the case at hand. E-mail also plays an integral role in today's digital age communications, understanding how to analyze e-mail messages to find clues.

CHAPTER 8

Web Browser and E-mail Forensics

Introduction

Internet apps that are already deployed on Windows might provide useful information about the user's prior computer activities. A Web browser, for example, is the default method to access the internet, and criminals use it to perpetrate internet-related crimes or to target other people online. Web browsers are used by internet users to socialize, shop online, send e-mails, and surf the Web, among other things. Because of this, criminal actors prefer to steal personal information such as account passwords using Web browsers. Web browser artifact analysis is an important aspect of any computer forensic investigation because it may help pinpoint the source of the breach of the user's prior behaviors in many circumstances.

Structure

This chapter presents and discusses the following in detail:

- Web Browser Forensics
- Google Chrome Browser Forensics
- Mozilla Firefox Browser Forensics
- Other Browser Forensics tools

Objectives

In this chapter, we will discuss how to look into various Web browsers for interesting clues that might help us solve the case at hand in today's digital era communications; how to examine user browsing patterns for clues and concealed Web browser data. For example, if we look at the suspect's Web browsers and discover their browsing history, saved passwords, or that he or she was downloading or searching for material on steganography and encryption tools, this is a strong indication that the user may be using these tactics to hide sensitive information.

Web browser forensics

Google Chrome, Safari, Samsung Internet, Firefox, Edge, Opera, UC Browser, and internet Explorer from Microsoft had the majority of the Web browser market share by the end of 2021. (Refer to *figure 8.1*). As a backdrop, the focus of this chapter is on analyzing artifacts from three main Web browsers: Google Chrome, Mozilla Firefox, and Microsoft Edge, using various digital forensic techniques.

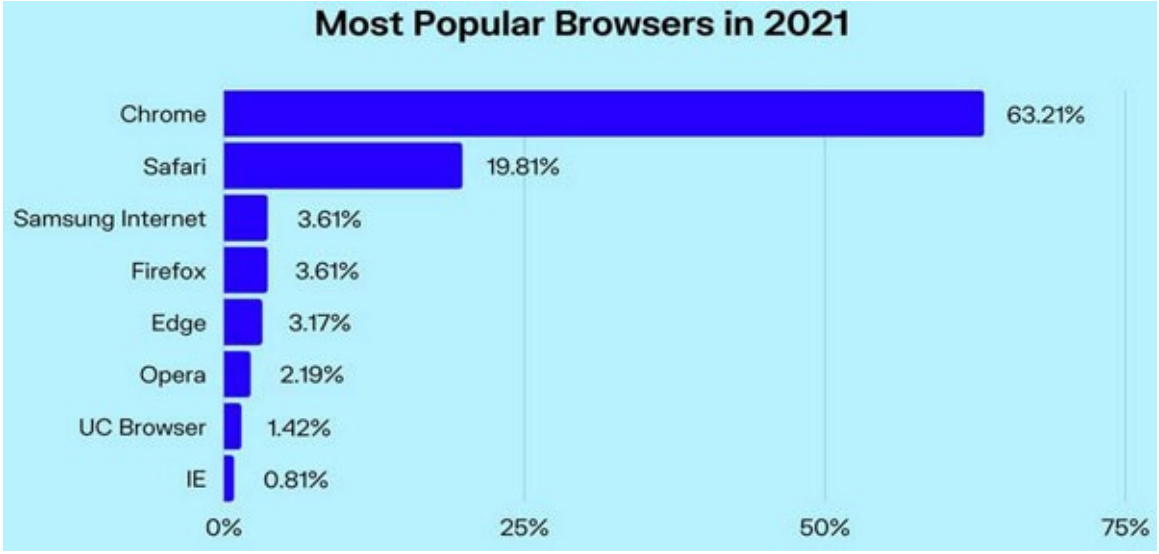


Figure 8.1: *Popular Web browsers in 2021 [1]*

In 2022, the market share varied slightly between the top contenders, namely, Chrome and Safari, while others remained consistent, as presented in *figure 8.2*.

Figure 8.2: *Web browser market share 2022 [2]*

Google chrome browser forensics

Most digital forensics investigators will come across Google Chrome in one of their examinations because it is the fastest and most widely used Web browser on desktops and laptops nowadays. Google Chrome is based on Chromium, a Google-developed open-source browser project. Because the Chromium project has yet to be released as a separate browser, we may refer to Google Chrome [4] as the project's public version. Vivaldi (**https://vivaldi.com**), Yandex browser (**https://browser.yandex.com**), Cent browser (**www.centbrowser.com**), and Opera browser (**www.opera.com**), to mention a few, are all based on the Chromium project. Most Web browsers based on the Chromium project will similarly store data; this allows examiners to use the same investigative techniques used to investigate Google Chrome to investigate these browsers, effectively making investigating Google Chrome a standard template for investigating most Chromium-based Web browsers. Chrome (created by *Google Inc.*) uses SQLite databases to store its configuration settings and user's private information, just like other Web browsers. Because these databases are files without extensions, you will not have any trouble opening them using the SQLite browser. To verify, go to the Google Chrome profile folder and make sure the option "**`All files (*)`**" is chosen, where you can see **`Events`**, **`Proxy`**, **`DNS`**, **`Sockets`**, and Domain security policy, as shown in *figure 8.3*.

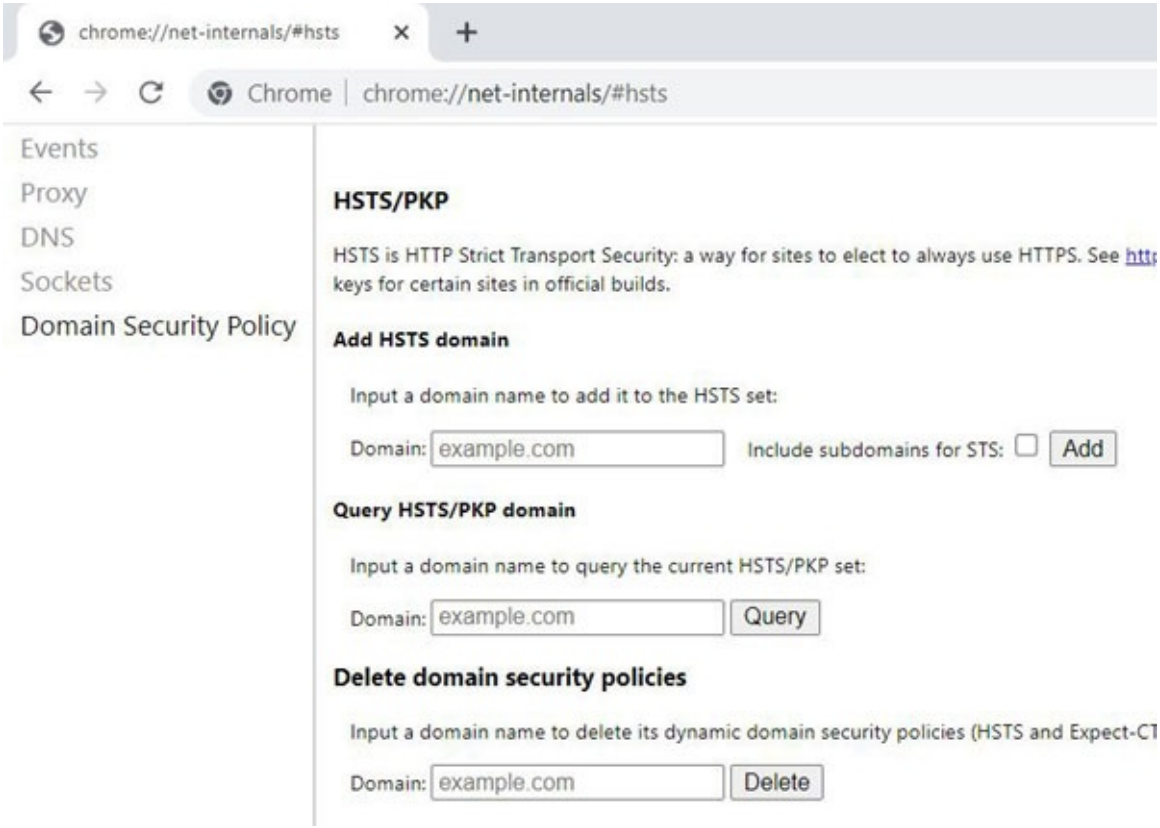


Figure 8.3: *Google Chrome internals [3]*

Google Chrome's setup settings, programs, bookmarks, and extensions are all stored in the profile. Google Chrome can have many profiles; however, as shown in *figure 8.4*, there is a default profile that can be located under **`C:\Users\<UserName>\AppData\Local\Google\Chrome\User Data\Default`**.

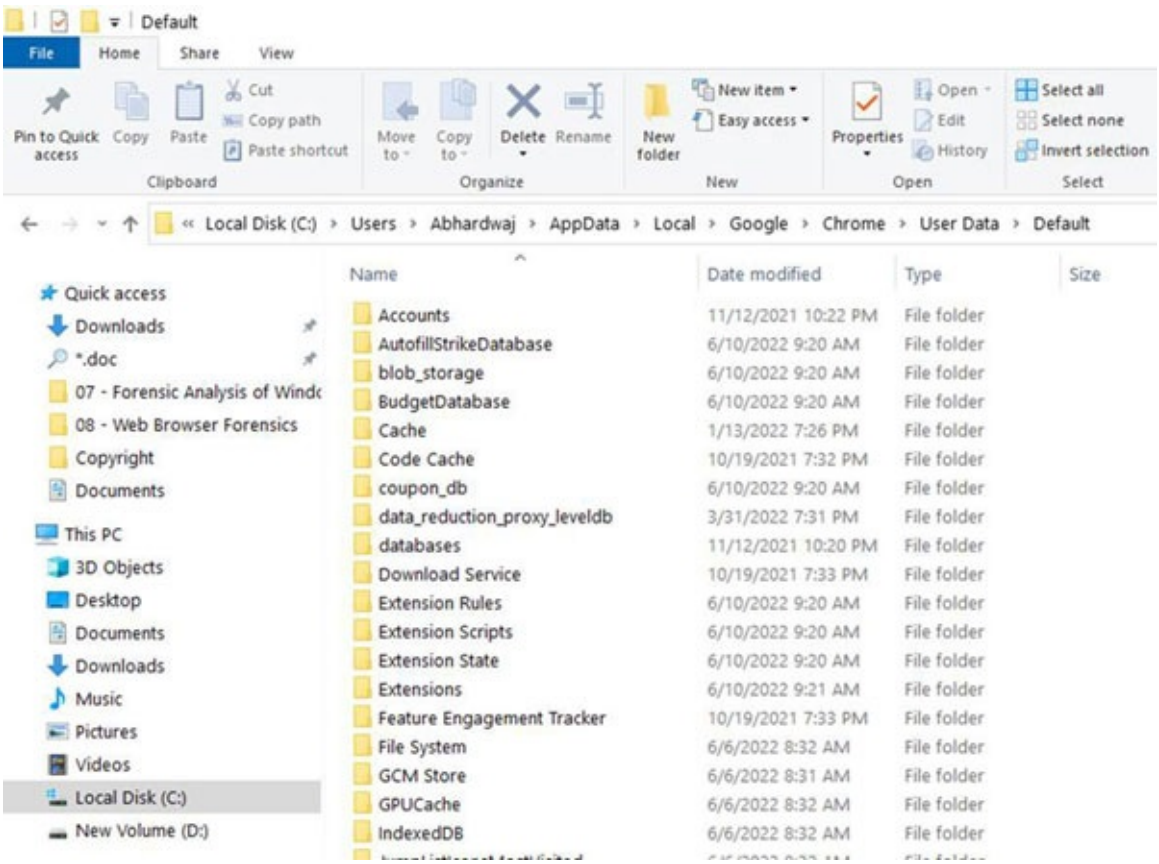


Figure 8.4: *Google Chrome user data*

If you have several profiles in Google Chrome, each one will have its folder with browser settings and

private data (passwords, browsing history, bookmarks, and so on) for the profile owner. Google Chrome does not provide any extra profiles with a name based on the user's login; instead, it gives them a generic name (for example, System Profile or Profile 1, Profile 2, and so on). Additional Chrome profiles can be located in the **`Users\UserName\AppData\Local\Google\Chrome\User Data\System`** Profile directory, as shown in *figure 8.5*. Then, in the resulting window, look for "Profile Path". Let us investigate the files stored within the Google Chrome profile(s) folder now that we know how to access it.

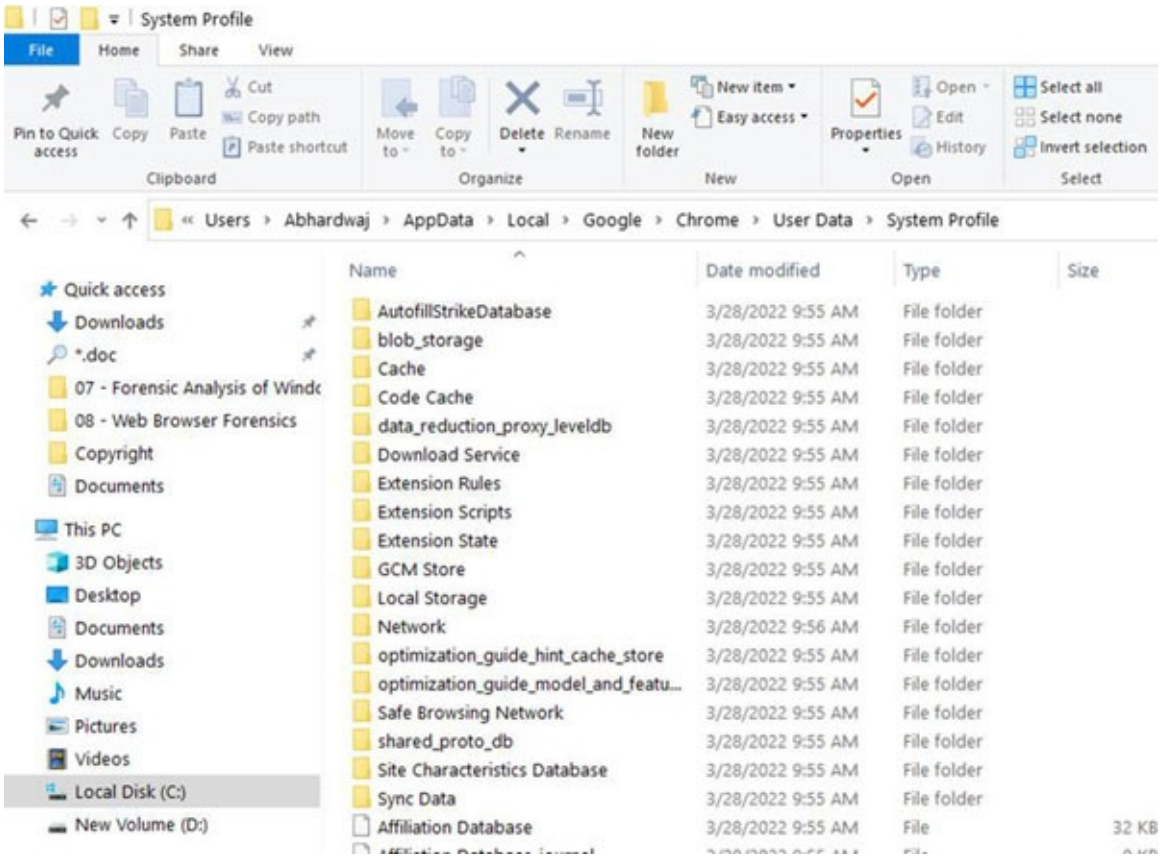


Figure 8.5: *Chrome system profile location*

To find the folder location of any Google Chrome profile (Refer to *figure 8.6*), start a Chrome window with the profile name/image in the top corner of the browser window, then type **`chrome:/version`** in the browser address bar.

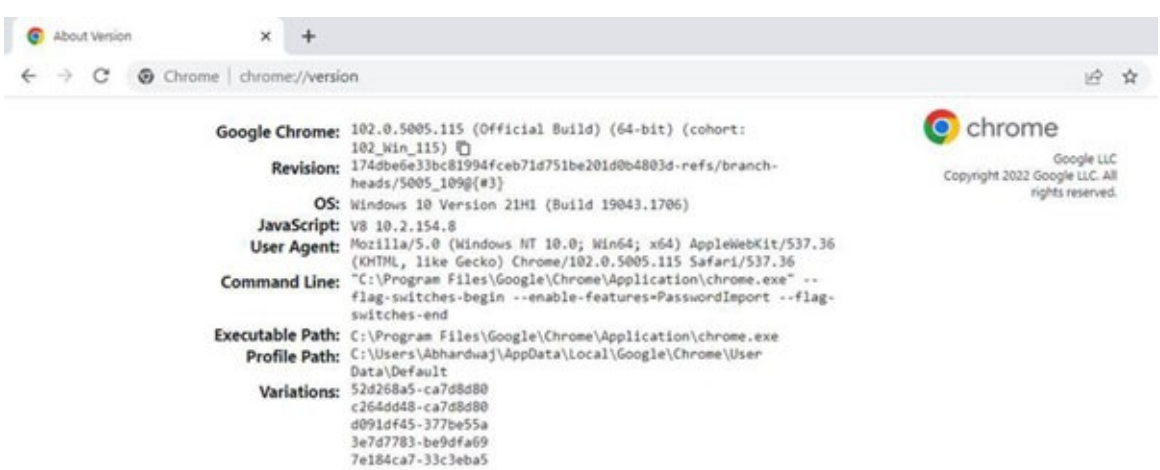


Figure 8.6: *Chrome version*

The **`History`** database file is placed under the Chrome user's profile and stores the user's browsing

history, downloads, keywords, and search terms. As seen in *figure 8.7*, this file may be inspected using DB Browser for SQLite. Go to the **`Downloads`** table under the **`Browse Data`** tab to find out when a specific file was downloaded and a wealth of other information on download history. The DB Browser for SQLite uses Google Chrome values stamps to display time information (also known as the Webkit format, which points to the number of microseconds passed since 00:00:00 UTC on January 7, 1971). Use the DCode tool to convert it to a readable format. ChromeHistoryView is a Nirsoft program that reveals Chrome history. This utility reads the Google Chrome Web browser's "History" file and can be downloaded from **www.nirsoft.net/utils/chromehistoryview.html**.

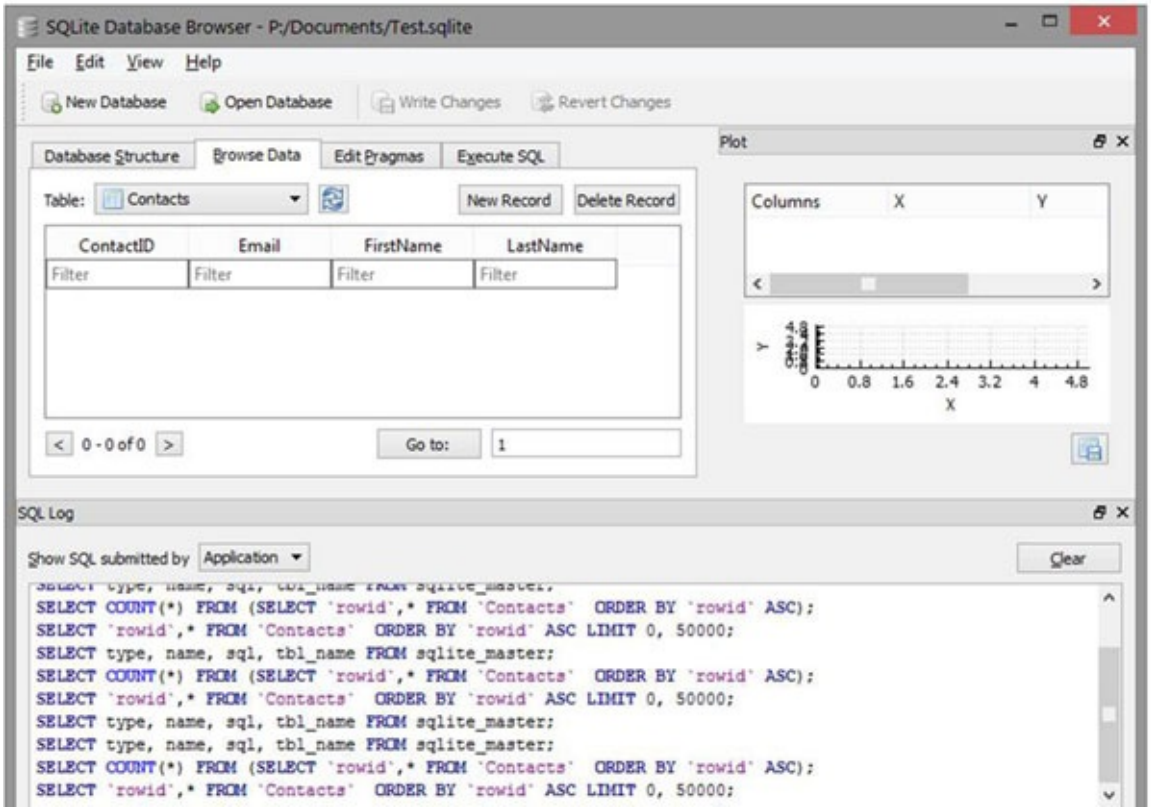


Figure 8.7: *SQLite portable visual schema*

When you use the internet, cookies are text files that contain little bits of information, such as a login and password, to identify your computer. HTTP cookies are a particular type of cookie that are used to identify specific users and enhance Web browsing. The server creates data in a cookie as soon as you connect. An ID that is specific to you and your computer is used to identify this data. Your computer and the network server exchange cookies, and when they do, the server reads the ID and knows what data to give you exactly. Google Chrome saves cookies information in the **`Cookies`** file under the Chrome user's profile; we can inspect the contents of the **`Cookies`** file with DB Browser for SQLite, just as we did with the **`History`** file previously, to see extensive information about stored Chrome cookies, as shown in *figure 8.8*.

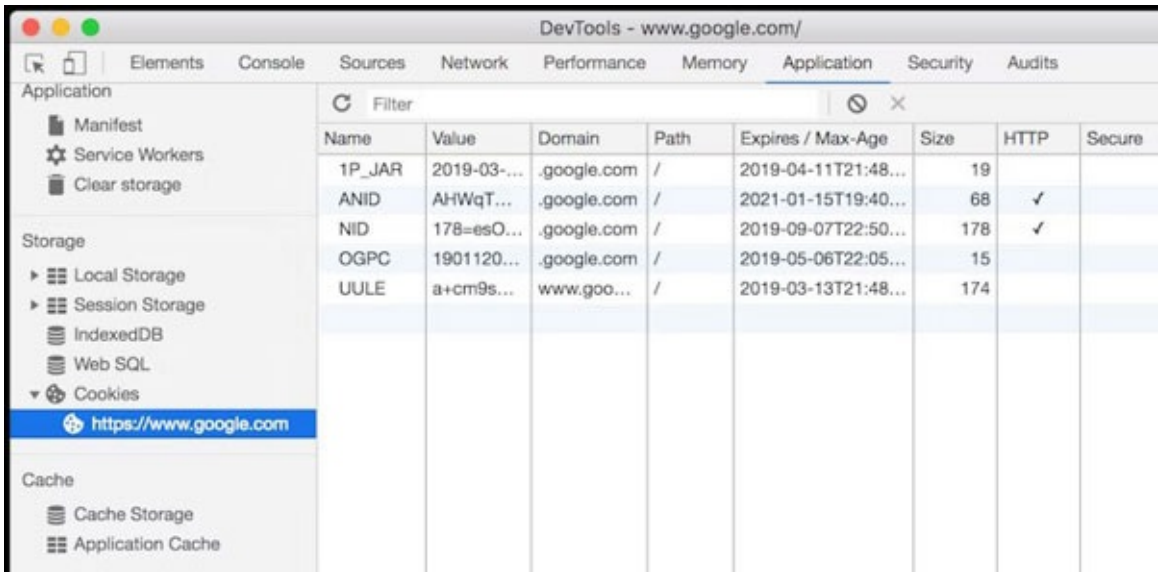


Figure 8.8: *View saved cookies*

Top sites and shortcuts

This database file contains a list of Google Chrome's most frequented websites. The information is saved in the thumbnail database, which has two tables: meta and thumbnails. This database is in charge of supplying Google Chrome's autocomplete feature while entering shortcuts (for example, a search keyword in the address bar and Web forms). There are two tables in this file: Meta and Omni box shortcuts. The autocomplete text and URLs are stored in the second table.

Login data

There are three tables in this database file: login, meta, and stats. For many online sites, the "login" database stores usernames and passwords (often encrypted), as well as other associated information. All usernames and passwords (in plain text) stored by the Google Chrome Web browser may be revealed using a portable tool from Nirsoft. Chromecast, as shown in *figure 8.9*, can be downloaded from **www.nirsoft.net/utils/chromepass.html**.

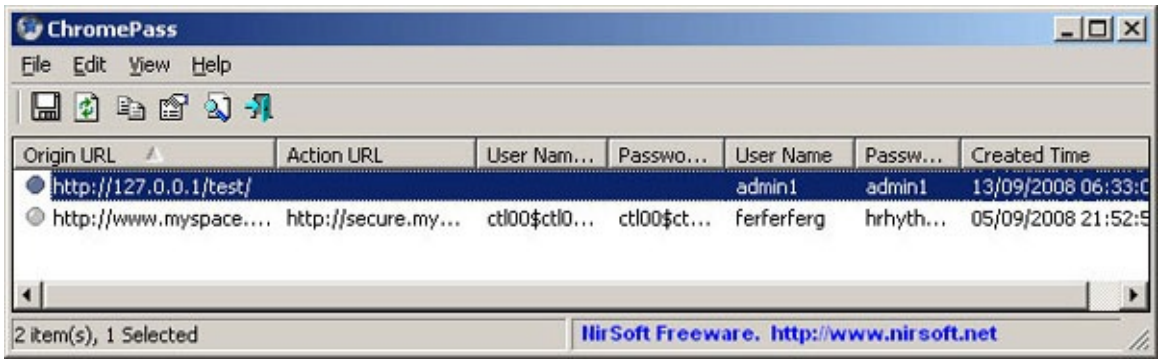


Figure 8.9: *ChromePass reveals all passwords stored by Google Chrome*

Web data

This function saves users' login credentials (without passwords, as Chrome moved login passwords to a separate file called "*Login Data*" in newer Google Chrome versions) so that the next time the user fills out a login form, searches keywords, and so on, Google Chrome will offer to autocomplete suggestions as

they type.

Bookmarks

A browser bookmark (sometimes known as a “favorite”) is a URL that a user saves for later access. In Google Chrome, the “Bookmarks database” file stores a user’s current bookmarks. We can open this file in Windows Notepad to see what is inside. We may use the *DCode* tool to transform the related “date added” value into a readable format to verify the date/time when a specific bookmark was added to Chrome; we have done this many times previously, as shown in *figure 8.10*.

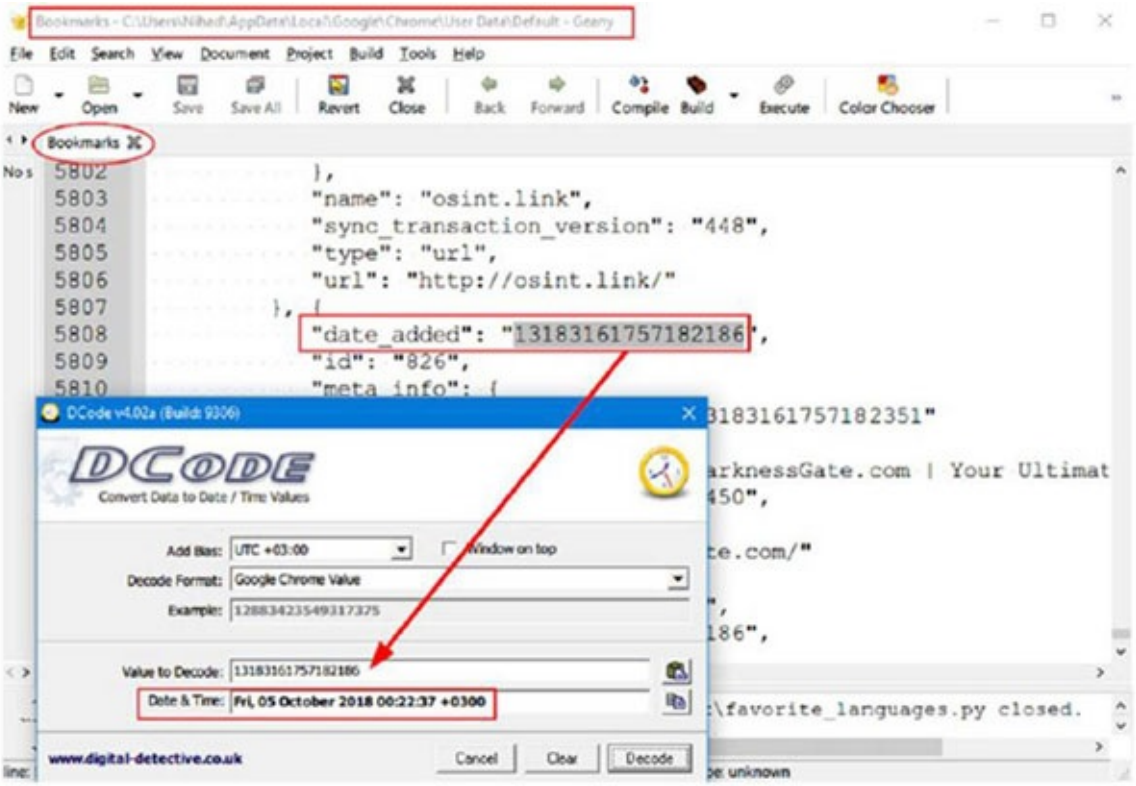


Figure 8.10: *Analyzing Google Chrome “Bookmarks”*

Bookmarks.bak

This database file contains recent Chrome bookmark backups; please notice that this file will be rewritten regularly each time Google Chrome is launched. This file has forensic importance because if a suspect deletes a certain bookmark(s) before shutting his or her Chrome browser, we can locate the deleted bookmark(s) in this file. To avoid overwriting this file, we should not run Google Chrome until we have a duplicate of it saved somewhere secure.

Cache folder

This folder stores frequently accessed static content, such as images and parts of HTML files, so that the next time a user visits the same website, the browser loads it faster because parts of the content are loaded from a local cache folder rather than downloading it from the Web site’s origin server. Using Nirsoft’s ChromeCacheView (**www.nirsoft.net/utils/chrome** cache **`view.html`**), we can automate the Google Chrome cache extraction procedure. This program examines the contents of the Google Chrome Web browser’s cache folder, as shown in *figure 8.11*, which is located in

`Users\UserName\AppData\Local\Google\Chrome\User Data\default\Cache`.

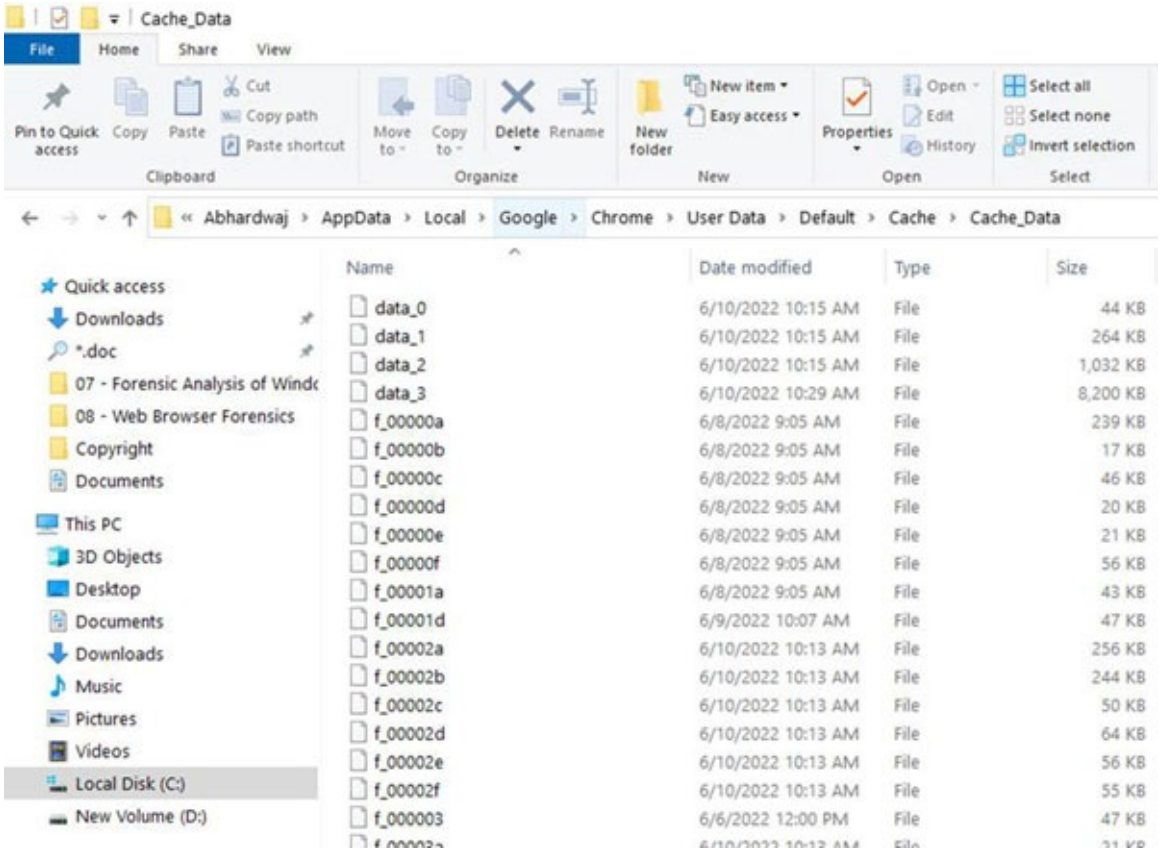


Figure 8.11: *View Google Chrome cache contents*

As we saw, Google Chrome stores quite a lot of personal information about its user. Investigating all these artifacts can help examiners to draw a complete timeline of a user's activities online in addition to understanding his/her intentions or interests by analyzing browsing history.

Mozilla Firefox Browser Forensics

Firefox [5] is a free, open-source Web browser created by Mozilla, and it is one of the most widely used browsers on the planet. Firefox does not use the Windows registry in the same manner that Internet Explorer does; instead, Firefox maintains its Web history, download history, and bookmarks in the **`places.sqlite`** database file. This file is located in your Firefox profile folder. By clicking the Windows key and going to **`%APPDATA%\Mozilla\Firefox\Profiles`**, as illustrated in *figure 8.12*, you can go to your profile. Your Firefox profile will display as a folder in the search results; click it to access it. You can also access the Firefox profile folder by pressing the Windows button + *R* and then typing the following in the Run window:

%APPDATA%. Click **`OK`** → Windows Explorer window will appear → Go to Mozilla → Firefox → Profiles

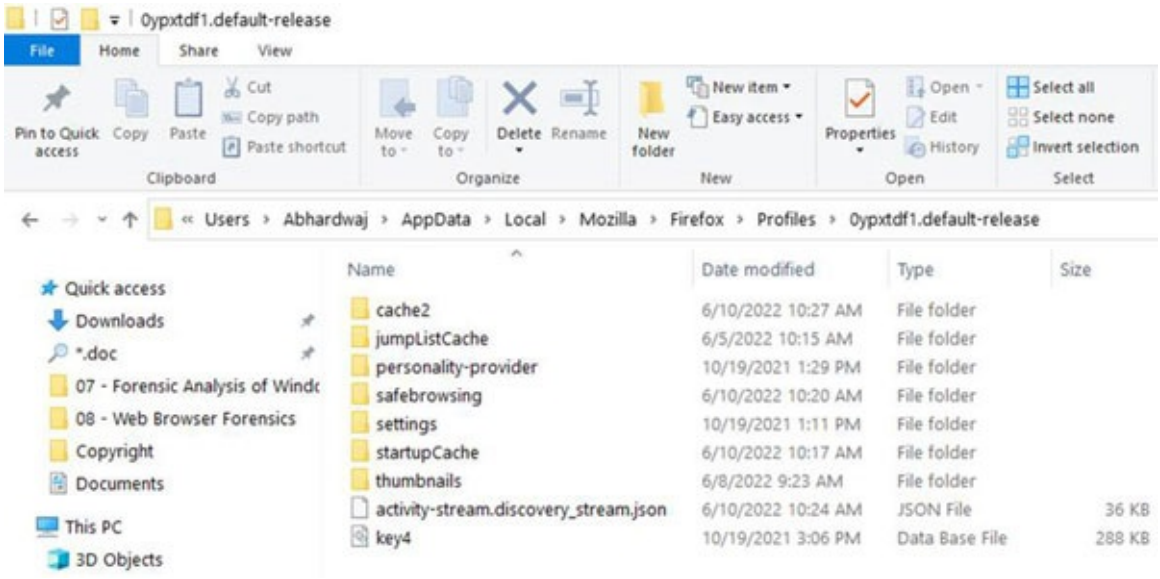


Figure 8.12: *Firefox browser folders*

To do Firefox Forensics, we may get information from the Mozilla Firefox browser's cache, which is located in the profile folder's cache folder. The cache information contains information on the user's browsing habits, bookmarks, and other pertinent information. Because we are focusing on Firefox Web browser forensics, let us go few relevant files briefly and present a few tools to help us automate our search.

- Bookmarks, visited websites, and download history is all stored in **`places.sqlite`**. The DB Browser for the SQLite application may be used to get information from the **`places.sqlite`** database file. We can use this tool to view target SQLite database tables and their content, just like we did before. The **`places.sqlite`** file is used by Firefox to keep all of a user profile's full history of visited sites. As shown in *figure 8.13*, the **`places.sqlite`** file is an SQLite database file that can be inspected using the SQLite database forensics tool.

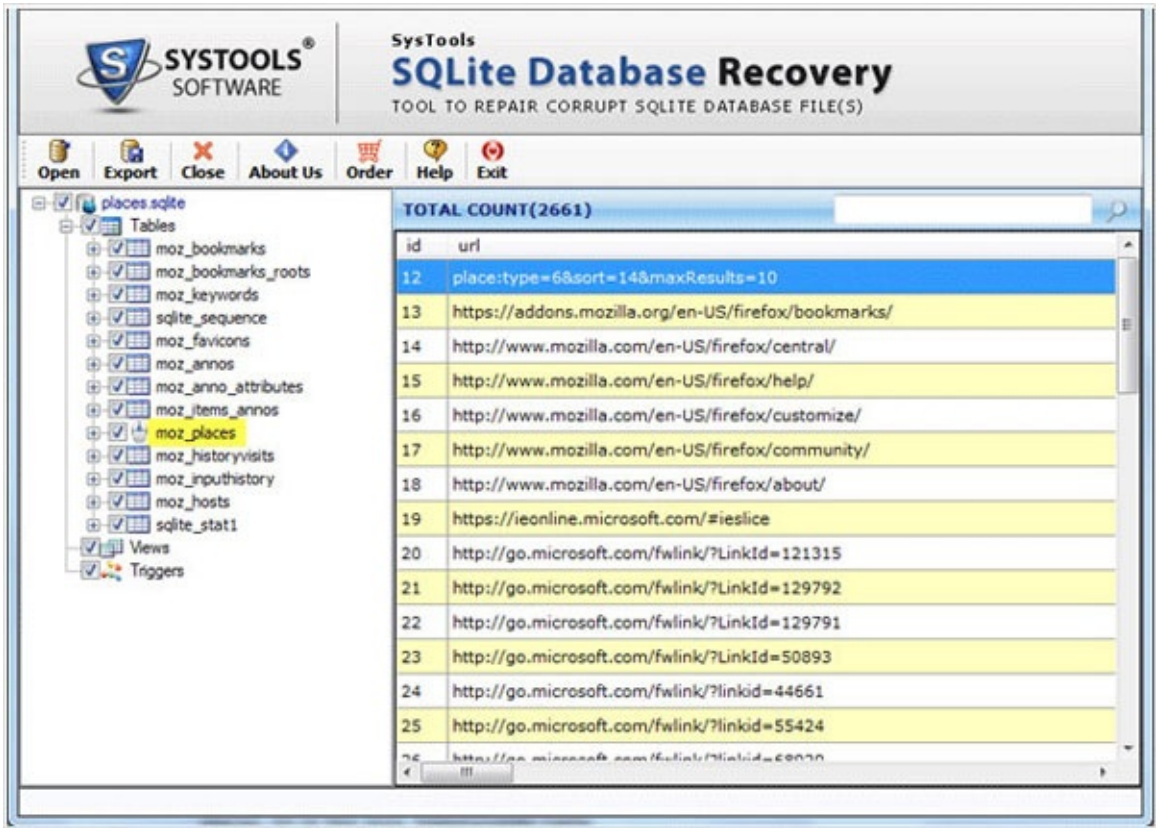


Figure 8.13: *Place.Sqlite tables and contents*

- **`Cookies.sqlite`**: Stores cookies left by previously visited websites (cookies are often used to save login usernames and passwords of previously visited websites and to preserve website preferences). The following tools can be used to retrieve information from the **`cookies.sqlite`** database file:
    - MZCookiesView (**www.nirsoft.net/utils/mzcv.html**). Displays all cookies stored in a Firefox cookie file; you can also export results into a text, XML, or HTML file.
    - DB Browser for SQLite.
- **`formhistory.sqlite`**: Stores your search keywords used in the Firefox search bar, and your searches are entered into Web forms.
- **`Key4.db`** and **`logins.json`**: Firefox keeps your passwords in this location. (The key database file was formerly known as **`key3.db`**; starting with Firefox version 58, the name was changed to **`Key4.db`**, but the **`logins.json`** file—which contains passwords in encrypted format—remains unchanged.) To view all usernames and passwords kept by Firefox, use PasswordFox, which can be downloaded from **www.nirsoft.net/utils/passwordfox.html**. If you run this application on the target system, it will show you the passwords for the current Firefox profile; if you want to see the passwords, see *figure 8.14*.

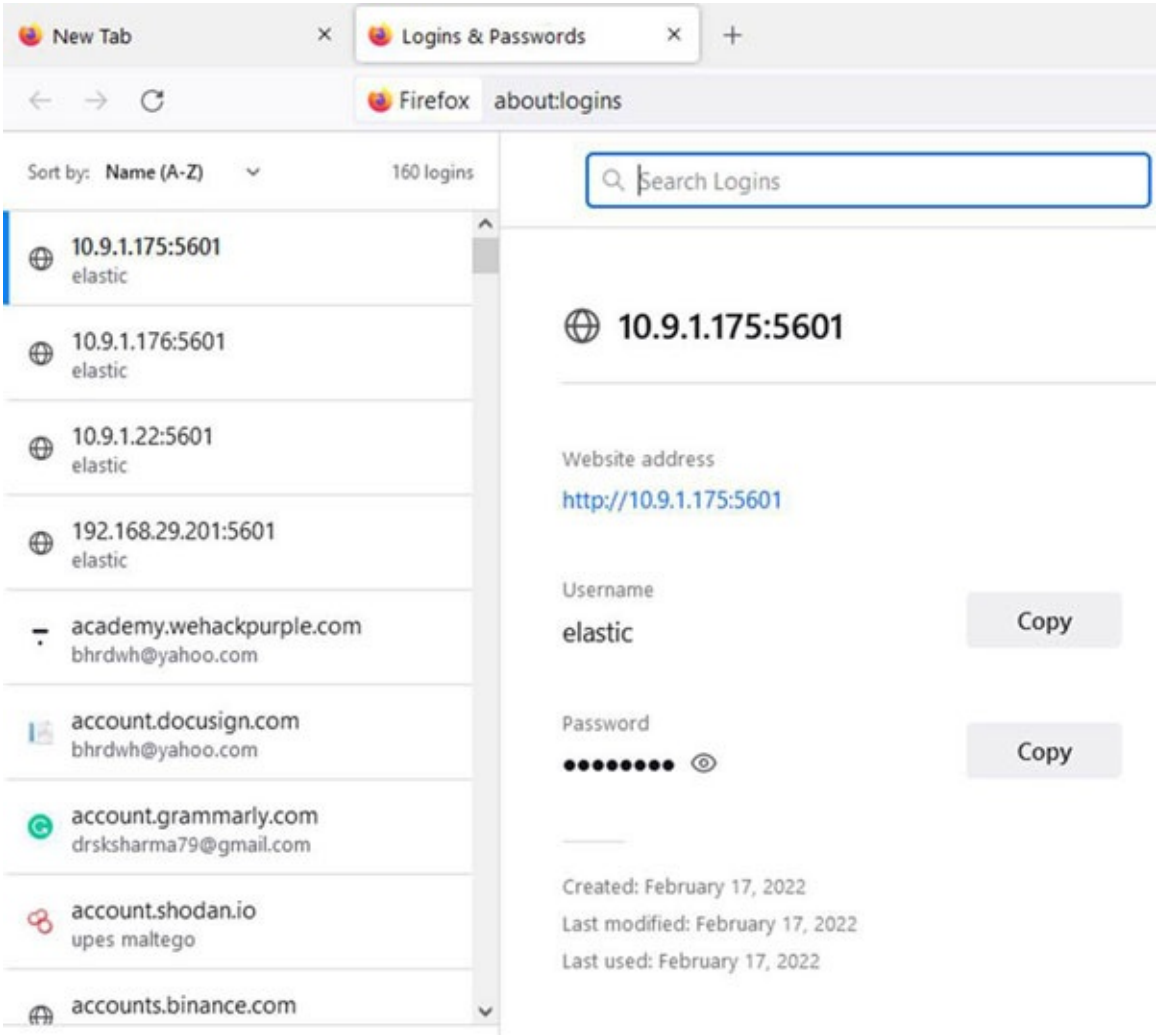


Figure 8.14: *Firefox passwords*

- **`permissions.sqlite`**: Stores Firefox permission for individual websites. For example, when

you allow a specific website to display pop-ups, Firefox saves this permission in this file. The same is true when installing an add-on from a particular website.

- **`search.json.mozlz4`**: Holds user-installed search engines.
- **`prefs.js`**: Stores Firefox preferences.
- **`addons.json`**: Views installed add-ons on Firefox.
- **Extension-data [Folder]**: Holds data generated by installed extensions (add-ons).

Microsoft Edge browser forensics

Microsoft Edge (codename Spartan) is the new default browser [6] for Windows 10 that replaces Internet Explorer. This is a lightweight Web browser that interacts with Windows 10's Cortana function, allowing users to do numerous activities (such as opening Web pages and conducting online searches) with just voice commands. We may expect more consumers to use Microsoft Edge instead of IE in the future; thus, understanding where this browser saves its data is critical for our forensics work. The Edge browser's configuration settings are stored in an ESE database, which can be found via **`Users\UserName\AppData\Local\Packages`**. *Figure 8.15* shows **`Microsoft.MicrosoftEdgexxxx\AC\MicrosoftEdge\UserDefault\DataStore\DBStorespartan.ed`**

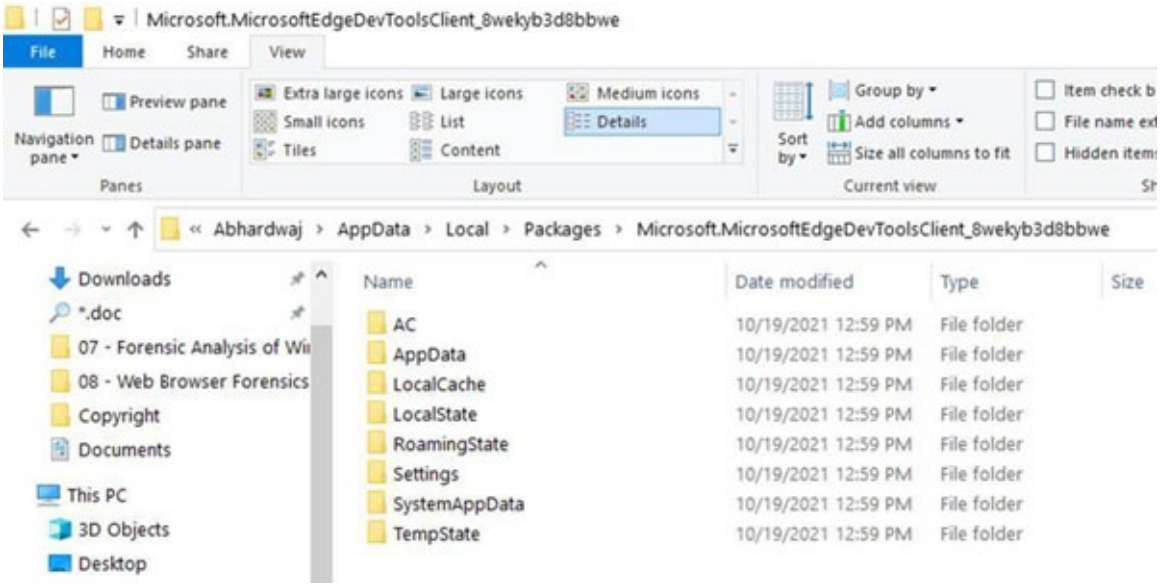


Figure 8.15: *Microsoft Edge folders*

Investigators may use Nirsoft's ESEDatabaseView (**www.nirsoft.net/utils/ese_databaseview.html**) to display data from the **`Spartan.edb`** database.

