

Unveiling Data Artefacts in Private Web Browsing A Comparative Digital Forensics Study

Supervisor: Dr. Saguna Saguna
Examiner: Professor Christer Åhlund

Brandon Spiteri

Information Security, master's level (120 credits)
2024

Luleå University of Technology
Department of Computer Science, Electrical and Space Engineering

[This page intentionally left blank]



Information Security, master's level (120 credits)

2024

Unveiling Data Artefacts in Private Web Browsing: A
Comparative Digital Forensics Study

Brandon Spiteri

braspi-2@student.ltu.se

Supervisors: Dr. Saguna Saguna

Examiner: Professor Christer Åhlund

Abstract

The Internet has brought invaluable advantages to society and has become a key component of our lives. Many things have been facilitated by this, for example, the ability to connect social to distant family members, ecommerce, business opportunities, or the possibility to learn through distance learning. However, it also raised concerns about user privacy and the issue of data harvesting during these online activities. Private Web Browsers aim to mitigate these concerns by reducing the amount of data that can be collected, yet a defined answer to what data artefacts can still be collected after a browsing session remains unclear, especially when digital forensics techniques are deployed. Data artefacts are residual information or traces that are left by using systems applications and devices.

This research aims to investigate these data artefacts by divining into the Computer Forensics branch and extracting data from Random Access Memory (RAM) and Hard Disk Drive (HDD). We note that these data artefacts may be affected by several variables which are the operating system, private web browser itself, state of the machine, and the state of the web browsing session. A series of experiments in a virtualized environment were conducted and the data collected was examined to determine what data artifacts were still recoverable and under which conditions.

Acknowledgements

I would like to express my deepest gratitude to my mother for her unwavering support, encouragement, and love throughout my academic journey. Her belief in me has been a constant source of strength and inspiration.

I am immensely grateful to my lecturers for their guidance, expertise, and dedication. Their mentorship has played a pivotal role in shaping my academic and intellectual growth. I am particularly grateful for professors Christer Åhlund and Saguna Saguna.

I also want to express my heartfelt appreciation to my girlfriend for her patience, understanding, and encouragement. Her unwavering support and belief in me have been a source of motivation during challenging times. I am grateful for her presence in my life and for the love and support she has consistently shown me.

I cannot forget to acknowledge my loyal companion, Oddie, for being my faithful study buddy and providing endless moments of joy and comfort. Your wagging tail and unconditional love have brought light into even the darkest of study sessions.

Finally, I am deeply grateful to all who have contributed to this thesis, directly or indirectly. Your support and encouragement have made this accomplishment possible.

Contents

Abstract.....	1
Acknowledgements.....	2
List of Figures	5
List of Tables	7
Abbreviations.....	8
Chapter 1 – Introduction.....	10
1.1 Introduction.....	10
1.2 Research Questions and Objectives	11
1.3 Significance of the study.....	12
1.4 Thesis outline.....	13
Chapter 2 – Key Concepts	14
2.1 Internet Privacy.....	14
2.2 Digital Forensics and Branches (Computer Forensics Focused)	14
2.3 Operating Systems	21
2.3.1 Different Types of Operating Systems	22
2.4 Non-volatile and Volatile Data	24
2.5 Machine State	30
2.5.1 Running, Shutdown, and Restart States.....	30
2.5.2 Sleep and Hibernate States.....	31
2.6 Web Browser Data Artefacts	32
2.6.1 Data Artefacts in Non-Volatile Memory.....	33
2.6.2 Data Artifact in Volatile memory data artefact.....	34
2.7 Private Web Browsers.....	34
2.8 Digital Forensics Tools.....	36
2.9 Preceding Literature Reviews	38
2.10 Knowledge Gap	40
Chapter 3 – Research Methodology	41
3.1 Overview	41
3.2 Experiment Explanation	44
3.3 Software and tools.....	46
3.4 Browsing Session Activity.....	48
Chapter 4 – Findings	51

4.1 Dead acquisition.....	51
4.2 Live Acquisition	60
Chapter 5 – Discussion	69
5.1 Discussion of Dead Acquisition	69
5.2 Discussion of Live Acquisition	70
Chapter 6 - Conclusions	74
6.1 Limitations.....	75
6.3 Future Work.....	76
Appendix	77
References	78