- Microsoft Edge cache content is stored at →
- \Users\ <UserName>\AppData\Local\Packages\Microsoft.MicrosoftEdge_*****\AC\#!001\MicrosoftEdge
- Microsoft Edge stores its browsing history in the same location (same database file) where IE version 10 and 11 store their data →
- \Users\<UserName>\AppData\Local\Microsoft\Windows\WebCache\WebCacheV01.dat
- The last browsing session of Microsoft Edge is stored at →
- \Users\

<UserName>\AppData\Local\Packages\Microsoft.MicrosoftEdge_****\AC\MicrosoftEdge\User\D

- Further analysis of Edge artifacts can reveal valuable forensics information; as we already saw, the valuable information is located in the Edge databases named **`spartan.edb`** and **`WebCacheV01.dat`** and in various locations inside its main folder, located at → **`\Users\<UserName>\AppData\Local\Packages\Microsoft.MicrosoftEdge_*****\`**

Other Web browser investigation tools

There are other general tools for investigating Web browser artifacts, mainly from Nirsoft.

- **`WebCacheImageInfo`** [7] can find and list all JPEG pictures with EXIF metadata saved in the cache folders of Internet Explorer, Firefox, and Google Chrome. As shown in *figure 8.16*, EXIF contains critical information about JPG photos, such as the camera type used to shoot the shot and the date and time the image was produced. Explore **www.nirsoft.net/utils/web** cache image **`info.html`** to get this utility.

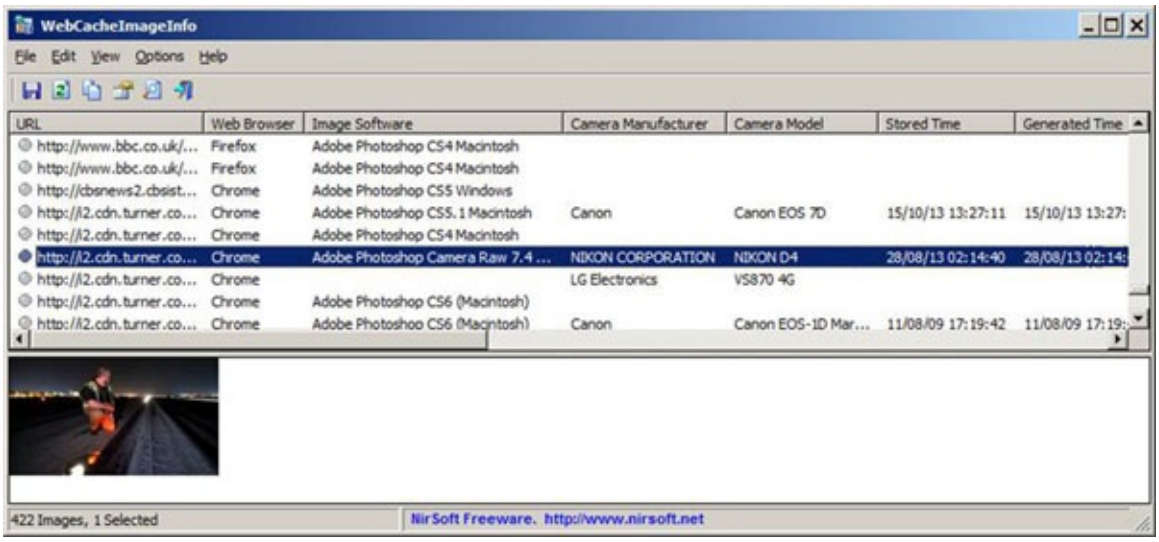


Figure 8.16: *WebCacheImageInfo*

- **`ImageCacheViewer`** [8] examines all main browser cache folders (IE, Firefox, and Google Chrome) and displays all pictures discovered therein, as seen in *figure 8.17*. Download this tool from **www.nirsoft.net/utils/image** cache **`viewer.html`**.

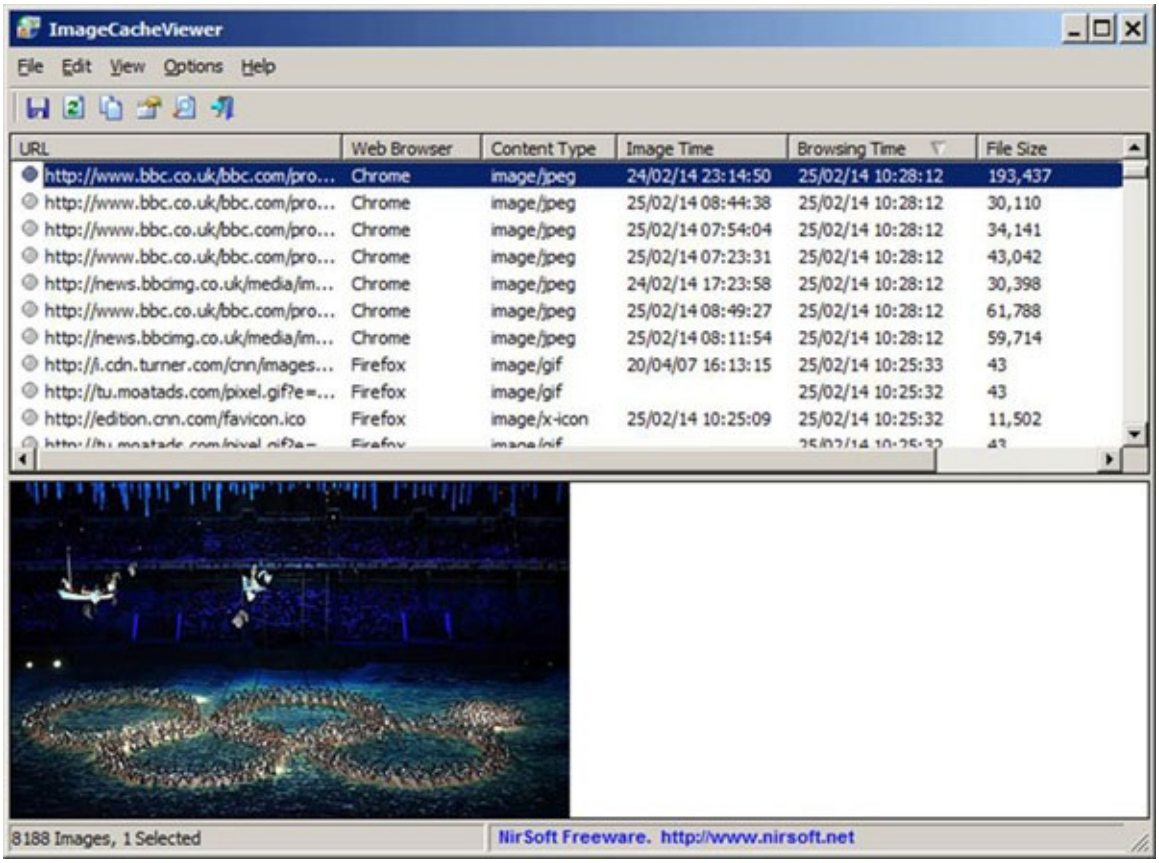


Figure 8.17: *ImageCacheViewer displays all cached images*

- All add-ons/extensions installed on all major Web browsers are displayed in **`BrowserAddonsView`** (Chrome, Firefox, and IE) [9]. If there are many profiles for Firefox and Chrome, the tool can display add-ons for all profiles, as shown in *figure 8.18*. This tool can be downloaded from **www.nirsoft.net/utils/web_browser_addons_view.html**.

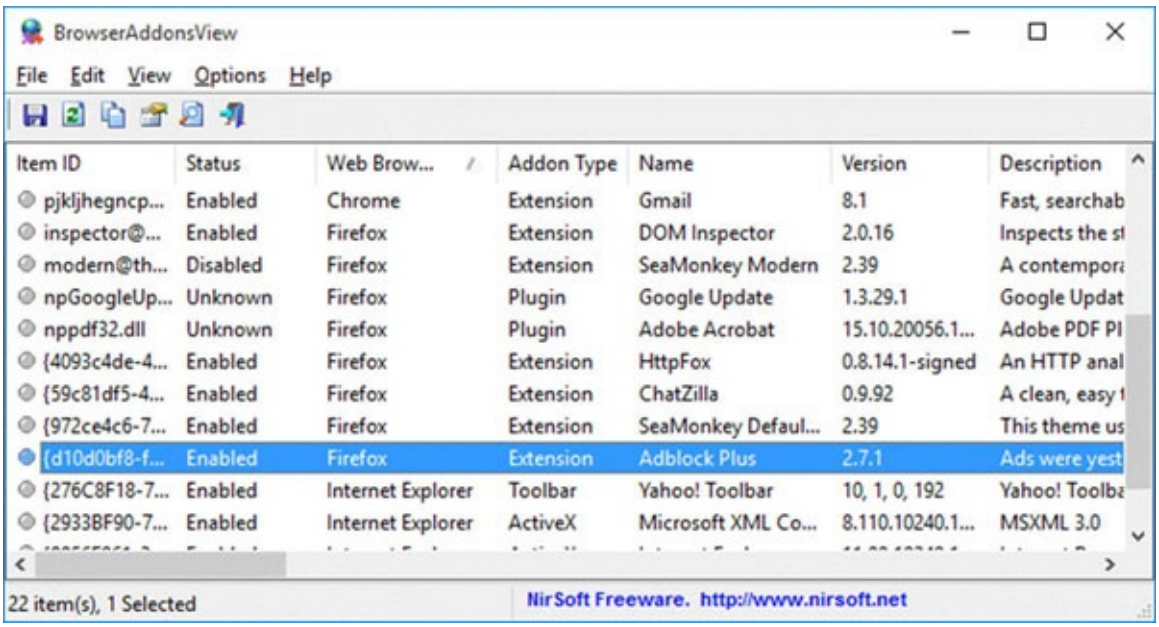


Figure 8.18: *BrowserAddonsView displays Web browser add-ons/plugins*

- As shown in *figure 8.19*, **`MyLastSearch`** [10] checks the online history in all main browsers (Chrome, Firefox, and Internet Explorer), as well as the cache folder, to obtain all prior search requests. This tool is useful for determining what a suspect is looking for at any given moment and what search engine can be used. It may be obtained from **www.nirsoft.net/utils/my_last_search.html**.

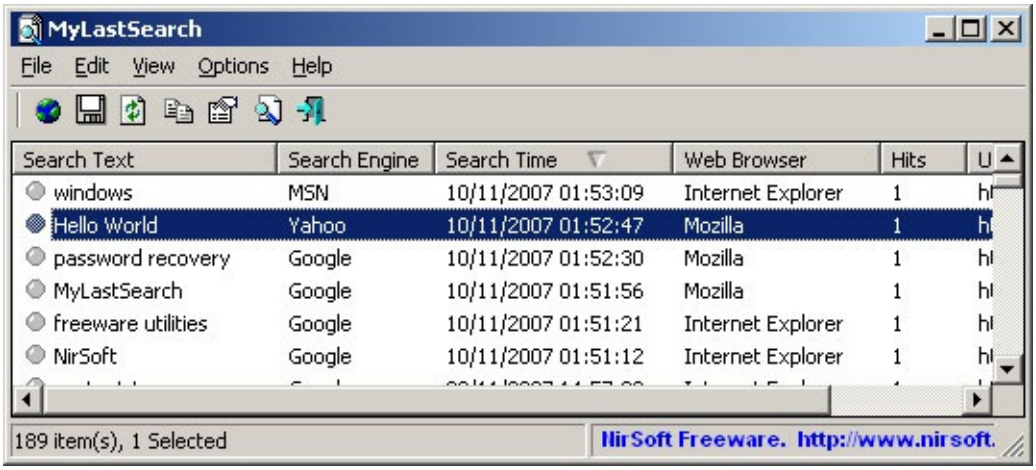


Figure 8.19: *MyLastSearch displaying cache and history of Web browser*

- As shown in *figure 8.20*, **`WebBrowserPassView`** [11] is a password recovery application that shows passwords saved in Internet Explorer (versions 4.0–11.0), Mozilla Firefox (all versions), Google Chrome, Safari, and Opera. This tool can be downloaded from **www.nirsoft.net/utils/web_browser_password.html**.

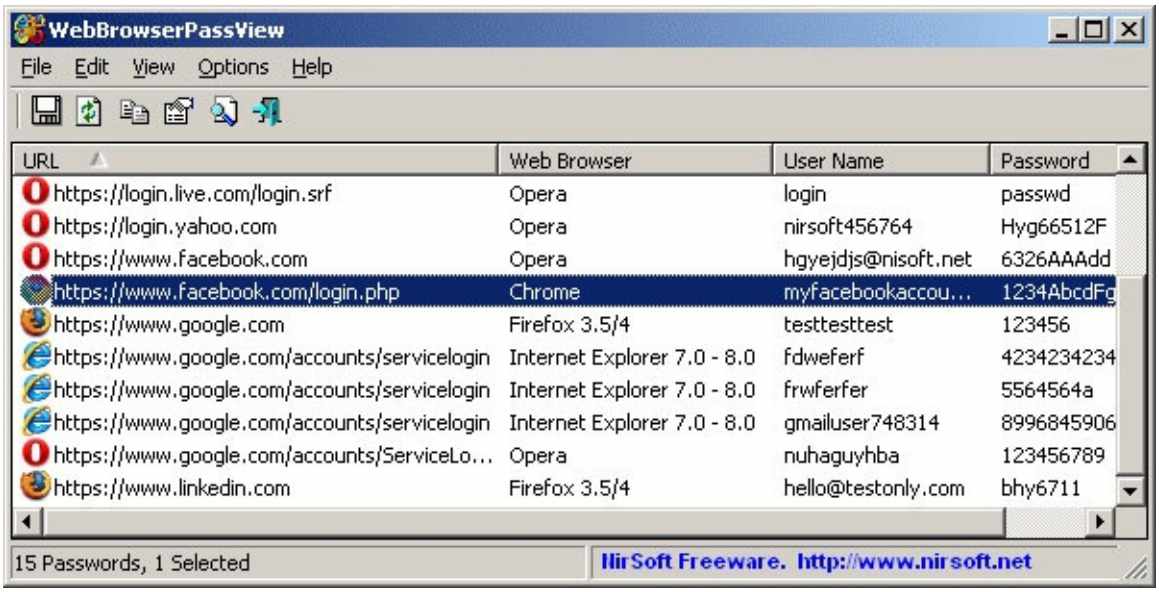


Figure 8.20: *MyLastSearch displaying*

- **`Web Historian`** [12] is a great Google Chrome browser extension tool for visualizing Web browsing history; it displays graphical circles of the number of days a website was visited (based on the time order of your website visits) and keyword search terms, as shown in *figure 8.21*, as well as the most active browsing hours of the day and days of the week. This can be installed from **www.webhistorian.org**.

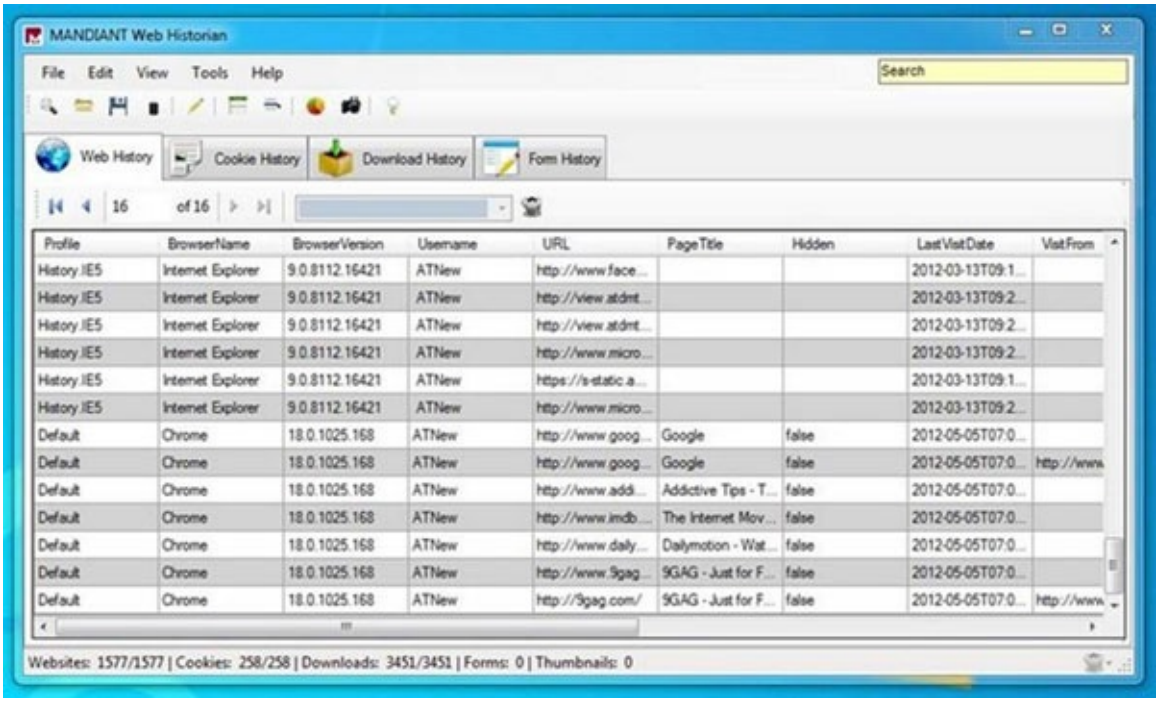


Figure 8.21: *Web Historian*

Conclusion

In this chapter, we discussed how to investigate the most used Web browsers—Google Chrome, Firefox, and IE/Edge—for forensic artifacts. The dependence in this chapter was on manual analysis, in addition to using some simple, free tools that can aid investigators in their forensics work for Web browsers. In the upcoming chapter, we will discuss investigating e-mail and e-mail crimes.

References

1. Statista. 2022. *Most popular internet browser versions 2021 | Statista*. [online] Available at: <**https://www.statista.com/statistics/268299/most-popular-internet-browsers/**> [Accessed 10 June 2022].
2. StatCounter Global Stats. 2022. *Browser Market Share Worldwide | Statcounter Global Stats*. [online] Available at: <**https://gs.statcounter.com/browser-market-share**> [Accessed 10 June 2022].
3. Support.google.com. 2022. *Where is the old "chrome://net-internals/#dns"—Google Chrome Community*. [online] Available at: <**https://support.google.com/chrome/thread/6943864/where-is-the-old-chrome-net-internals-dns?hl=en**> [Accessed 10 June 2022].
4. Google.com. 2022. *Google Chrome—Download the Fast, Secure Browser from Google*. [online] Available at: <**https://www.google.com/chrome/?brand=YTUH&gclid=CjwKCAjwtIaVBhBkEiwAsr7-c1ldO0--hADU1LJ_MWuAVshwPWkqYO7mF0G1A6CXvId-on7kuELhCxoC_q0QAvD_BwE&gclsrc=aw.ds**> [Accessed 10 June 2022].
5. Mozilla. 2022. *Download the fastest Firefox ever*. [online] Available at: <**https://www.mozilla.org/en-US/firefox/new/**> [Accessed 10 June 2022].

6. 2022. [online] Available at: <**https://www.microsoft.com/en-us/edge**> [Accessed 10 June 2022].
7. NirSoft. 2022. *Display images information stored in Web browser cache*. [online] Available at: <**https://www.nirsoft.net/utils/web_cache_image_info.html**> [Accessed 10 June 2022].
8. NirSoft. 2022. *View images in the cache of your Web browser*. [online] Available at: <**https://www.nirsoft.net/utils/image_cache_viewer.html**> [Accessed 10 June 2022].
9. NirSoft. 2022. *Web browser add-ons/plugins viewer*. [online] Available at: <**https://www.nirsoft.net/utils/web_browser_addons_view.html**> [Accessed 10 June 2022].
10. NirSoft. 2022. *MyLastSearch: View your search engine query in Google and others*. [online] Available at: <**https://www.nirsoft.net/utils/my_last_search.html**> [Accessed 10 June 2022].
11. NirSoft. 2022. *Recover lost passwords stored in your Web browser*. [online] Available at: <**https://www.nirsoft.net/utils/web_browser_password.html**> [Accessed 10 June 2022].
12. Historian, M., Security, C. and Historian, M., 2022. *Mandiant Web Historian Free Download*. [online] apponic. Available at: <**https://mandiant-web-historian.apponic.com/**> [Accessed 10 June 2022].

CHAPTER 9

E-mail Forensics

Introduction

With the use of internet increasing over the years, the number of e-mails sent and received globally has increased. Although roughly 306.4 billion e-mails were estimated to have been sent and received each day in 2020, this figure is expected to increase to over 376.4 billion daily e-mails by 2025. In the previous chapter, we discussed investigating Web browsers; now, in this chapter, we will learn about e-mail forensics. From a digital forensics viewpoint, we are concerned about finding and recovering e-mails from a suspect forensic image file/device, analyzing the e-mail header, extracting useful information from it like IP address and date/time when a particular e-mail was sent, and finally, tracing e-mail back to its origin (the sender).

Structure

This chapter presents and discusses the following in detail:

- E-mails around us
- E-mail communication steps
- E-mail protocols
- Examine e-mail headers
- Determine sender's geolocation and time zone

Objectives

Not many of the younger generations will remember when the postman was one of the only widely accessible means of sending letters long distance. As time passed, e-mail technology was invented and refined, which made letter sending much easier, faster, and more convenient. Today in 2022, approximately 333.2 billion e-mails are sent per day, which works out to well over 3.5 million e-mails per second. That is a lot of e-mails. As a result, those statistics prove that e-mail is the most used communication medium for business interactions. E-mail is also one of the cheapest marketing tools used by all sorts of businesses—from small brands to industry giants.

E-mails around us

Key daily e-mail statistics for 2022 are as follows:

- A total of 99% of e-mail users check their inboxes at least once a day
- In total, 333.2 billion e-mails are sent per day

- Over 3.5 million e-mails are sent per second
- A total of 20% of Americans check their e-mail inbox more than five times a day
- A total of 85% of all e-mails are spam e-mails
- In total, $7,000 was earned per day by sending bulk e-mails
- A total of 100 million spam e-mails are blocked by Gmail every day
- A total of 31 billion e-mails bounce per day
- A total of 58% of Gen Z check e-mail multiple times in a day
- A total of 66% of Gen Z receive less than 20 e-mails daily
- Plain text e-mails got a 42% open rate compared to the HTML e-mails

E-mails have become the primary means of communication in today's digital age; for instance, it is rare to see a person who owns a computer, smartphone, or tablet without having an active e-mail account. This is a huge number already and yet is continuing to increase steadily as more people around the world enter the digital era. With the prominence of the internet, e-mails have emerged as the most popular application for business communication, document transfers, and transactions from computers and mobile phones. With this emergence, e-mail security protocols have also been implemented to mitigate the illegitimate actions of criminals, such as business e-mail compromise, phishing e-mails, and ransomware. However, there comes a time when specific e-mails need to be examined and data extracted for legal matters such as civil litigation and legally aided criminal investigations. This is where e-mail forensics is applied. Digital evidence in the form of e-mail data can be crucial in civil and criminal cases. However, be sure it is extracted in the correct manner using e-mail forensics.

There are two standard methods to send/receive e-mails: the first one is using an application to send and receive e-mails (for example, e-mail clients like Mozilla Thunderbird), and the second is using a Web interface browser to access your e-mail account (for example, Gmail, Yahoo, and Outlook). E-mail can be mainly abused through the following:

- Sending spam and phishing e-mails
- Using it to commit e-mail harassment crimes
- Invading other user's privacy by stealing their e-mail login credentials

E-mail forensics is the analysis of e-mail and its content to determine the legitimacy, source, date, time, the actual sender, and recipients in a forensically sound manner. The aim of this is to provide admissible digital evidence for use in civil or criminal courts. Before we begin our discussion on how to analyze and track e-mails, let us give some important prerequisite information regarding how e-mail communications and protocols.

E-mail communication steps

To demonstrate how e-mail delivery works refer to *figure 9.1* in the following example:

- Neha composes an e-mail (*neha@ignou.com*) using her computer for Karan (*karan@darkgate.com*); the message first needs to be sent to Neha's sending SMTP server (*smtp.ignou.com*) using the SMTP protocol.
- The sending server performs a lookup to find the mail exchange record of the receiving server

(*darkgate.com*) through the DNS protocol on DNS *mx.darkgate.com* for the domain *darkgate.com*.

- The DNS server responds and gives the mail exchange server mx.darkgate.com for the domain *darkgate.com*.
- Now, the sending server will establish an SMTP connection with the receiving server and send the e-mail to Karan's mailbox on the receiving server.
- The receiving server will receive the incoming e-mail and store it in Karan's mailbox.
- Karan can either download the e-mail message from his mailbox on the receiving server into his e-mail client (for example, Mozilla Thunderbird) on a local machine using POP3 or IMAP protocols, or he can use Webmail (through a Web browser) to read the e-mail directly on the receiving server.

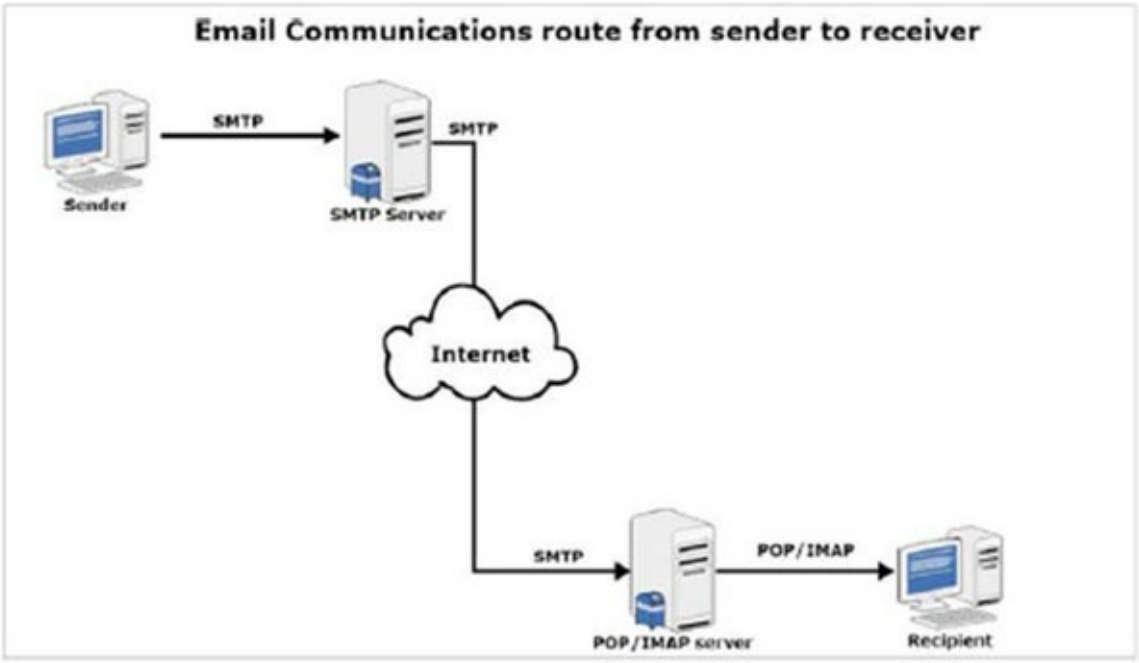


Figure 9.1: *E-mail communication steps*

E-mail protocols

In the previous section, we have mentioned many names of e-mail protocols that facilitate e-mail delivery. *Table 9.1* lists the main protocols used in e-mail communication and the role of each one.

E-mail protocol	Role performed
SMTP	Simple Mail Transfer Protocol: transfers e-mail messages from sender (client) to destination (servers) to other servers in between.
POP3	Post Office Protocol v3: is used by sender clients to down incoming e-mails from their mailbox on the server to their local systems/hard drive. Outlook or Thunderbird is used as the client application.
IMAP	Internet Message Access Protocol: similar to POP3, but this allows storing a copy of the e-mail message on the server that is being downloaded.
HTTP (S)	Hyper Text Transfer Protocol: is the Web browser interface used by the client when accessing Web-based e-mails, for example, Outlook Web Access, Gmail, and Yahoo.

Table 9.1: *Common e-mail protocols*

Examine e-mail headers

When examining e-mails for forensic information (for example, to see where the e-mail comes from), the needed information is already stored within it, specifically in the e-mail header section. An e-mail header stores a wealth of forensically useful information about an e-mail under investigation, such as the path it took over the internet to arrive, stop points/delays made during e-mail delivery, and the IP address of the machine that sent this e-mail, in addition to the client (for example, e-mail program) who sent this e-mail and the type of OS used (in some cases). Note that most of the information (including the technical information) in the e-mail header can be forged! Tech-savvy criminals can conceal the origin of their e-mails and even make it similar to an original e-mail that they are trying to reproduce (for example, phishing e-mails); however, the role of a forensic examiner is to gather the information in the e-mail header and examine it thoroughly, as it can lead to something useful for solving the case at hand.

Reveal header information

Before examining e-mail headers, viewing e-mail headers using popular Webmail services (Gmail, Microsoft Outlook) and e-mail clients (Thunderbird and MS Outlook) is presented as follows.

View Gmail headers

To display Gmail headers, access the target Gmail account using a Web browser → open the e-mail whose header you want to view. Next to Reply → click the down arrow (see *figure 9.2*) and click **`Show original`**.

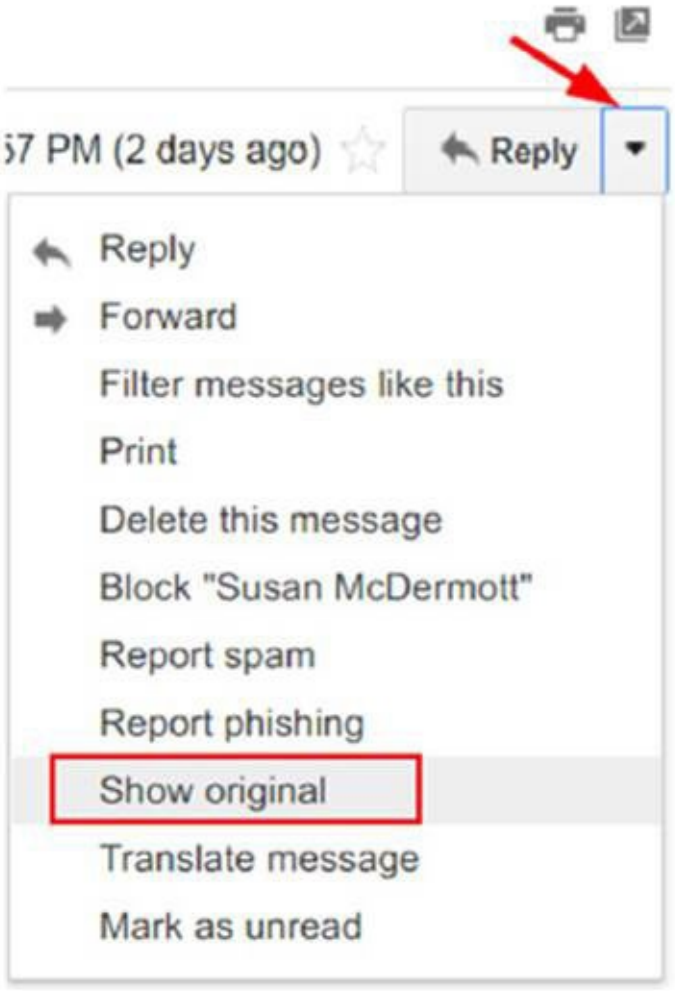


Figure 9.2: *Gmail message header*

View Outlook mail header

To display the Outlook Webmail header, access the Outlook account using your preferred browser. Open the e-mail whose header you want to view, and next to **`Reply`** | click the down arrow in the top right-hand corner of the e-mail. Click on **`View message source`** (see *figure 9.3*).

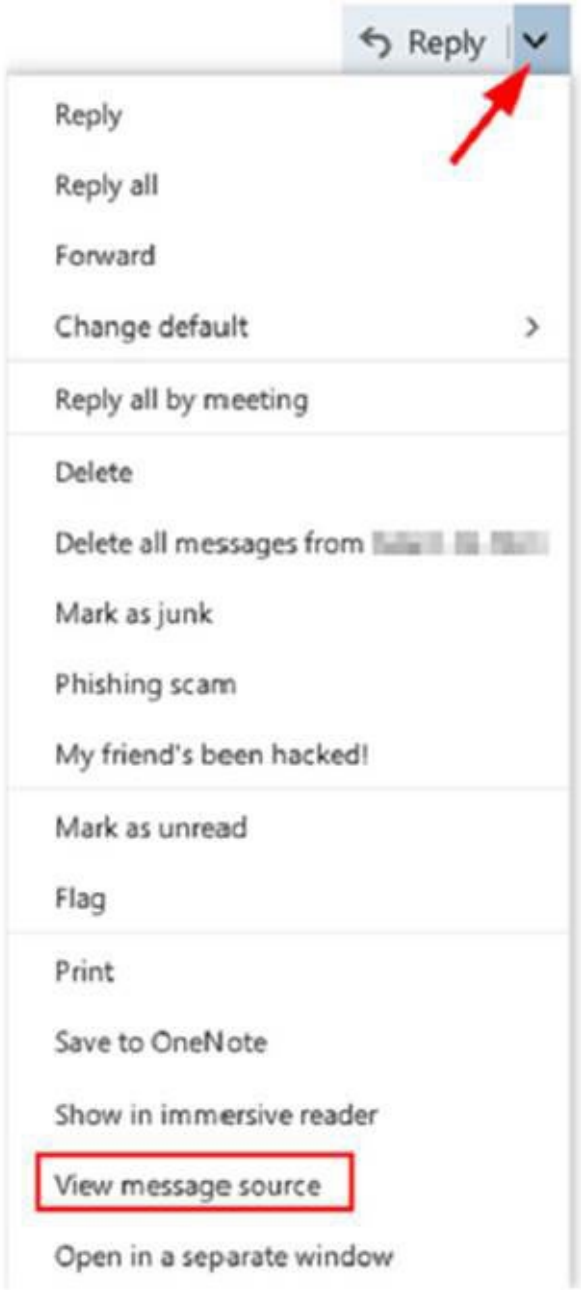


Figure 9.3: *Outlook mail message source*

View Mozilla Thunderbird headers

To display e-mail headers using the Mozilla Thunderbird e-mail client, open Thunderbird and then open the message whose header you want to investigate in a new window by double-clicking over it. Move to **`View`** | **`Headers`** | **`All`**. Another option to view the header is to open the message in a new window, click on "**`More`**" button on the top right message window and select **`View Source,`** as presented in *figure 9.4*.

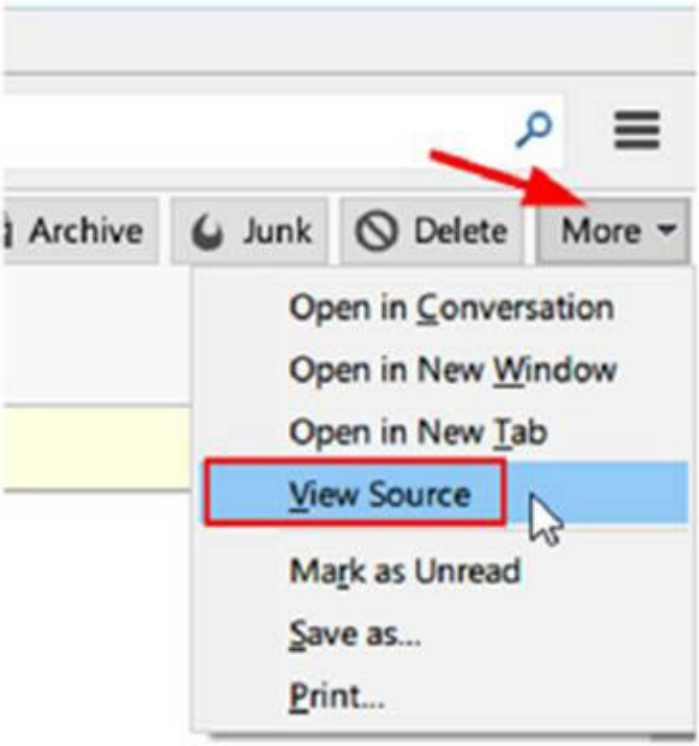


Figure 9.4: *Mozilla Thunderbird e-mail client header*

View Outlook mail client header

To display a full e-mail header using the Microsoft Outlook mail client, open MS Outlook and go to the e-mail whose header information you want to view. Double-click this e-mail to open it in a new window. Then click **`File`** | **`Properties`**. The header info is located in the **`Internet headers`** box, as presented in *figure 9.5*.

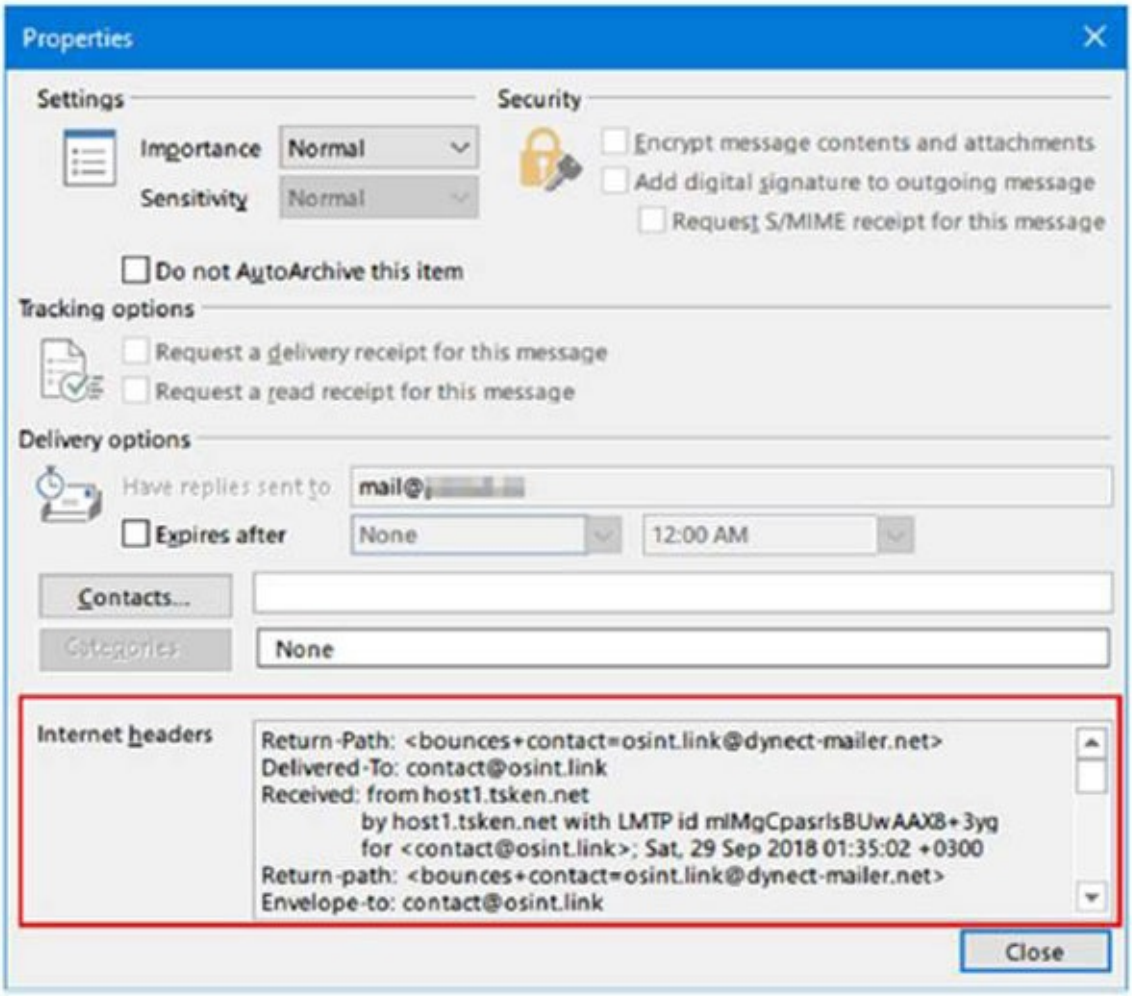


Figure 9.5: *MS Outlook e-mail client header info*

Analyzing e-mail headers

Now that we know how to reveal an e-mail header, we can begin analyzing it. Keeping in mind that the preferred method to read an e-mail header is from bottom to top. *Figure 9.6* is a sample e-mail header from a message received using the Gmail service.

```
Delivered-To: bhrdwh@gmail.com
Received: by 2002:a05:7110:2051:b0:17b:82b6:92a4 with SMTP id y17csp4915825gea;
        Wed, 1 Jun 2022 04:03:15 -0700 (PDT)
X-Google-Smtp-Source: ABdhPJx/bxm7AhXeLuqNRZo/Sl54oNnNiWfBXQHKJBTY+w3+Wy2428w76x8IpAPW4qZK2Q/AWhY+
X-Received: by 2002:a25:bc83:0:b0:65c:deb1:8c7 with SMTP id e3-20020a25bc83000000b0065cdeb108c7mr15739920ybk.451.1654081395842;
        Wed, 01 Jun 2022 04:03:15 -0700 (PDT)
ARC-Seal: i=1; a=rsa-sha256; t=1654081395; cv=none;
        d=google.com; s=arc-20160816;
        b=E5g0/KTgftDW6Bp6G1NnWxCX4ntJiGhS+4oPNx6vujr9TzIGiBhTiFKGkdL7IMxqGd
         IOQV7WwHLAIEN1xmuDOq/lw6FC+oqMkEOMgvz01aXAeO/Besd2oLK0/9evb59NtwXzKy
         /A0hUv3YtjraE4W1CaB/ulEryX5lLHVGfDhfW/yL8XteNNTp9pfiWnQs50Z3bfgzUSpA
         dxfqfRurU5F++OfUA1CDhiCXEkHfxkm0VCXMmF3kfSlc87OvD365D+vxcBbRgP5h3JU2
         jgjJW57Q7J3YS7Og4sgCBow4TS1uoBUGWcTpImu4hKO1W/VMRkPfS4nCBWDFEik+0zSk
         jdmQ==
ARC-Message-Signature: i=1; a=rsa-sha256; c=relaxed/relaxed; d=google.com; s=arc-20160816;
        h=mime-version:list-unsubscribe-post:list-unsubscribe:list-id
         :feedback-id:message-id:date:to:reply-to:from:subject:dkim-signature
         :dkim-signature;
        bh=/xghABQ2OO+Glmm7kHatzUxkhc1R10acvYI/MGQiJXM=;
        b=NLQqRhcGbtfjSsJ8ycX2sBHtef9bDoPVVEkDRNqfmz7yTQBJca6FootOjqato4Cvtc
         RI75axkVJMJES6Z5eOjrPwGrietzxyirRJ7EHDsZikHFSIUj1gatih+bkANsVKHV7Bvb
         96XcFdwkFxe0PAYRKCmQGRc5fAH0bMnlndHw0rjQgX04+nxG62TcUxSgUsHmp3uaOAkp
         vpBnaesSgkcglFLbC0j1DZ44ljzjMsKf+v6Mt5MqcGTfgq+xtTo5kpif3mF1dxgmOCwW
         /UQOeVxs+80omv28TUXLvOF9RvPc/Qjj/FsaFSma6KZI4825gNW3vJmiuyh6WRtwm+M2
         XaKg==
ARC-Authentication-Results: i=1; mx.google.com;
       dkim=pass header.i=@gmail.mcsv.net header.s=k1 header.b=l3ncpqQ1;
       dkim=pass header.i=@os.me header.s=k1 header.b=Iq3n8yEf;
       spf=pass (google.com: domain of bounce-mc.us3_121135914.4739978-4e5456ea05@mail65.sea91.rsgsv.net designates
148.105.15.65 as permitted sender) smtp.mailfrom=bounce-mc.us3_121135914.4739978-4e5456ea05@mail65.sea91.rsgsv.net
Return-Path: <bounce-mc.us3_121135914.4739978-4e5456ea05@mail65.sea91.rsgsv.net>
Received: from mail65.sea91.rsgsv.net (mail65.sea91.rsgsv.net. [148.105.15.65])
        by mx.google.com with ESMTPS id 64-20020a2519430000000b0065ce3999848s11473008ybz.30.2022.06.01.04.03.15
```

Figure 9.6: *Sample e-mail header; always read it from bottom to top*

When an e-mail travels through the internet, each mail server the e-mail passes through will add a piece of information to the header, so the preceding e-mail header screen capture can contain more information about the e-mail client and OS used to send the message. See *figure 9.7* for a partial e-mail header sent from a Mozilla Thunderbird e-mail client using a Windows 10 machine.

```
Date: Sun, 7 Oct 2018 00:22:02 +0300
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:45.0) Gecko/20100101
 Thunderbird/45.8.0
```

Figure 9.7: *Mail header*

An e-mail header can expose additional information about the message, such as the name and version of the e-mail client used to create and send the message, as well as the operating system used to compose and deliver the message. Manually analyzing e-mail headers might be difficult for novices, but there are many programs and online services that can extract important information from e-mail headers for you. Let us start using the Message header, a free online tool available at **https://toolbox.googleapps.com/apps/messageheader**.

To use this tool, copy the target e-mail headers and paste them into the **`Paste e-mail header here`** box before clicking **`Analyze the header above`**. The tool will examine the message header and display the names of all attachments, as well as the path the message followed from the originator to the receiver (see *figure 9.8*) and any delays that may have occurred during delivery.