List of Figures

Figure 1 Digital Forensics Branches	17
Figure 2 Operating System Components	21
Figure 3 Market share of the desktop operating system 2024.	22
Figure 4 Operating System Market share of mobile and tablet operating systems 2024	23
Figure 5 System Bus	26
Figure 6 HDD components	28
Figure 7 Browsers in scope	35
Figure 8 Windows 11 Experiment Overview	43
Figure 9 Ubuntu 23.10 Experiment Overview.	44
Figure 10 Data Acquisition Scope.	51
Figure 11 Bookmarks Extracted from Microsoft Edge Running on Windows OS.....	52
Figure 12 Gmail bookmark.....	53
Figure 13 Web download file "Syllabus_FMISA.pdf"	53
Figure 14 Bookmarks Extracted from Ubuntu Running Microsoft Edge	54
Figure 15 Bookmarks extracted from Windows OS and Firefox.....	54
Figure 16 Web History in Windows and Firefox OS.....	55
Figure 17 Web browsing history extracted from Ubuntu and Firefox	55
Figure 18 Bookmarks extracted from Ubuntu and Firefox.....	55
Figure 19 YouTube bookmark extracted from Windows OS and Google Chrome	56
Figure 20 History extraction from Windows OS and Google Chrome.....	56
Figure 21 Web download from Windows OS and Google Chrome	56
Figure 22 Bookmarks Extraction from Ubuntu and Google Chrome.....	57
Figure 23 History extraction from Ubuntu and Google Chrome	57
Figure 24 e-Courts Bookmark Extracted from Windows OS and Brave Browser	58
Figure 25 Web History Extracted from Windows OS and Brave Browser	58
Figure 26 Download History Extracted from Windows OS and Brave Browser	58
Figure 27 Web browsing history retrieved from Ubuntu and Brave Browser	59
Figure 28 Bookmarks retrieved from Ubuntu and Brave Browser	59
Figure 29 Web Content Extracted from Microsoft Edge Running on Windows 11.....	61
Figure 30 URL return for Experiment 5.....	62

Figure 31 Image obtained from Experiment 13 CDN.....	63
Figure 32 Web page content of E17	64
Figure 33 Email Address Extraction from E29.....	64
Figure 34 URL Extraction from E29	64
Figure 35 Image extraction from E2	65
Figure 36 Extraction of credentials from E3.	66
Figure 37 E-mail content extracted from E27.....	67
Figure 38 Facebook chat messages extracted from E31	67
Figure 39 Images extracted from E4.....	68

List of Tables

Table 1 Kernel Types of Operating Systems.	24
Table 2 Data Artefacts.	32
Table 3 System Requirements.	42
Table 4 Machine States Explanation.	43
Table 5 Tools used Overview.	47
Table 6 Search Query results.	48
Table 7 Dead Acquisition Results.	52
Table 8 Microsoft Edge Experiment Results.	61
Table 9 Mozilla Firefox Experiment Results.	65
Table 10 Google Chrome Experiment Results	66
Table 11 Brave Browser Experiment Results.	68
Table 12 Total Data Artefacts Extracted.	74
Table 13 Total Data Artefacts per OS.	75
Table 14 Effects of the machine state on data artefacts.	75

Abbreviations

PII	Personal Identifying Information
USB	Universal Serial Bus
IDE	Integrated Drive Electronics
SATA	Serial Advanced Technology Attachment
CKC	Cyber Kill Chain
CPU	Central Processing Unit
I/O	Input/Output
HDD	Hard Disk Drive
SSD	Solid-state Drive
RAM	Random Access Memory
NFT	Network Forensic Tools
NFP	Network Forensic Process
AI	Artificial Intelligence
DBF	Database Forensics
DBMS	Database Management System
IoT	Internet of Things
IaaS	Infrastructure as a service
PaaS	Platform as a service
SaaS	Software as a service
DFI	Digital Forensic Investigator
DFP	Digital Forensic Process
OS	Operating System

GB Gigabytes

KB Kilobytes

MB Megabytes

ROM Read Only memory

BIOS Basic Input/Output System

TB Terabyte

DNS Domain Name Service

GUI Graphical User Interface

VM Virtual Machine

CDN Content Delivery Network

Chapter 1 – Introduction

This chapter serves to describe the overall objective of this thesis. First, an introduction to the subject will be highlighted to give a general overview of what domain will be investigated. The objective and scope of the thesis will follow, so a clear identification of key areas is further highlighted. The next subsection will include the motivation for this area and its significance. Finally, an overview of the outline will be explained.

1.1 Introduction

The increase of digital assets throughout the years has brought the utilisation of the Internet in everyday life. This has changed the way individuals communicate, interact, and conduct daily activities around the world. A single click of a button will immerse the user in vast repositories of knowledge, connections, online shopping, and convenience that was once only imaginable [1]. Privacy concern has been raised due to the amount of data that can be harvested during a browsing session. Various software now offer the possibility to browse the Internet in private mode. This is referred to as private web browsing. As with everything, this can be used for legitimate cases or for malicious activities such as cyberattacks, data leaks, and cybercrime [2].