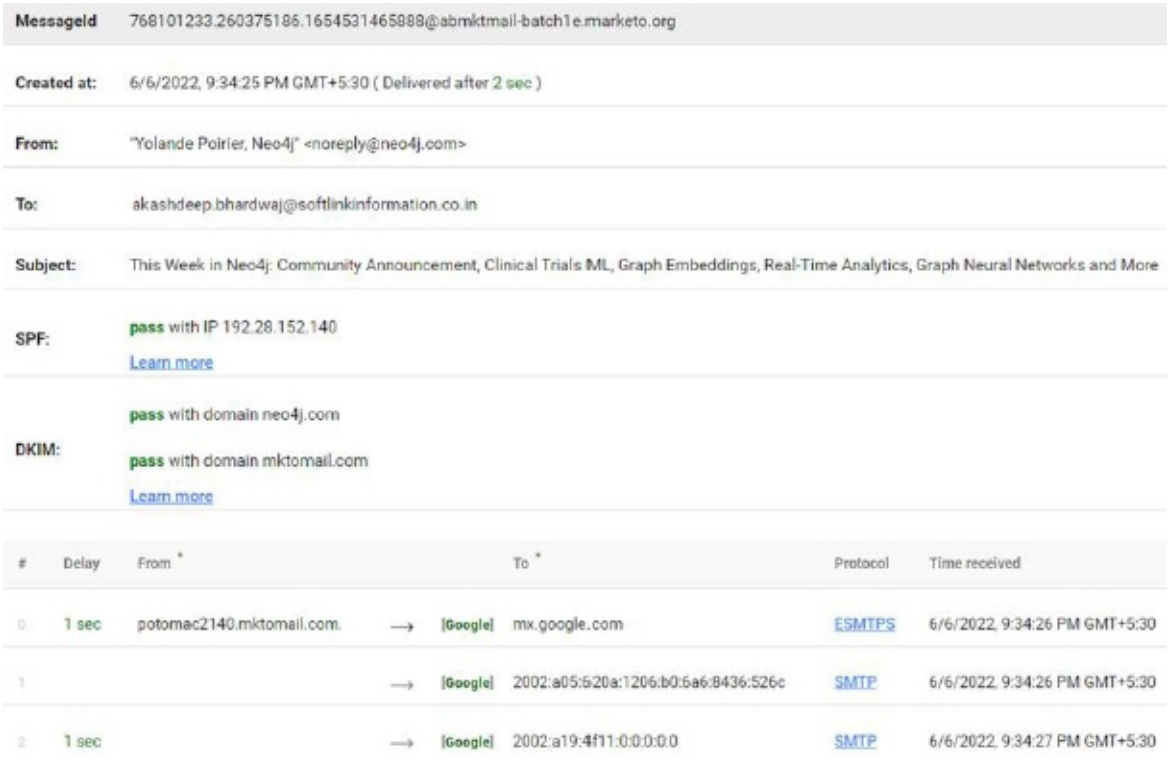


Figure 9.8: *Message route from sender to receiver revealed by the Message header*

There are various other tools to analyze e-mail headers; some of the most popular are as follows:

- E-mail header analyzer

  Access at **https://mxtoolbox.com/EmailHeaders.aspx**

  This is an online tool for parsing e-mail headers, making them human-readable. To use this tool, just go to the website and paste the target e-mail header info. The result will show—among other info—the path this e-mail took over the internet in addition to any delays that may have happened,

as illustrated in *figure 9.9*.

Headers Found

Header Name	Header Value
Delivered-To	akashdeep.bhardwaj@softlinkinformation.co.in
X-Google-Smtp-Source	ABdhPJz4knDQi6dmCwGngggDWfqi04FrXH20lyzWcR1K9f6kUqKOZvry4rd+N5WqhTgT1tXHwMP
X-Received	by 2002:a05:620a:1206:b0:6a6:8436:526c with SMTP id u6-20020a05620a120600b006a68436526cmr14540071qkj.669.1654531466660; Mon, 06 Jun 2022 09:04:26 -0700 (PDT)
ARC-Seal	i=1; a=rsa-sha256; t=1654531466; cv=none; d=google.com; s=arc-20160816; b=psPUB2y32754Sys9mgZ2BW62qM3pd1O+QCpaPGqHGimNLSGZJGXxrSnefKokkcUO5 eat6CjlRZ9uFQ28UFhniFeVy5O5Ej4DetgOZxlM9HtLRZJE/KKRkeyae38JztJ30EWkM S8MOJZJ/hh+mr9mQsmHf6h6McEwAV+0o6kzk1fC+wNg3mCvjPZMSkfXFKnDsNhn/lV58 vL8uJlf+L6thhXFg+8nplpX7PqbKXAZeFXk37xT55gP+uQrAHLQoHPhYxy2cxHC3+grf ivn9/c7ivmYi3za3A7q4Ox15pdJWhA2qi9gPavS2B26h+u+aSHfpxBApWD8iQYiELI0I ZeMw==
ARC-Message-Signature	i=1; a=rsa-sha256; c=relaxed/relaxed; d=google.com; s=arc-20160816; h=list-unsubscribe:mime-version:subject:message-id:to:reply-to:from:date:dkim-signature:dkim-signature; bh=PfvOtkdlN4BqNDCKK4qESxkN2kuAtKZaVDzFhcT0/0=; b=nVKDvh4gFhetgQb8aL9ROWMJWU9hQH/ES41g+cFWu/OE6wGDx6KF0WmYjMZk/rYI60 TV3Mh9szChI3cySuZfvKdY3/50nMwbUuJY/y3E+dCA0p6xHb3dfRPvGLjViPyfsezpr0 HLT3JUUPuh1KQrI78czYkjvWHoqi1SiG9uG5oUMQ2P+vsY9t8LXnuU5ZeixgAMZiUmiE va8VHvq6TznR7Rd7zt1zrHOsTwAAr0f1rFI1ZWaDpKjMi8gURvd2AGV81jjMC33fWLWq LiEXXaFznOjNEi2r1Cv/38VZ6+LD0emUWi2XuSdX5k60iTHGE+eqVV0nUc2JiEpYQ1EmD k9aQ==
ARC-Authentication-Results	i=1; mx.google.com; dkim=pass header.i=@neo4j.com header.s=m1 header.b=ccSUPvlf; dkim=pass header.i=@mktomail.com header.s=m1 header.b=XafDpvAC; spf=pass (google.com: domain of 710-rrc-335.0.143313.0.0.55127.9.7110292@potomac1050.mktomail.com designates 192.28.152.140 as permitted sender) smtp.mailfrom=710-RRC-335.0.143313.0.0.55127.9.7110292@potomac1050.mktomail.com; dmarc=pass (p=QUARANTINE sp=QUARANTINE dis=NONE) header.from=neo4j.com

Figure 9.9: *MXToolbox e-mail header analysis*

- eMailTrackerPro

  Access at **www.emailtrackerpro.com**

  This is a commercial tool (it offers a 15-day fully functional trial version) for tracking e-mails using e-mail headers. To use this tool to track a particular e-mail header, download and install the tool from the website as any Windows app. To launch the app → click **`Trace Headers`** to analyze the e-mail headers as presented in *figure 9.10*. 3. A new window will appear, here, paste the target e-mail header and press the **`Trace`** button to start tracing. Click **`My Trace Reports`** to view the detailed tracing report.

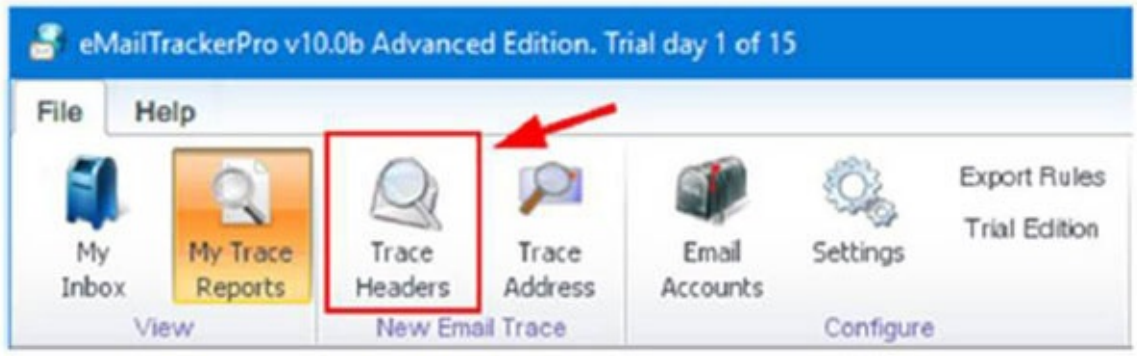


Figure 9.10: *eMailTrackerPro to track e-mail headers*

Determine the sender's geolocation and time zone

As seen previously, the sender's IP address can be extracted from the e-mail header (refer to the line that begins with "Received: from" in the e-mail header); now, use this IP address to determine the geographical location of the sender. There are many online services that can be used to map IP addresses to geographical locations, such as the one mentioned as follow:

- **IP2Location**: is an IP Geolocation Web service as a REST API providing a service to do a reverse IP location lookup for ISO3166 country code, region or state, city, latitude and longitude, ZIP/Postal code, time zone, **Internet Service Provider** (**ISP**) or company name, domain name, net speed, area code, weather station code, weather station name, **mobile country code** (**MCC**), **mobile network code** (**MNC**) and carrier brand, elevation, usage type, address type, and IAB

category. To search, submit a query string and a set of parameters to the IP2Location™ REST API service, and you will get the IP location lookup within seconds. Search results are derived from the IP2Location™ index of over 4 billion unique IP addresses, supporting IPv4 and IPv6 supported, as illustrated in *figure 9.11*.

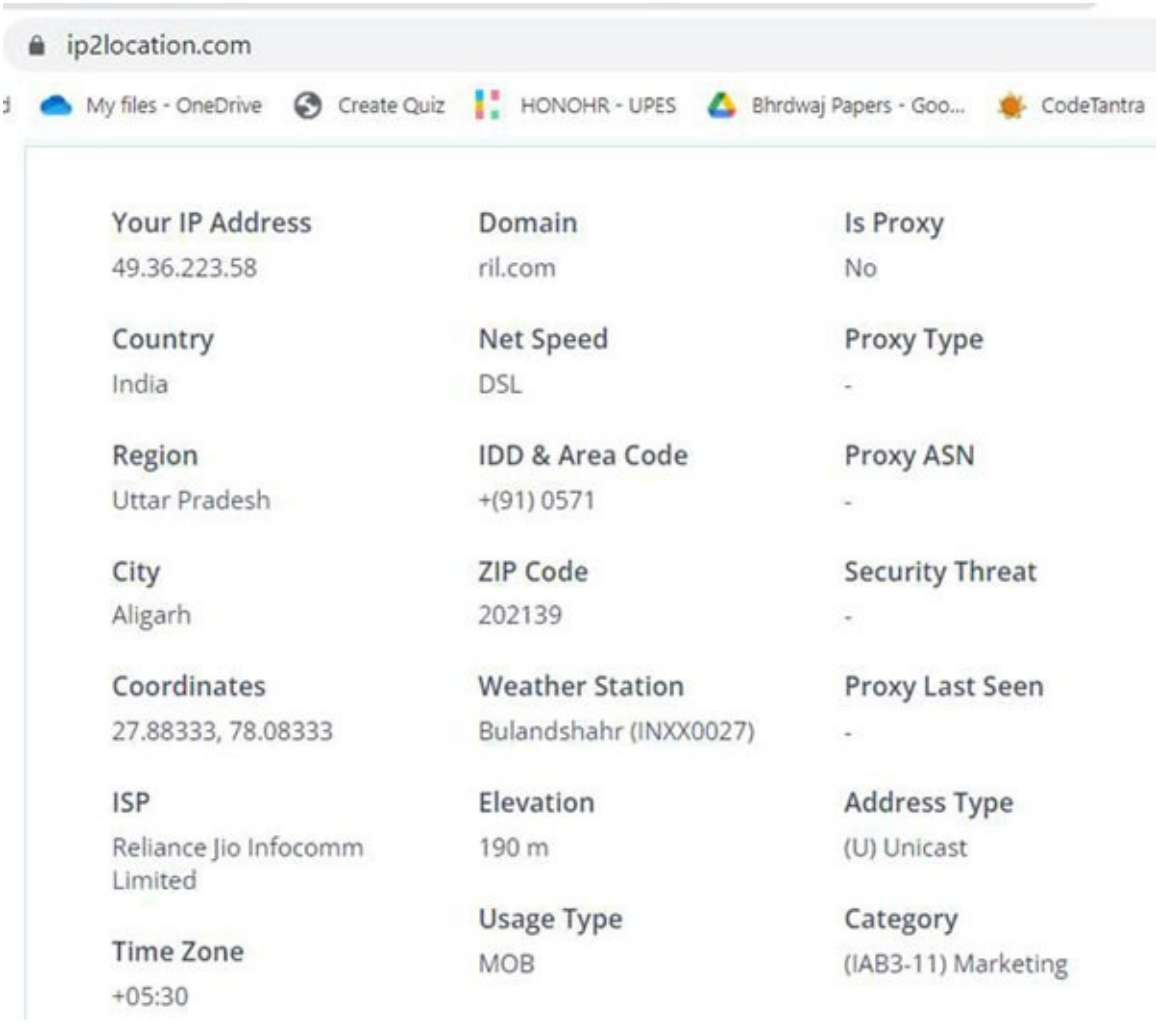


Figure 9.11: *IP2Location for IP address to geographical location*

- Ultimate IP Address Tracker helps identify users, collect online details, and get IP numbers. Investigators can view, download, and process enriched IP tracker data, as illustrated in *figure 9.13*. Use the IP tracker with an IP address to identify and collect online details based on the IP number. Advanced technology combined with cookies allows you to identify visitors. Enrich, view, download, and process IP tracker data with Opentracker, which records each unique user and their IP address, as presented in *figure 9.12*.

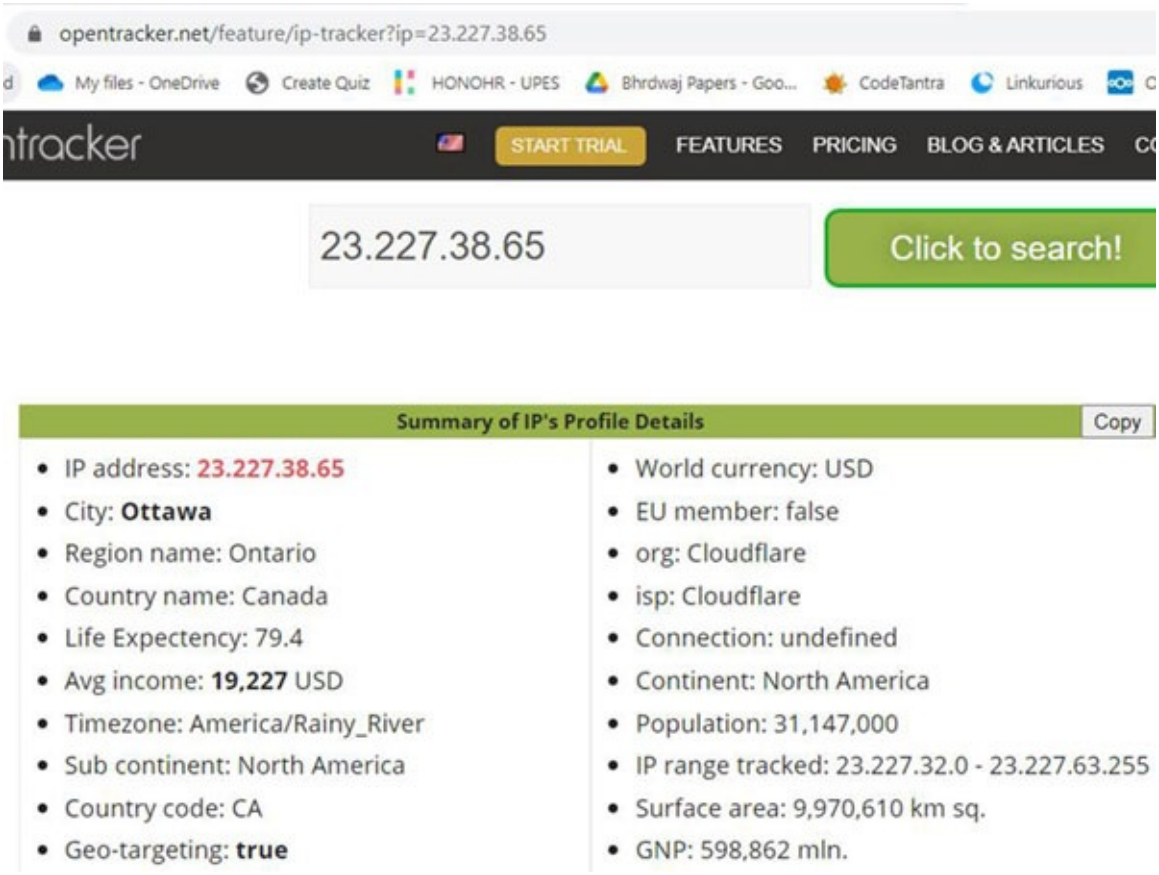


Figure 9.12: *Open tracker as IP tracker*

In some cases, the sender's IP address can be missed or not included in the message header. This is true when the sender uses a Webmail service like Gmail to send e-mails. In this case, we can determine the sender's location by checking the sender's computer time zone information. To learn this piece of information, use Google's Toolbox Message Header and check the **`Created at:`** field, as shown in *figure 9.13*.

Figure 9.13: *Revealing the sender's local time zone*

Conclusion

The majority of cybercrimes include the use of the internet in some way to commit or aid a criminal act. This leaves digital traces on both the client and the Web server, such as browser history, prior searches, download history, and login credentials to social media accounts (for example, IP address and timestamp information). The primary areas where internet-related forensics artifacts can be found have been covered in this chapter on Web browser forensics. We will continue our study in the upcoming chapter, looking at

how anti-forensics might be employed to obstruct inquiries and then prepare forensic reports.

CHAPTER 10

Anti-Forensics Techniques and Report Writing

Introduction

In previous chapters, we explored how to capture and create forensic images from memory or hard drives, how to gather data from Web browsers, and how to examine e-mails. We may now proceed to the next phase of the forensic investigation, which entails searching the photographs we have gathered for relevant hints. All analytical work should be done on the forensics image only to prevent accidentally destroying original evidence and rendering the entire inquiry useless in a court of law. The original suspect equipment should not be tampered with by forensic examiners. In this previous chapter, we will learn how to report about forensic images and anti-forensics techniques.

Structure

This chapter presents and discusses anti-forensic techniques and report writing for digital forensics cases. The reader will be able to learn about the following aspects in this chapter:

- Anti-forensics
- Digital Steganography
- Text Steganography
- Image Steganography
- Audio-video Steganography
- Network Steganography
- Metadata manipulation
- Encryption techniques
- Anonymity techniques
- Report writing

Objectives

As was already mentioned, digital forensics, also known as cyber or computer forensics, is a subfield of forensic science that gathers, examines, and presents digital evidence for use in legal proceedings or remedial action using scientific rigor, methodology, and knowledge to help solve crimes and incidents. The basic goal of digital forensics is to thoroughly examine digital evidence and make it ready for presentation in a court of law. Digital forensic investigators use a range of forensic tools to gather, store, and assess digital evidence. They will draw conclusions from their research and present them to those who will act on them.

On the other hand, anti-forensics science is a set of strategies for thwarting forensic investigation. It aims to stymie and mislead investigations by making it difficult, if not impossible, to access and understand digital evidence. To frustrate forensic investigators and extend the time needed to conduct the initial examination, anti-forensics techniques aim to destroy or conceal digital evidence. The best definition of anti-forensics techniques came from Dr. Marc Rogers of Purdue University, who defined them as *attempts to negatively alter the existence, amount, and/or quality of evidence from a crime scene or to make the analysis and investigation of evidence difficult or impossible to complete*. In this chapter, we will discuss the idea of digital anti-forensics tactics and how they can be used to influence the course of a forensic investigation, making it very challenging to perform a digital investigation or even to gather sufficient evidence to support a claim in court.

Anti-forensics techniques

Other than a criminal and terrorist organization, anyone (organization or individual) wishing to maintain their online privacy and/or securely destroy their private data can use anti-forensics techniques for good. This includes military personnel, law enforcement officials, politicians and diplomats, researchers, journalists and whistleblowers, business corporations, and internet users seeking privacy. In terms of digital forensics, examiners should be aware of such methods and how they work so, they can react appropriately if they come across them while conducting investigations. Anti-forensics has developed into a broad field with methods that address many different computer security issues. Data hiding techniques (also known as Steganography), data destruction techniques (also known as anti-recovery), data encryption, cryptographic anonymity, and direct attacks against computer forensics tools are all examples of digital anti-forensics techniques that forensic examiners come across during a digital investigation.

Digital Steganography

The science of Steganography involves hiding a hidden message within a conventional file and maintaining its secrecy throughout transmission. Steganography techniques have been used since the dawn of time to deliver secret espionage information from spies and to relay covert signals during armed battles. In earlier steganography techniques, physical objects (such as paper, eggs, invisible ink, and even human skin) were used to cover the secret message. However, as computing technology and internet communications advance, modern strategies increasingly focus on using digital data (whether photographs, videos, music, or even a file system like Windows NTFS) and IPs to communicate secret signals between communicative parties.

Any digital file type can theoretically be used to conceal a secret message because most digital steganography methods do not alter the appearance of the overt file being used as a carrier, making it impossible for outside observers to identify the message's intended recipient. Generally, there are three ways to include a hidden message in a digital file as follows:

- **Injunction**: This method allowed us to bury a hidden message in a trivial, unreadable section of the obvious file. This method uses a hidden message included after the file's end marker (EOF). The overt file's appearance and quality are unaffected by this process.
- **Substitution**: Using this method, we swap out pointless portions of the overt file with those in the covert message. Because we are only exchanging bits in this method rather than adding anything new, the overt file size will not increase, making it more secure than the prior one.

- **Generation**: This method, which involves creating a new file that includes the concealed message, is the best one for achieving digital Steganography.

Steganography methods are frequently categorized based on the type of carrier file (overt file), which is used to hide the secret message.

Text Steganography

Text Steganography uses words to conceal sensitive information. A few possibilities include reducing text size to 1 pixel, using MS Word's hidden text capability, and inserting spaces between words and/or one or two spaces at the end of each line to hold hidden bits. In the form of text Steganography, known as watermarking, concealed data is inserted into an open file (such as an image or audio file) and can only be retrieved by the owner using a secret key. The primary drawback of this sort of digital file, as compared to others, is the limited amount of hidden data that can be embedded in the text. Text steganography can be created by simply changing a text's orthography while keeping its visual appearance. By changing the wording of words that are crucial to the plan, we can avoid being detected by the monitoring equipment used by some nations' governments to censor or block internet traffic. A free online tool called Spider Army (**http://txtn.us**) transforms Unicode text into a different string that visually resembles the original. In plain sight, hidden messages can be found in tweets on Twitter. Your tweet's characters are swapped out for Unicode homoglyphs that seem similar in order to conceal your concealed content.

Image Steganography

The most used file format for secret message encryption is this one. As we all know, in the current digital age, people routinely exchange and post photos online (for example, on social media sites); the significant volume of daily digital image exchanges makes this file type less suspicious to outside observers. In image Steganography, a secret message is hidden within an image file using a Steganography technique. The recipient receives the output, known as a stego-image, and uses a similar procedure to extract or decode the secret message from the overt file. Data concealed in visuals is not without its limits. You should always work with a copy of acquired digital evidence in order to prevent unintentionally erasing the evidence. For many implementations of an image's Steganography algorithms, updating the image (resizing, changing the format, editing, or cropping it) will destroy the secret message contained within it.

Audio-video Steganography

Digital audio recordings can be used to conceal sensitive information via audio Steganography. The most popular audio steganography program is MP3stego, which uses the MP3 audio file format to conceal the concealed data like MP3Stego at **www.petitcolas.net/steganography/mp3stego**. Video files can be encrypted with the same methods used to encrypt music and photo files. Video files can store a lot of hidden data without lowering the quality of the original video because they are made up of a series of audio and image files (overt files).

Network Steganography

Experiments demonstrate that it is possible to embed secret messages using networking protocols like the

TCP/IP family. Numerous networking protocol design elements make this possibility possible. Covert TCP is an illustration of a program used to hide data within networking protocols (TCP/IP header).

Metadata manipulation

In computer forensic investigations, metadata timestamps in digital files are crucial because they allow investigators to focus their search on a period of time that is relevant to the case at hand (for example, before or after an incident took place). For instance, a suspect has the ability to alter any digital file's four primary timestamp attribute values when using the Windows NTFS file system. The following are these values:

- **File created**: The hard drive's "created" time can be found here.
- **Last time the file was accessed**: This is the date and time. The access time will change when a user opens or moves a file from one place to another. Antivirus software can alter the access times of files while it scans them.
- **File modified**: This refers to when a file's data was altered.
- **MTF last written**: When right-clicking a file and choosing `Properties` in Windows Explorer, this attribute does not show up; however, some computer forensic programs, such as EnCase and AccessData FTK, can view it. The portable utility BulkFileChanger from Nirsoft may update the primary date attributes of any NTFS file, such as the created/modified/accessed date, in addition to other file attributes (**www.nirsoft.net/utils/bulk** file `changer.html`) (such as Read Only, Hidden, and System). Simply run the tool, choose the file whose attributes you want to edit from the list, and then click the `Clock` button on the program toolbar menu to change attribute values, which will allow you to change the time attribute.
- While altering a file's date/time stamp is still a useful defense against computer forensics, keep in mind that a skilled investigator may be able to confirm or refute this suspicion by doing a thorough examination of the file's secret timestamp characteristics in the MFT. This will make it quite evident that the subject file's time characteristics have been manually modified.

Encryption techniques

Although the goal of Steganography is to conceal secret information by making it difficult to detect, encryption techniques accomplish a similar task by rearranging the data. Information is concealed using encryption, which renders it unintelligible to unintended recipients. Today's IT systems heavily rely on encryption, and public and private companies must secure their data in transit and at rest. Without the suspect's participation, a forensic investigation of encrypted devices will undoubtedly be difficult, time-consuming, or perhaps impossible due to the widespread use of encryption programs, some of which have quite powerful capabilities. When the key or password was unavailable, authorities frequently paid substantial costs to decrypt very valuable incrementing data. A key is a group of bits used by an algorithm in cryptography to change information from plain text to cypher text and vice versa. This aims to jumble the information so that only those with the proper key to decode it may see it and recover the needed information.

Disk encryption using open-source tools

Different encryption tools are already accessible; some of them are free, and some are paid for. The venerable program TrueCrypt is the most widely used open-source encryption program (used for disk and file encryption). The development of TrueCrypt abruptly stopped in 2014, but its widespread use and sophisticated encryption capabilities have prompted other developers to start new forks of this project.

The following are the main projects based on this tool:

- VeraCrypt is based on TrueCrypt 7.1a and may be found at **www.veracrypt.fr/en/Home.html**. It is an encryption disk that is open source and free and works with Windows, Mac OS X, and Linux. In addition to having the ability to encrypt the full hard drive and portable storage devices like USB sticks and external HDDs, VeraCrypt also enables the creation of a hidden volume (which is contained within a virtual encrypted disk) and a secret operating system partition.
- CipherShed (**www.ciphershed.org**): This fork of TrueCrypt is also available. It keeps the existing TrueCrypt container format's backward compatibility.

Anonymity techniques

When sending or receiving messages over open networks like the internet, digital anonymity works by obscuring any traces between the sender and message recipient. It encrypts messages using a combination of encryption techniques and uses cryptographic anonymity software to mask your identity while it is being transmitted. Users can protect their online privacy when using anonymous networks like the TOR network by hiding their real IP address from prying eyes like the ISP. The TOR network is the most well-known anonymous network.

You must use the TOR Browser, which can be downloaded from **www.torproject.org/projects/torbrowser.html.en**, to access the TOR network. When a person takes stringent security precautions, it is nearly impossible to track them through the TOR networks. I2P and Freenet are two further anonymous networks (**https://geti2p.net/en**; **https://freenetproject.org**).

Using anti-forensics tactics does not have to be difficult because much common software, like Web browsers, can be set up to automatically forget a user's previous actions with only a single click. For instance, several online browsers have added a unique setting called Private Browsing (Firefox) or Incognito Mode (Google Chrome). When a user closes the browser, Firefox Private Browsing automatically deletes all browsing history, form and search bar entries, download lists, passwords, cookies and cached Web content, offline Web content, and user data from their computer. The private mode also blocks ads with hidden trackers, making tracking a user across various websites very challenging.

You can open a new private Firefox window by going to the right corner of the main Firefox window and clicking New Private Window. Alternatively, you can press the *Ctrl+Shift+P* button combination to open a new private browsing window. Google Chrome has a similar feature called Incognito Mode that can be accessed from the Chrome menu on the top right corner (new Incognito window; see *figure 10.1*).

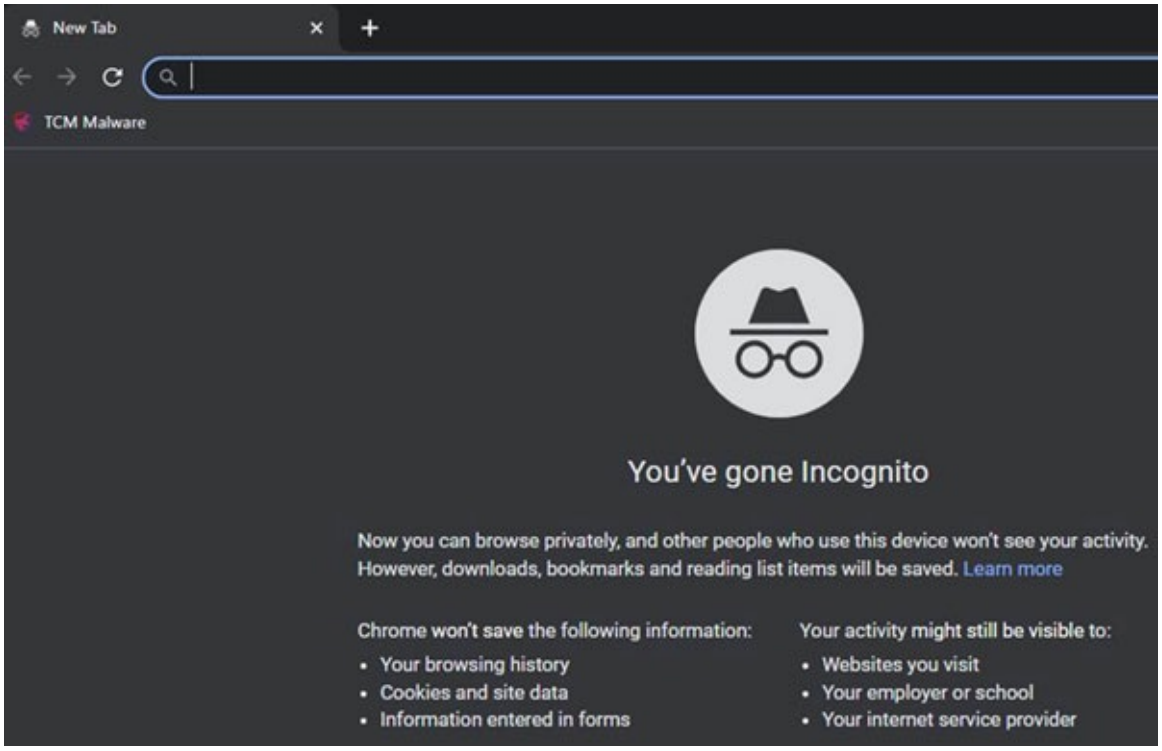


Figure 10.1: *Google Chrome Incognito Mode window launched*

Digital forensic reports

The creation of a report is the last stage of any inquiry, during which the investigator provides the results of the digital forensics analysis to the organization that was the target of the cyberattack or, in the case of a public investigation, to the court. The format and content of the report will be determined by the investigator's experience, writing style, the nature of the crime or incident, and the IT proficiency of the reader(s). Unlike public investigations (a jury in a criminal or civil procedure), which require as much detail as possible to buttress the examiner's claim, investigating civil situations may not necessitate giving substantial technical details. The digital forensic examiner/analyst must use plain language and clear, understandable technical words when writing a good digital forensics report; examples and figures are excellent for illustrating how a specific technology functions. To ensure that your report is understood by top management (in corporate conflicts) and judges/juries, keep in mind that it should be understandable by diverse user segments with varying IT skills (in public investigations). A timeline of events can help you comprehend what happened in order; many computers forensic suites automatically build one and let the examiner add tags and comments where they are needed.

The following are suggested components of any digital forensics report:

Investigator Information: Brief information about the examiner(s) who handled the case and what his/her role was in the investigation.

- **Case description**: Brief description of the case/incident (for example, investigating Arthur's USB thumb drive for possible violations of company IT usage policy) and what was required from the examiner in this regard.
- **Investigation**: This is the longest part of the report; the investigator should explain the forensic acquisition process, indicate what tools and techniques were used to analyze digital artifacts, and give some details about the technical procedures undertaken by the examiner(s) to extract data relevant to the case in hand. He/she also needs to submit a copy of the chain of custody forms and describe the methodology applied to handle and retrieve digital evidence.

- **Summary of findings**: This is the conclusion of the report; it presents the investigator's opinion about the innocence or guilt of the accused party, and it may also include a note to expand the investigation to other areas if needed.
- **Explanation of terms**: Like unallocated disk space, device configuration overlay, slack space. These terms should be described in simple language with examples so a non-technical audience can understand them.

The report is typically submitted with the digital evidence on a CD or DVD (that is, write-once media). The original evidentiary media should be kept private and should not be used to demonstrate digital evidence since doing so could cause the original evidence to be lost or altered. More anti-forensics tools will be created as computing technology develops. For computer forensic investigators, having up-to-date skills will present genuine hurdles. Practitioners in the field of cyber forensics must continue their education and keep up with the most recent technologies and methods on the market. Computer forensics software should include anti-forensics skills to automatically identify technical defenses used by criminals to obfuscate their criminal activity.

Conclusion

The reader will be able to learn about reporting, which is a crucial aspect of any forensics investigation. Although cases involving internal investigations of civil cases like company policy violations require less detail in their reports and can focus relatively more on the final output or summary, public investigations that involve courts typically require more technical details and thorough descriptions of the methodology used to collect and analyze the digital evidence. Typically, non-technical audiences get digital forensics reports, so be sure to write in plain language that judges, juries, and management staff can understand.

The creation of a thorough investigative report and the presentation of the key observations in a concise and understandable format are useful for reference and documentation purposes regardless of who oversaw the investigation (civil or public entity) you are writing a report about, even though that may not have been required by the reporting entity. We examined how to prepare an investigative report to summarize the results of our research to bring our book to a close.

CHAPTER 11

Hands-on Lab Practical

Introduction

In this chapter, we present hands-on labs that the learners need to study and replicate on their systems. To implement these labs, Kali Linux (latest version)/Windows 10/11 operating systems are required to be run in the virtual environment. Do not install the Digital Forensics tools and run these on the main physical machine.

Lab 1: FTK imager

FTK imager is an application that makes precise reproductions of the current findings without really altering them. The original evidence's image is kept, which enables us to copy data much more quickly so that it may be saved and subjected to additional analysis. The FTK imager also gives us access to its built-in consistency testing feature, which creates a hash report that aids in comparing the evidence's hash before and after the image of the evidence is created. A hard drive's data collection is referred to as an image or maybe a forensic image when done as part of an ongoing investigation. One of the most important tasks in a digital forensic process is generating a forensic image. However, for this imaged disk to function, the hard drive must be used. The disk image files must be opened and put on the drive using an imaging application; one cannot recover a hard drive by just putting the disk image files on it. Several disk images can be stored on a single hard drive. Flash drives with more storage space can also be used to store disk images.

As the forensics picture gives us a way to preserve the integrity of the evidence and make sure it has not been tampered with, we frequently need to confirm its integrity. Even while the FTK imager is typically used as an imaging system and previewing tool, it has significant features to help forensic investigators with various issues when they examine digital devices. So, let us get going and explore the choices that FTK imager provides on the `File` tab.

Creating a Forensics Image

1. After installing it, launch FTK Imager by AccessData to see the window pop-up that represents the tool's first screen, as shown in *figure 11.1*.

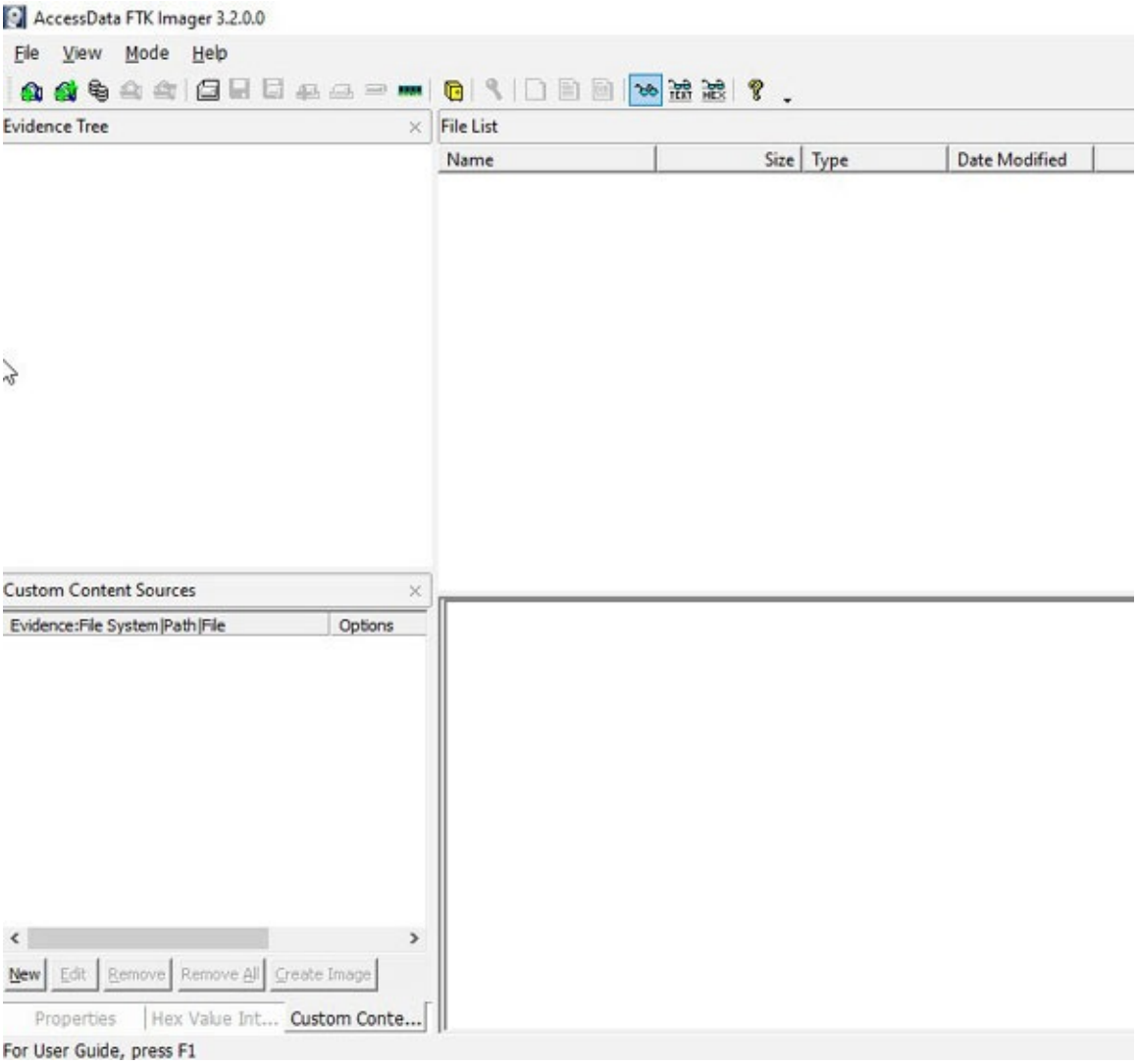


Figure 11.1: *Interface of FTK imager*

2. Now, to create a Disk Image as depicted in *figure 11.2*. Click on **`File`** | **`Create Disk Image`**.

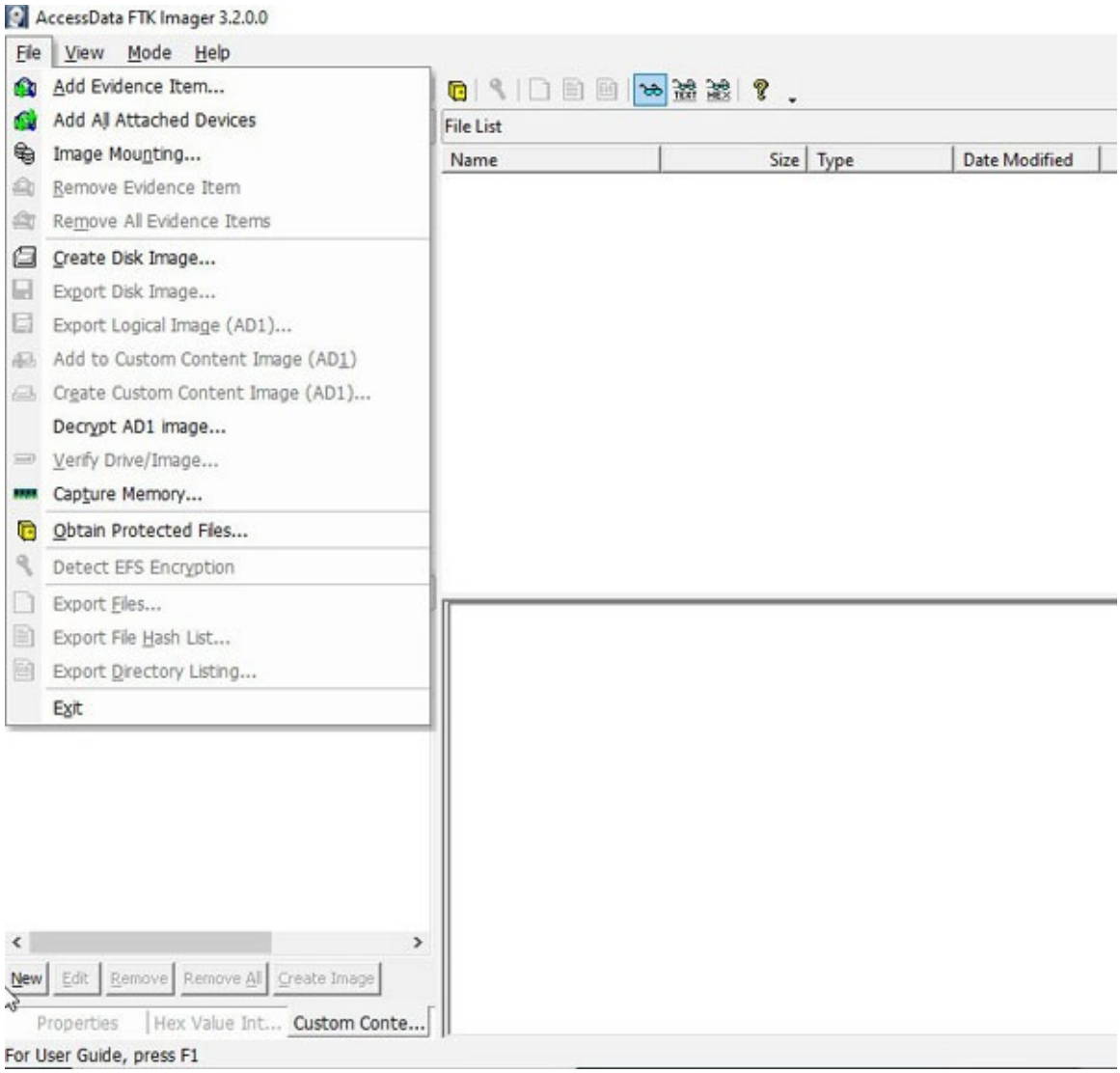


Figure 11.2: *Select create Disk Image*

3. Now that you have a disk, you can select the source, as presented in *figure 11.3*. According to your evidence, it may be a logical or physical drive. The main piece of hardware or internal equipment used to retain, access, and organize data is called a physical drive.

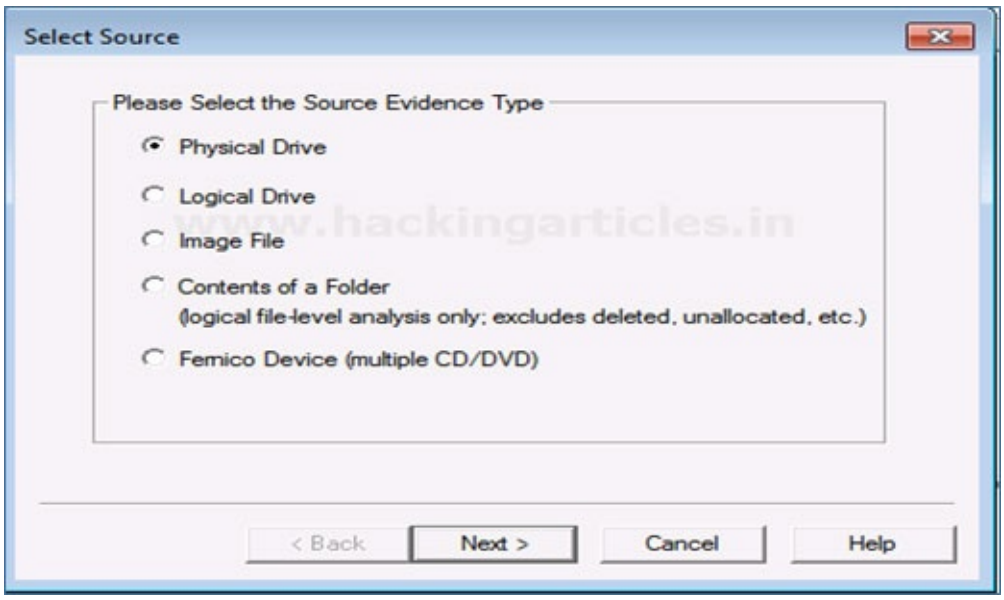


Figure 11.3: *Select source evidence type*

4. A drive space called a *logical drive* is typically built over a physical hard disk, as shown in

figure 11.4. Because it acts independently, a logical disk has its own characteristics and features.

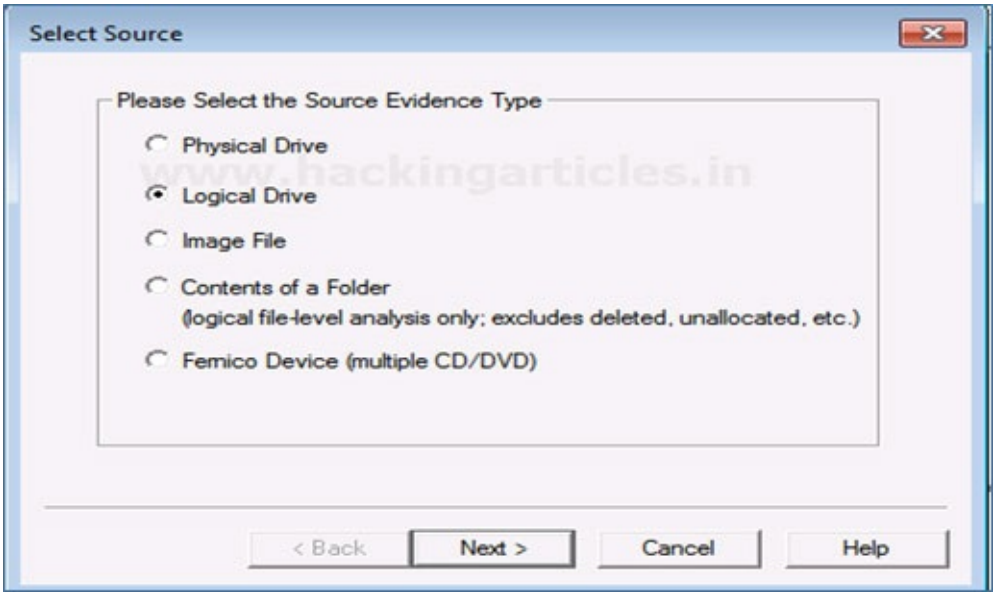


Figure 11.4: *Selecting the logical drive*

5. Now, choose the source of your drive (as shown in *figure 11.5*) that you want to create an image copy of.

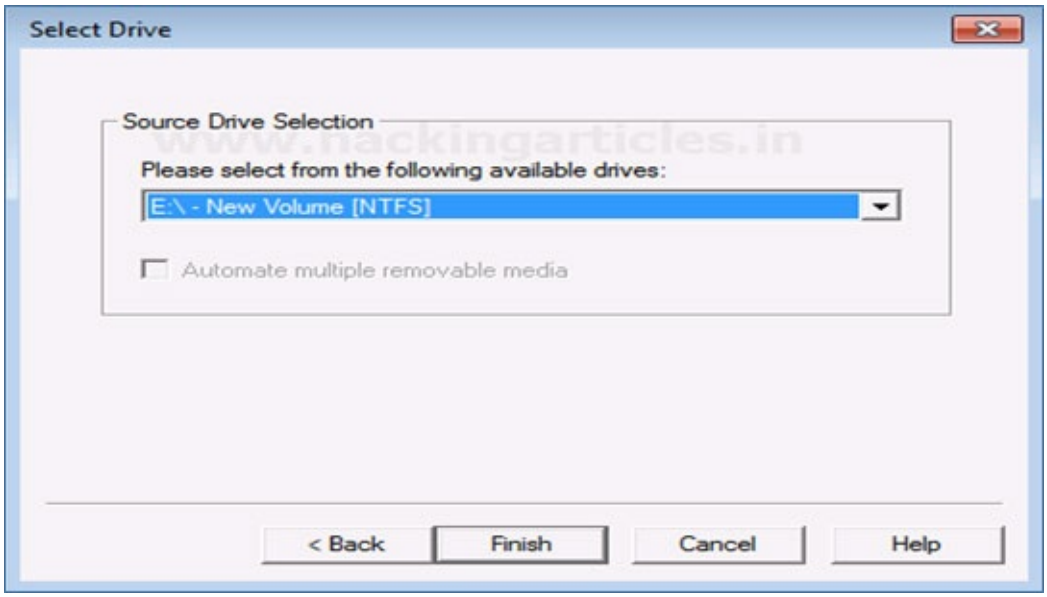


Figure 11.5: *Selecting the source drive*

6. Include the destination path for the newly produced picture, as shown in *figure 11.6*. To avoid evidence loss, it should be duplicated onto a different hard drive, and several versions of the original evidence should be made.

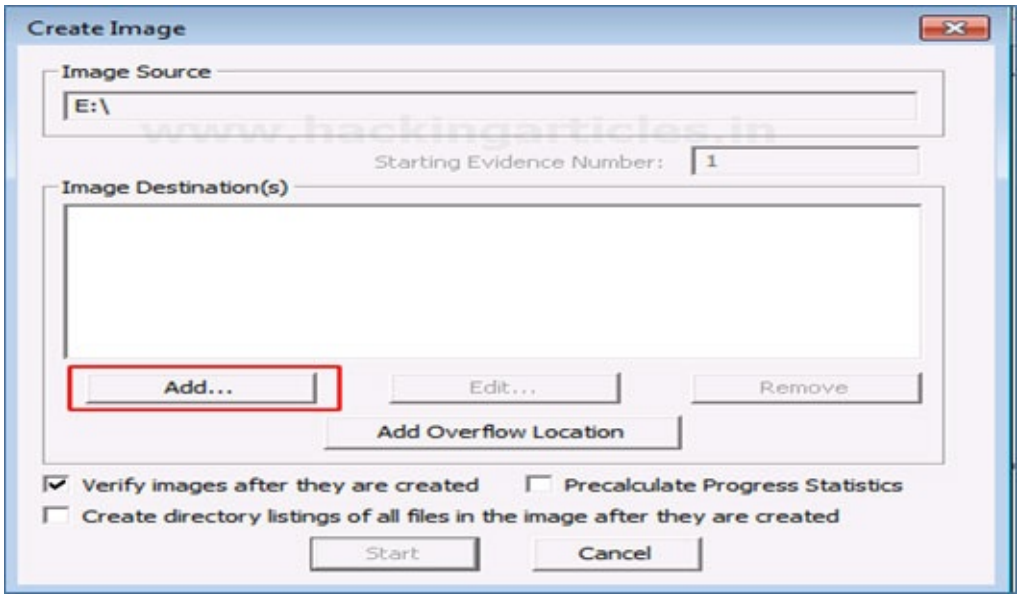


Figure 11.6: *Creating image*

7. Choose the picture format that you wish to use. The picture can be created in the following formats:

   - **Raw(dd)**: This is a bit-by-bit duplicate of the original evidence that has not been altered in any way. They do not have any metadata in them.
   - **SMART**: It is an outdated image format for Linux that was formerly quite popular.

   EnCase Evidence File, often known as E01, is a widely used image format that is compatible with the Autopsy format and also creates and maintains metadata about the imaging process. The acronym AFF stands for Advanced Forensic Format, an open-source file format, as selected in *figure 11.7*.

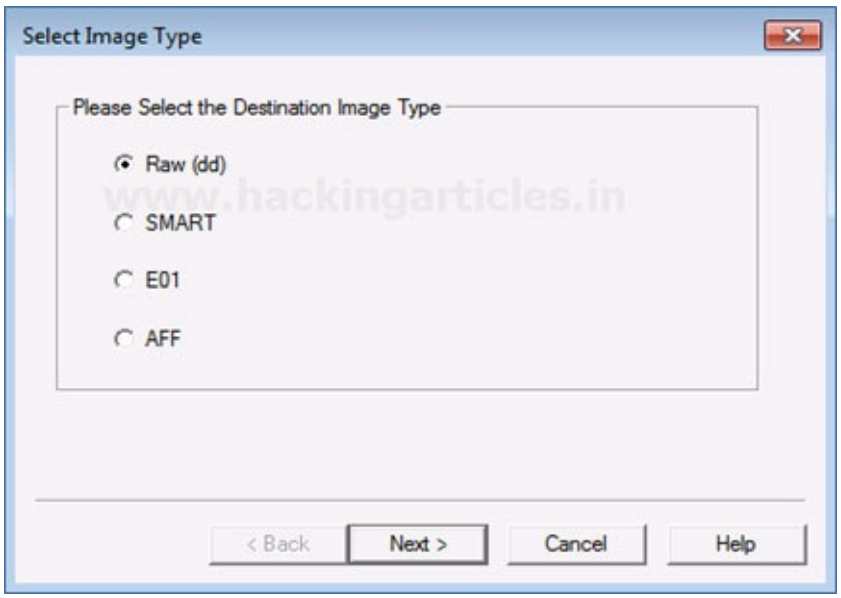


Figure 11.7: *Selecting the destination image type*

8. Now, add the details of the image to proceed (as shown in *figure 11.8*).

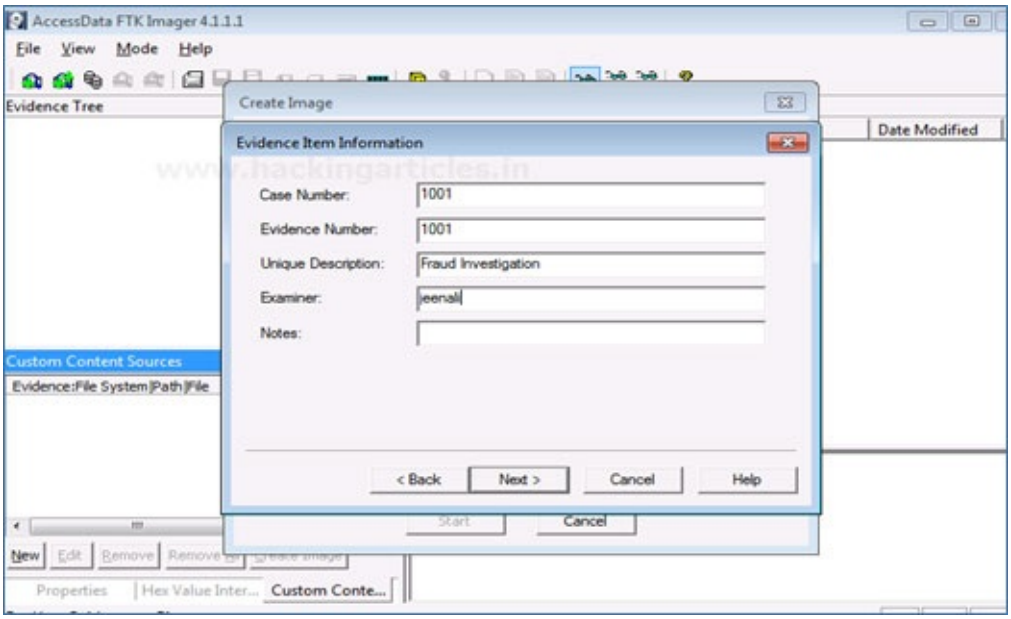


Figure 11.8: *Add evidence item information*

9. Now, finally, add the destination of the image file (as shown in *figure 11.9*), name the image file, and then click on **`Finish`**.

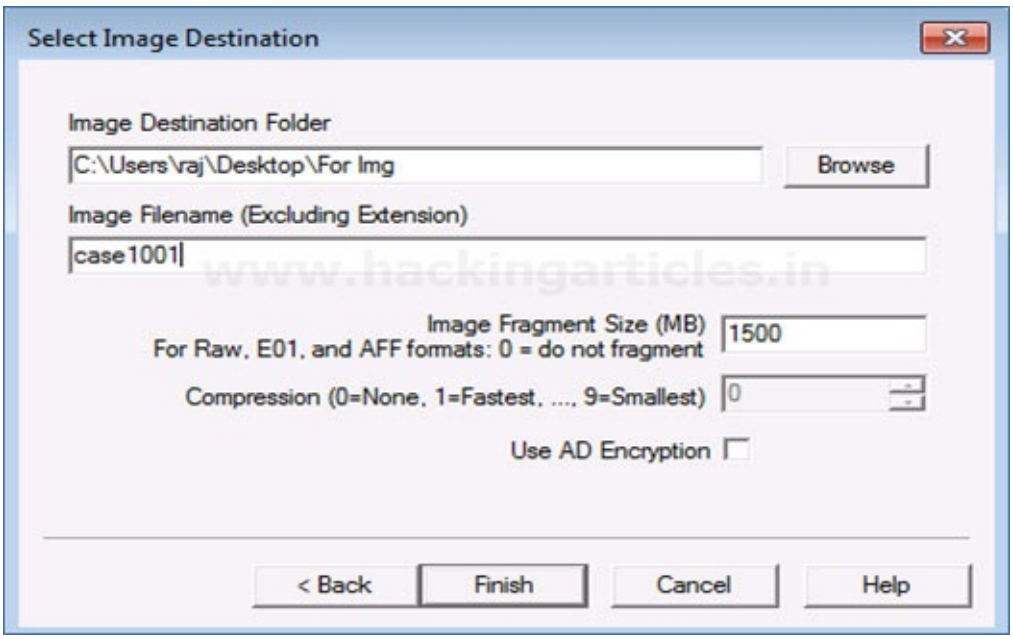


Figure 11.9: *Select image destination*

10. Once you have added the destination path, you can now start with the Imaging and also click on the verify option box to generate a hash (as shown in *figure 11.10*).

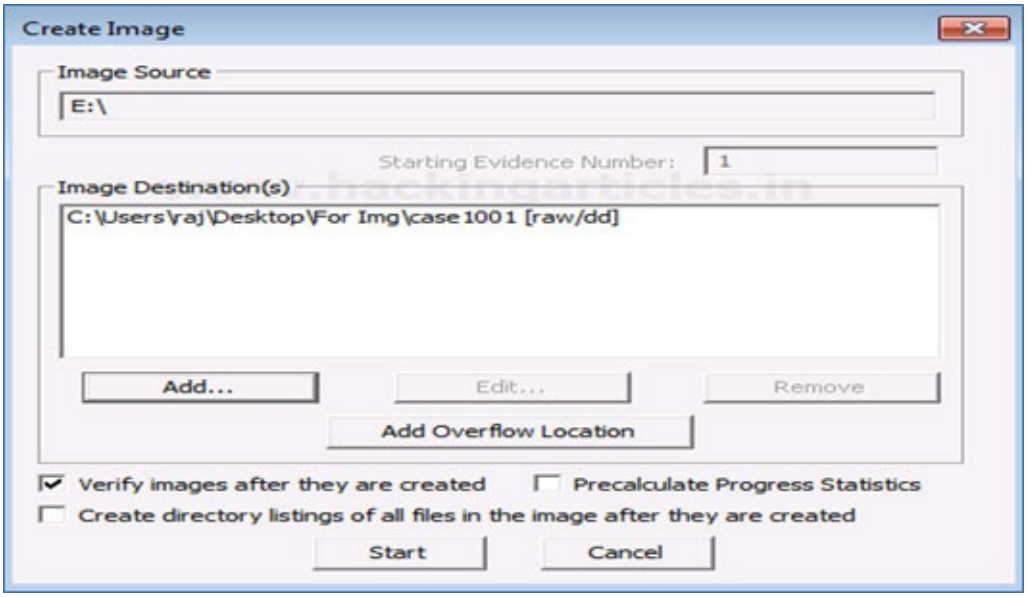


Figure 11.10: *Create image*

11. Now, let us wait for a few minutes (the progress is shown in *figure 11.11*) for the image to be created.

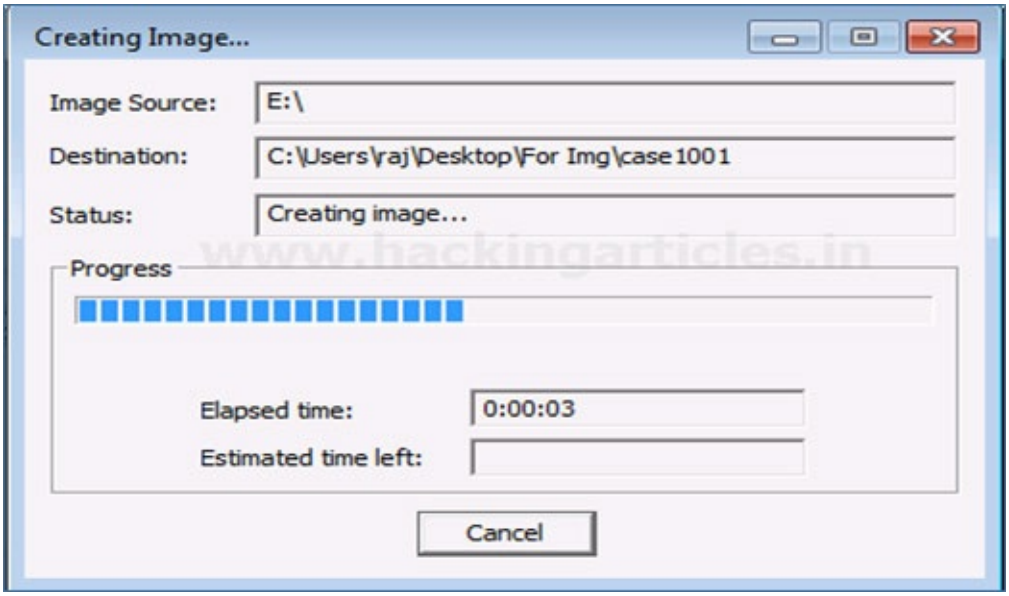


Figure 11.11: *Preparing the image*

12. After the image is created, a Hash result is generated, which verifies the MD5 Hash, SHA1 Hash, and the presence of any bad sector, as displayed in *figure 11.12*.

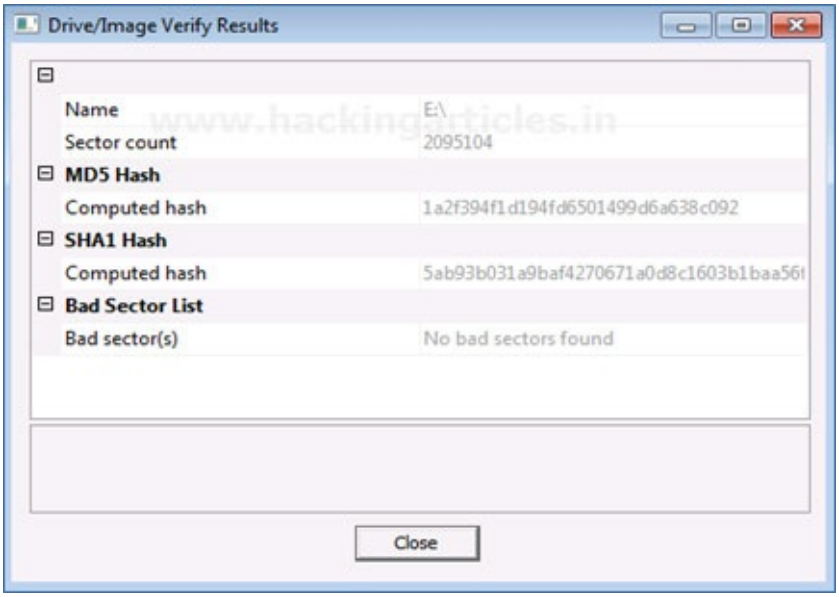


Figure 11.12: *Image summary*

Lab 2: Magnet RAM capture

In this exercise, we will discover multiple methods for capturing RAM memory for examination. Take a little time to learn about them all, as diverse situations need unique actions. Random Access Memory is referred to as RAM. It is known as the computer's primary memory that renders it crucial for a system to function. The user can save data they are now using or going to use in the system's RAM. However, because RAM is volatile, any data that is kept there would be lost the moment the power is off. RAM is accessible and user-friendly because it can be read and written too.

RAM capture is a crucial activity because researchers have come to learn that a variety of facts may be found in volatile memory. This information can be helpful in an examination and further enable an investigator to determine what programs a suspect or attacker was using at the time of the attack. It is also conceivable that online attackers using RAM rather than the system might have some data or tools saved there.

Lunar Sols Dump No inquiries are posed to the user end regarding the executable's combination of Windows 32-bit and Windows 64-bit. It is a little utility that makes it simple to store the RAM capacity of your computer. Although it is a terminal application, there is no need to learn a guest command-line switch or open a command line.

1. Ultimately, all that is required of us is to create a clone (as shown in *figure 11.13*) of the memory space in the current working directory and simply double-click the program.

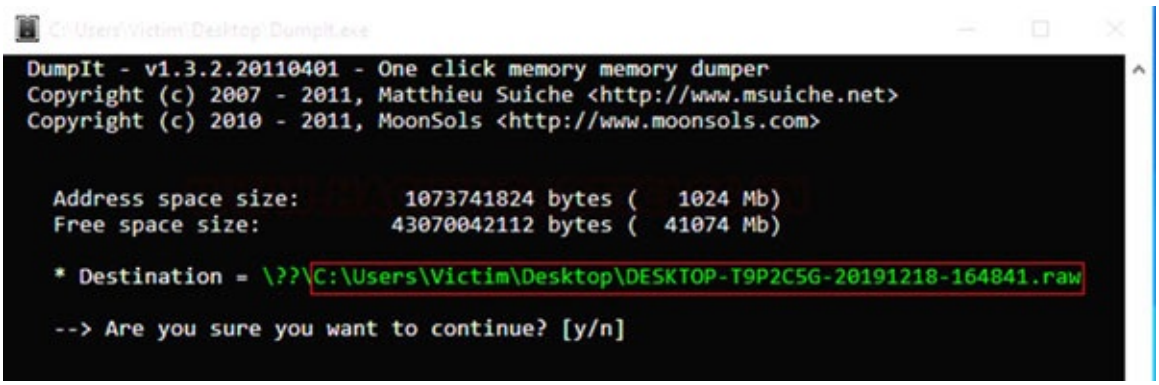


Figure 11.13: *Using DumpIt*

2. As we can see in the preceding picture, this software has already given us the location of the image we will produce through this procedure and is asking the user if they want to proceed or not. We must hit “y” to continue if we wish to (as shown in *figure 11.14*).

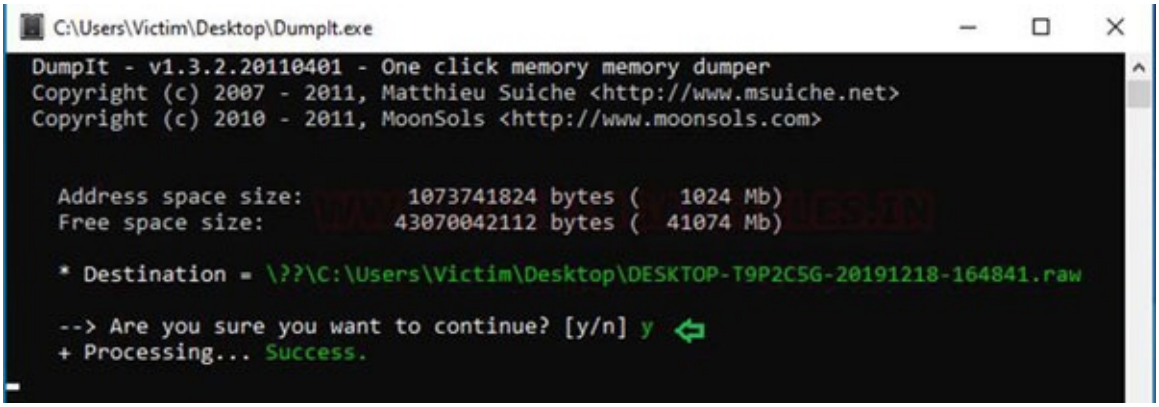


Figure 11.14: *Choose yes*

3. If we start the process, then after it is finished, it displays the message **`Success`** if we were successful.

   We may now verify the software’s suggested path to see if we were successful in capturing the RAM (the raw dump is shown in *figure 11.15*).

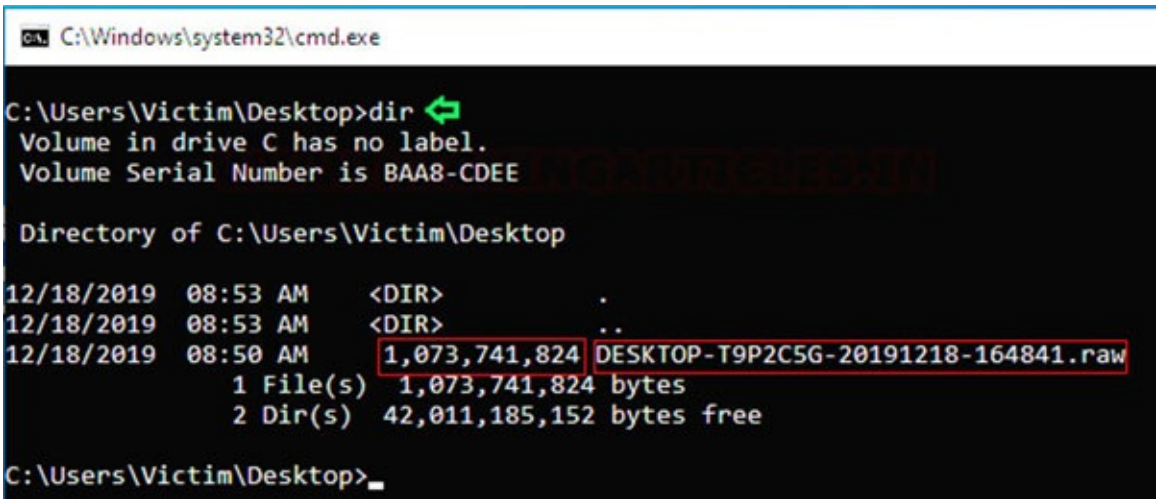


Figure 11.15: *Type dir*

Now that the RAM picture has been properly produced, we can see that our memory has been correctly recorded.

Now, we will use Magnet Forensics to capture the RAM, as displayed in *figure 11.16*.

Magnet Forensics is a free RAM capturing or memory imaging program that is used to record the actual memory of a suspect’s computer. This tool enables investigators to examine and retrieve important data that can only be discovered in the computer’s memory. Because of Magnet Ram capture’s minimal memory space, investigators may use the technique even as data is being rewritten in memory. Memory data may be recorded in raw format, and their analysis is simple.

In the forensic inquiry, this photograph may be used as proof. Processed evidence includes a program that is operating on the computer, network connections, signs of malware infection, registry hives, login information, encrypted files and keys, and so on.

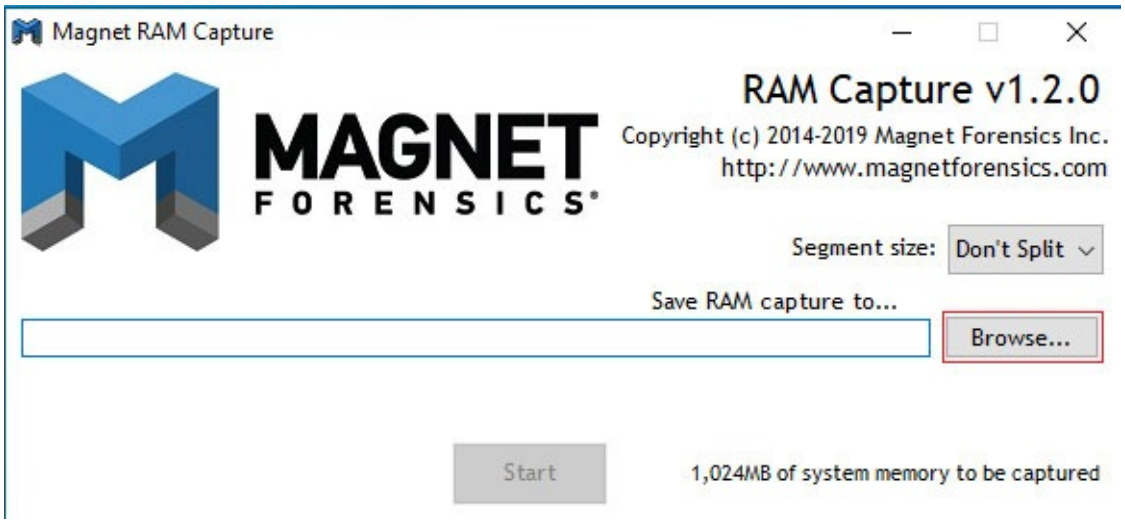


Figure 11.16: *Magnet forensics interface*

4. Now that the RAM image has been properly produced (as displayed in *figure 11.17*), we can see that our memory has been correctly recorded.

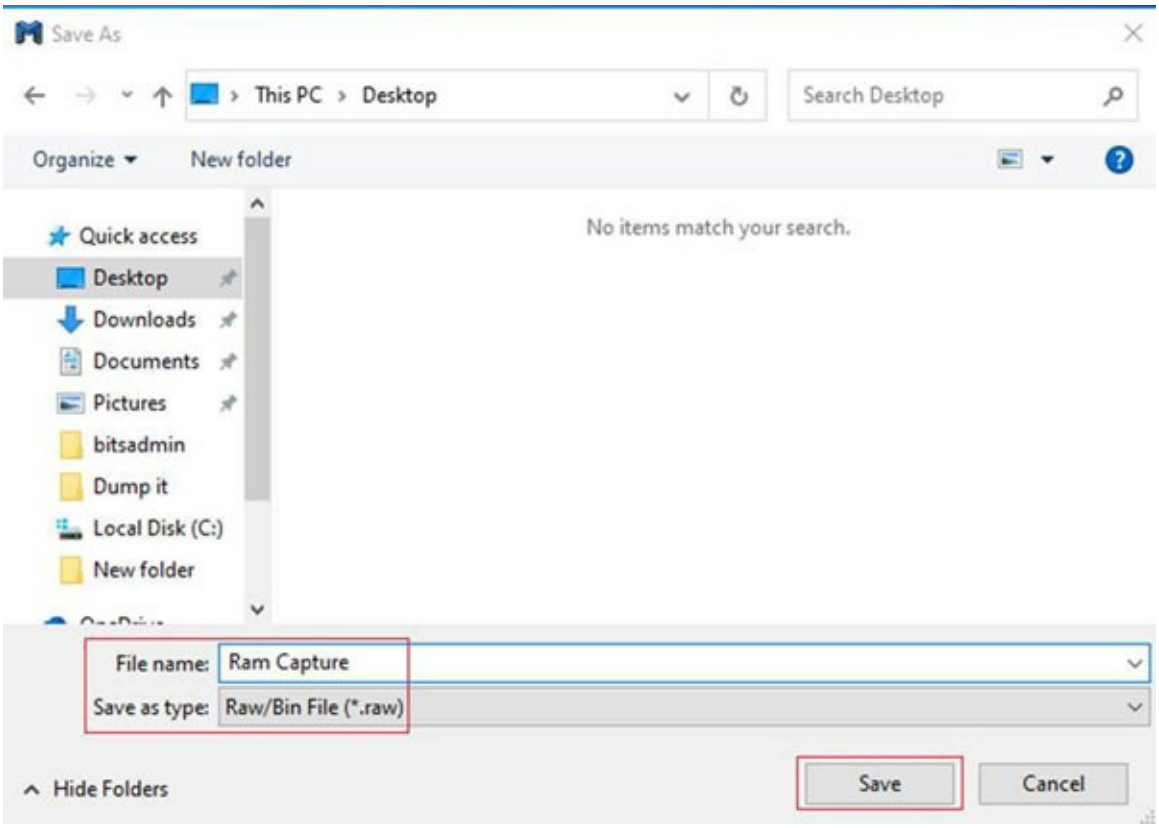


Figure 11.17: *Saving the RAM capture*

5. As seen in the preceding figure, we must specify the memory image's name and the structure in which we wish to record it.

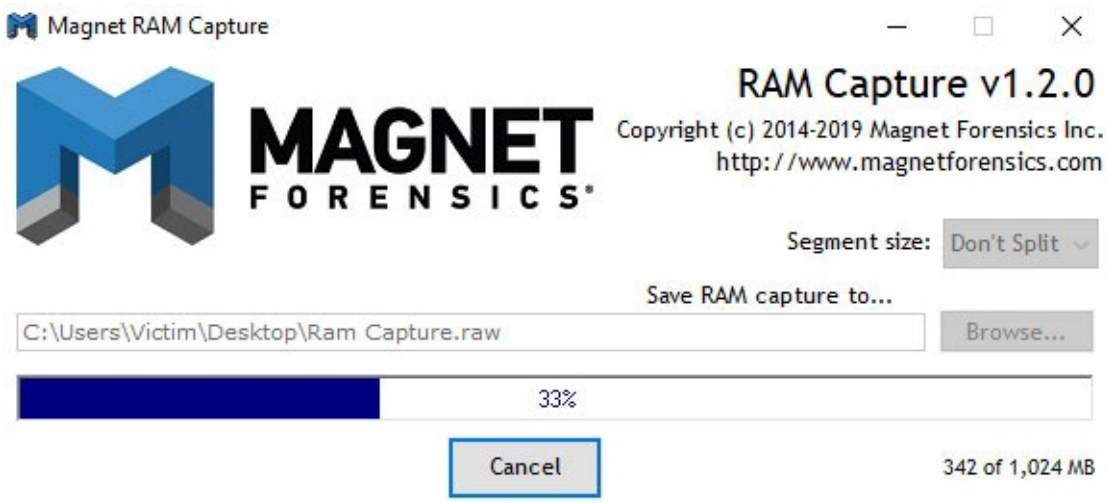


Figure 11.18: *Progress of the RAM capture*

6. After giving the aforementioned information, we now start the process of collecting the memory image (as presented in *figure 11.19*). The length of time it takes to finish the operation determines the size of the storage.

Figure 11.19: *RAM capture successful*

7. When the operation is finished, a pop-up notification confirming the process was successful (as displayed in *figure 11.20*) and gave us the path address where our previously given recorded memory is located is displayed.

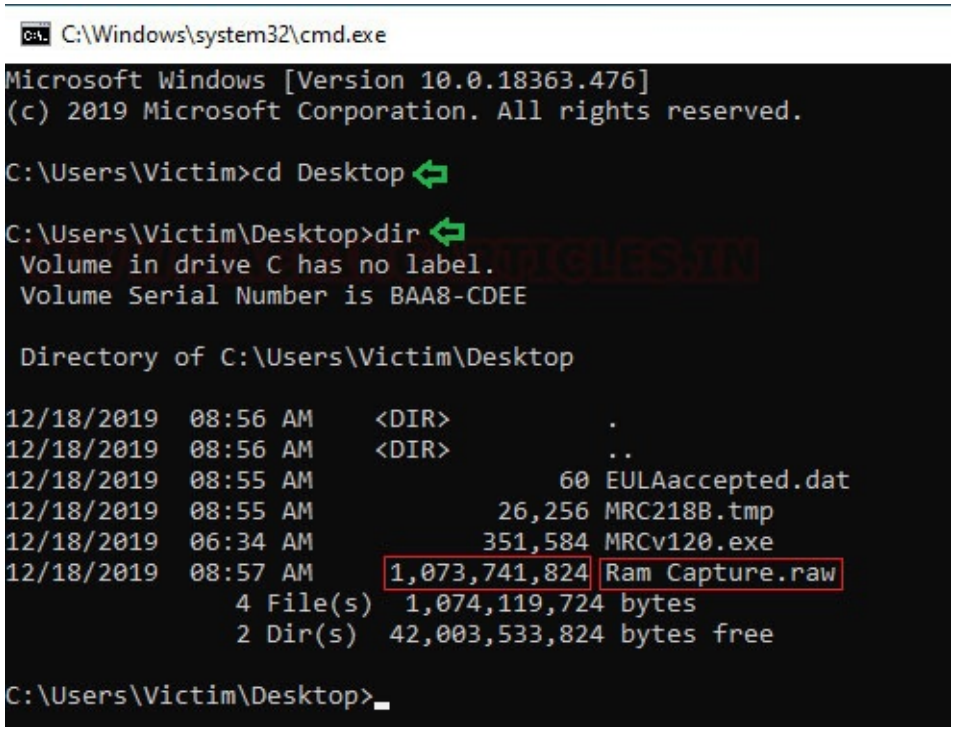


Figure 11.20: *Displaying captured RAM dump*

As you can see in the preceding screenshot, our image was successfully formed; thus, we can now verify our located path to determine if our memory image was made or not.

Lab 3: Memory forensics

An open-source memory forensics tool for malware investigation and incident response is called volatility. For analyzing RAM in 32-bit and 64-bit systems, it is among the greatest open-source software tools. Linux, Windows, Mac, and Android analysis are all supported. It may be used with Windows, Linux, and Mac operating systems and is based on Python. It can analyze various dump types, including raw dumps, crash dumps, VMware dumps (`.vmem`), virtual box dumps, and more. *Aaron Walters*, a software engineer and a businessman, developed volatility based on his academic work in memory forensics.

Objective: To perform memory forensics using volatility on Linux.

1. Download the volatility package (as displayed in *figure 11.21*) from **https://www.volatilityfoundation.org/releases**

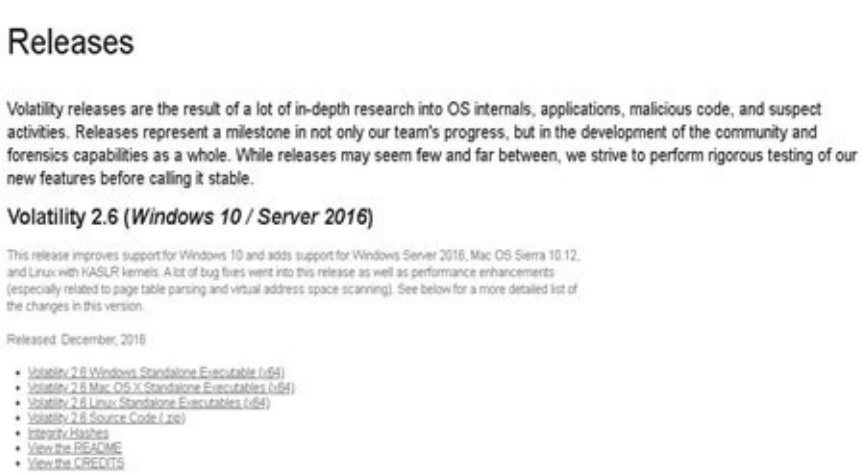
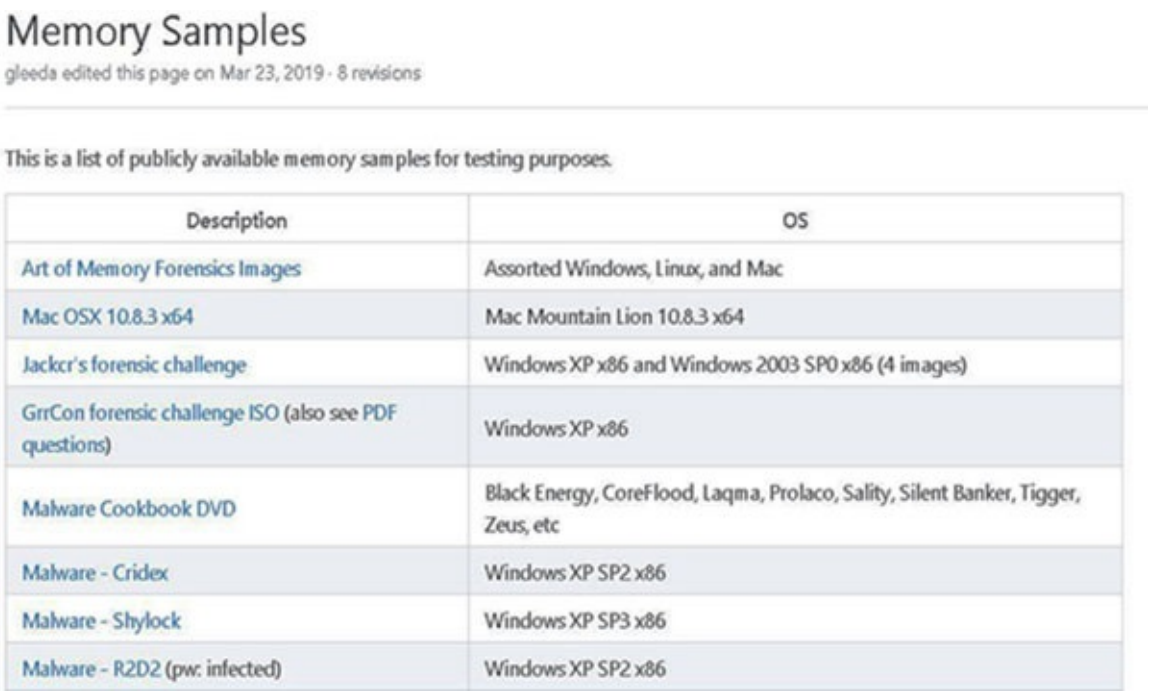


Figure 11.21: *Volatility framework*

2. Download Malware—Cridex as Sample Memory (as shown in *figure 11.22*) from **https://github.com/volatilityfoundation/volatility/wiki/Memory-**



Memory Samples

gleeda edited this page on Mar 23, 2019 · 8 revisions

This is a list of publicly available memory samples for testing purposes.

Description	OS
Art of Memory Forensics Images	Assorted Windows, Linux, and Mac
Mac OSX 10.8.3 x64	Mac Mountain Lion 10.8.3 x64
Jackcr's forensic challenge	Windows XP x86 and Windows 2003 SP0 x86 (4 images)
GrrCon forensic challenge ISO (also see PDF questions)	Windows XP x86
Malware Cookbook DVD	Black Energy, CoreFlood, Laqma, Prolaco, Sality, Silent Banker, Tigger, Zeus, etc
Malware - Cridex	Windows XP SP2 x86
Malware - Shylock	Windows XP SP3 x86
Malware - R2D2 (pw: infected)	Windows XP SP2 x86

Figure 11.22: *Memory samples*

3. Open the terminal window (as shown in *figure 11.23*) and check for the various options and supported plug-in commands available in volatility Command: **`volatility -h`**.

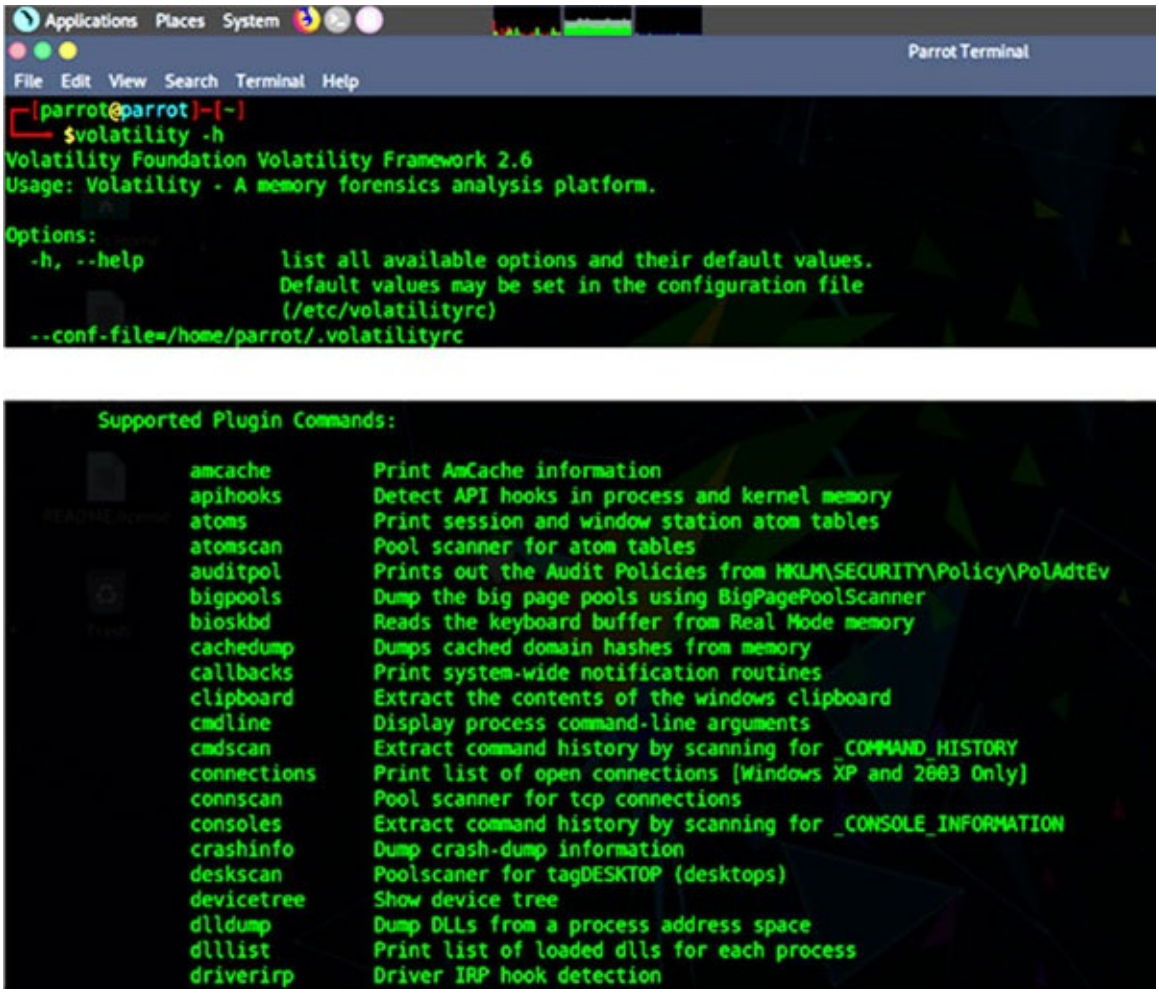


Figure 11.23: *Volatility Help page*

4. Explore some plug-ins (shown in *figure 11.23*) available in volatility by using the commands mentioned previously using **`Cridex.vmem`** as memory sample with screen shorts are given as follows:

5. **`imageinfo`**: It is to identify the supported "profiles" (shown in *figure 11.24*) for the dumped memory image.

   `Command: volatility -f cridex.vmem Imageinfo`

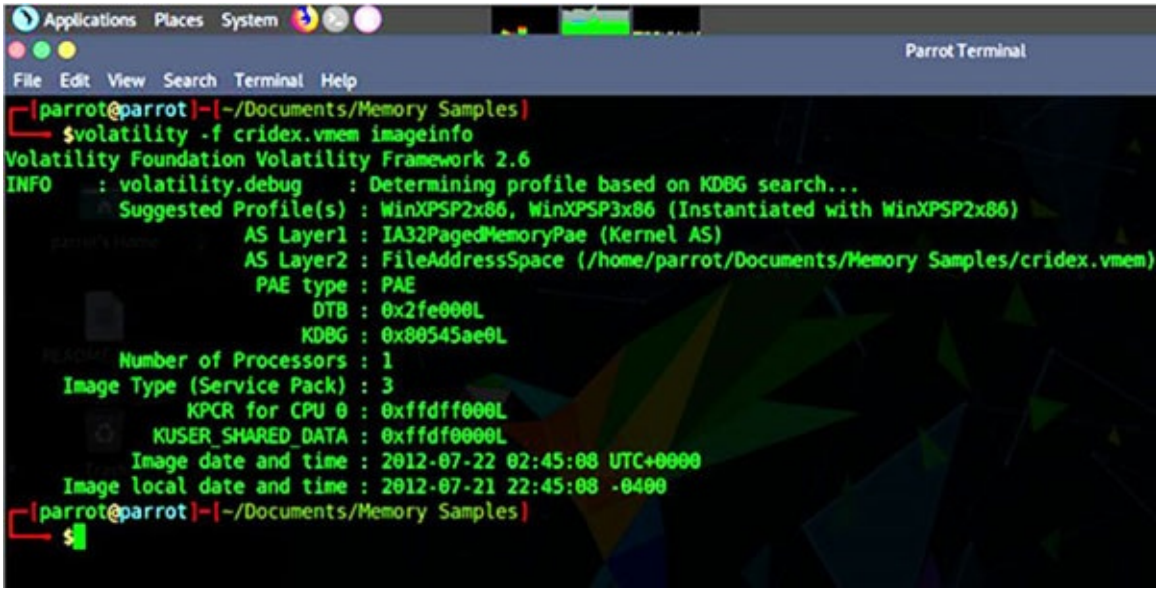


Figure 11.24: *Getting the image information*

6. **pslist**: This plug-in of the volatility tool shows the processes (displayed in *figure 11.25*) and gets a list of the running process in the memory dump.

   Command: `volatility -f cridex.vmem --profile=WinXPSP2x86 pslist`

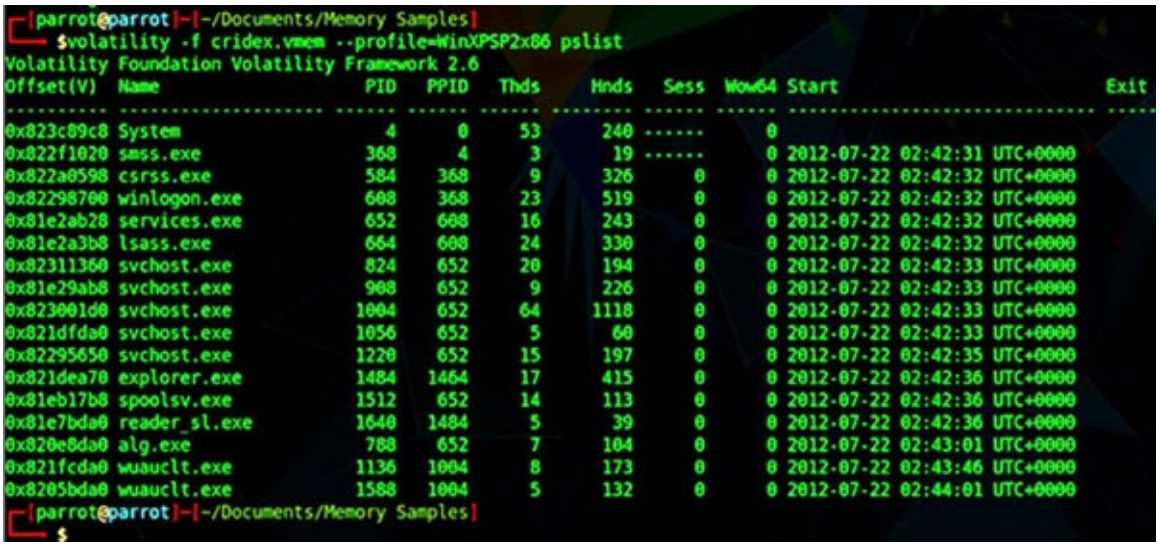


Figure 11.25: *Getting the list of running processes*

7. **psscan**: This plug-in of the volatility tool (shown in *figure 11.26*) can find processes that were previously terminated (inactive) and processes that have been hidden or unlinked by a rootkit.