When this tool is used for any malicious intent, law enforcement agencies are required to deploy specific techniques to aid in solving the crime. These techniques are called digital forensics, which encompasses several different branches, each having a specific domain. Generally, the main branches can be classified as cloud, IoT, mobile, database, network, and computer forensics [3]. Several data media, such as hard disk drives and random access memory, fall into the category of Computer Forensic. Non-volatile data refer to data that survive reboot even when no power is applied for a prolonged period. On the other hand, volatile data is generally lost when no power is applied after a certain time. The major differences are that volatile memory is faster when compared to nonvolatile memory, however, the amount of storage is significantly less[4], [5].

Even if Private Web Browsers are not an anti-forensic technique itself, they can still hinder the job of a Digital Investigator. Data artefacts extracted from these browsers can be less when compared to how the browser operates normally. These data artefacts are also affected by the operating system itself due to the storage and organisation of data. Other

factors could be the private web browser itself, the state of the machine, and the state of the browser. The state of the browser can be either open, which means that it is currently active, or closed, which means that the browser was active but is not anymore. The state of machine can be either of the following:

- Running State (Active State / On)
- Shutdown state (Turned off)
- Restart state (rebooted)
- Sleep state (standby state)
- Hibernate State

These affect what data artefacts can be retrieved due to RAM and how computer systems work in terms of processing data.

1.2 Research Questions and Objectives

The main general research question of the study could be defined as 'During a digital forensic investigation, what data artefacts could be found when using a private web browsing session using different operating systems and machine states'. However, this main question has been broken down into three smaller questions to facilitate the research process. The three research questions are as follows:

- Are there any data artefacts left on the device when different popular private Web Browsers are used, and if so, what are they?
- What are the differences between operating systems in terms of the data artefacts found when using Private Web Browsers?
- How do variations in experimental scenarios or states impact the types of data artefacts discovered on digital devices?

Question 1 aim will be to investigate whether popular private web browsers will leave any data artefacts on the device that it was used to navigate on the Internet. If there are any data artefacts found, would this mean that these private web browsers are in fact not private or less than perceived. Also, if data artefacts are found, which private web browser offers the best privacy when compared to each other.

The aim of Question 2 is to investigate two different Operating Systems and to be able to compare if the Operating System will affect any data found. If a private web browser is running on a Linux based operating system or a Windows based operating system, will they return the same results? This would mean that the same Private Web Browser will be evaluated on a different operating system and the results may vary, this could mean that the said Private Web Browser achieves better privacy when running on Operating System A than on B.

The aim of Question 3 is to finally investigate all these variables by running different experiments as they may return different results. An experiment would be simulating a state of the Operating System and the state of the browsing session itself. These may return different data artefacts which can be found in either volatile memory, nonvolatile memory, or both.

1.3 Significance of the study

Given the unique combination of variables used in this research, the resulting findings will be particularly difficult to find elsewhere. The specific pairing of Private Web Browsers with both Linux and Windows based operating systems will be examined to determine any differences of the data artefacts found when using the same Private Web Browser but different Operating system.

Digital forensics is a technique for extracting valuable information during investigations, and one will greatly benefit from the insights gained through this research. Furthermore, this study has the potential to assist the general user in selecting a private web browser that aligns best with their privacy preferences. Finally, we think that it is extremely important for the user to know what data artefacts could left on their device from a Private Web Browsing session, while also important for Digital Forensics Investigators to know what data artefacts they could possibly extract. This study will serve as a helping aid for law enforcement agencies and other professionals which specialise in the field of digital forensics.

1.4 Thesis outline

This section gives an overview of the thesis structure and chapters.

- Chapter 1 Introduction – this is the introduction section of the thesis where an overview of the major key concepts, along with the research question, are highlighted.
- Chapter 2 Key Concepts - This section will go in depth of the main concepts by presenting literature review and background studies conducted in the area. Concepts such as the difference between Digital Forensic branches, Operating Systems, machine states, data artefacts, and private web browsers are reviewed.
- Chapter 3 Research Methodology – In this section a methodology is explained in which the experiments were conducted. This includes what websites were in scope, the user interaction, the different states of the machine, the private web browser, and the operating system.
- Chapter 4 Findings – Once the experiments were carried out, a digital investigation was carried out and the findings were recorded. To facilitate the process, the experiments were divided into 2 categories, dead acquisition and live acquisition. Dead acquisition focused on data extraction from non-volatile media while live acquisition focused on volatile data. The findings were presented in a high-level table and an explanation was provided, along with evidence of the data artefacts.
- Chapter 5 Discussion – In this section, the findings of part 4 were evaluated and the meaning of the results was extracted. The differences between the results and which operating system performed the best compared to the batch of experiments was also highlighted. A total of four batches were needed to systematically examine the results obtained.
- Chapter 6 Conclusion – is the final part where a conclusion is drawn to answer the research questions presented in this thesis along with future work and possible research areas.