   Command: `volatility -f cridex.vmem --profile=WinXPSP2x86 psscan`

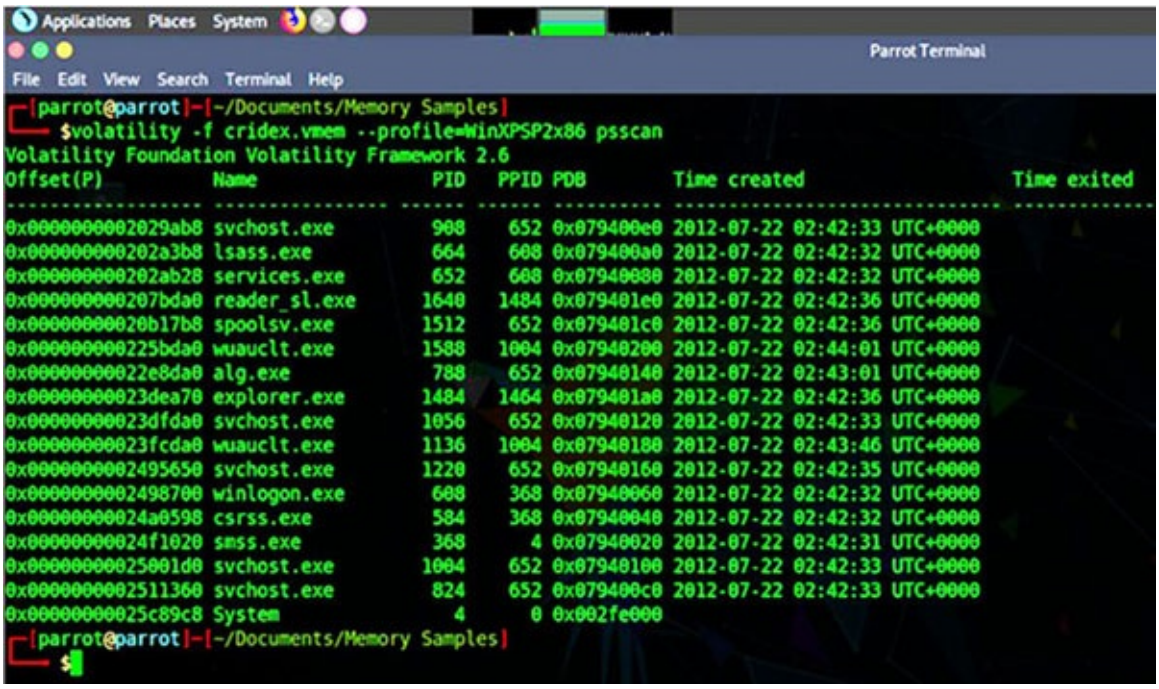


Figure 11.26: *Running the process scan*

8. **pstree**: This plug-in of the volatility tool shows processes (as presented in *figure 11.27*) with their PID and PPID's. The purpose is to display the processes in a tree (parent/child) format. This will show the linking of the process with the parent process.

   Command: `volatility -f cridex.vmem --profile=WinXPSP2x86 pstree`

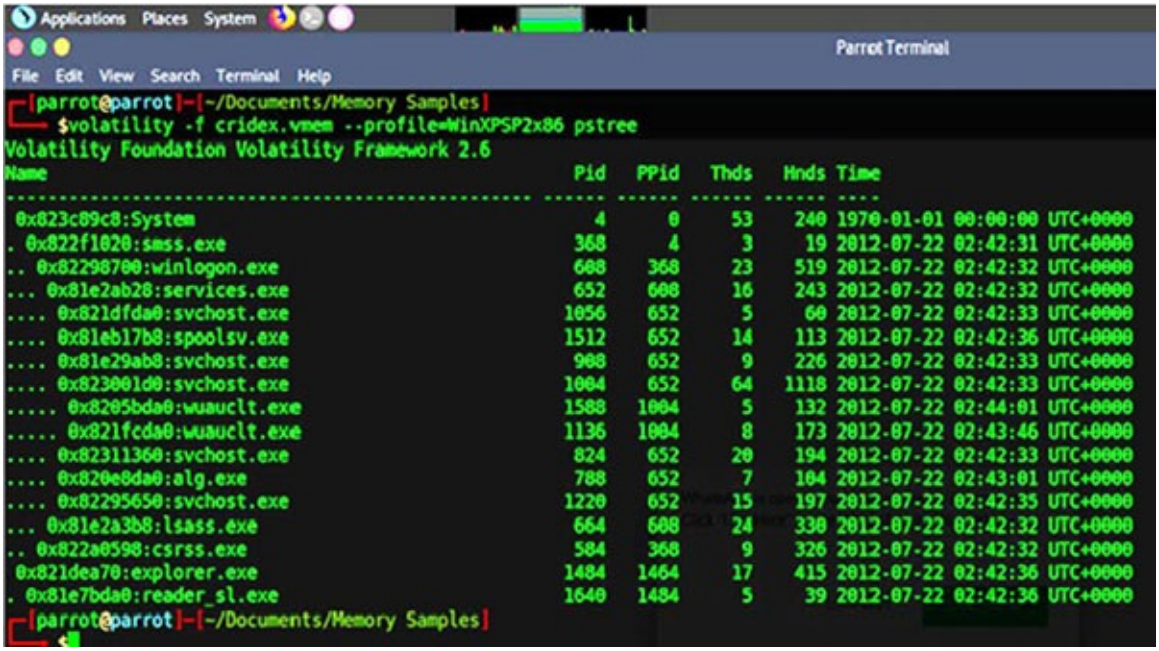


Figure 11.27: *Generating the process tree*

9. **`procdump`**: This plug-in (shown in *figure 11.28*) of the volatility tool is used to dump a process's executable.

   Command:

   ```
   volatility -f cridex.vmem --profile=WinXPSP2x86 procdump -p 908 --dump-dir=/home/parrot/Documents/file /home/parrot/Documents/executable.908.exe
   ```

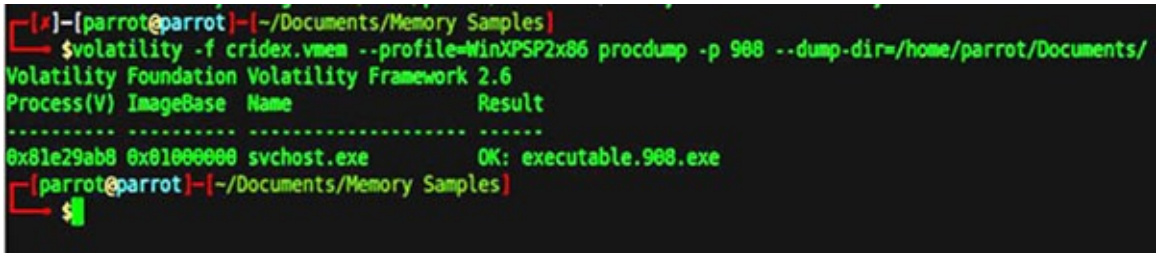


Figure 11.28: *Generating the procdump*

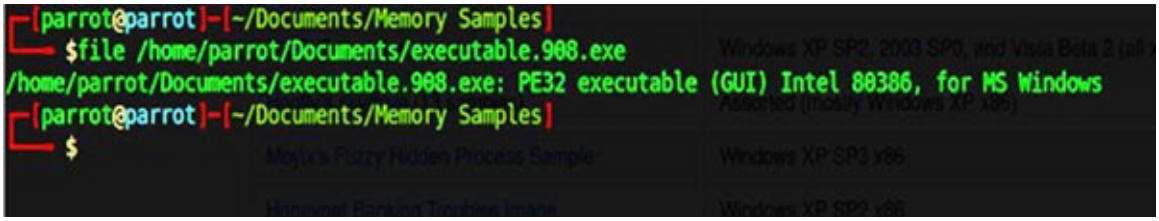


Figure 11.29: *Executable file created*

10. **`memdump`**: This plug-in (shown in *figure 11.30*) of the volatility tool is used to extract all memory resident pages in a process (see memmap for details) into an individual file.

    Command:

    ```
    volatility -f cridex.vmem --profile=WinXPSP2x86 memdump -p 908 --dump-dir=/home/parrot/Documents/file /home/parrot/Documents/908.dmp
    ```

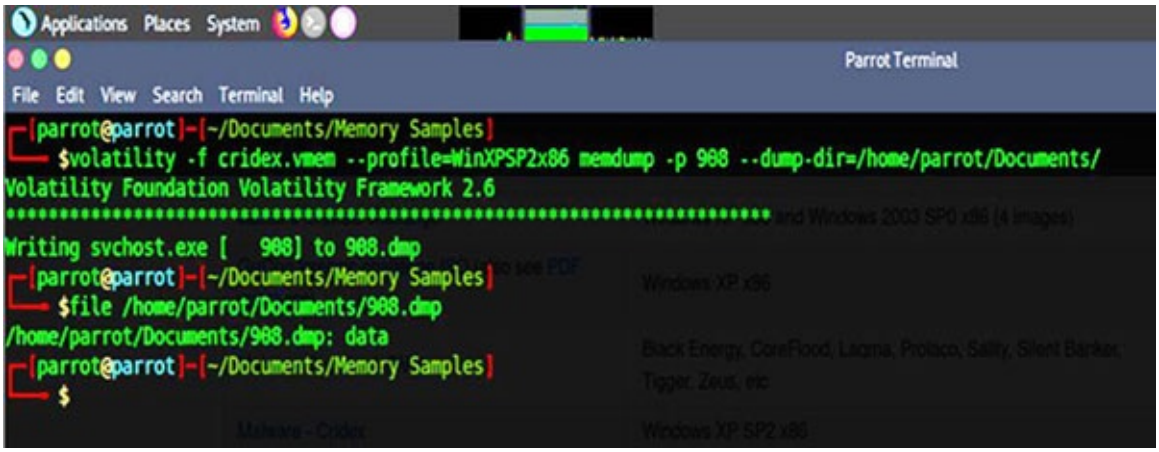


Figure 11.30: *Executable memory dump*

11. **Kernel debugger** (**KDBG**) block address and the right system profile may be positively identified using the plug-in known as kdbgscan. It only searches for KDBG header signatures connected to the volatility profiles. This mainly helps to clear up any misunderstandings that can arise if the Pslist plug-in fails to display any processes in the process list. If a KDBG with an incorrect PsActiveProcessHead pointer is discovered earlier in a sample, this can take place.

    Command: `volatility -f cridex.vmem --profile=WinXPSP2x86 kdbgscan`

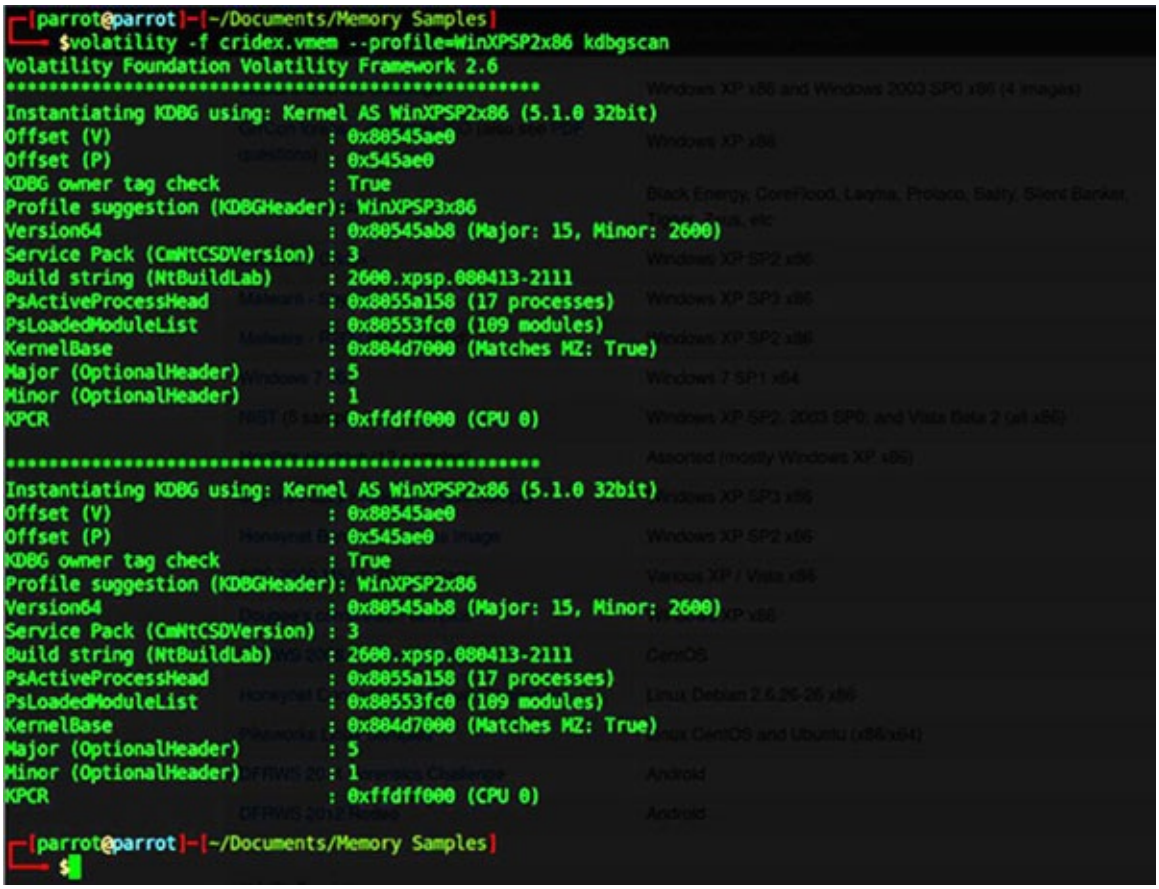


Figure 11.31: *Executable KDBG scan*

12. **Kernel Processor Control Region** (**KPCR**) structures may be found with the plug-in kpcrscan. The kernel uses a KPCR as a data structure to hold processor-specific information. Kpcrscan looks for possible KPCR values and dumps them. On a multi-core system, each CPU has its own KPCR. The CPU, which is a single-core processor, is shown in *figure 11.32*.

    Command: `volatility -f cridex.vmem --profile=WinXPSP2x86 kpcrscan`

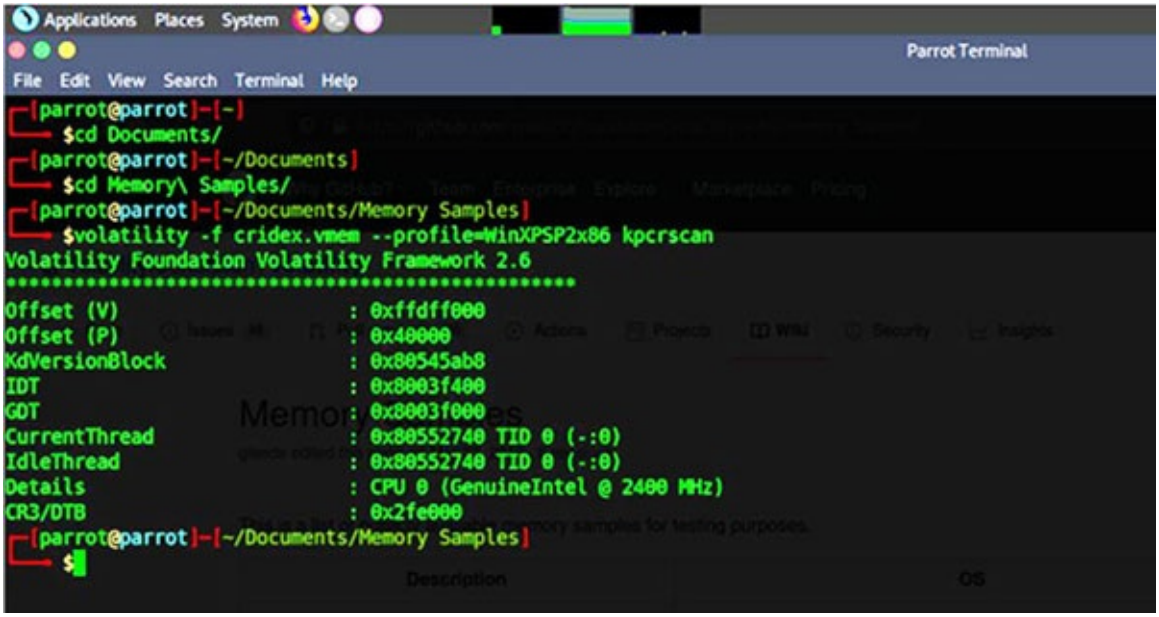


Figure 11.32: *Executing KPCR scan*

Lab 4: Malware analysis

The term "*Malicious Software*" is referred to by the single-word term "*Malware*." The word "*malware*" refers to a broad range of dangerous software created by online criminals. Cyberattacks are now affecting an increasing number of internet users, and enterprises of every size are increasingly a target. The malicious software allows for backdoor entrance into computers, allowing for the theft of many types of data, including private information. Understanding how a piece of malware works and its potential consequences is the process of malware analysis. It is important to understand that malware can have a wide range of activities and that malware code can vary greatly. These could manifest as Trojan horses, worms, malware, and viruses. Each sort of malware collects data about the infected machine without the user's knowledge or consent.

Aim: To perform malware analysis of Pony malware in windows.

Pony virus, often referred to by the names Pony Stealer, Pony Loader, FareIT, and other variations, is a password-stealing program that has the ability to decrypt or unlock credentials for over 110 different apps, including VPN, FTP, e-mail, instant messaging, Web browsers, and many more. Pony Stealer is incredibly harmful, and once it takes over a computer, it turns it into a botnet that it may exploit to spread to other computers.

Pony is more than simply a tool for stealing credentials or using bitcoin. Actually, it is a botnet controller that preys on Windows computers. In order to create and administer its botnets, it includes a control panel, database and user administration, logging, and statistics.

Prerequisites:

1. Download the Pony malware sample from **https://any.run/malware-trends/pony**.
2. Extract the file (as shown in *figure 11.33*) from password protected zipped file.

Figure 11.33: *Malware is password protected*

3. Use the extracted file carefully to get info regarding the pony malware using the tools listed as follows:

    - Download the HxD hex editor. Link: **https://mh-nexus.de/en/hxd**
    - Download Exeinfo PE, PE file identification program. Link: **https://exeinfo-pe.en.uptodown.com/windows#:~:text=Exeinfo%20PE%20is%20a%20program,exe%2**
    - Download PeStudio, Malware Initial Assessment Tool. Link: **https://www.winitor.com/**

    Note: Guidelines for malware analysis.

    - Virtual machines (VMs) should always be used for malware analysis. VM should be updated and must be used in Host-Only network configuration.
    - No USB drive should be plugged into the VM.
    - No important data should be present on the VM.
    - Disable all the shared folders between the Host and VM.
    - Always use compressed and password-protected malware samples only to prevent the random execution of malware.

Performing Malware Analysis:

HxD

Hex editors are among the most straightforward malware analysis tools, yet they may be quite effective. A hex editor like HxD is made to display both the ASCII interpretation and the file's raw hexadecimal format.

1. Open the malware in the HxD hex editor.
2. Look for 4D 5A (signature) in the first Offset(h) under 00 and 01 (first two bytes).
3. 4D 5A refers to a **Portable Executable** (**PE**) file, i.e., (`.exe`) or (`.dll`) files.
4. Search for the notice "*program cannot be run in DOS mode*" under the Decoded text, verifying it is a PE program.

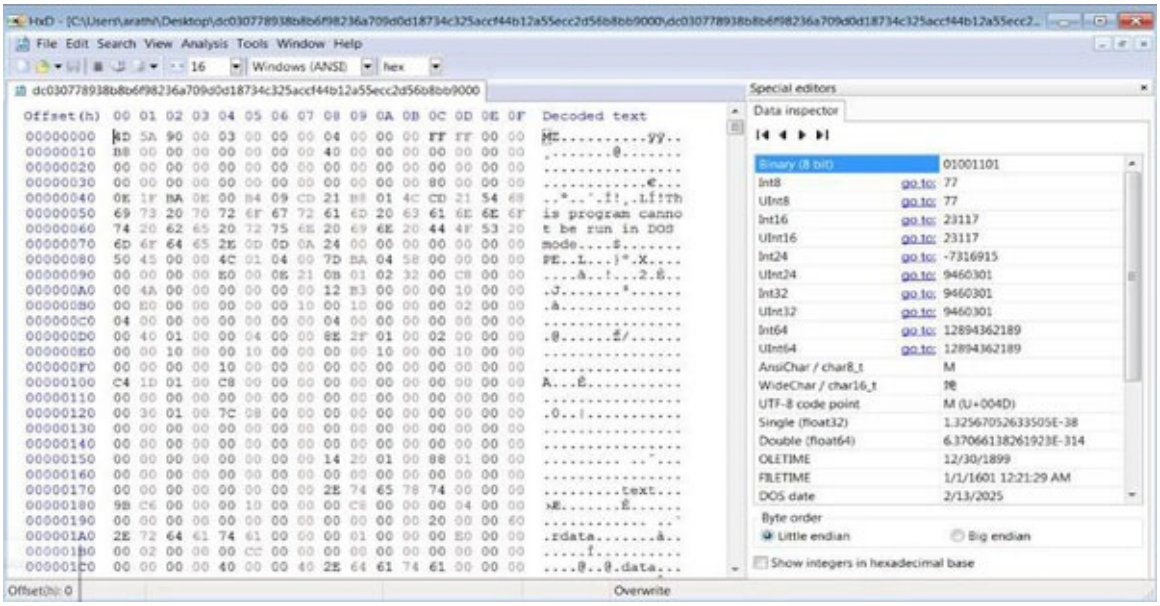


Figure 11.34: *Opening the malware in HxD*

Exeinfo PE

A little software called Exeinfo PE may be used to display numerous details about any executable file. This tool assists in checking all the attributes of (**.exe**) files and verifying them. You may alternatively rename the file, run the **.exe** directly, or just delete it. The precise size and location of the entrance are additional pieces of information. To put it simply, you have access to a vast array of options for editing any Windows executable file.

1. Open the malware in ExeInfo PE.

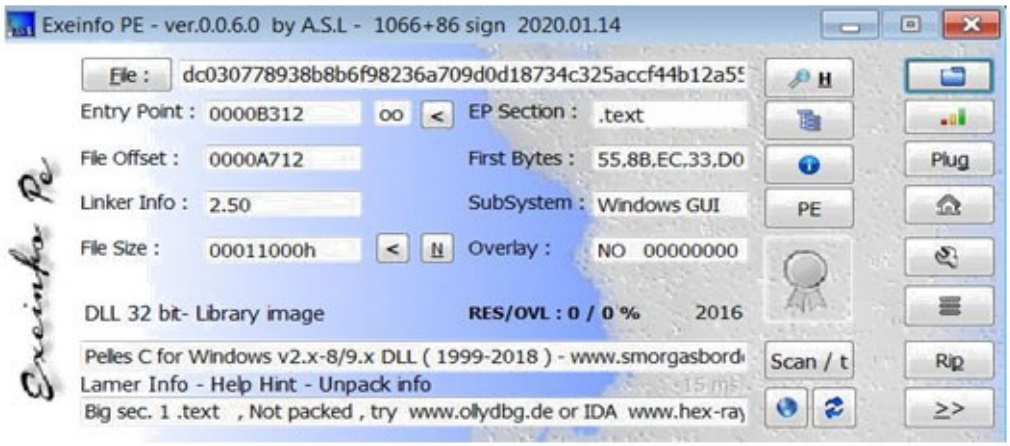


Figure 11.35: *Opening the malware in Exeinfo PE*

2. As no packing information is displayed, hence, no packer is used to pack the malware (not packed) to hide its extinction—detailed view of the section viewer.

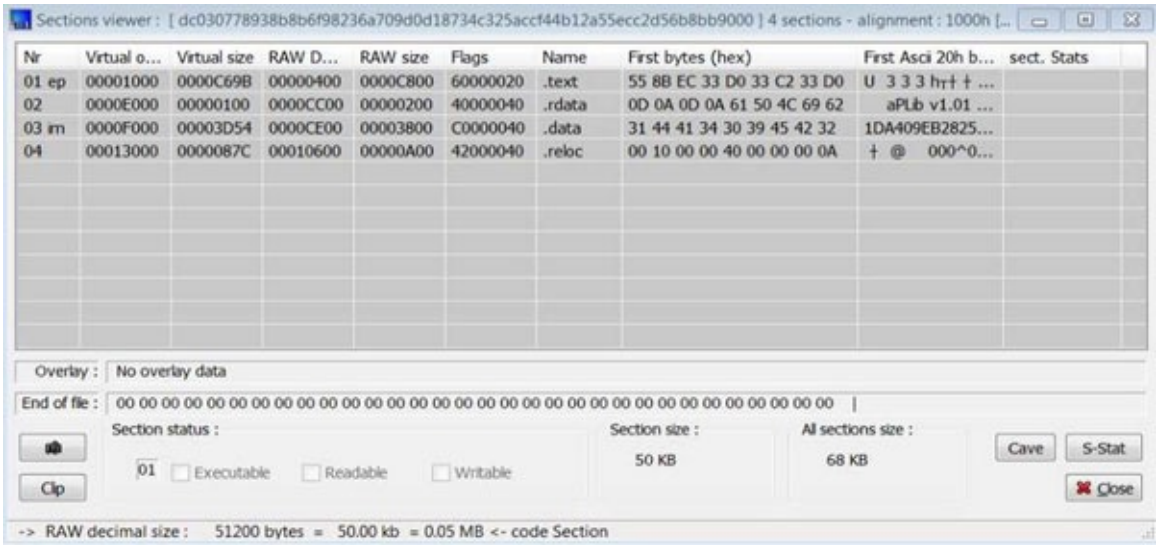


Figure 11.36: *Opening the malware in HxD*

3. Detailed view of PE header section.

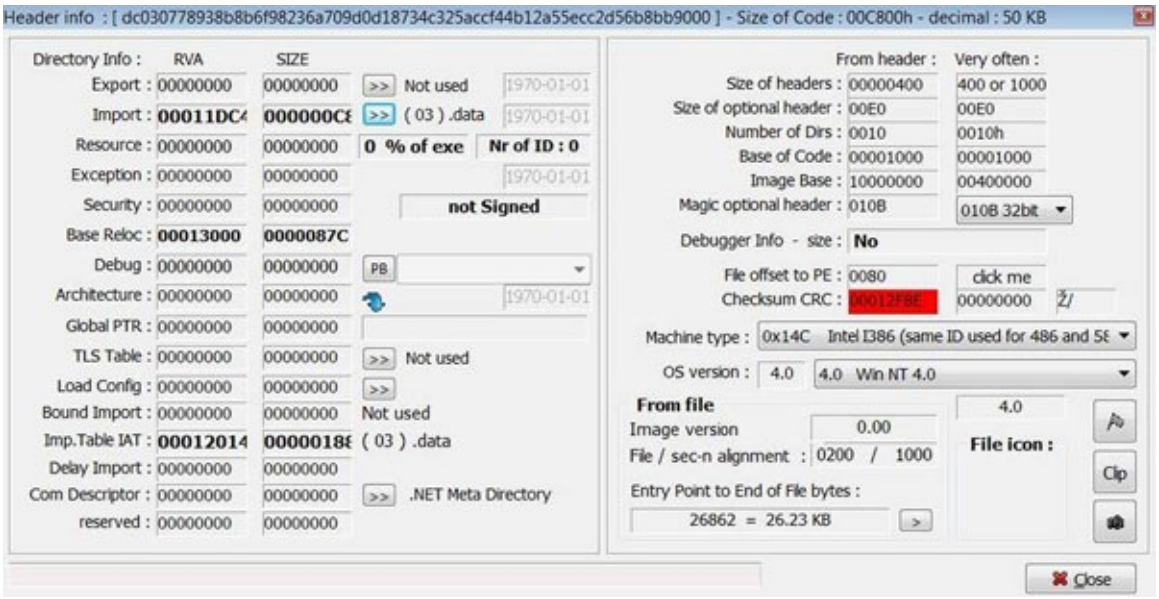


Figure 11.37: *Analyzing the PE header*

PeStudio

PeStudio is a utility tool performing the static analysis of any Windows executable binary (like malware). A file being analyzed with PeStudio is never launched. PeStudio seeks to identify suspicious artifacts in executable files to facilitate and quicken Malware Initial Assessment.

1. Opening the malware in PeStudio.

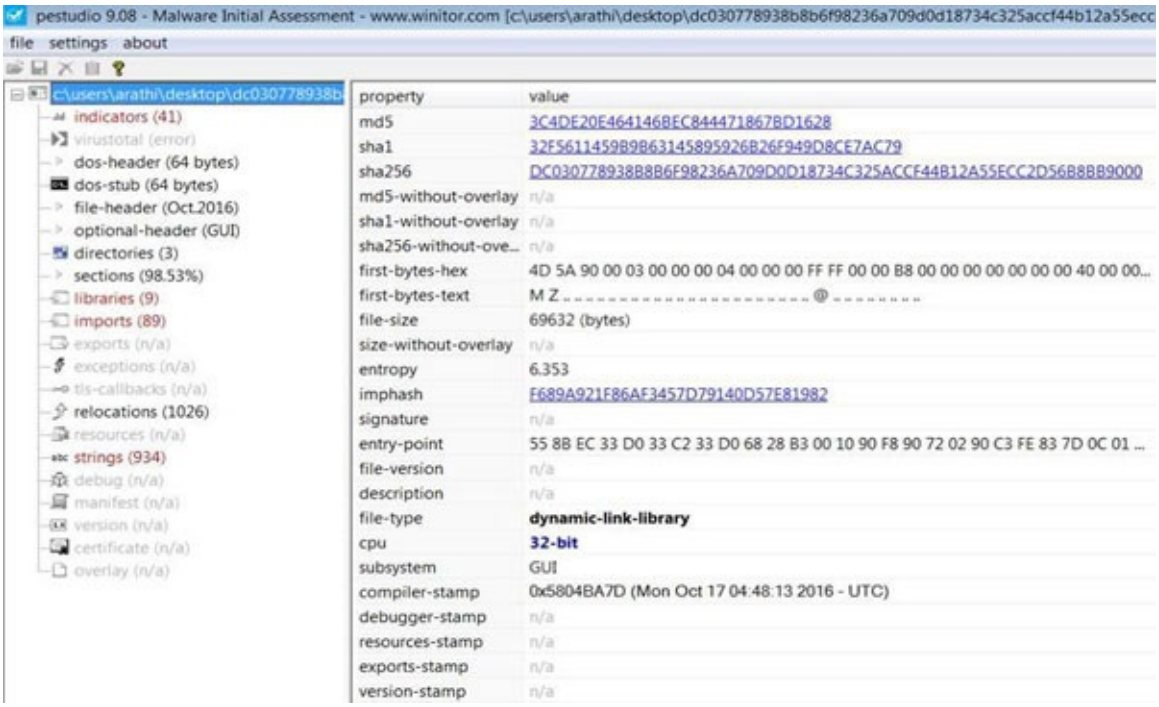


Figure 11.38: *Opening the malware in PeStudio*

2. Basic info is displayed initially; browse different sections to get more information about the malware.
3. In the indicators section, the backlisted file referencing objects and accessing URLs are displayed, which might signify the information collected by the malware is sent over the internet.

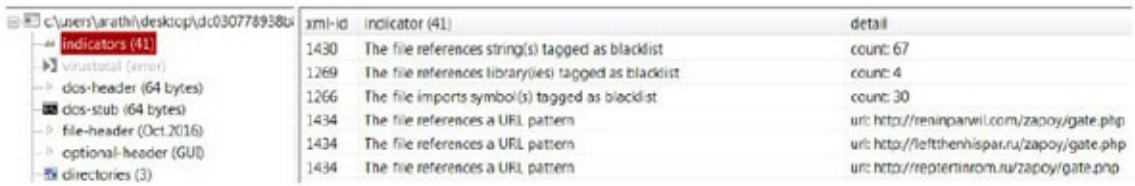


Figure 11.39: *Highlighting the indicators*

4. In the strings section, all the suspicious strings are detected by the tool according to the predefined set of strings present in the tool.

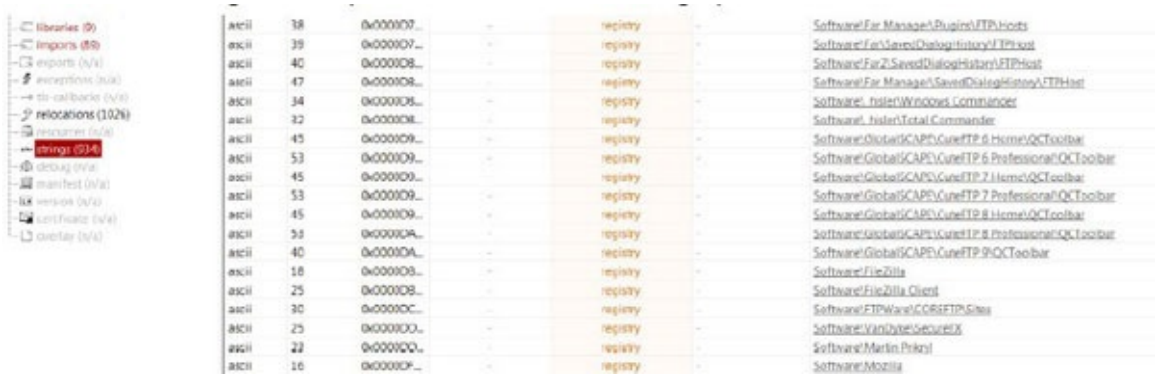


Figure 11.40: *Analyzing the strings*

5. We can have file names, URLs, IP Addresses, and Registry Keys as strings, which provide valuable information about the malware functionality.
6. In the libraries section, the list of libraries is displayed which are required to be loaded (dll) and blacklisted the suspicious ones according to the predefined tool data.

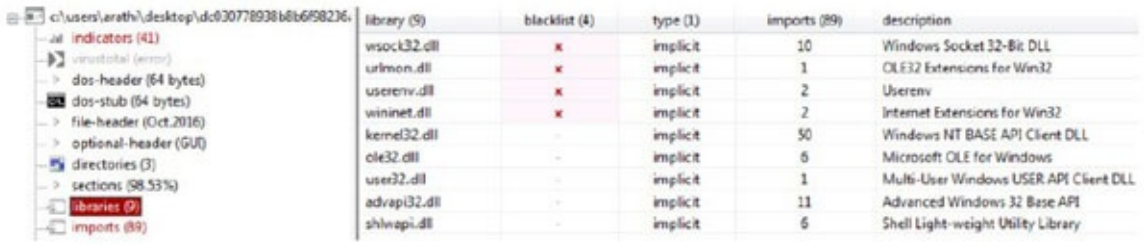


Figure 11.41: *Analyzing the libraries*

7. In the imports section, the list of import functions used by the malware is displayed, and blacklisted the suspicious ones according to the predefined tool data.

Figure 11.42: *Analyzing the exports*

Identification of the malware:

- For accurate identification of malware samples, use unique Hashes instead of using file names. Hashes can be used to identify malware online using malware analysis sites like Virus Total.

  Link: **https://www.virustotal.com/gui/**

- Search the file hash of the malware in the search tab of the Virus Total website. The file is considered malicious by 63 out of 69 different antiviruses, which is a strong sign of the file is malware.

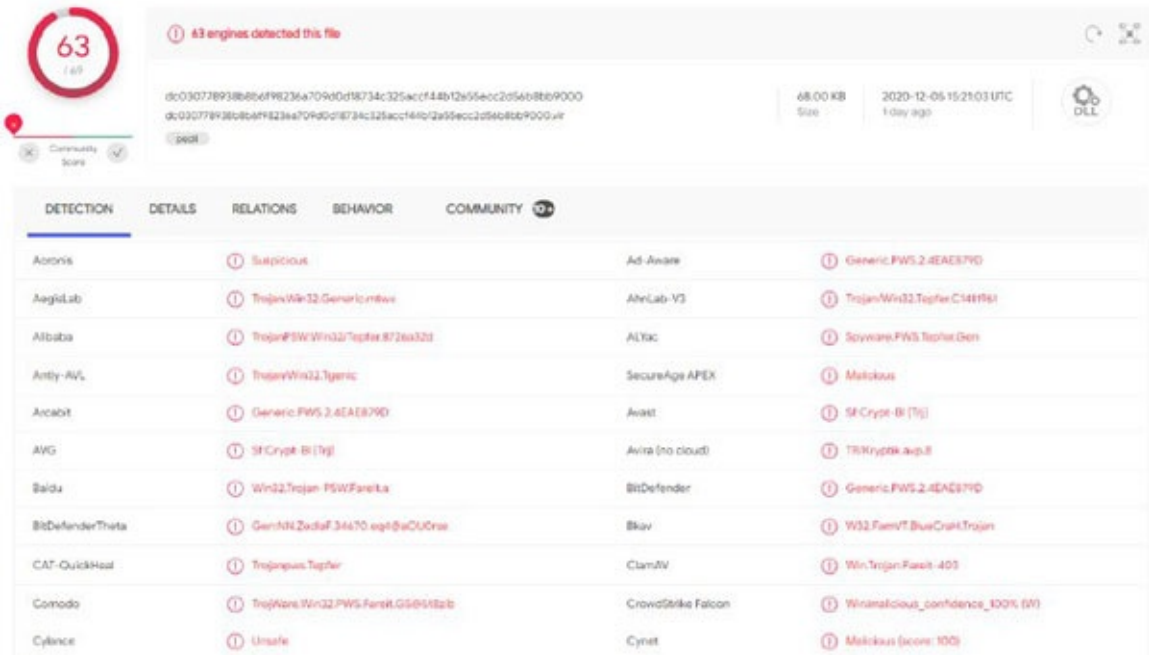


Figure 11.43: *Analyzing the virus total report*

Lab 5: data hiding—Steganography

Objective: Learn Steganography using Kali Linux

A steganography application called Steghide can conceal data in several types of picture and audio files. Because the color- and sample-frequencies are left unchanged, the embedding is immune to first-order analytical techniques. Steganography is the art of concealing a secret message within a larger, seemingly unimportant message. This is a step forward over standard encryption, which only hides the original message. Steganography attempts to hide the fact that communication even exists. Steganography communications frequently seem like something different than the original (secret) message, such as an image, music, bigger text, and so on. The embedded data can be compressed, encrypted, and automatically checked for integrity using a checksum, among other features. For usage as a cover file, the JPEG, BMP, WAV, and AU file types are supported. The structure of the secret data is not constrained.

Steganographic terminology includes the following:

- **Plaintext**: The initial, necessary hidden message.
- **Ciphertext**: Secret communications are frequently first encrypted using conventional techniques. The Ciphertext is the name of an encrypted communication.
- **Covert Text**: A bigger, innocuous-looking piece of data that serves as a container for plaintext and Ciphertext. This might be a graphic, audio file, text, and so on.
- **Stegotext**: Information produced as a result of encoding plaintext or Ciphertext within the covert text.

Steps to perform:

1. Use Linux or Kali Linux OS to run Steghide; if not present, install using `sudo apt install steghide`. Display a help screen. No arguments are required.

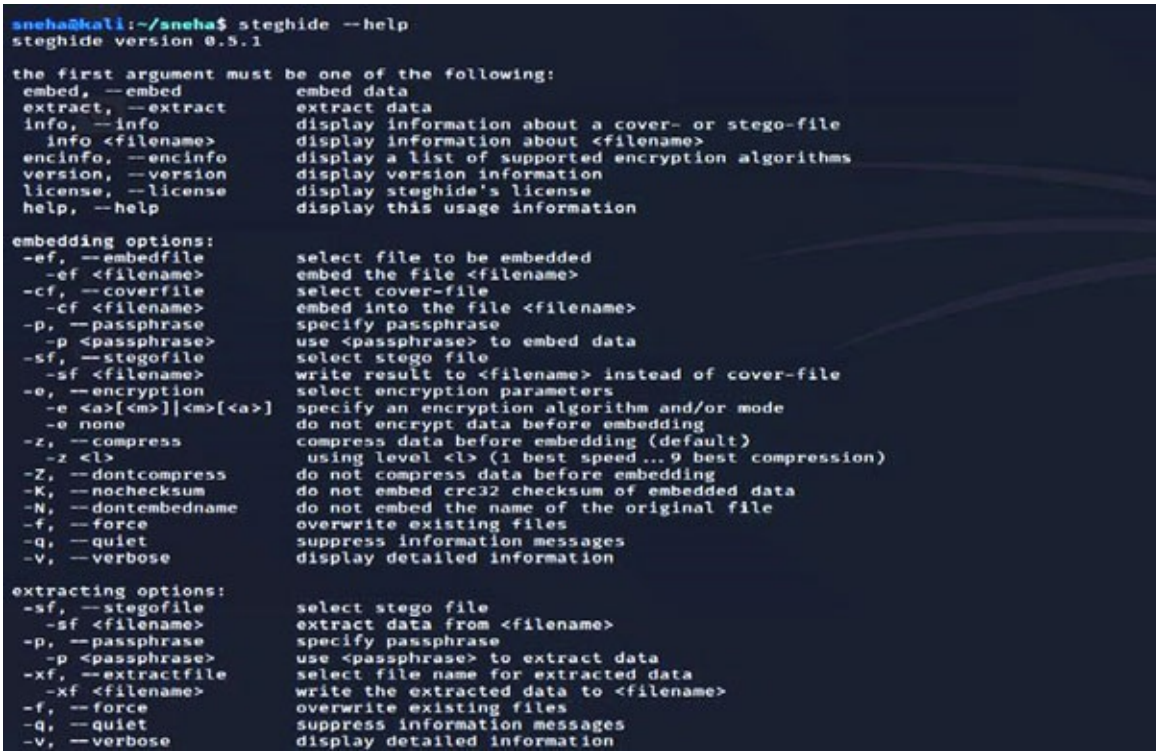


Figure 11.44: *Exploring the steghide help*

2. To embed text in images using steghide, enter the text in a file created using the CAT command.

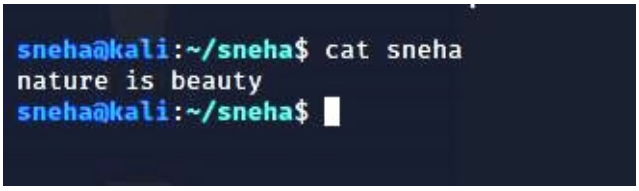


Figure 11.45: *Checking the file*

3. Once data is entered into the file. Now, let us embed the text in the `image.jpg`. We need to run this command `steghide embed -cf image.jpg -ef sneha`. In the command. It will ask for a passphrase, enter it and enter it again. We should use the embed command if we want to embed secret data in a cover file. It has arguments: `-ef`, `--embedfile` filename
4. Indicate which file will be inserted (the file that contains the secret message).

   Note that the stego file contains the actual file name embedded by steghide. The default behavior when extracting data (see as follows) is to store the embedded file under its original name in the current directory. Steghide will read the sensitive information from standard input if this option is left out or if the filename is set to -. -cf, filename—coverfile
5. Name the cover file in which the data will be embedded.

   One of the following formats—AU, BMP, JPEG, or WAV—must be used for the cover file. Based on header data, the file format will be automatically determined (the extension is not relevant). Steghide will read the cover file from standard input if this argument is not provided or if the filename is empty.

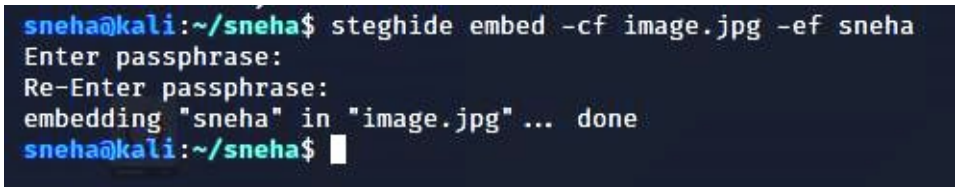


Figure 11.46: *Embedding the file*

6. The text in the image has been embedded; let us extract that embedded text from the image.

   We need to run this command steghide extract -sf image.jpg. It will ask for a passphrase to enter it. The data has been extracted from the file. If you have received a file that contains a message that has been embedded with steghide, use the extract command to extract it. The following argument: `-sf, --stegofile` *`filename`*
7. Name the stego file (the file that contains embedded data). Steghide will read a stego file from standard input if this argument is not provided or if the filename is.

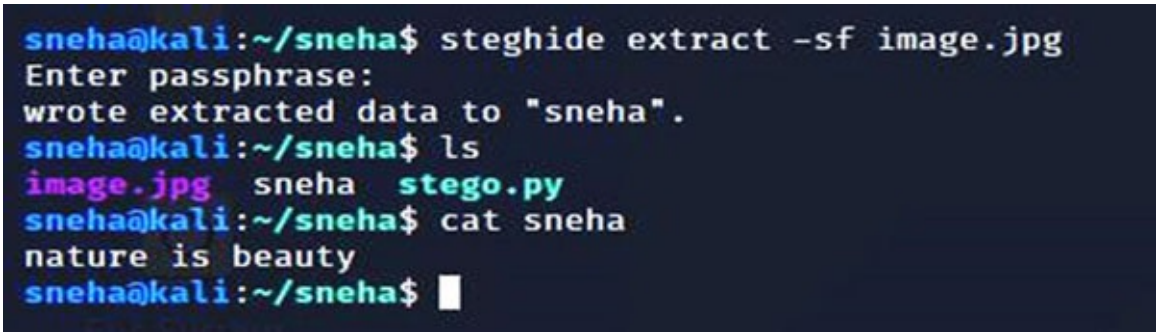


Figure 11.47: *Extracting the file*

Steganography on Windows

1. Download and install OpenStego

   This app provides two main functionalities:

   - **Data Hiding**: It can hide any data within a cover file (for example, images).
   - **Watermarking (beta)**: Watermarking files (for example, images) with an invisible signature. It can be used to detect unauthorized file copying.

2. The interface of Open Stego:

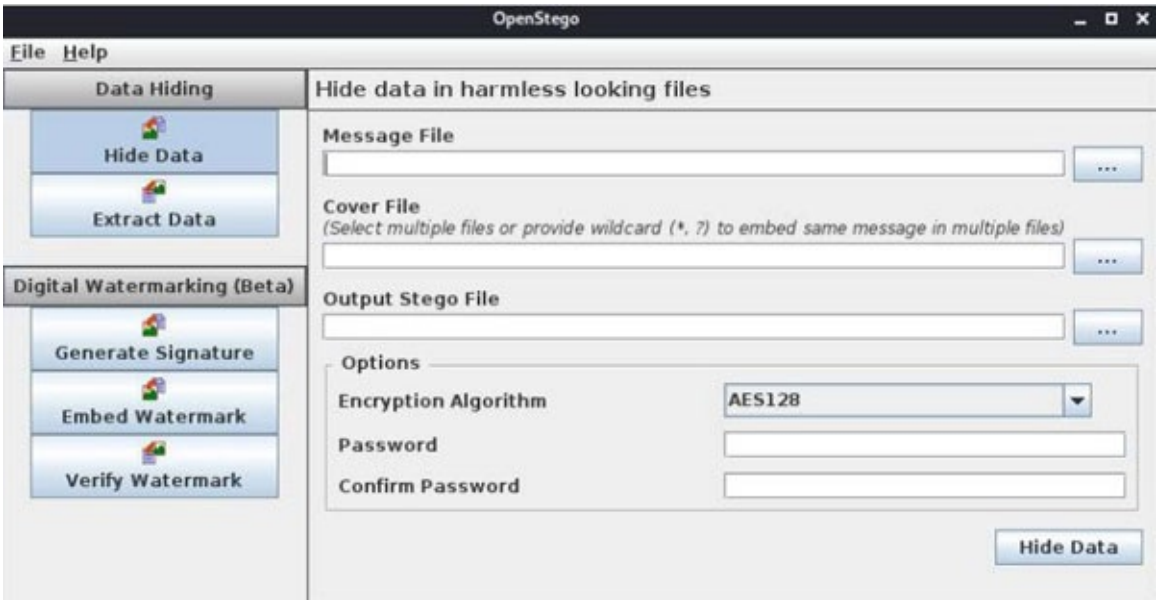


Figure 11.48: *OpenStego interface*

3. Hiding text in an image and checking the size before embedding text.

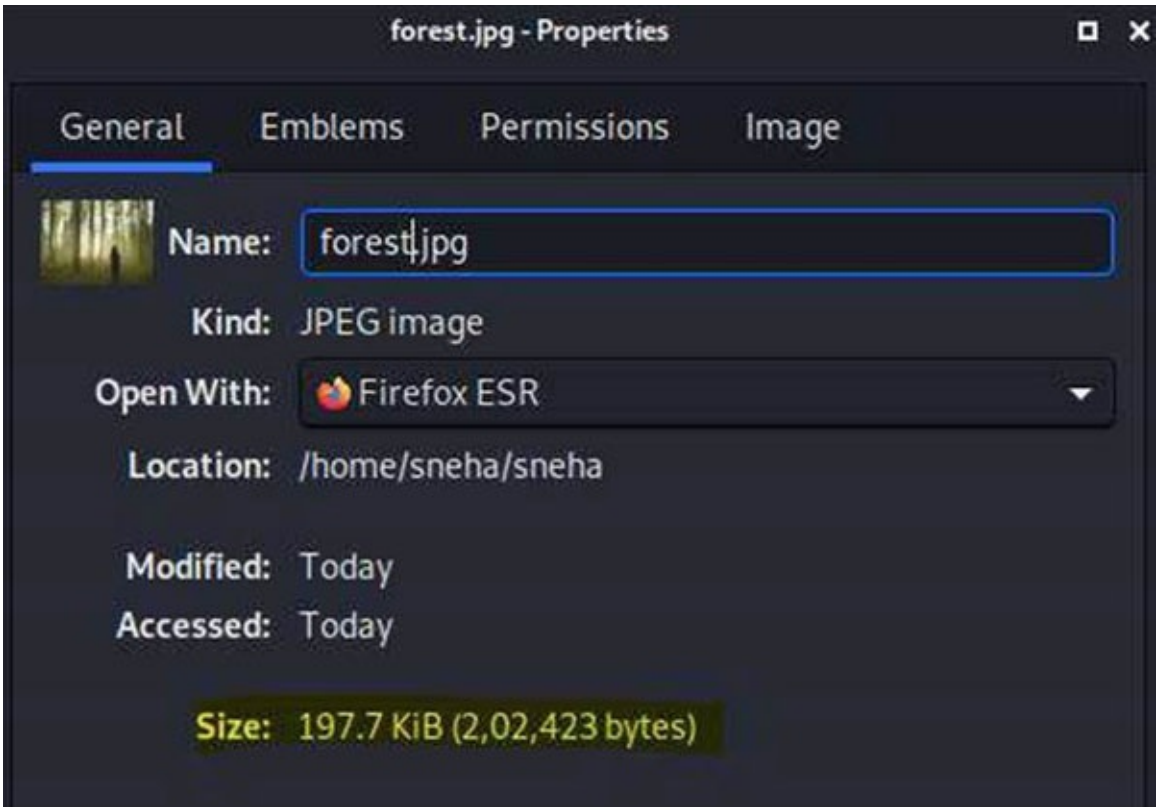


Figure 11.49: *Hiding the text*

4. The cover file can be of types `.bmp`, `.gif`, `.jpeg`, `.jpg`, `.tiff`, `.wbmp`, and so on.

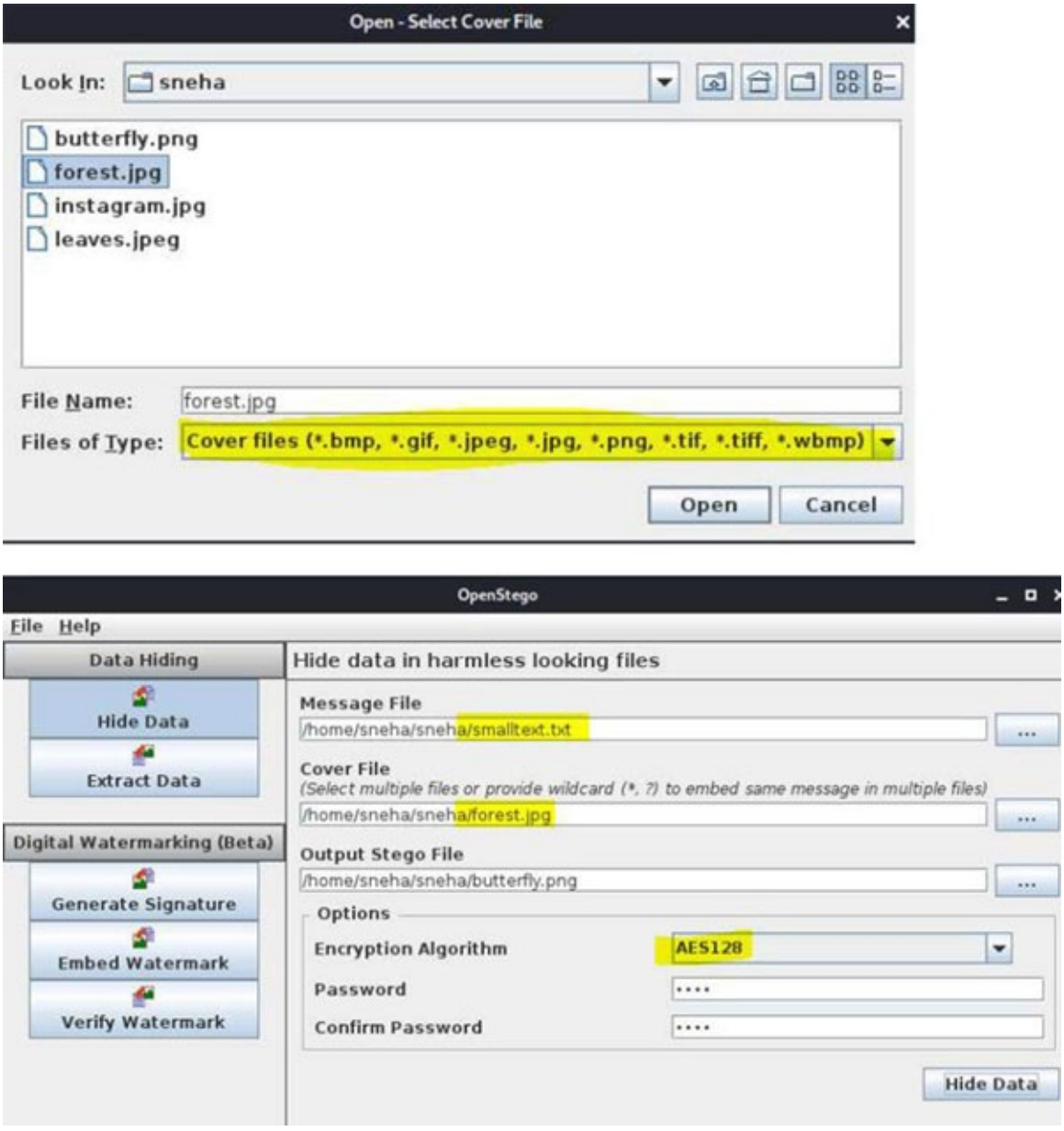


Figure 11.50: *Using the encryption*

5. The output is named as `butterfly.png`.

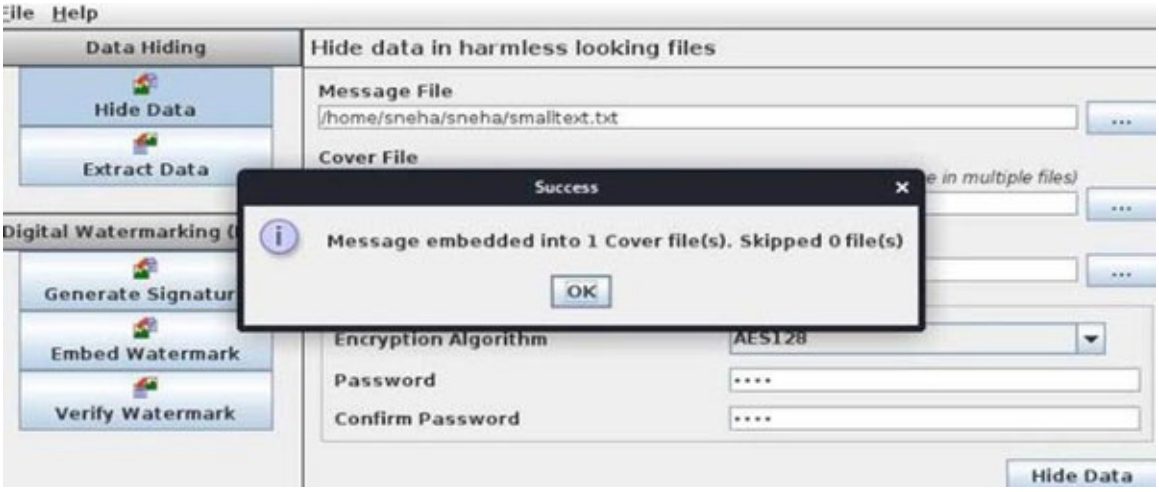


Figure 11.51: *Successfully embedded*

6. After embedding the text in the file, the size of the image changed.

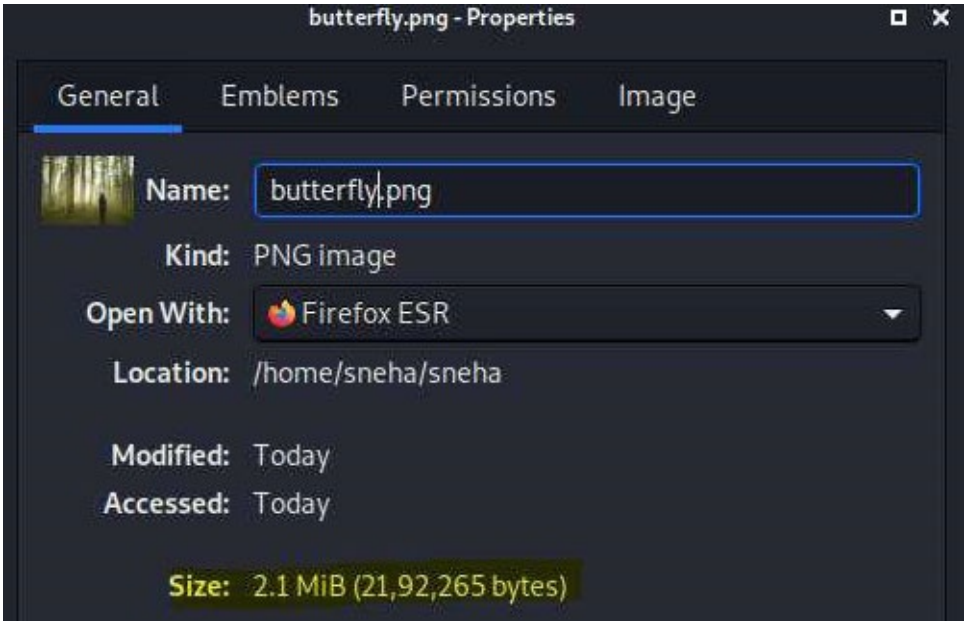


Figure 11.52: *Checking the properties of the file*

Observations

- The size of the image increased from 197KiB to 2.1MiB after embedding the text. Earlier, the image was of `.jpg`; now, the extension has been changed to `.PNG`.
- No distortion in the image, which is eye-catching

Hiding image to the text file

1. Hiding pdf file into image.

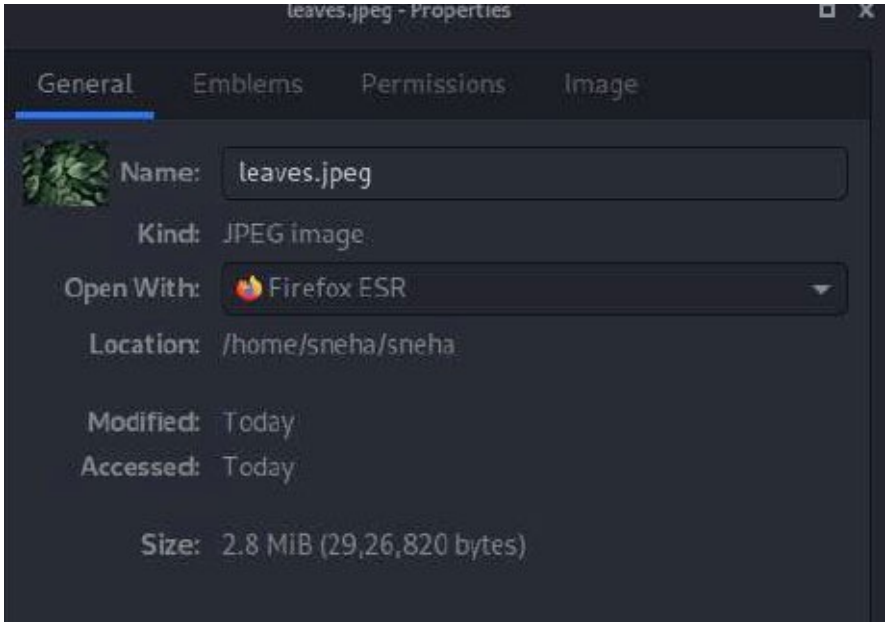


Figure 11.53: *Checking the files*

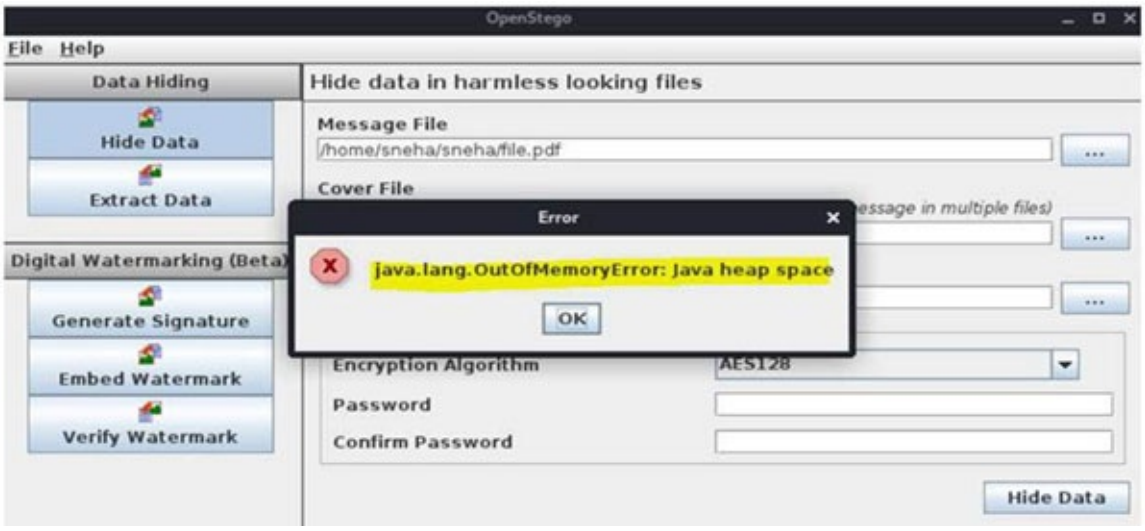


Figure 11.54: *Error*

Observation

The size of the pdf is 6MiB, and the size of the image is 2MiB. As the size of the image is less, the pdf cannot be embedded in it as it shows the error `java.lang.OutOfMemoryError`.

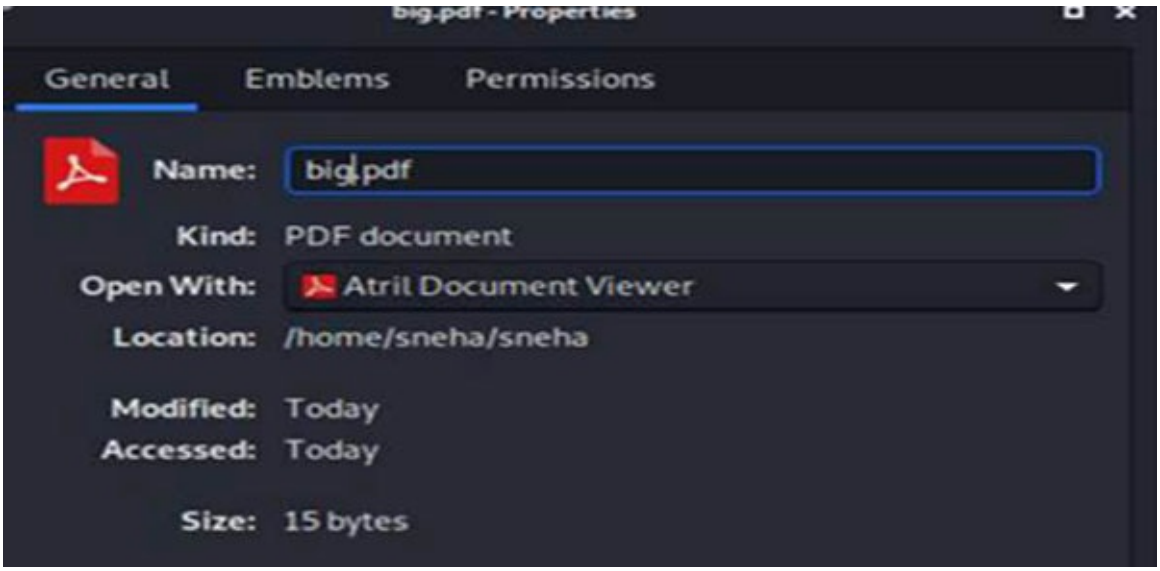


Figure 11.55: *Checking file properties*

Figure 11.56: *OutofMemory exception*

Observations

Even though the size of the pdf is as small as 15 byes, still it can be hidden in the image. So, pdf cannot be hidden in the image using Open Stego.

Hiding a zip file in an image:

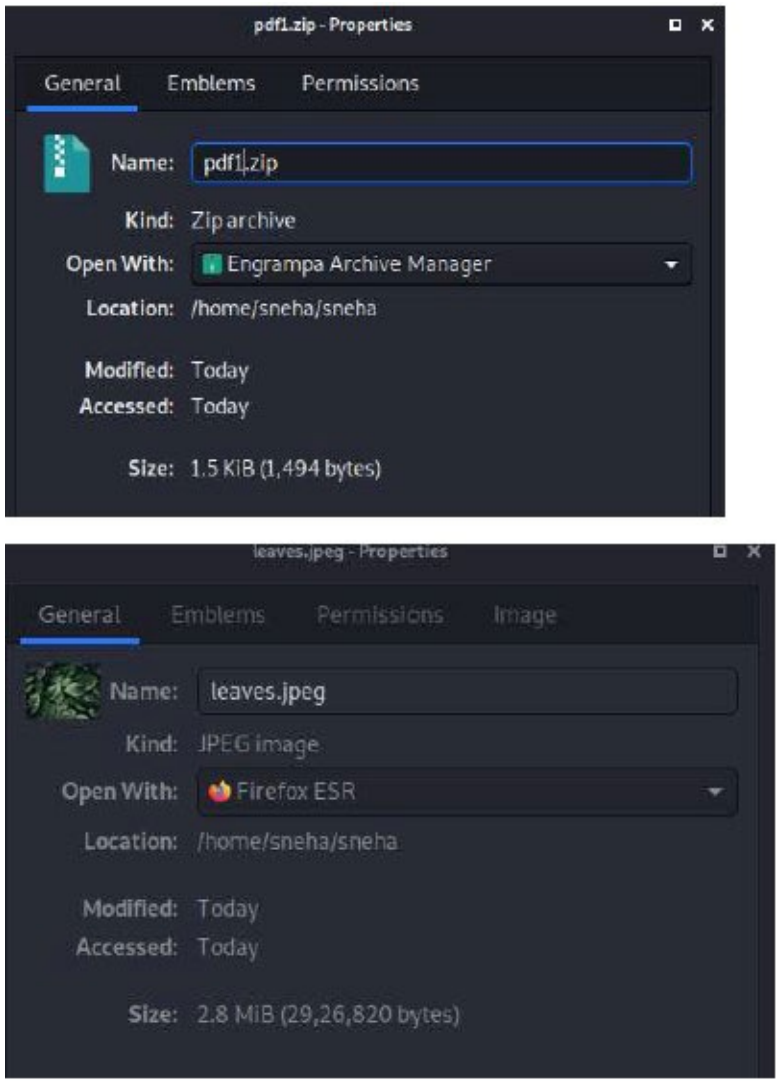


Figure 11.57: *Hiding zip file in an image*

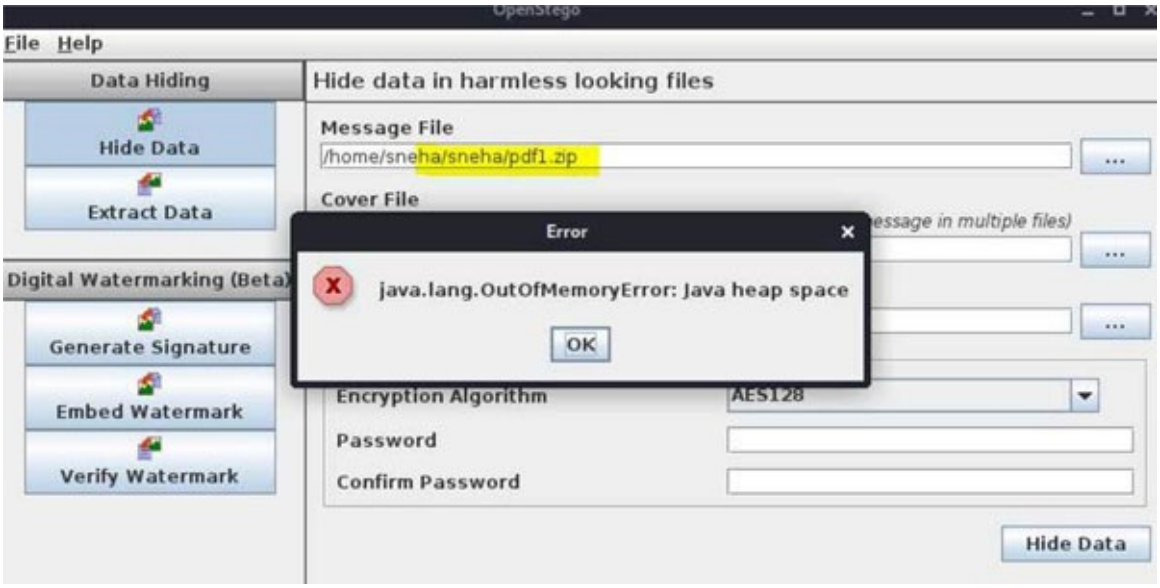


Figure 11.58: *Showing exception*

Observation

We cannot hide a zip file in an image using open stego.

Extracting data from the image

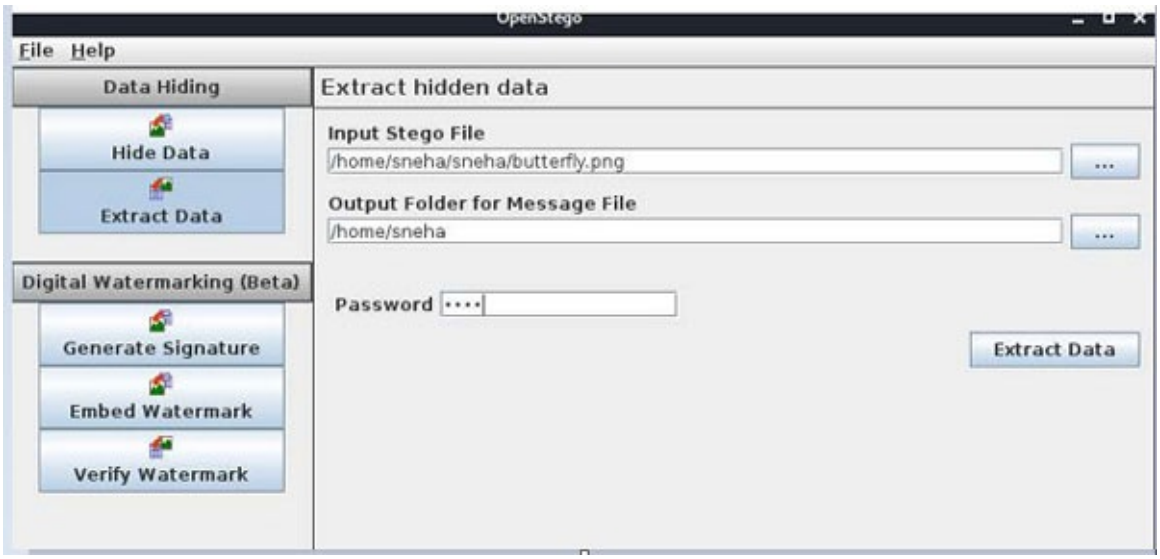


Figure 11.59: *Setting password*

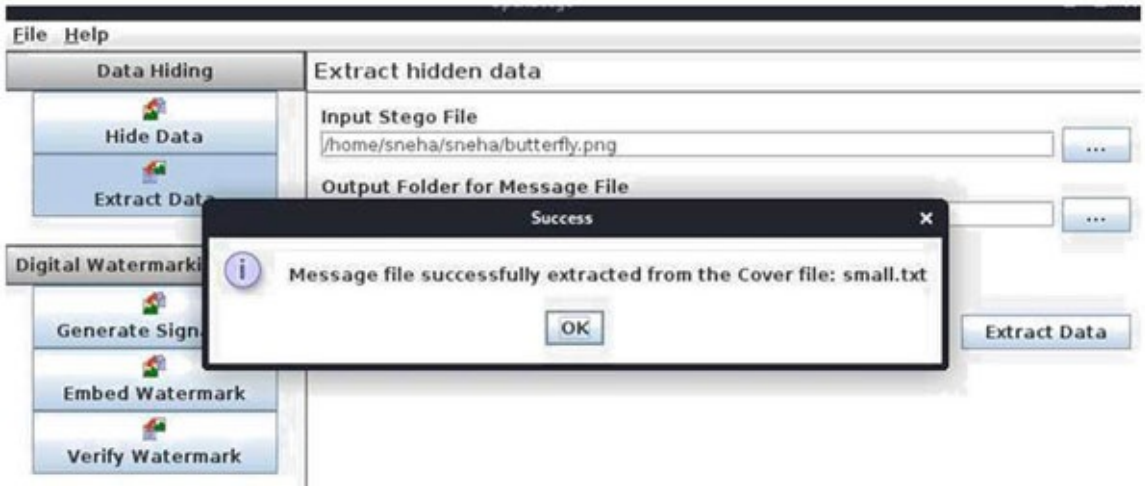


Figure 11.60: *Successful extraction*

Extracted file

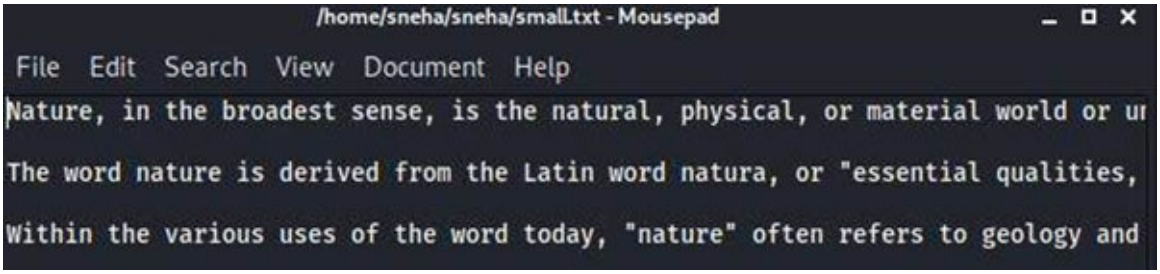


Figure 11.61: *Extracted file*

Lab 6: Recovering deleted files

The most basic of the abilities required for a forensic investigator is undoubtedly retrieving deleted files. As you are aware, until they are replaced, "*deleted*" files are still there on the storage media. Simply by

deleting such documents, the cluster becomes ready for overwriting. This implies that even if the accused erased evidence files, we would still be able to recover them up until the point at which the file system overwrites them. The Sleuth Kit, an open-source tool for finding and retrieving deleted files, will be used in this experiment. We will use the Sleuth Kit with our Windows testing environment, even though it was originally designed for Linux. We will use the GUI interface called Autopsy that was created for TSK.

1. Install it on your system.

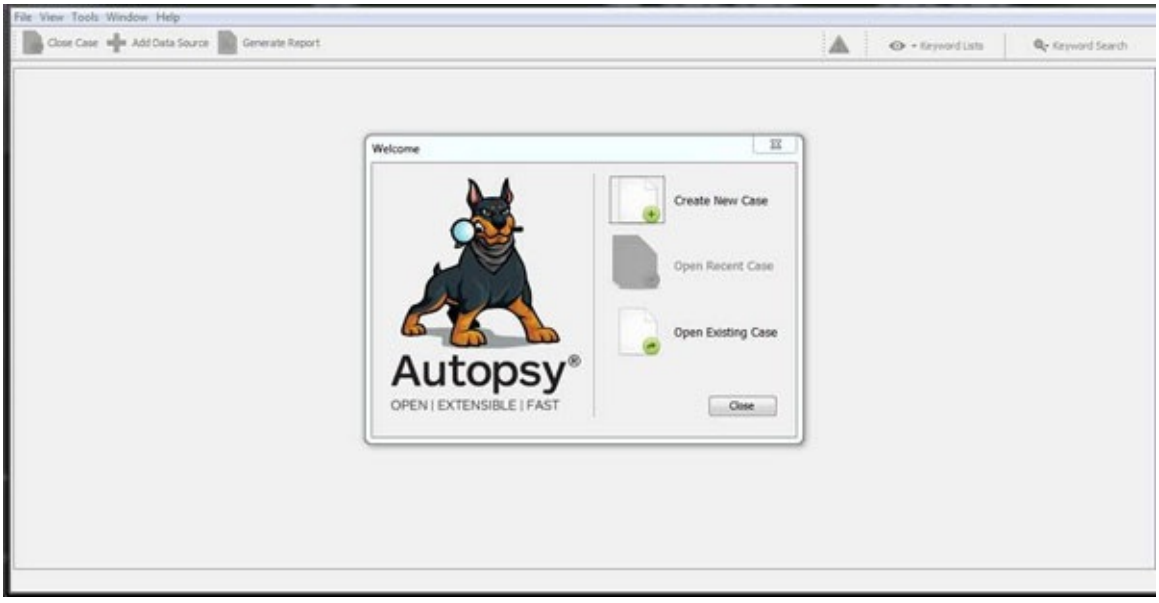


Figure 11.62: *Autopsy interface*

Following the installation and launch of Autopsy, you will see a screen similar to the one seen previously.

2. Next, choose **`Create New Case`**.
3. Once you do, a new window requesting a name for your new case and the directory in which you want to keep your cases will appear. Type `New Case 101` into the `C:\\Cases base directory`.

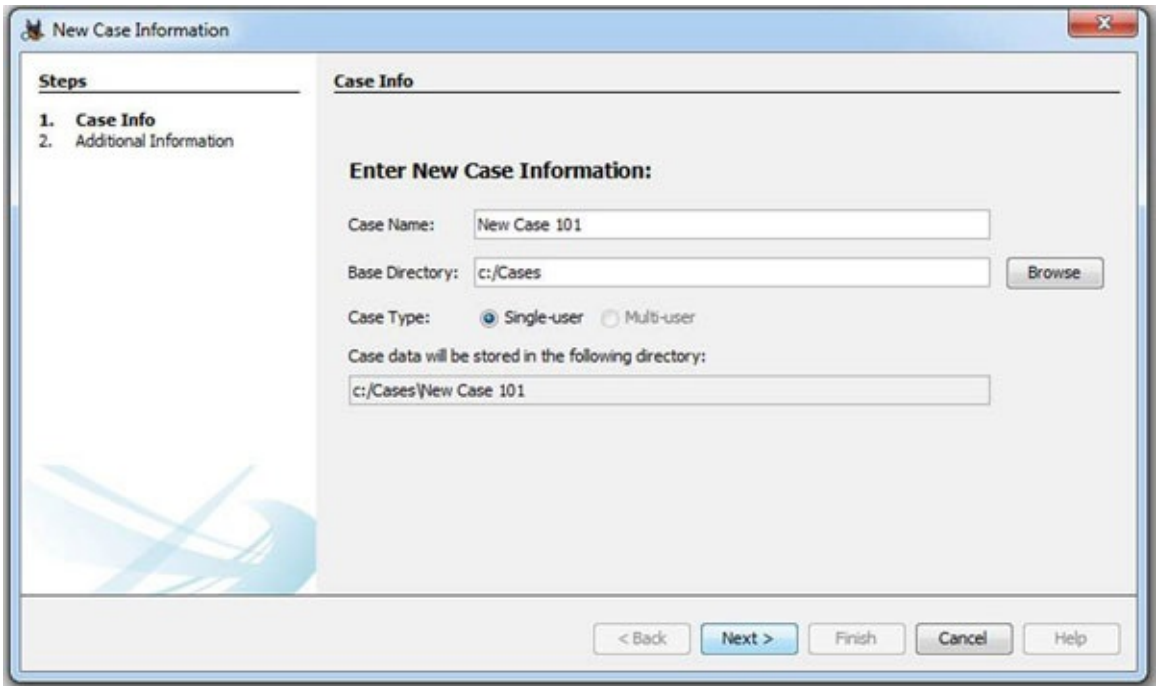


Figure 11.63: *Entering the new case information*

4. Click **`Next`** now.
5. A new window will pop up, asking for the case number and examiner's name. Give it a case number of 101 and the examiner's name or initials.

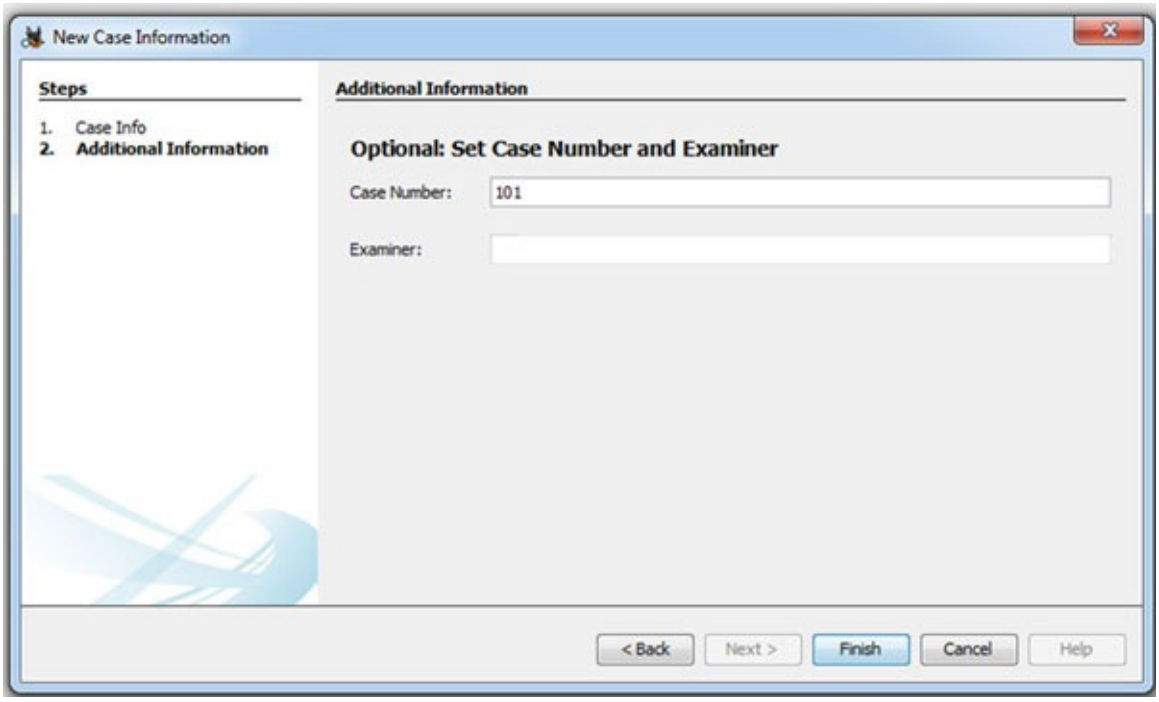


Figure 11.64: *Set case number*

6. Select **`Finish`**.
7. Then, select **`Add New Data`** from the menu in the upper left corner. When you do, a pop-up labeled **`Add Data Source`** will appear. Select **`Image File`** and afterward browse to the image file you prepared in Module 1 because we will be using the one from that module. My copy is located in the location c:forensic pictures. Yours might be distinct.

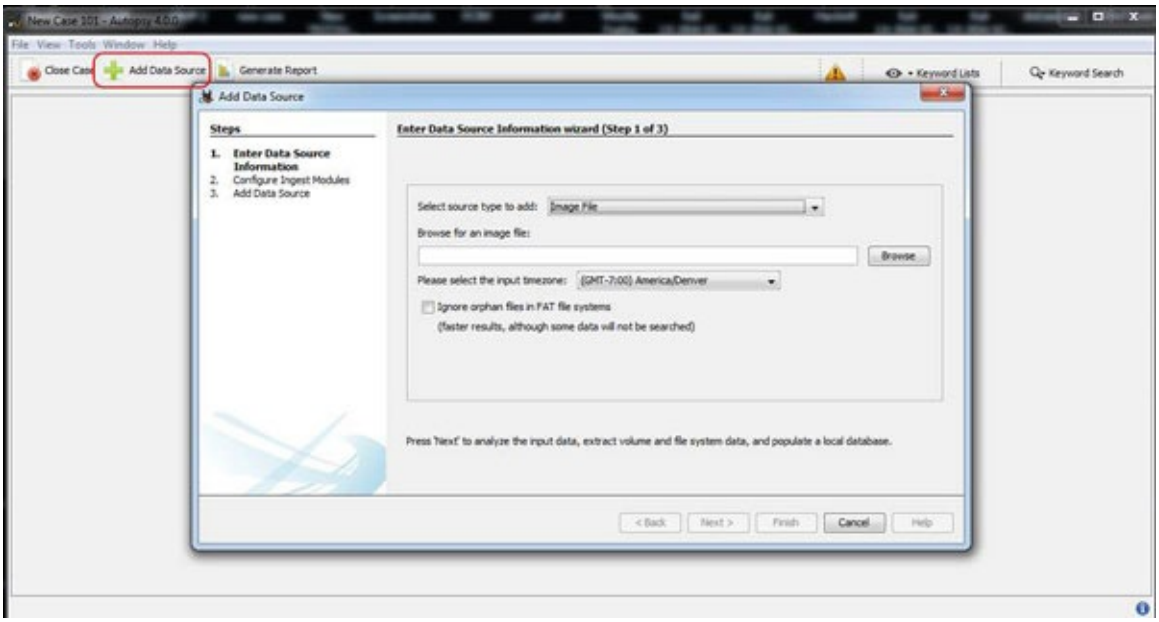


Figure 11.65: *Add the data source*

8. Now include our first. The image.dd.001 from the first lesson in this series.

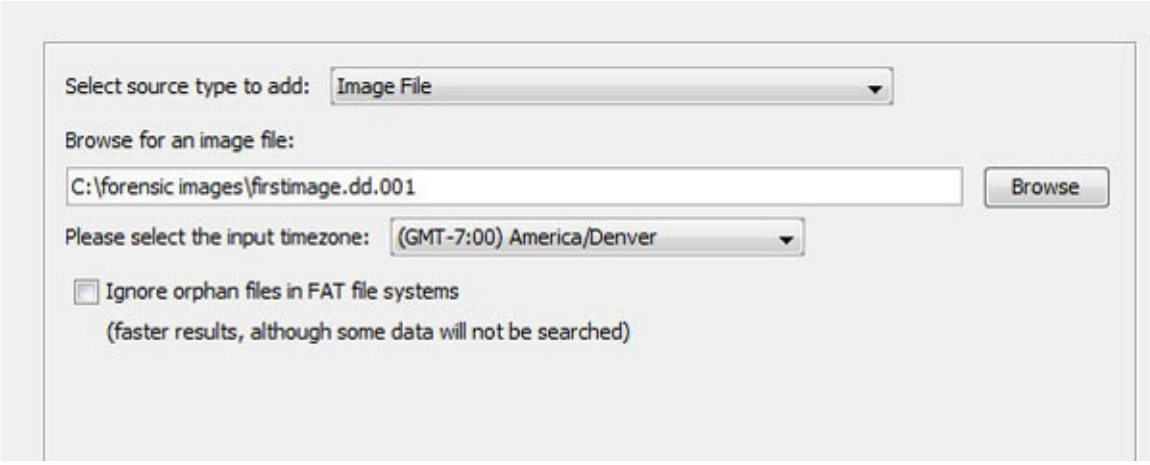


Figure 11.66: *Adding the source file*

9. Clicking next after adding the photograph will trigger Autopsy to start analyzing it. A screen similar to the one shown as follows will eventually welcome you.
10. Select **`Finish`**.

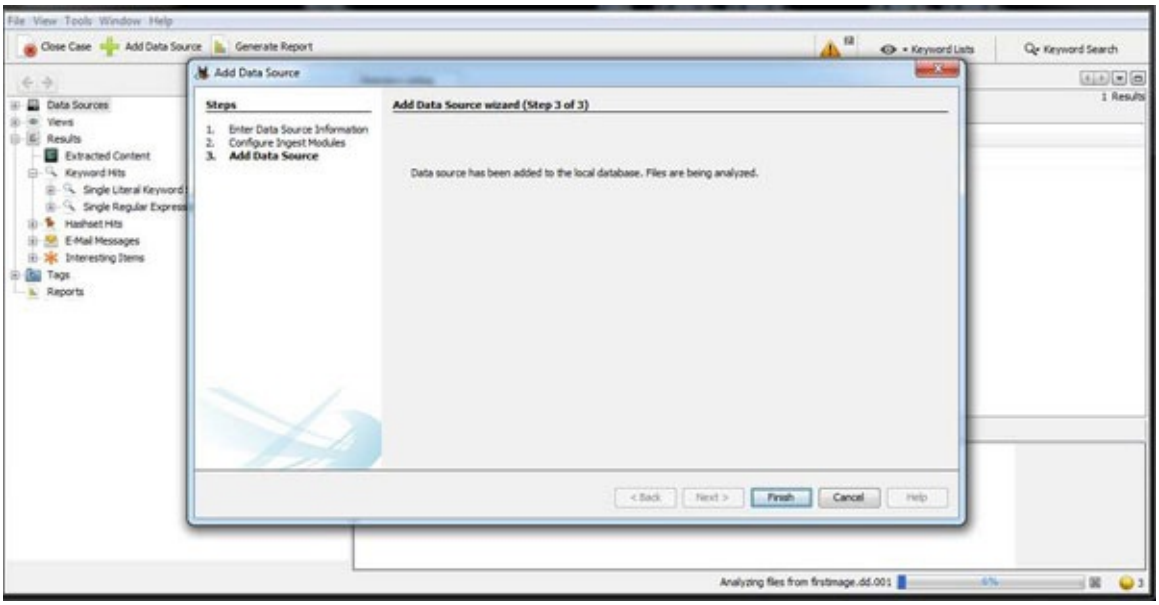


Figure 11.67: *Add data source*

11. You ought to now see a screen similar to the one as shown in *figure 11.68*. You should see your `firstimage.dd.001` as your data source; take note.

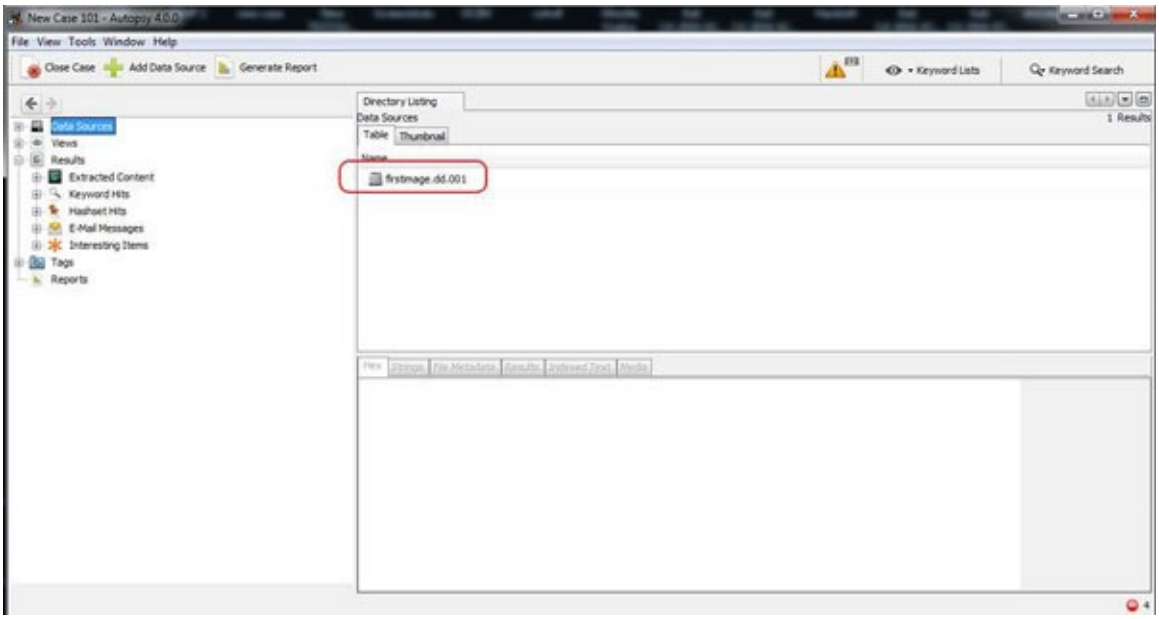


Figure 11.68: *First image added*

12. An Autopsy will list all file kinds and the quantity of files in each group if we expand **`File Types`** in the object explorer. As you can see in the image shown as follows, I selected the **`Images`** file type, and Autopsy displayed every Image file.

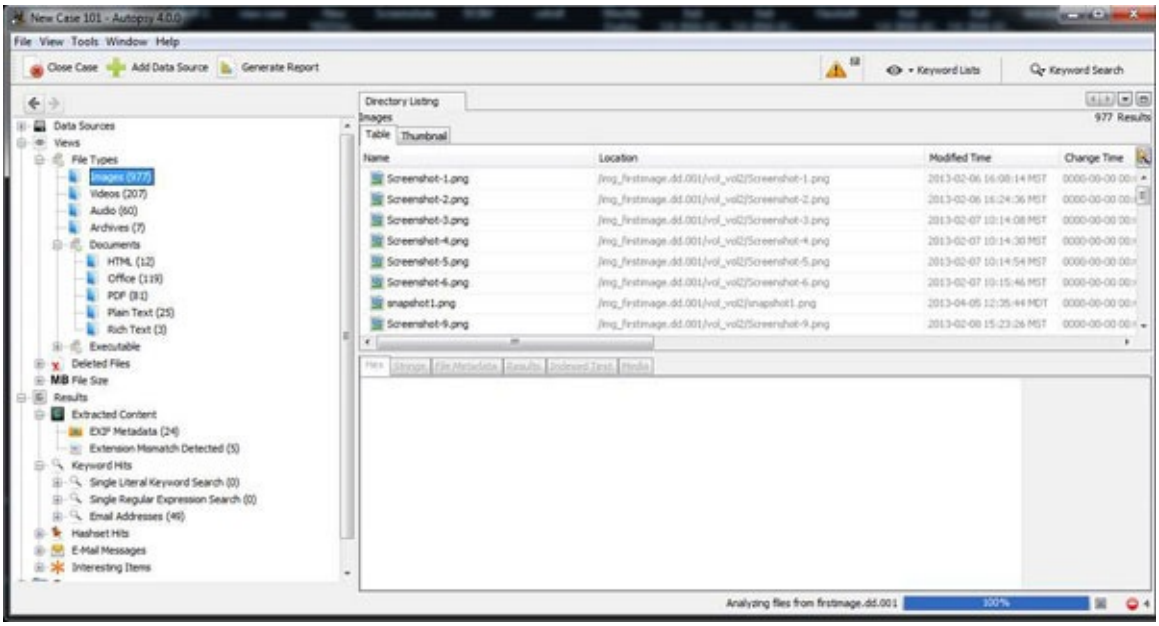


Figure 11.69: *Exploring the images*

13. A little further, below in the object explorer, we can see a File Type named **`Deleted Files`**. When we click on it will display all the deleted files.