Chapter 2 – Key Concepts

2.1 Internet Privacy

The Internet has brought numerous advantages to our daily lives and has become an integral part of society. Easy access to information, expanded economic prospects, enhanced social connectivity, e-Commerce, and broader educational opportunities are a tiny fraction of these advantages [1]. Internet privacy, sometimes referred to as 'online privacy', essentially refers to the right an individual person has regarding the storage and purpose of such data. One of the issues of the Internet is the privacy of the user and what data is mined when surfing the Web. These can range from non-personal identifying information such as your actions on the site visited or more critical data such as personal identifying information (PII). As users are becoming more educated about how online companies use the data they collect, the cost of their privacy is being evaluated [6].

When browsing the Internet, various data artefacts are extracted from the session, which could mean that every click, search, or interaction with the browser could be monitored and monetised. This can raise several privacy issues, including tracking and surveillance. Tracking is when you browse the internet and ads start showing up based on previously visited searches. This is because of cookie profiling which is used to track your activities online and creates a profile associated with you and your browsing habits which is a serious invasion of privacy. Surveillance can be imposed by governments on citizens online presence, which can have benefits such as aiding law enforcement. However, this can be another privacy concern, but some instances, such as the UK's Investigatory Powers Act, authorise this legally [7]. Communication service providers are also required to retain customer internet connection records for a year, these can be obtained by the government authorities to be used in investigations.

2.2 Digital Forensics and Branches (Computer Forensics Focused)

Digital forensics is, as the name suggests, a forensic investigation on media that stores any form of digital data in one way or another. These techniques are usually used in investigations led by law enforcement or incident response teams. This technique may be used in a court of case, and thus it is extremely important to follow the correct procedure so

that the evidence extracted will be valid. According to [8] there is a process to gather such information. They specify that the process is composed of the following:

- Identification of digital evidence
- Preservation of digital evidence
- Analysis of digital evidence
- Digital evidence presentation

Also, according to [9] the process is very similar in most cases after they have analysed various number of models and framework for digital forensics. This is due to the fact that all frameworks will have overlapping strategies since the foundations are the same.

Digital evidence identification – during this step both have specified that it involves the identification of potential sources which can include digital evidence. These can be a wide range of items, but in a nutshell, these can be all electronic devices that have some form of memory associated with it.

Preservation of digital evidence - as with all other evidence, it is extremely important to preserve the evidence without altering it. A copy of the original data is extracted to be examined to prevent alteration of the original copy. When copying this data there is a possibility of writing to the disk itself, and thus a write blocker is used for such purposes. As described by [3], when talking about a write blocker, there are essentially two types, which are hardware and software. These prevent any write commands from being written to disk and thus maintain data integrity. Keeping the integrity of the data will ensure that the hash of both the original and the copy matches. A hash is generated by a hash function, which is a mathematical algorithm that is applied to the data and a value called 'hash' is deduced. A change in any value will result in a completely different hash, and thus the original and the copy will not match.

- Hardware write blocker - These are physical devices that are placed between the data being copied and the data source from which the forensic extraction will take place. To prevent data from being written, the write blocker will allow read-only access. These can take different forms, an example would be Universal Serial Bus (USB), Integrated Drive Electronics (IDE), or Serial Advanced Technology Attachment (SATA).

- Software write blocker – is like their hardware counterparts but are not as effective as them. They are software or applications which create a virtual write blocking environment that intercepts the write commands. Another issue is that they rely on the host operating system, and thus some issues may arise. For these reasons, it is not advisable to use such means in a court of law, as the evidence may be contested, since there is the possibility of it not being equal to the original.

Digital Evidence Analysis - In both articles, this process is found to be an integral part of the part of the technique for obvious reasons. Once the data is ready for examination, it will be analysed to determine different artefacts depending on the scope of the analysis. The authors in [8] have done a great job of identifying the process that is required during this stage. They identified that there are multiple steps during a cyber attack that are referred to as the 'Cyber Kill Chain' (CKC) by Lockheed Martin. This CKC highlighted the different phases that take place during a cyber-attack, and then by making use of the framework they proposed, it was highlighted the steps to analyse such phases. It is important to mention that there are different categories of artefacts to be found during a Digital Forensic Investigation. This 'D4I' incorporates the selection of the artefact category when starting the analysis process.