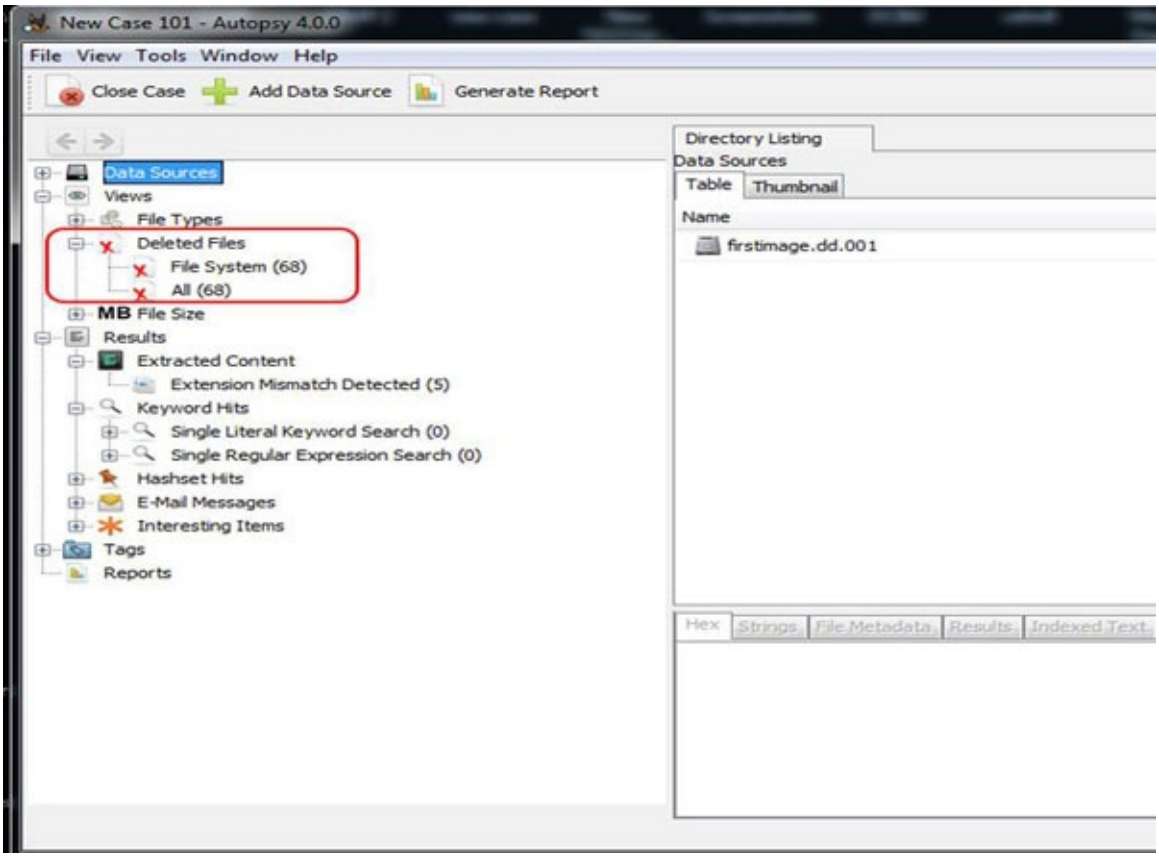


Figure 11.70: *Exploring the deleted files*

14. In the lower right pane, which appears when we click on a removed file, we can perform some analysis. Hex, Strings, File Metadata, Results, and Indexed Text are among the categories you will find. When you select the **`File Metadata`** tab, the file's metadata, which includes the name, type, size, updated date, and generated time, will be displayed (MAC).

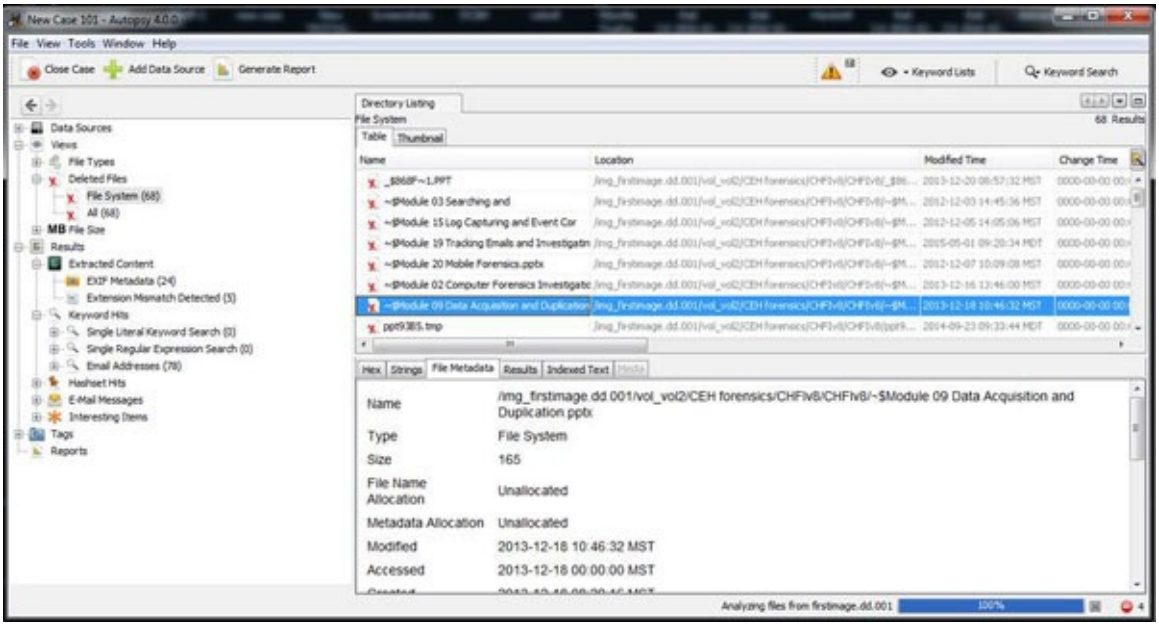


Figure 11.71: *Exploring the file system*

15. Now, choose "`Export`" from the context menu when right-clicking on the deleted file to recover it. This will bring up a window similar to the following:

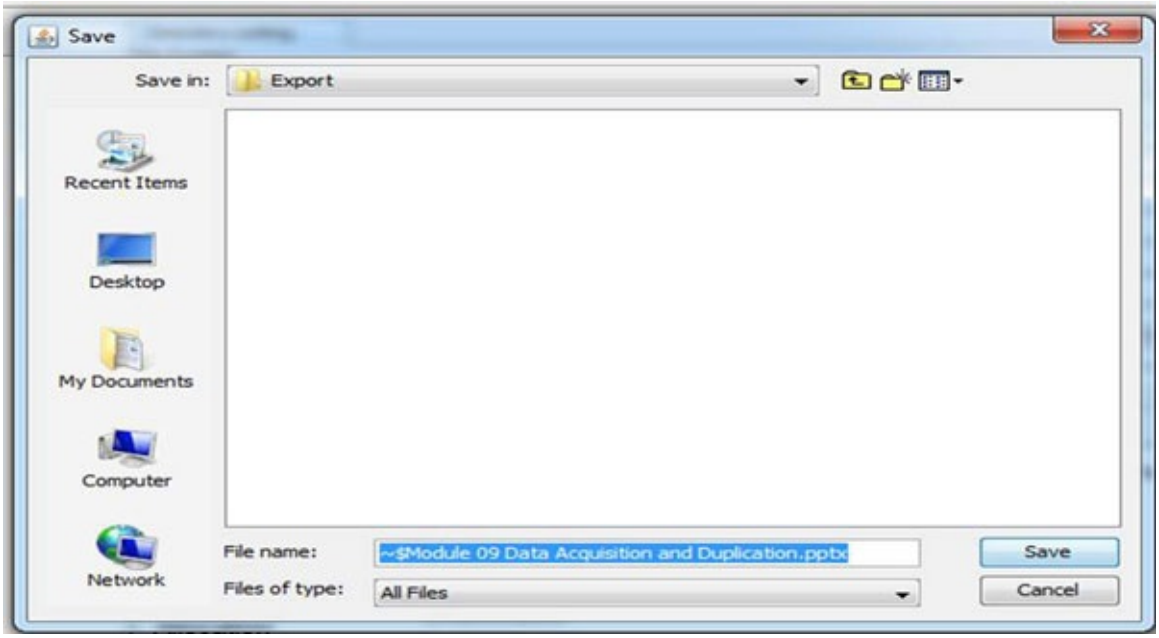


Figure 11.72: *Saving the file*

16. Save the deleted file to the Export directory without delay.
17. Browse to C: Cases New Case 101 Export to locate the transferred or deleted file.

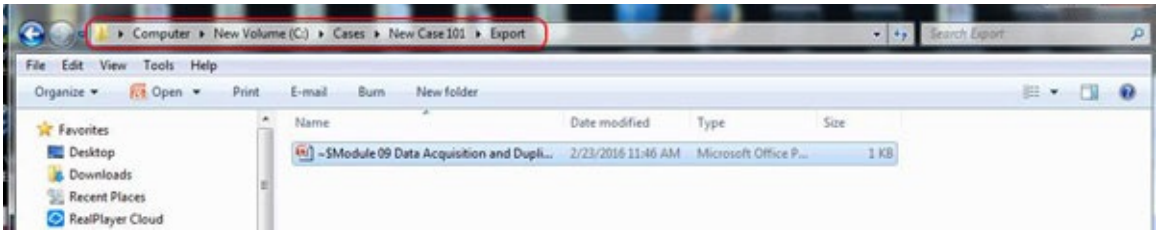


Figure 11.73: *Exported file*

Now that file can be double-clicked to launch the proper program. Key evidentiary files will frequently get deleted by suspects in an effort to hide their trails. As forensic investigators, we are aware that those files can still be recovered up until the point at which the file system overwrites them. Retrieval of these deleted files is simple with programs like Autopsy and almost any other forensic suite (Encase, ProDiscover, FTK, Oxygen, and so on).

Lab 7: Finding key evidence

In this lab, file analysis will be done using Autopsy. To put it another way, we will use Autopsy to look for keywords, file kinds, metadata, and so on that may be helpful in locating proof to back up our claims.

1. First, launch Autopsy and open our **`firstimage.dd.001`** files.

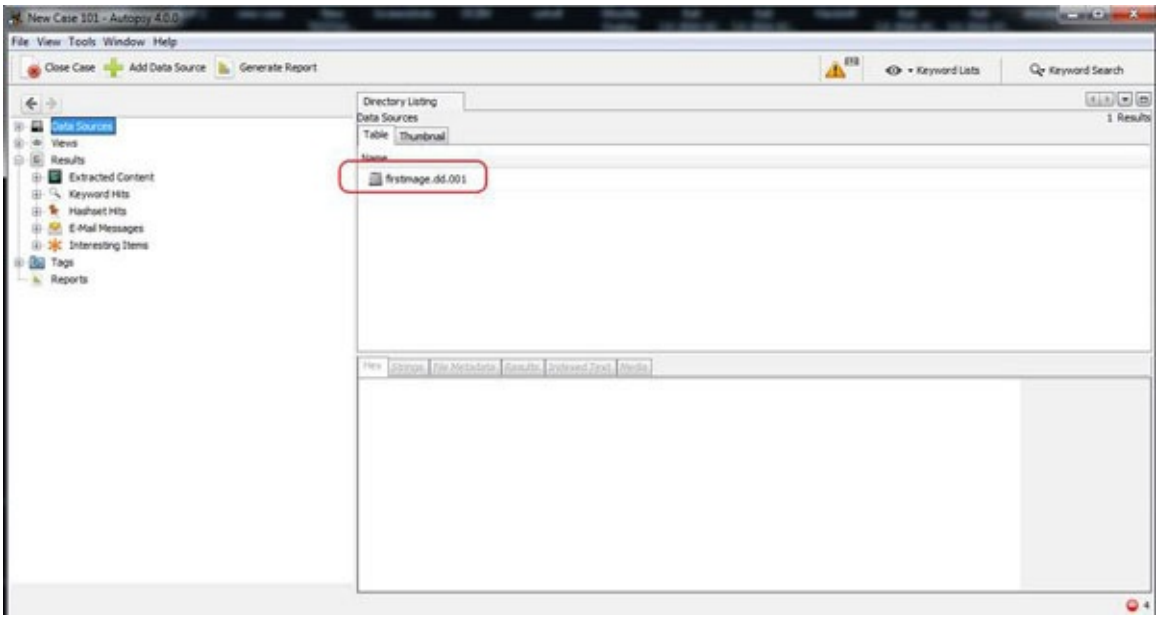


Figure 11.74: *Launching Autopsy*

This image will be automatically indexed by Autopsy for analysis.

After it has finished its investigation, it will classify each file according to its type. An Autopsy has classified all file types, including pictures, movies, music, papers, executable code, and deleted data, as you can see in the following snapshot:

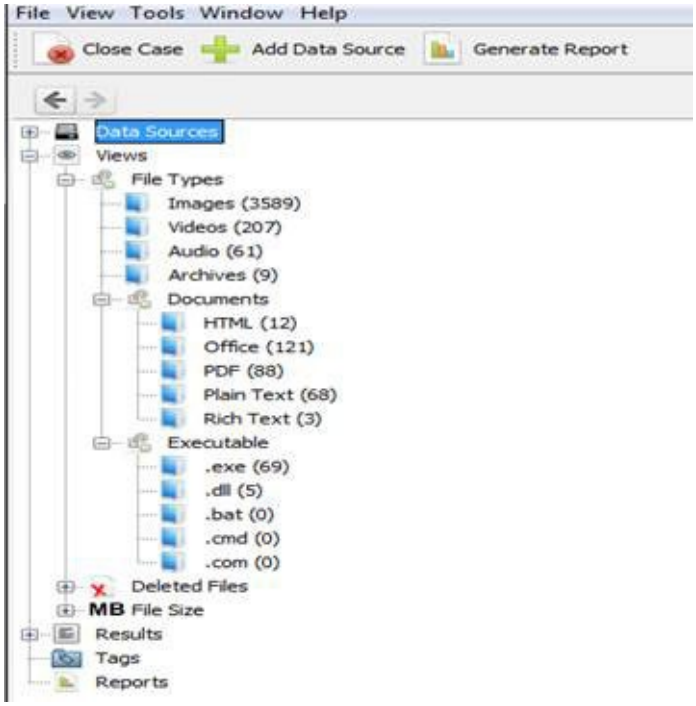


Figure 11.75: *Exploring hierarchy*

2. Search for Keywords

   Suppose we are searching for files with the word "*forensics*" in them. The Search box is located in the upper right corner of our computer. We can type this word there and then click on **`Search`**.

   Afterward, Autopsy will start looking for that term in every file. As you might expect, this keyword in a real investigation would likely be specific to the investigation, such as "*ransom*," "*extortion*," "*sex*," and so on.

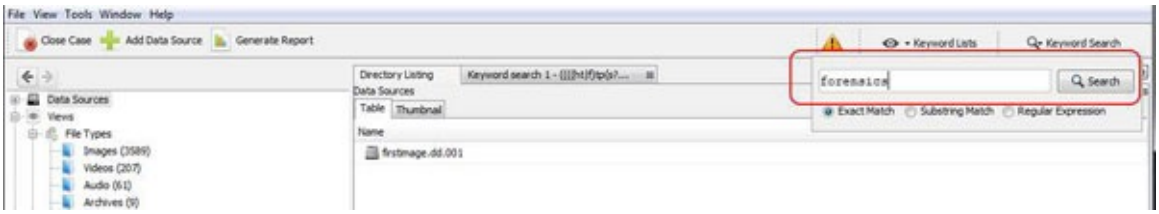


Figure 11.76: *Search for forensics*

Once Autopsy has finished its search, the main table pane will show each file that matches that term. Then, we can click on those files to read them in more detail.

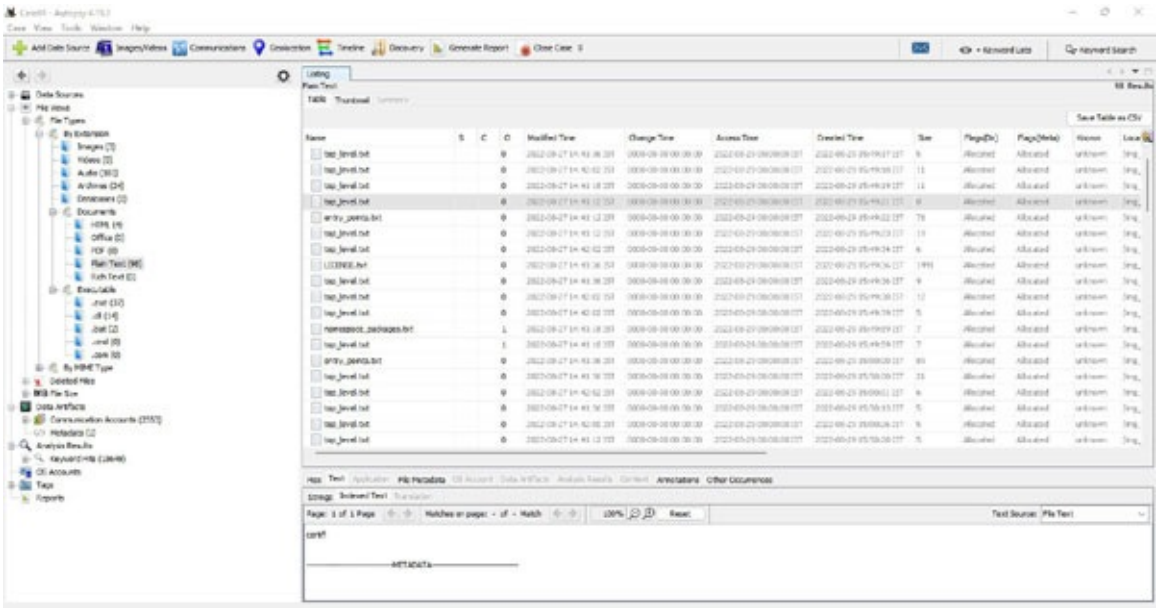


Figure 11.77: *Exploring forensics*

The **`Indexed Text`** tab, which is located in the lower right window, will then show us every occurrence of that phrase and underline it for us.

Third Step: Specialized search methods

We can do extremely specialized searches during autopsies, which may be crucial to our inquiry. These could be IP addresses, URLs, e-mail addresses, or phone numbers.

On this page, click the eye in the upper right corner. The pull-down window shown as follows should appear. We can search for here:

1. Phone numbers
2. IP addresses
3. e-mail addresses
4. URL's

Let us look over these files to see if there are any URLs that could help us determine what the accused was doing when the computer was taken. Select the checkbox next to URLs in the pull-down menu. It will fill in the URLs in the regex phrase.

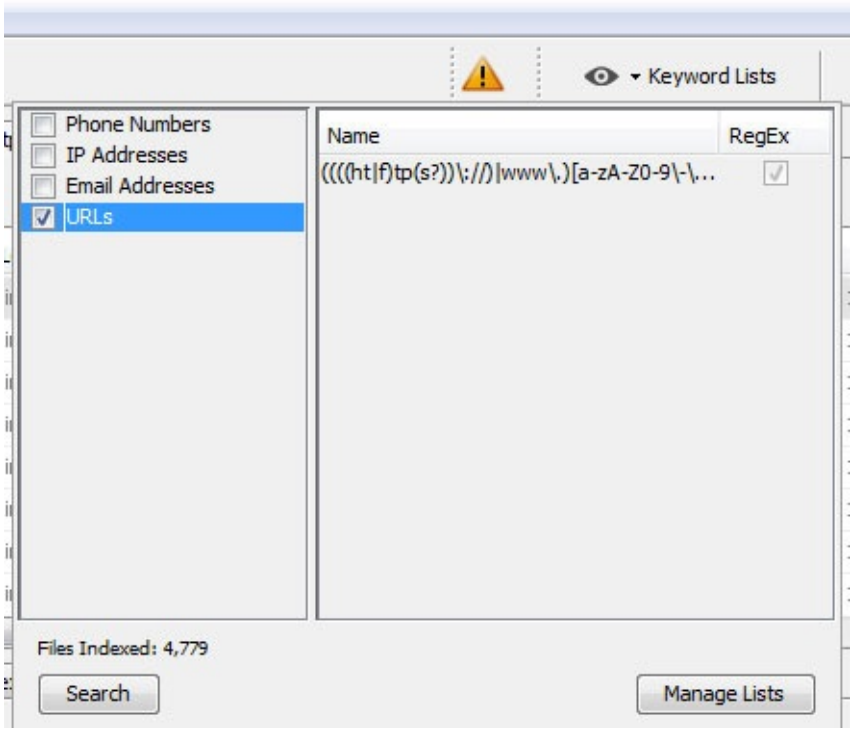


Figure 11.78: *Exploring URLs*

3. Next, click on **`Search`**, and Autopsy will begin looking through every file for that text pattern. The regular expression it is looking for is displayed in the right-hand window.

   Regular expression searches are very CPU intensive and slow, so be patient.

   When it is done doing this search, it will display the results like the following:

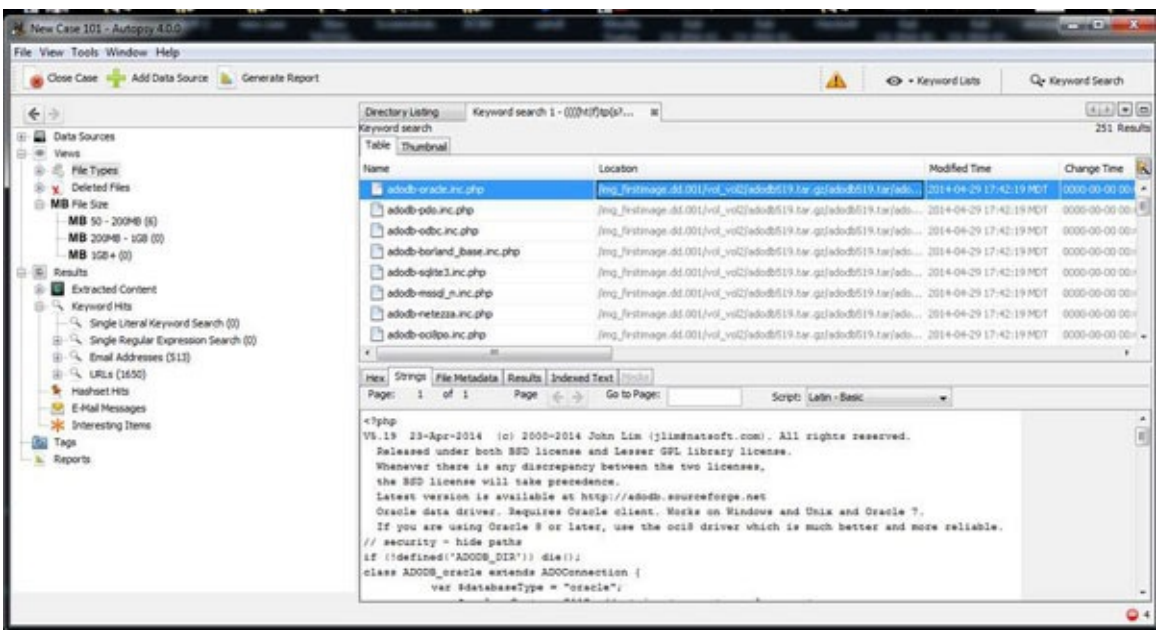


Figure 11.79: *Exploring file types*

As you can see previously, Autopsy displays every file that it found a URL.

In addition, we can use this method to find e-mail addresses as well, but when Autopsy did its initial analysis, it categorized all the e-mail addresses it found in the Explorer pane, as seen as follows.

An Autopsy is a powerful tool for doing forensic image analysis. Among the many things, it is capable of doing are deleted file analysis, file type analysis, keyword analysis, and finding such key artifacts as URL's, e-mail addresses, IP addresses, and phone numbers. In addition, we can create custom regular

expressions to search for just about any text pattern one can imagine.

Lab 8: Analyzing the registry for evidence

Although almost all Microsoft Windows customers are aware that their computer has a registry, very few of them are aware of what it does or how to change it. The registry may be a gold mine of information for forensic analysts about what, when, where, and that something happened on the system.

On a Windows computer, components of a piece about users, technology, and software are kept in a database called the registry. Even though the registry was created to set the system, in order to do so, it collects a vast amount of data on usage patterns, smart devices, software used and when, and so on. The forensic investigator can use all of these information to follow the who, what, where, and when of a forensic inquiry. Knowing exactly where to look is the key.

Root directories exist within the registry. The term *hives* refer to these root folders. Five registry hives are present as follows:

1. The loaded user profiles are all stored in HKEY USERS.
2. HKEYCURRENT USER stores information about the user currently logged in.
3. HKEYCLASSES ROOT stores configuration details about the file-opening program.
4. KEYCURRENT CONFIGURE stores start-up hardware information.
5. HKEYLOCAL MACHINE stores hardware and system configuration.

The Windows directory/subdirectory architecture and the registry's organizational structure are quite identical. Subkeys come after the five root keys, also known as hives. You may occasionally have sub-subkeys. The contents pane then shows the definitions and values for these subkeys. The numbers are sometimes only 0 or 1, indicating on or off, but they can also carry more complicated information, typically shown in hexadecimal.

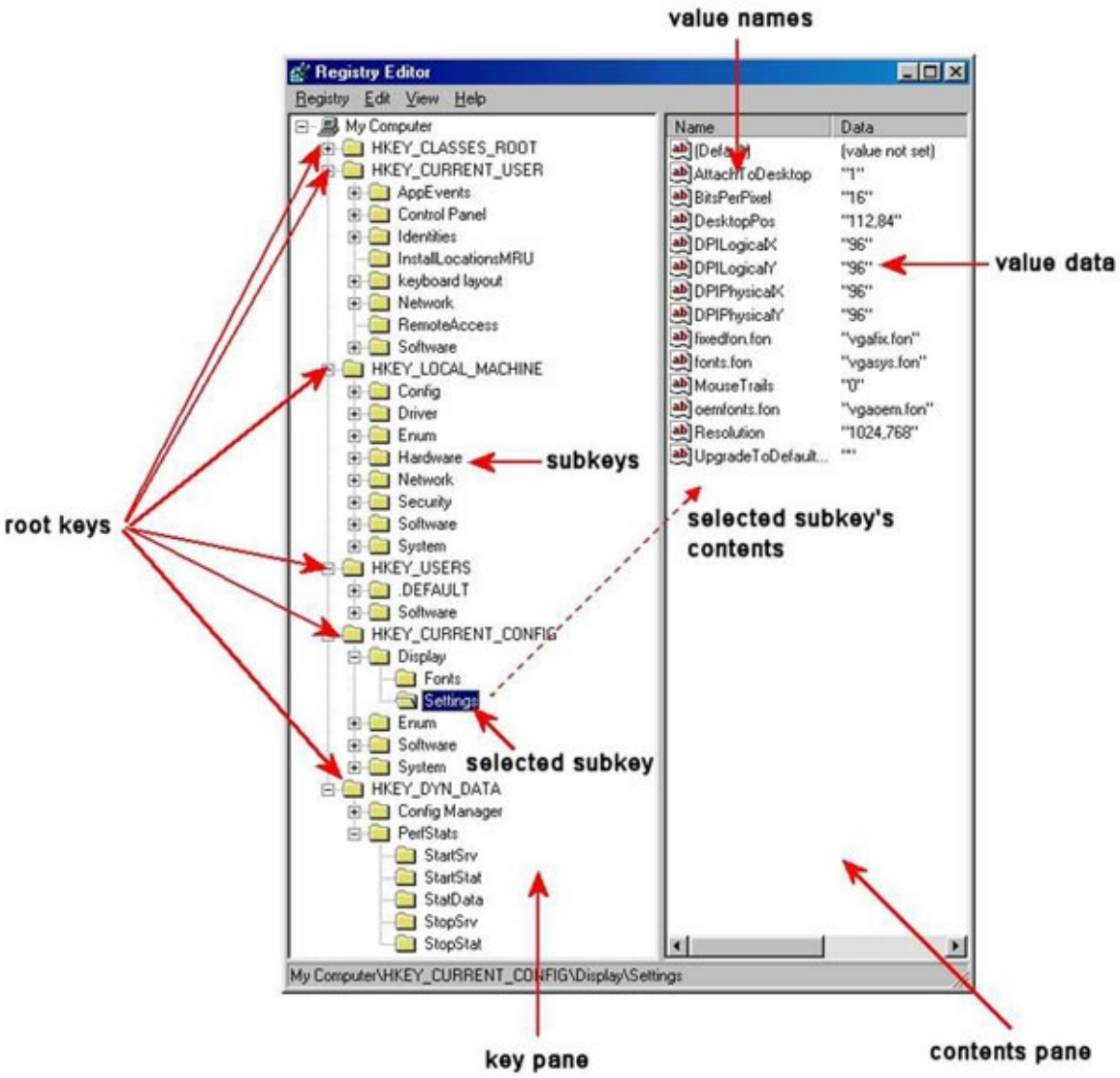


Figure 11.80: *Highlighting the components*

We can examine the registry on our own machine using the Windows regedit tool, but not in forensic mode. To launch the properties window like the one shown in *figure 11.81*, simply put regedit into the search box and click on it.

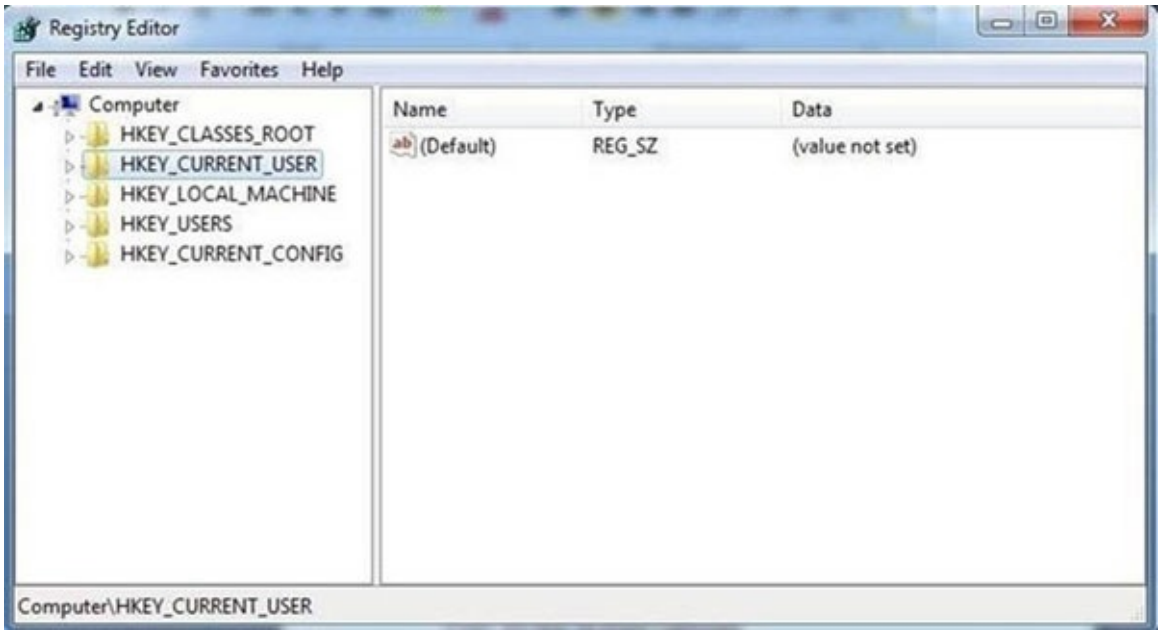


Figure 11.81: *Exploring current user*

As a forensics expert, the register can prove to be a gold mine of details on who, what, where, and whenever something occurred on a system that can conclusively connect the offender to the conduct in issue.

The register contains data, including the following:

- Customers and the latest time they used the scheme.
- The most used technology, whichever devices are mounted to the scheme, include the identifiers of flash drives, hard drives, phones, tablets, and so on.
- The duration of the scheme is connected to a particular wireless access point.
- What and when documents were made accessible.
- A list of any searches performed on the device.

Mobile evidence in the registry

A local wireless network adapter is often compromised by hackers, who then use it to enter networks. If the IP address is tracked in this manner, it will point to the neighbor's or another wireless AP rather than theirs.

For instance, *John Borrell III*, a participant of Anonymous, broke into the computer systems of the Utah Chiefs of Police and the Salt Lake City Police Department in January 2012. When the FBI was contacted to conduct an investigation, they were able to track the hacker's IP address to Toledo, Ohio's Blessed Sacrament Church's Wi-Fi AP. The wireless AP password of the church appeared to have been hacked by the hacker, who was using it to access the internet *anonymously*.

Ultimately, the FBI was able to identify the perpetrator using a variety of investigation methods, most of which included meticulous, low-tech investigative work. The fact that *John Borrell* had boasted about his skill as a hacker on Twitter was helpful. *Mr. Borrell* was ultimately found guilty and given a two-year federal prison term.

By looking into Mr. Borrell's register, the FBI was able to show that he had been associated with the church AP when they located him and took his computer. It was just necessary for the forensic investigator to examine the registry at this place:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows
NT\CurrentVersion\NetworkList\Profiles
```

You may get a list of the GUIDs of the wireless access points the device has connected to there. When you click on one, details like the SSID name and the hexadecimal time of the last connection are displayed. Therefore, despite Mr. Borrell's initial denial of involvement in this hack, the evidence was convincing, and he ultimately entered a guilty plea.

The attacker connected to the **`HolidayInnColumbia`** SSID in November 2014, as shown in the following screenshot:

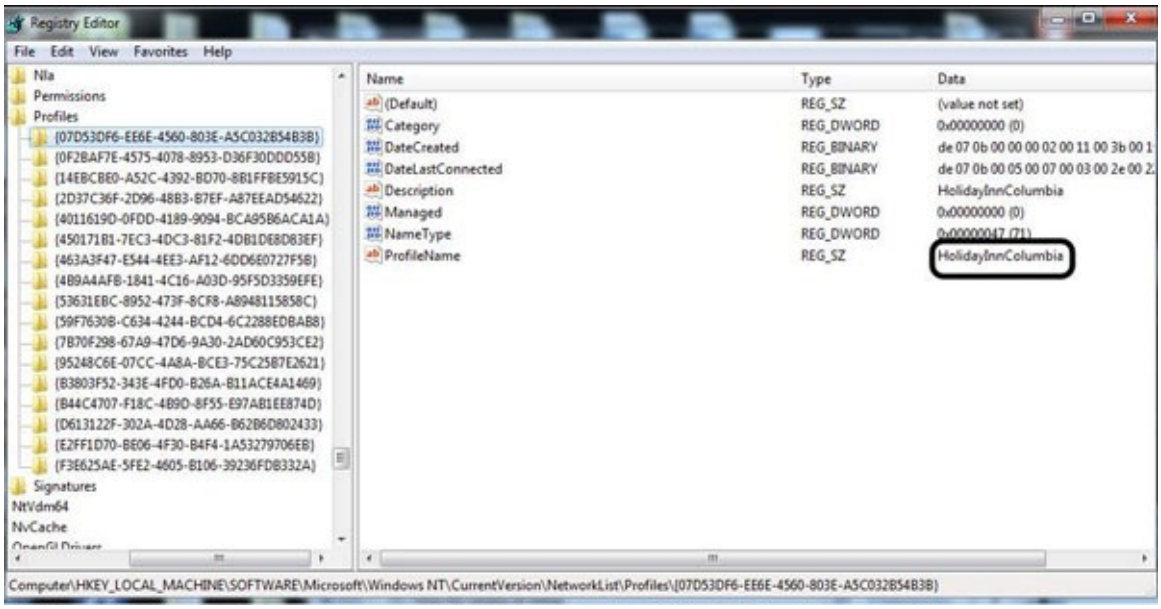


Figure 11.82: *Exploring registry*

The Windows registry keeps a lot of records of user activity. Most of the time, such registry entries are created to improve Windows' performance. These keys serve as a kind of road map for the user or attacker's activity as a forensics expert.

The **`RecentDocs`** key is one of such keys. By file extension, it keeps track of the latest documents used or opened on the system. It is accessible at:

- HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs

  Because each key can store up to 10 documents, the most recent Word documents, for example, would have **`the.doc`** or **`.docx`** augmentation depending on the version of the word they were created in. The most recent 10-word documents are presented under this key when we navigate to the **`.docx`** extension.

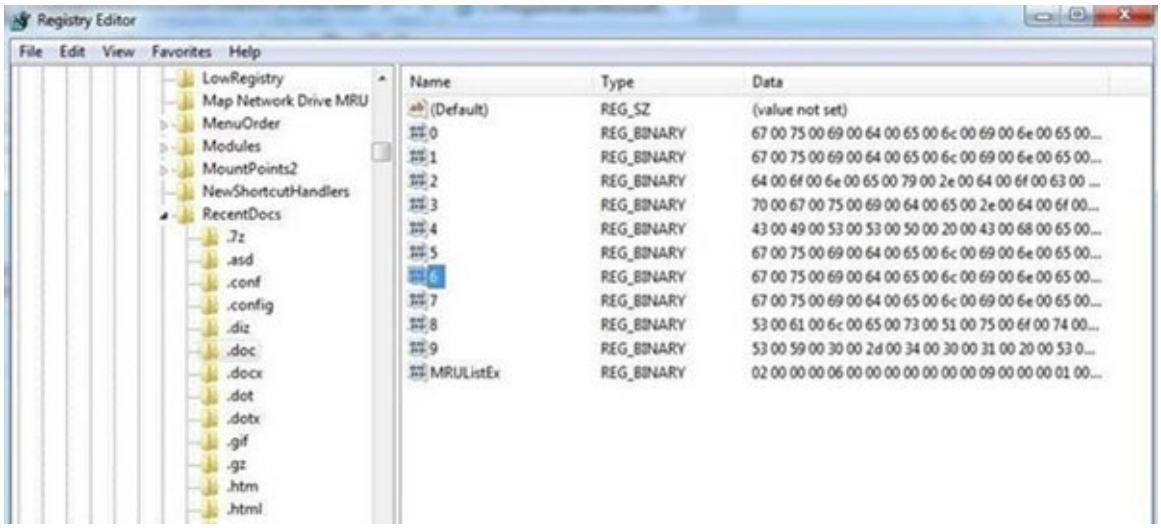


Figure 11.83: *Exploring registry*

The details about the document are shown when we tap on one of those keys, as seen in *figure 11.84*. The article's data may be seen to the left in hex and to the right in ASCII. This paper was an outline for a Metasploit course, as evidenced by this.

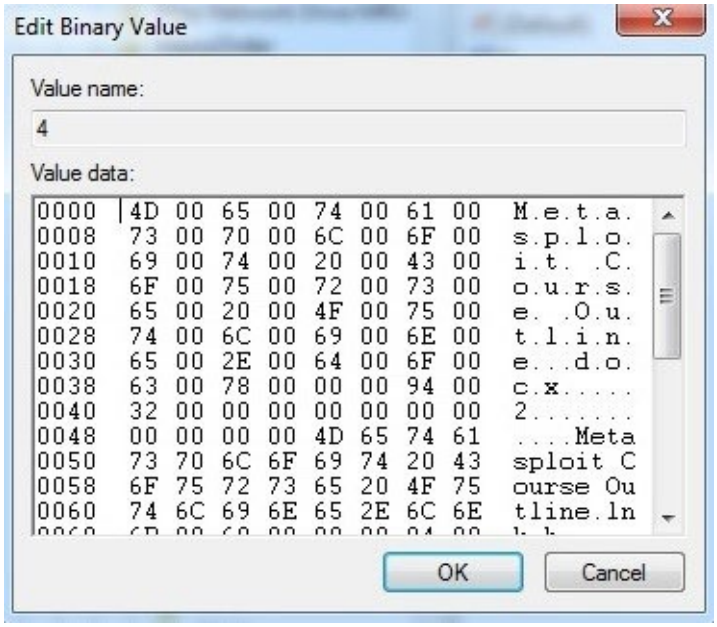


Figure 11.84: *Editing binary value*

a.tar file may occasionally be uploaded by an attacker; therefore, it is an excellent spot to examine for breach-proof. Generally speaking, a Windows computer will not have **`a.tar`** file extension; therefore, the existence of an item here warrants additional examination. Examine the files in **`the.tar`** key to determine whether they include any information about the attack or the perpetrator.

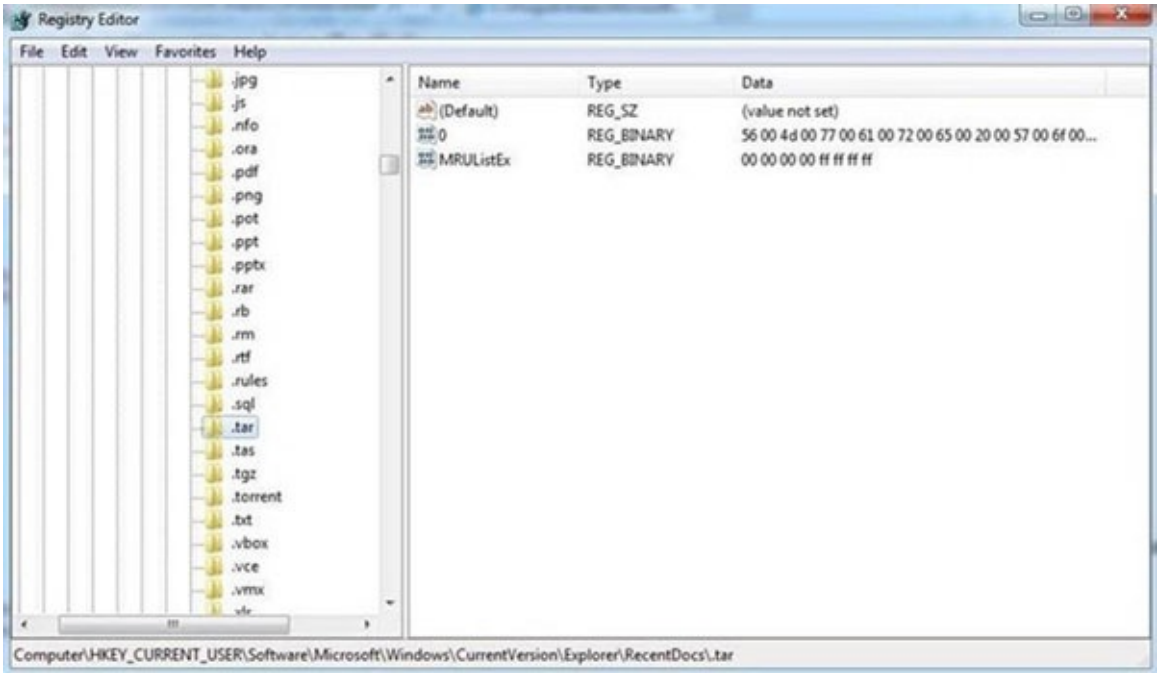


Figure 11.85: *Exploring current user*

In civil or policy violation investigations, evidence might be found in various graphic file extensions such as **`.jpg`**, **`.gif`**, or **`.png`**.

TypedURLs key

When the user types a URL in Internet Explorer, this value is stored in the registry at:

`HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\TypedURLs`

The last URLs that the user viewed with IE are listed when we open that key in the registry. In civil or

policy violation inquiry sorts of cases, this may show the source of harmful software that was used in the breach or what the user was searching for/at.

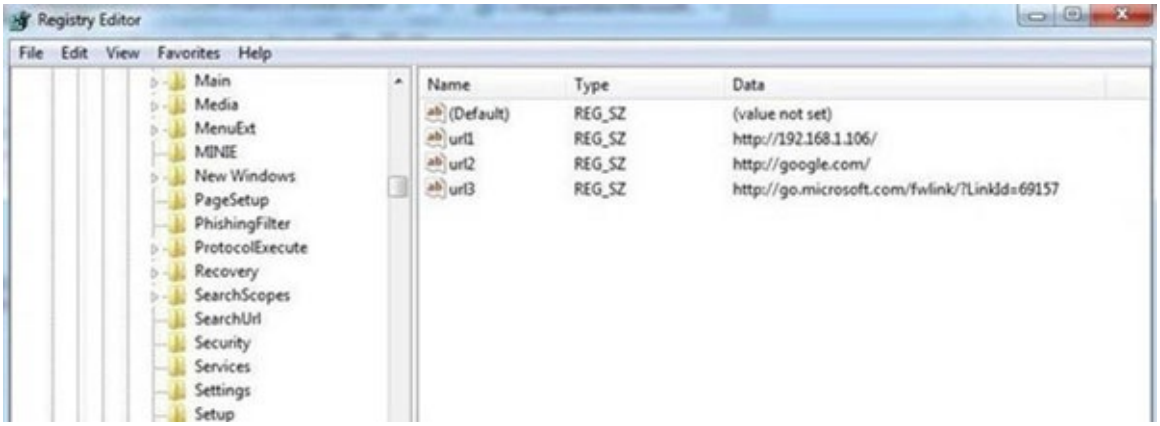


Figure 11.86: *Exploring the typedURL*

Lab 9: Analyzing Windows pre-fetch files for evidence

The pre-fetch technology was undoubtedly not implemented by Microsoft for forensic examination but rather to enhance Windows' speed. Pre-fetching files that the system expects the user will require and loading them into memory makes "*fetching*" the files faster and more efficient, as the name of the system suggests. It is a form of artificial intelligence that gets things prepared for you in advance of what you might need next.

The pre-fetch mechanism in Windows has the advantage of being able to expose a great deal of information about what the user was doing, even if they were clever enough to try to hide their activities. From Windows XP and 2003 (they are the same builds), pre-fetch has been a feature of the Windows operating system and is preinstalled in Windows 10. On the server side, application pre-fetch has to be set in the registry for Windows Server 2003, 2008, and 2012.

Metadata about the used files are contained in these pre-fetch files. This metadata contains details like the most recent date the program was used, the location where the program files were saved, how frequently the application was used, and many other details that might be helpful to a forensics expert. This information can be crucial in attempting to establish that a suspect really used a program that was used in the crime, such as the browser or a program used to create documents, like a word document, even after the program had been deleted from the computer.

The pre-fetch files are located in the **`C:\Windows\prefetch`** folder. Here is an illustration of query processing files from a Windows 7 system, which is still the most used operating system worldwide.

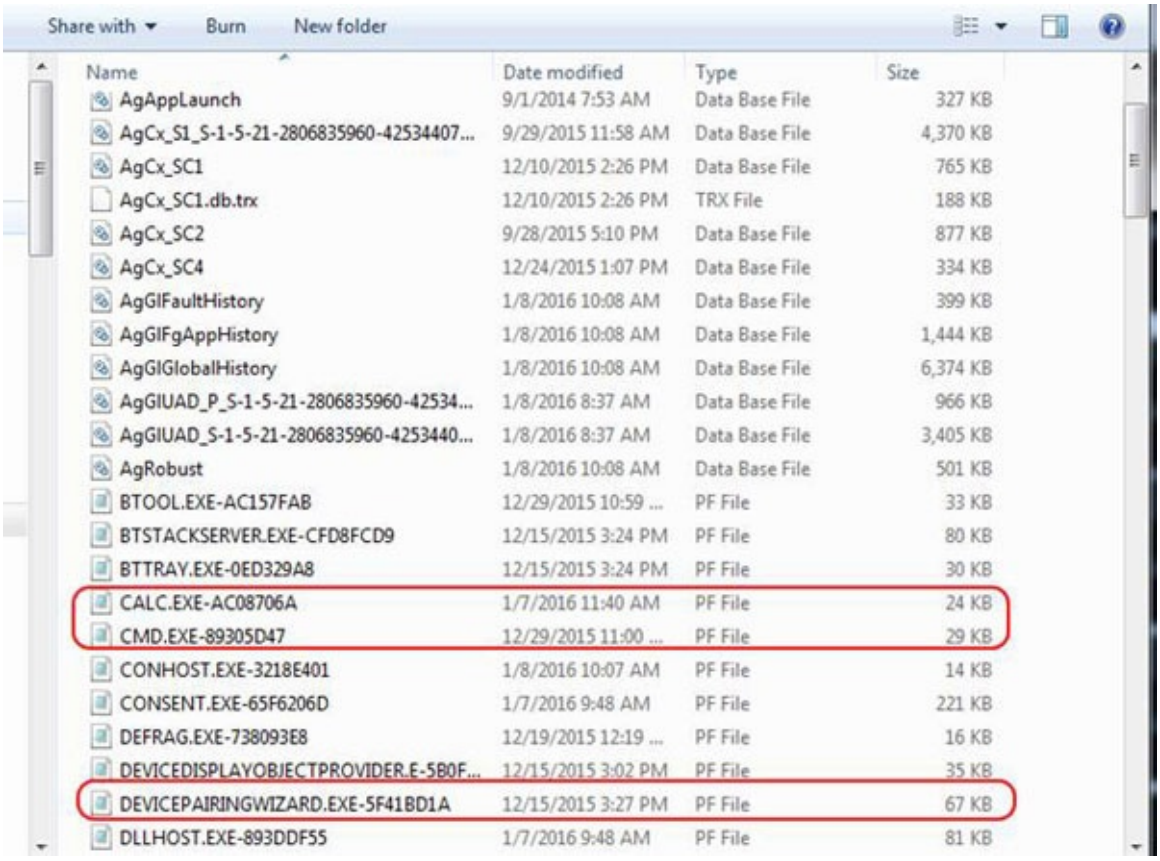


Figure 11.87: *Highlight the findings*

As you can see, in this **`c:\windows\prefetch`** directory, we can see all of the pre-fetch files. These are the files ending in .pf, but we can also see database files (.DB) as well. For now, let us focus on the **`.pf`** files. In the first group of circled files seen previously, you can see the pre-fetch files for both **`calc.exe`** and **`cmd.exe`**. In the next column, we can see the last date they were modified, which usually means the date those applications were last used.

We can see the pre-fetch file for the "DEVICEPAIRINGWIZARD" program in the second highlighted section. The last time it was used was on 15 December 2015. The Wizard for Bluetooth pairing is this. Imagine a situation in which a suspect denies ever using or understanding the Bluetooth technology that was used in the crime. This documentation would convincingly refute such a claim in court.

Although seeing the pre-fetch files will provide us with some fundamental information, we must parse the file to obtain all of the data it contains. Although there are several tools that can parse these files, and they all perform very well, Nirsoft's free WinPrefetchView is probably the most user-friendly and has a pleasant, straightforward GUI.

1. Run it after downloading and installing it. It will seize every pre-fetch file and parse it, as seen in *figure 11.88*.

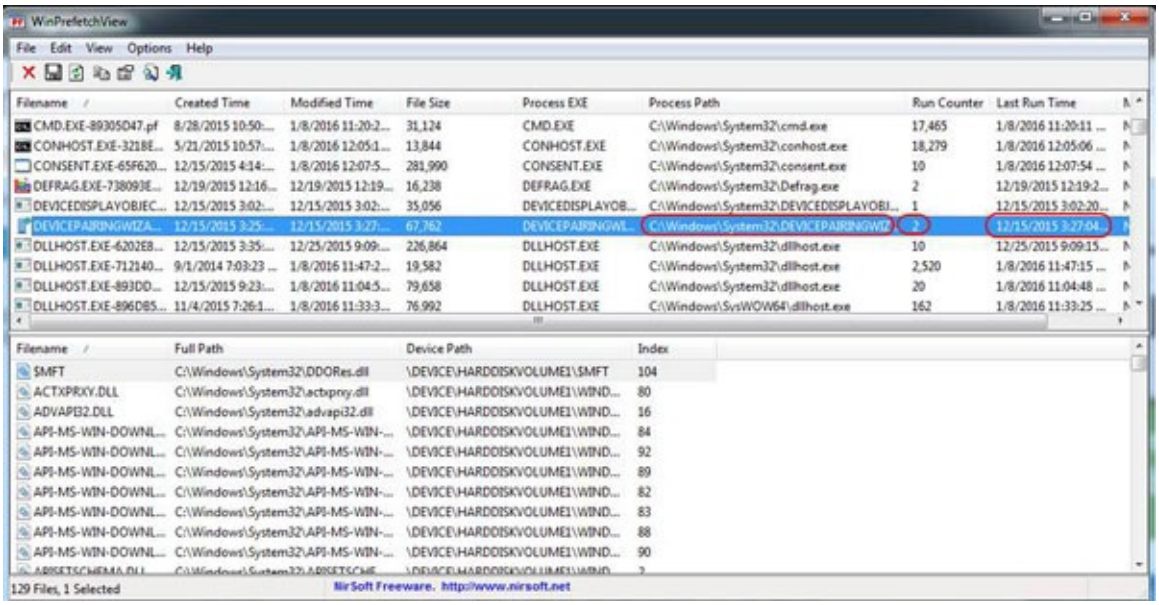


Figure 11.88: *Exploring WinPrefetch*

Look at that Bluetooth Device Pairing Wizard, shall we? In the aforementioned screenshot, I have circled it. We can see the location of **`the.exe`** file, how frequently it has been executed, and when it was last executed. All of these information can be crucial in a forensic inquiry. Also, observe that each file used by the application is listed in the lower pane along with its path.

2. We may generate a chronology of the events that happened on that system by sorting by the **`Last Run Time`** tab.

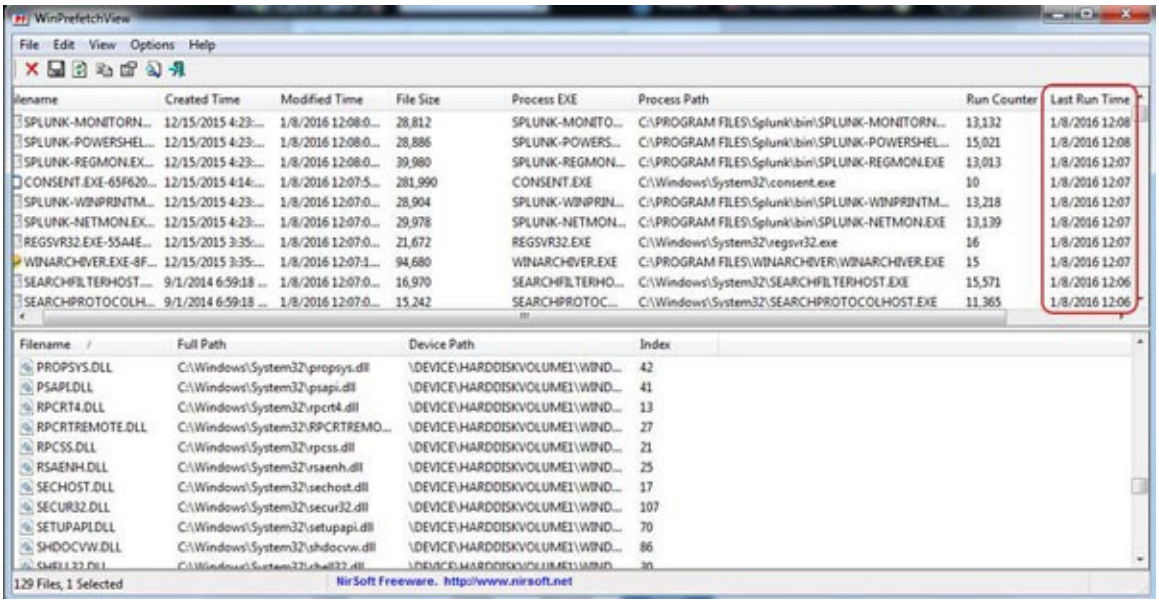


Figure 11.89: *Exploring the last runtime tab*

The suspect might turn off pre-fetch if they were really tech-savvy. Additionally, they might disable or suspend superfetch, a feature that was added to Windows Vista and functions similarly to pre-fetch but preloads apps depending on your usage habits.

3. Go to Computer Administration, click on **`Services`** on the left, double-click Superfetch, and then pick the appropriate option to disable or stop it.

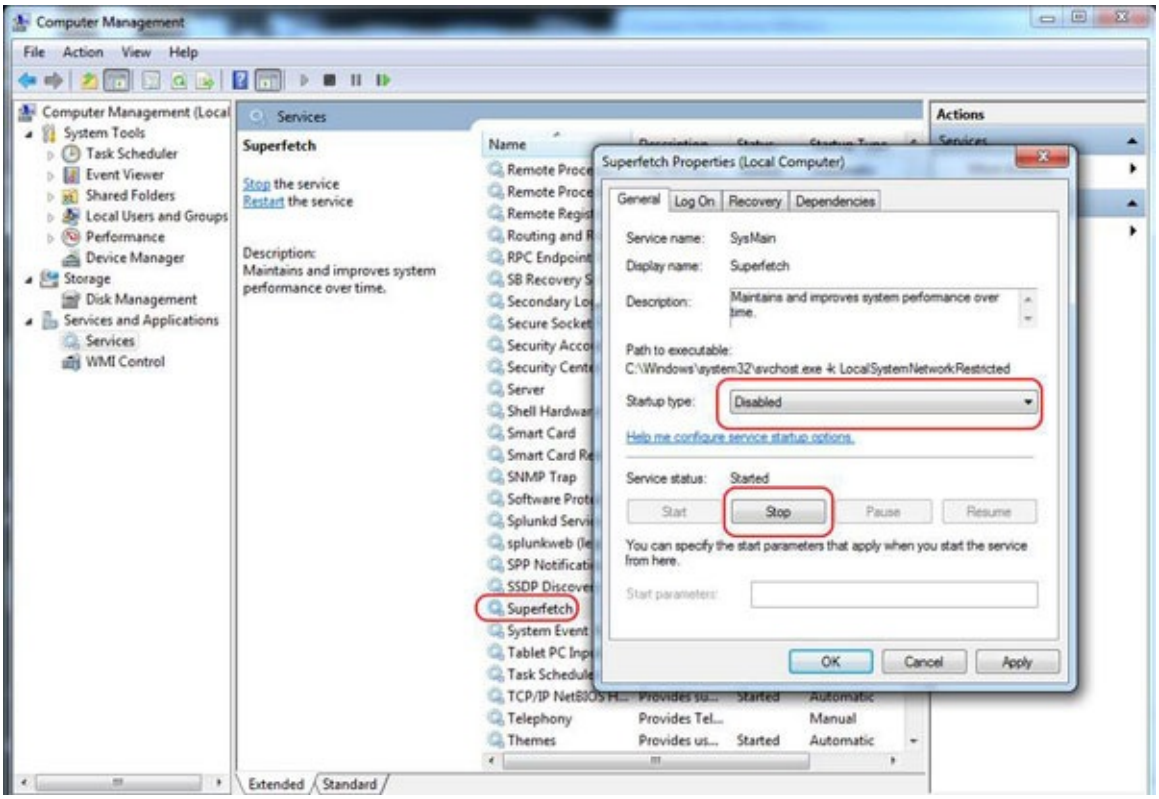


Figure 11.90: *Disable the service*

Even if the program has been deleted from the system, pre-fetch files might be crucial in a forensic examination to show that the suspect really used it.

Lab 10: Browser forensics

A suspect's Web browser may frequently provide us with a wealth of information about their online activities prior to the system's acquisition. Because the browser is used for so many online activities, we ought to be able to locate artifacts or proof of almost all the suspect's actions in the browser and its related directories and data sets.

In this lab, we will look at what information we may gather from the browser and how it can be applied to a criminal or civil inquiry.

Internet Explorer (IE)

Let us start using Internet Explorer from Microsoft. With the exception of later versions of Windows 10, where Edge is the default browser, it is downloaded on every individual Windows PC and is extensively used. It is the only computer permitted in many academic and business settings.

Based on the Windows version, Internet Explorer stores its information in several locations. Let us start by examining the most recent iterations after 2000.

Windows 2000 and XP

The following places provide proof of the user's online activities:

```
%systemdir%\Documents and Settings\%username%\Local Settings\Temporary Internet Files\Content.ie5
```

```
%systemdir%\Documents and Settings\%username%\Cookies
```

```
%systemdir%\Documents and Settings\%username%\Local Settings\History\history.ie5
```

Windows Vista & 7

Starting with Windows Vista and 7, the file path differs somewhat. The files for IE are located here:

```
%systemdir%\Users\%username%\AppData\Local\Microsoft\Windows\Temporary Internet Files\
```

```
%systemdir%\Users\%username%\AppData\Local\Microsoft\Windows\Temporary Internet Files\Low\
```

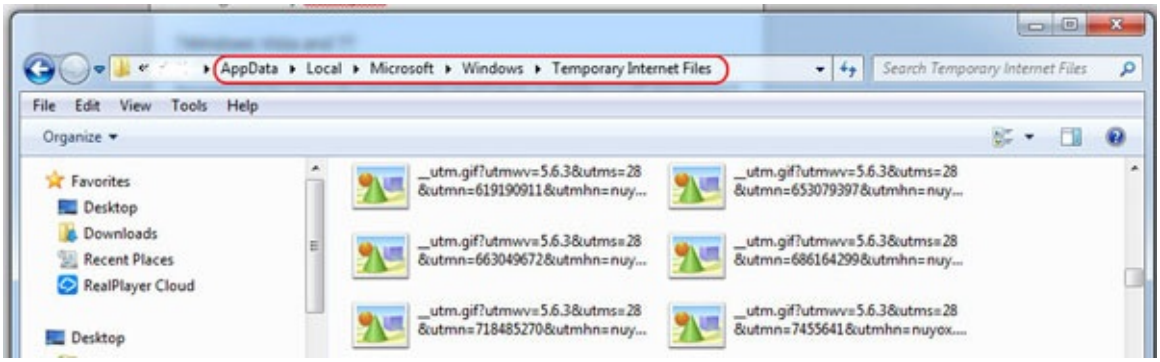


Figure 11.91: *Exploring temporary internet files*

Please be aware that the hidden folders **`AppData`** and **`Temporary Internet Files`** exist.

Mozilla Firefox

The majority of the data is kept in SQLite databases with Mozilla Firefox and its several versions (Iceweasel in Kali is one). Depending on the operating system, we can locate the databases in various places.

The following is the path to the database in Windows (XP, Vista, and 7), Linux, and Mac OS X.

Windows XP

```
C:\Documents and Settings\<username>\Application Data\Mozilla\Firefox\Profiles\<profile folder>\places.sqlite
```

Windows Vista & 7

```
C:\Users\<user>\AppData\Roaming\Mozilla\Firefox\Profiles\<profile folder>\places.sqlite
```

GNU/Linux

```
/home/<user>/.mozilla/firefox/<profile folder>/places.sqlite
```

Mac OS X

```
/Users/<user>/Library/Application Support/Firefox/Profiles/default.lov/places.sqlite
```

Mozilla stores the user's data in a database system that has the following structure:

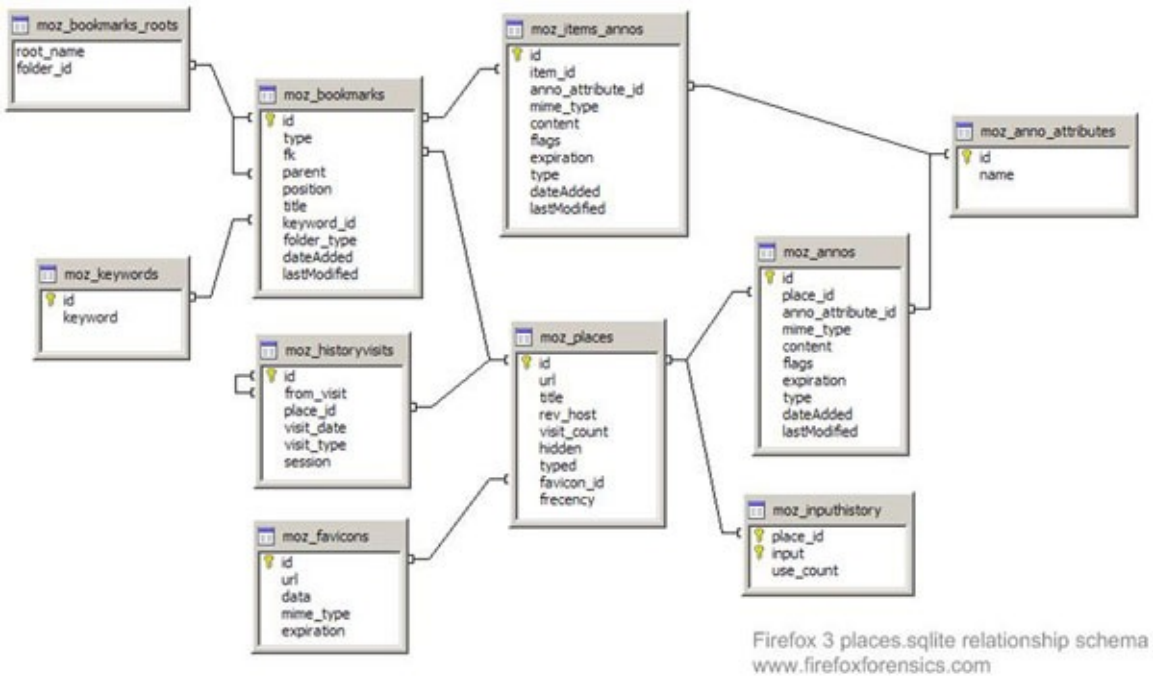


Figure 11.92: *Mozilla user data structure*

1. Using SQLite to find browser evidence in Mozilla.

   Numerous browsers, programs, and portable devices that need a compact, lightweight relational database now use SQLite. It is growing in popularity among mobile phones and wireless apps due to how lightweight it is. Given that it is becoming increasingly prevalent, it is imperative for any qualified forensic investigator to get familiar with it.

   We will require an SQLite browser in order to browse or access the data in these SQLite databases.

2. Load the database file into the SQLite browser

   Browse to the site mentioned previously for the operating system you are researching after installing the SQLite browser. Since it is a Windows 7 PC in this instance, I browse to:

   `C:\Users\<user>\AppData\Roaming\Mozilla\Firefox\Profiles\<profile folder>\`

   There will be a lot of files that end with "*sqlite*." These are the tables in the database that Mozilla uses to keep track of users' browsing habits.

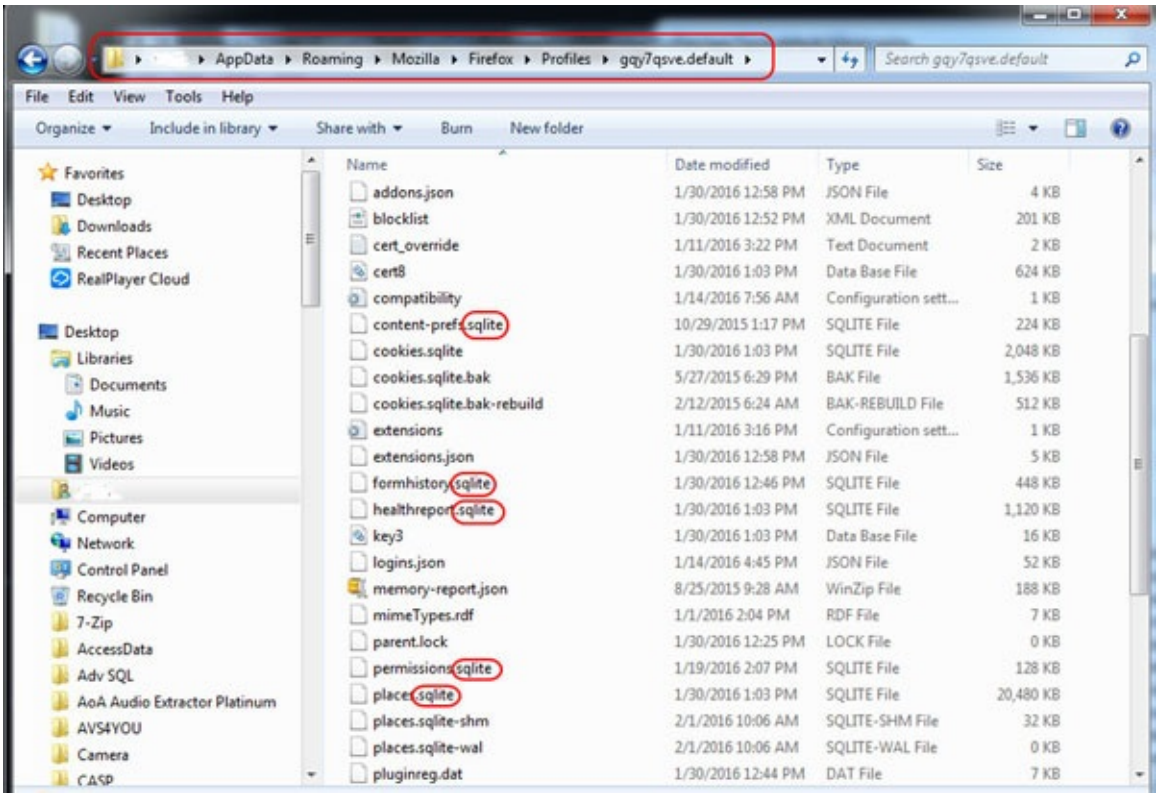


Figure 11.93: *Highlighting SQLite*

Let us launch the SQLite browser and access that database. Then select the far-left **`Database Structure`** tab. All 13 tables in the database are shown in the screenshot, as shown in *figure 11.94*. Keep in mind that every table name starts with "moz."

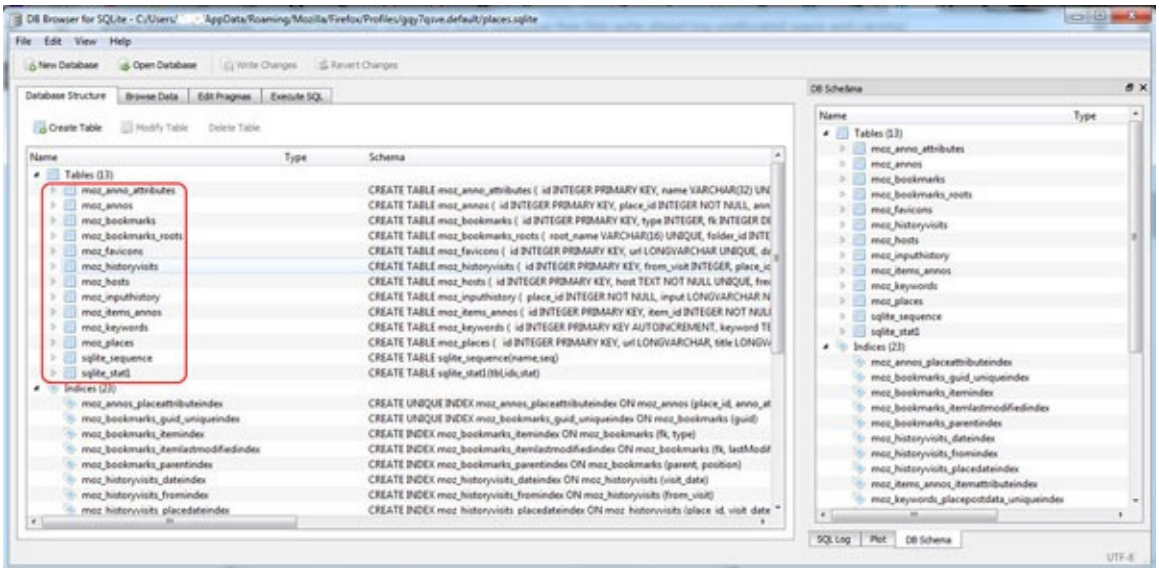


Figure 11.94: *Highlighting Moz*

The data in the chosen table will be seen in the SQLite browser if we pick the **`Browse Data`** option. The **`moz anno attributes`** are what we are looking at in the following screenshot:

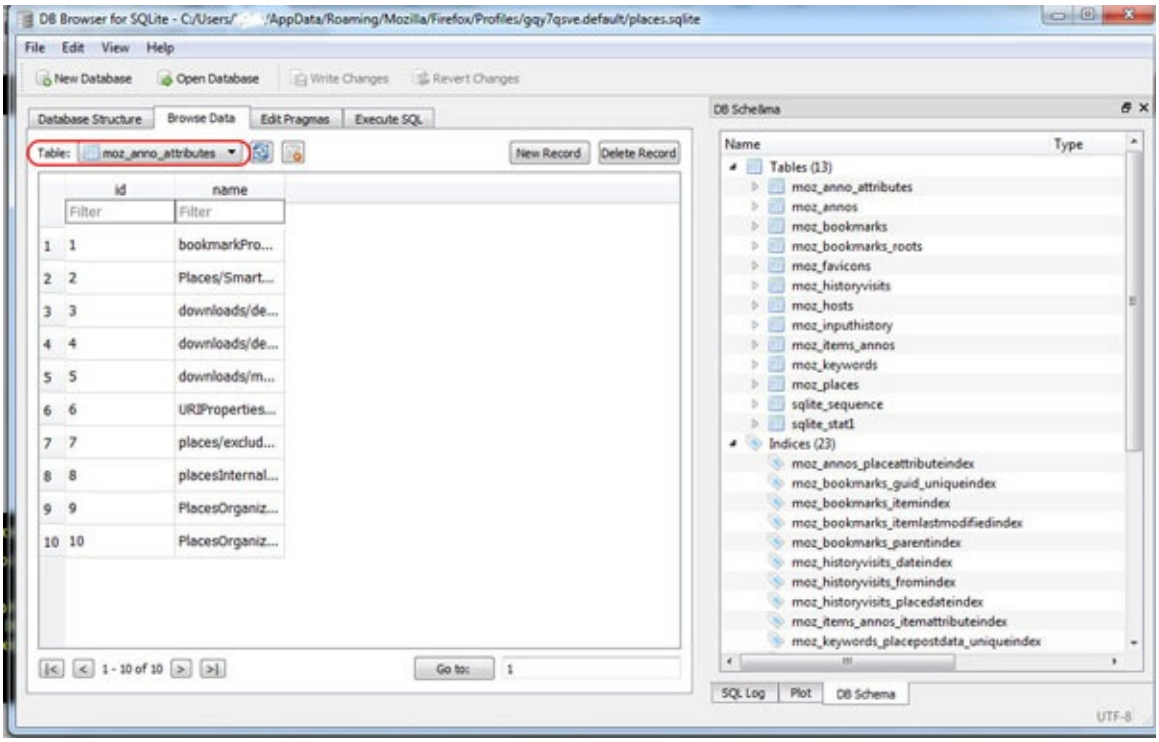


Figure 11.95: *Highlighting Mozilla attributes*

3. Querying the database

   You must be familiar with some fundamental SQL syntax in order to use the SQLite browser to discover evidence successfully.

   Let us search for the input that the user submitted into the browser in the Moz input history database. The **`Execute SQL`** tab should be clicked to launch a SQL query window. Then, by inputting, we might create a broad SQL query to locate all the input data:

   ```
   SELECT * FROM moz_inputhistory
   ```

   To run the query after typing it, click the play (>) icon.

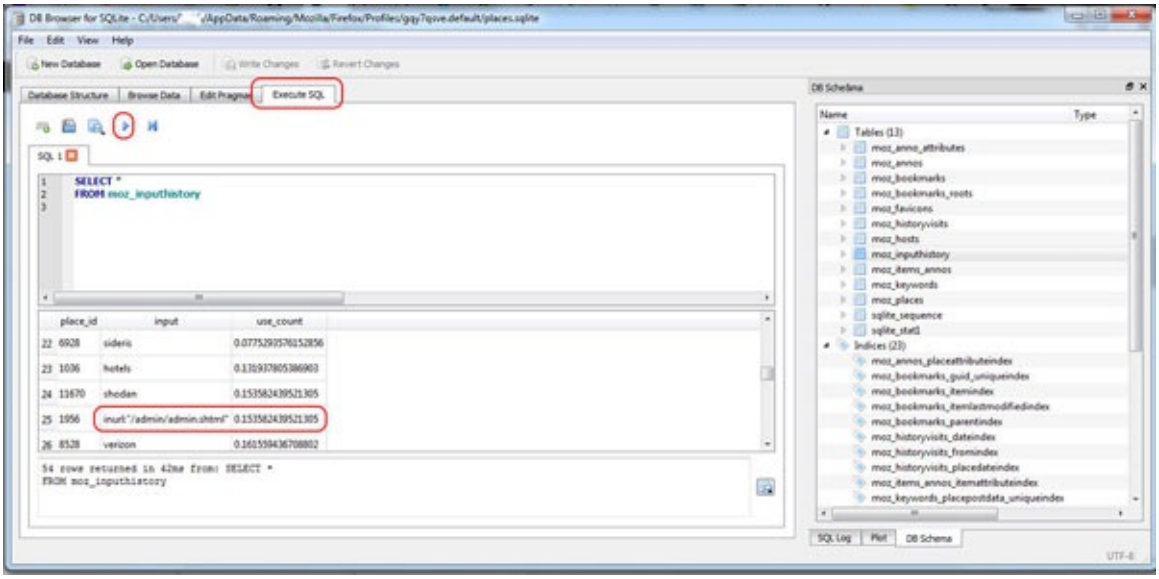


Figure 11.96: *Execute SQL*

As you can see, the suspect was typing some suspicious Google hacks using the keyword **`inurl`** and looking for admin directories. Hmm... we may be on to something here!

4. Finding specific user input

Let us assume that this was a case where the employee is suspected of having downloaded pirated files from a torrenting site (in many companies and institutions, this is prohibited activity, and in many countries, it is illegal). We could be very specific in our SQL query to find where the suspect may have input “tor.” We could find every occurrence where they typed “tor” querying the input history with:

```
SELECT * FROM moz_inputhistory WHERE input like ‘%tor%’
```

This query will provide us all columns (**`SELECT *`**) from the input history table (**`FROM moz_inputhistory`**) where the typed input is like “tor” (WHERE input like “%tor%”). Note the wildcards (%) before and after tor. This indicates that we are looking for anything before tor and anything after tor.

This query should provide us with results of any input by the user that has “tor” anywhere in it.

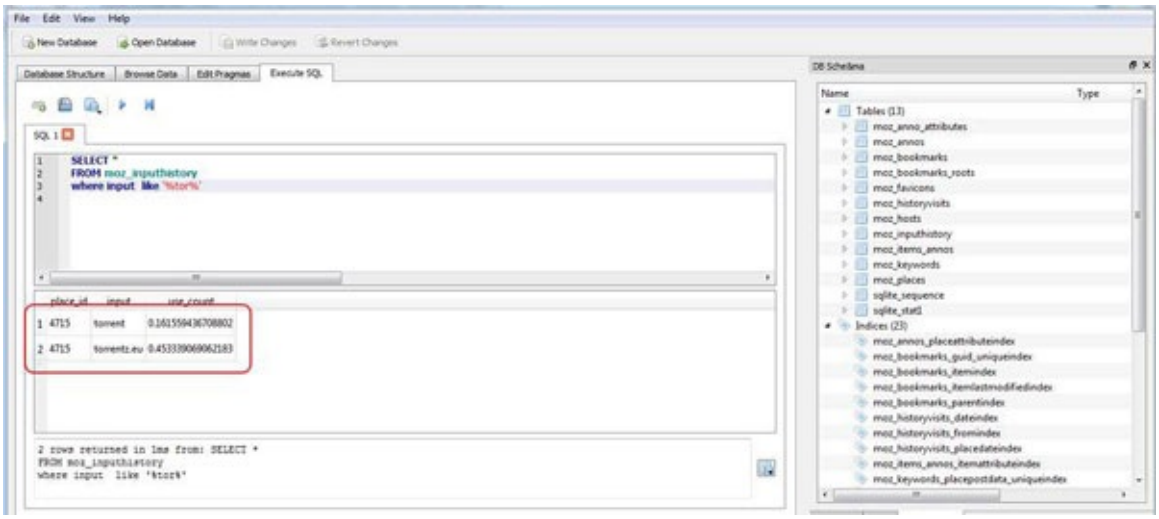


Figure 11.97: *Highlighting details*

The preceding screenshot shows two instances when the suspect/user entered “tor,” as you can see. This may be sufficient proof that the suspect was hunting for torrent sites, but it is possible that we need to explore a little further to locate the URLs of the sites in his location’s history (Moz places table).

Lab 11: Extracting EXIF data from graphics files

In many cases, when a computer, phone, or mobile device is seized for evidence, the system will have graphic images that might be used as evidence. Obviously, in some cases, these graphic images may be the evidence, such as in child pornography cases. In other situations, the graphic images may tell us something about where and when the suspect was somewhere specific.

Most digital devices “stamp” information on these graphic images that can tell us a lot about the who, what, when, and where the pictures were taken. This information is known as EXIF data and can very often be useful to the forensic investigator. **Exchangeable image file format** (**EXIF**) is a standard set by the digital camera industry to identify formats for digital images and sound files. This information includes camera settings, time, date, shutter speed, exposure, whether a flash was used, compression, the name of the camera, and other information critical to viewing and editing the image in image-editing software. This information can also be useful to the forensic investigator.

Originally developed for JPG and JPEG file formats, some other formats use EXIF data as well, but this data is not available for PNG and GIF image file types. There are numerous applications that can extract this EXIF data from graphic files. Nearly every one of the major forensic suites (EnCase, FTK, Oxygen,

and so on) has this capability built in.

1. Install ExifReader first

   Once ExifReader has been downloaded, clicking on the application will launch a neat and straightforward GUI similar to the one seen as follows:

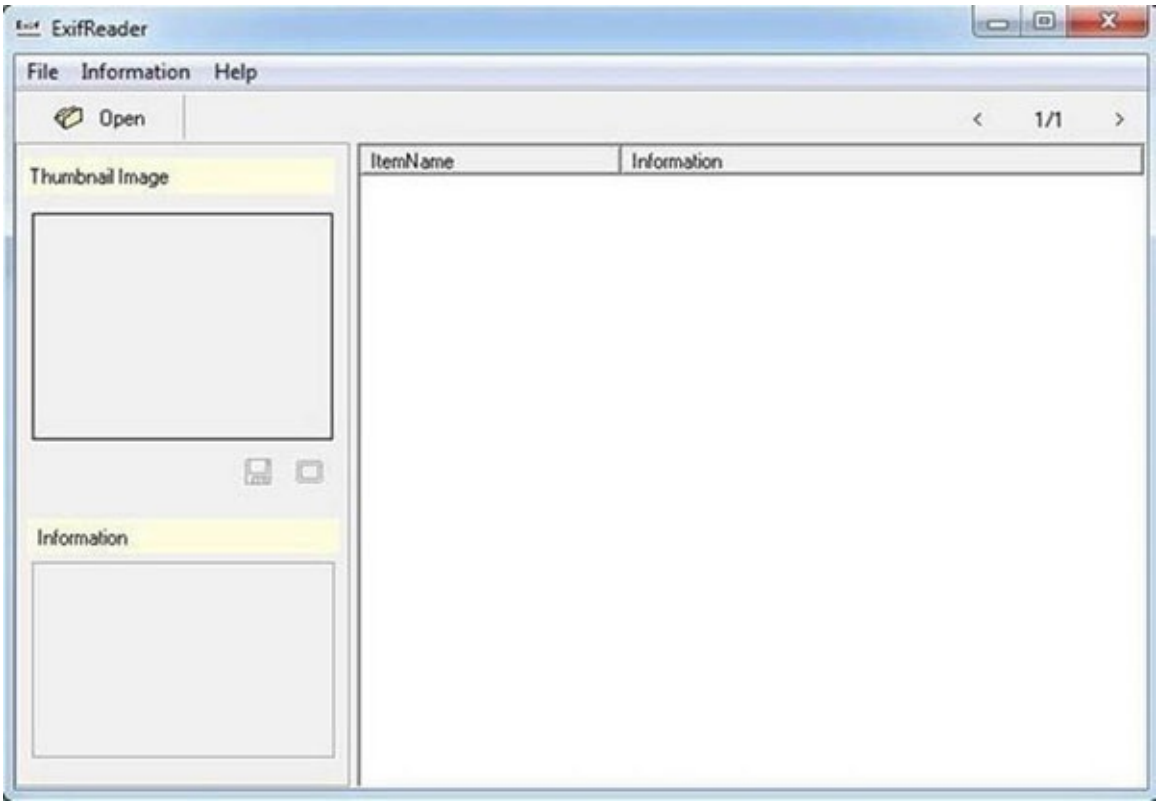


Figure 11.98: *ExifReader interface*

2. Try clicking the "**`Open`**" button at this point and navigate to the media or system that contains the pictures. Let us just use JPEG and JPG because they have the most data.
3. Open a picture file in step two.
4. ExifReader loads the image into the thumbnail on the left and displays the EXIF information further down the page when you open a photo file.

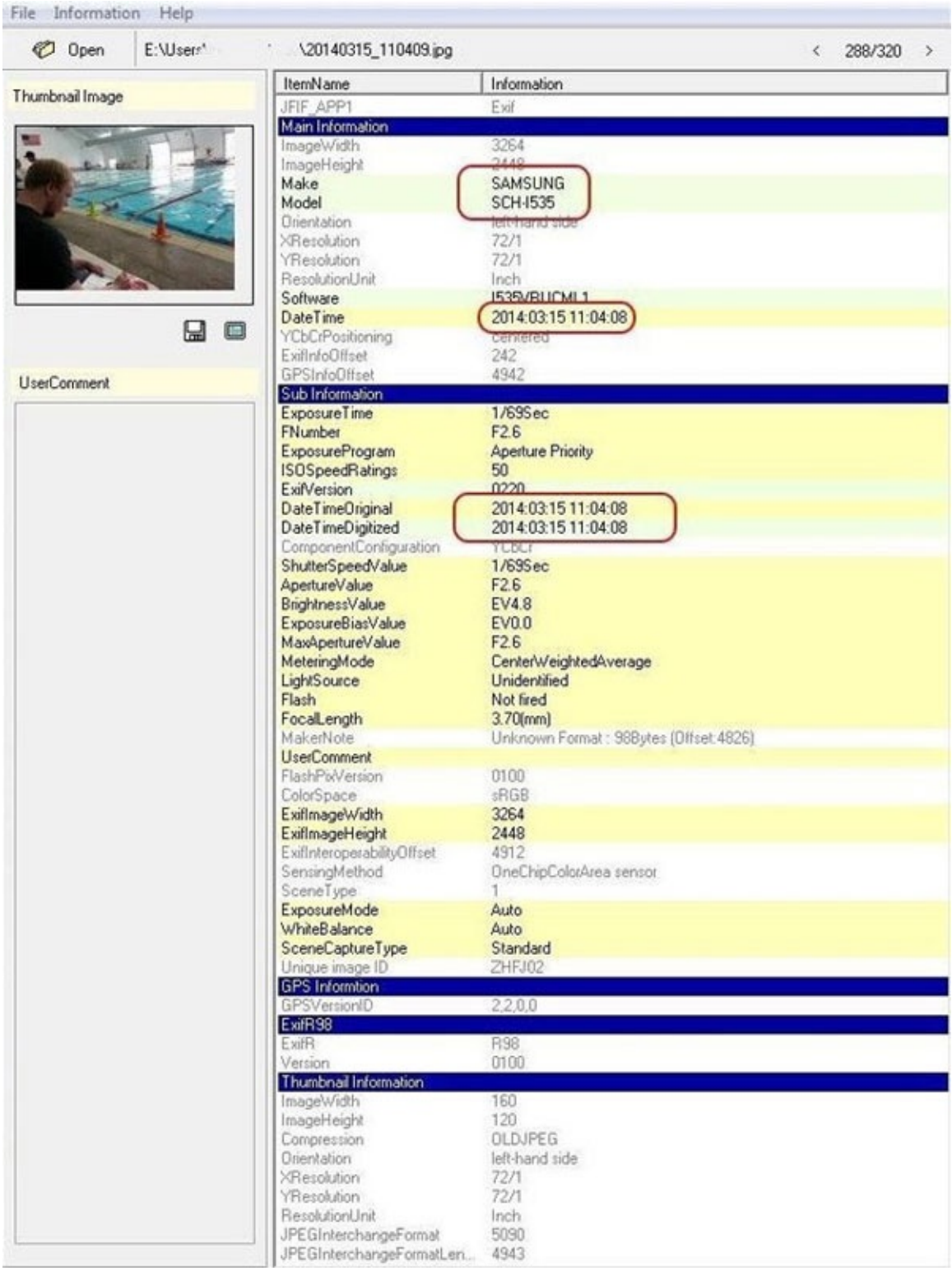


Figure 11.99: *Highlighting EXIF information*

Keep in mind that the photo on the suspect's laptop was shot on 15 March 2014, at 11:04 a.m., using a Samsung phone, version SCH-I535. The majority of the additional data in the EXIF data is concerned with the technical specifications of the equipment and cinematography. The majority of the material is just somewhat useful to forensic investigators. We would know the precise GPS coordinates of where the photo was taken if the device had GPS turned on at the time.

Extraction of EXIF from a **third image**

Let us try a different image. This will be another JPG file. Regrettably, the thumbnail does not show up when we open it. This occurs with a large number of images, yet the EXIF information is still visible even when the thumbnail is not.

As you can see in *figure 11.100*, it reveals that the image was captured on 18 February 2007 using a Nikon camera, model E3100.

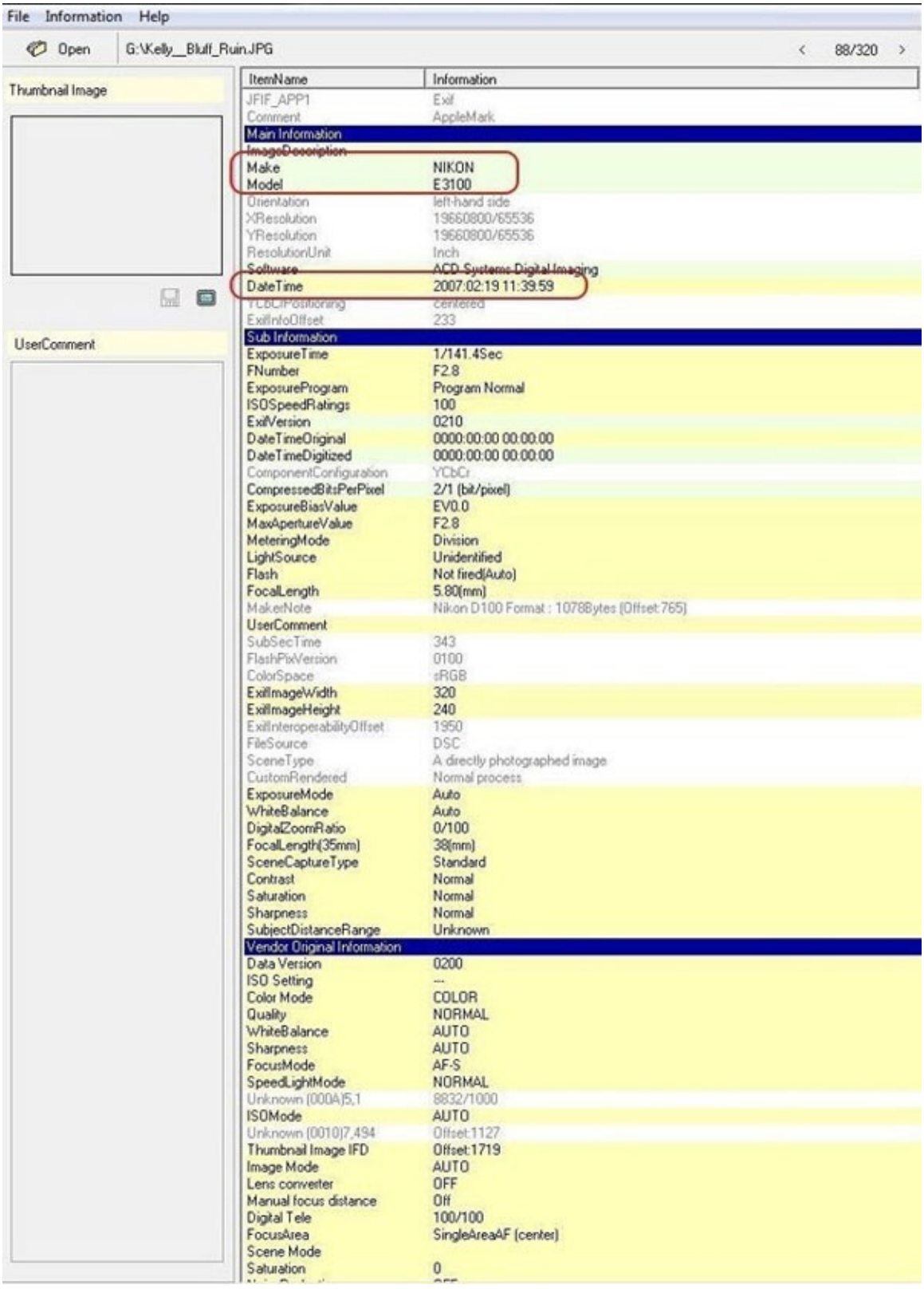


Figure 11.100: *Highlighting the NIKON camera model*

5. GPS data extraction

   We will check to see if we can recover the GPS information for the photo's shooting location from it.

Figure 11.101: *Highlighting GPS details*

6. Since this Apple iPhone had location services turned on, the GPS data, which is stated in latitude and longitude, can be found near the bottom of the EXIF data. Now that we have this information from the EXIF, we can use Google Maps or another mapping program to determine the precise location where this photo was shot.

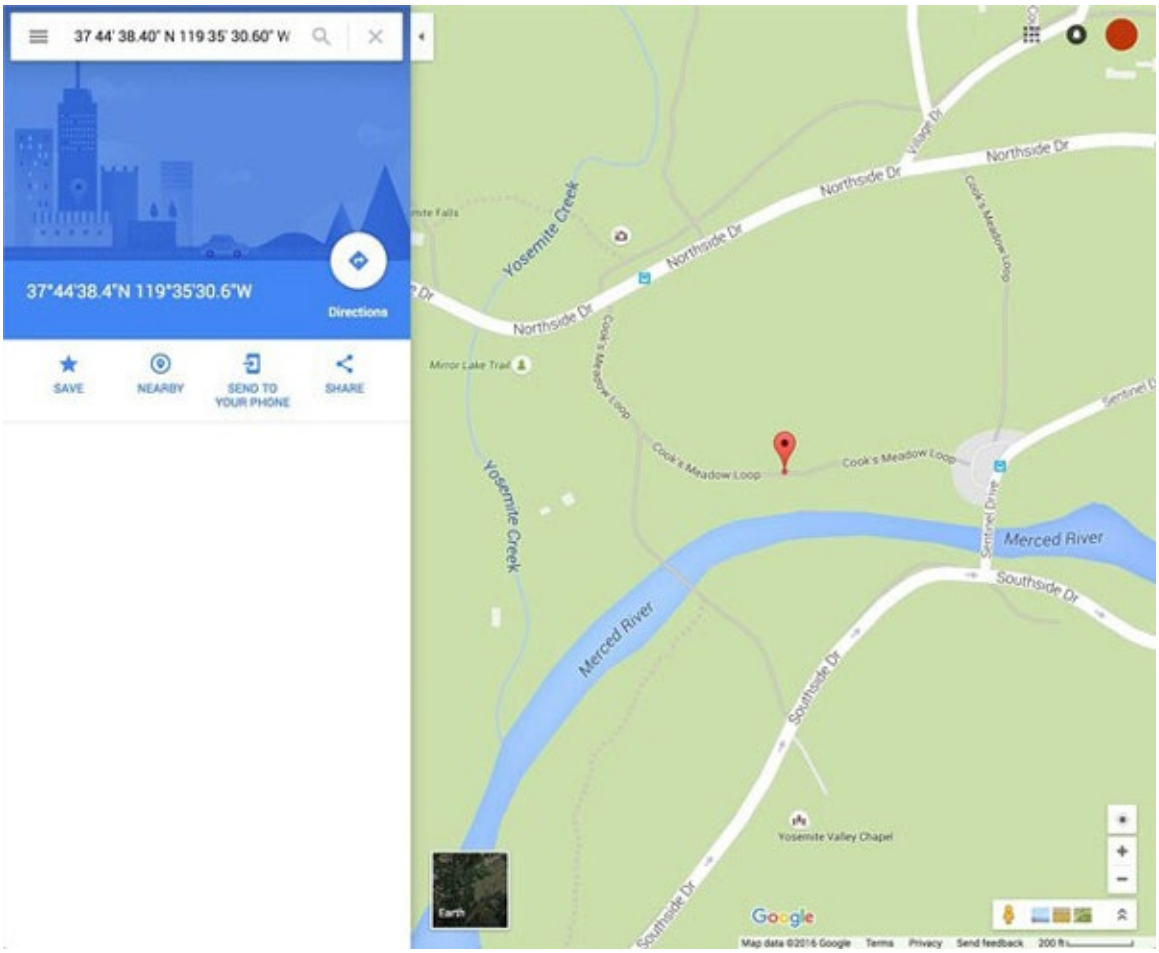


Figure 11.102: *GPS data*

When working with graphic picture files, forensic investigators frequently discover important information about the file and the perpetrator in the EXIF data. This data includes the brand and model of the camera, the day and time the shot was taken, and possibly even the location. All of this data might be helpful in a digital forensic inquiry.

Index

A

B

C

D

E

F

G

H

I

K

L

M

N

O

P

R

S

T

U

V

W

X

Y