Presentation of digital evidence - At the end of the process, digital evidence must be presented to a court of law or to the technical personnel that need to act. Documentation will be crucial in this part to determine where and how the artefacts were extracted. When presenting these data to a court of law, the digital investigator must make sure that that was preserved correctly and that chain of custody was established.

Digital Forensics is very vast, and thus it is divided into branches to undertake an investigation at the highest level. These branches may overlap in some instances, meaning that a digital investigator may have to deal with multiple branches per incident. However, the main five branches are computer forensics, mobile device forensics, network forensics, database forensics, IoT forensics and cloud forensics [3] [10]. Figure 1 demonstrates the six main branches of Digital Forensics.

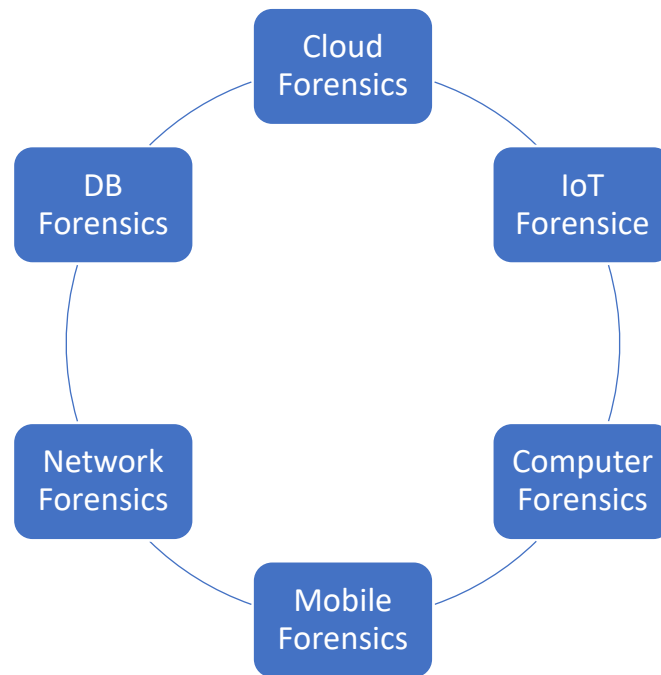


Figure 1 Digital Forensics Branches.

Computer forensics seeks evidence in computers and digital storage devices in order to locate, save, retrieve, and evaluate digital data for use in court cases. It is similar to data recovery in that it is used in both civil cases and computer criminal investigations, but it places more emphasis on generating a legal audit trail. Evidence maintains a clear chain of custody by adhering to established protocols, particularly those pertaining to other digital evidence [11]. This field looks at data artefacts in the hard disk drive (HDD) and random access memory (RAM). Digital forensics tackles the rise in crimes involving computers by locating and tracking down the perpetrators. Computer forensics is crucial to maintain online evidence in the face of ongoing dangers of data breaches, and plays a crucial role in preventing cybercrimes through a variety of tactics and instruments within the broader field of cyber forensics [10], [11].

Mobile forensics is another branch of digital forensics that focusses on recovering data artefacts from a wide array of mobile devices. Throughout the years, devices have been shrinking in space and becoming more portable, we also have seen an advancement of Internet of Things devices along with wearables such as medical equipment, NFC smart rings, and smart watches [12]. According to research, the number of smartphone users has increased to 6.25 billion in recent years due to cost being more affordable and the development of applications such as commerce, health e-learning, finance, social networks,

games, entertainment platforms, etc. [13]. Similarly, the number of tablets, portable gaming systems and notebooks have also found their way in everyday life, which also fall within the mobile forensics branch.

These devices can contain a vast amount of data artefacts that are of particular interest to a digital forensic investigator. Data such as call logs, text, emails, photos, media, app data, text documents, and browsing history can prove to be invaluable during an investigation and could aid in the solving of crimes, however, this field proves to have several challenges. Mobile forensics inherit the weakness that such devices are battery powered and a certain amount of data artefacts may be lost if this battery is depleted. Another weakness is that because these devices are portable, encryption and security features are more widely adopted, which can hinder the data acquisition process. Other weaknesses are the range of operating systems found and data fragmentation and overwriting due to the limit amount of storage available [14], [15].

The next branch that forms part of digital forensics is network forensics, which is a unique field specialising in monitoring and analysing the traffic that is generated by machines on a network. The focus can be defined as the gathering of information about network traffic which can be used either to provide legal evidence or for the detection of intrusions and malicious behaviour. This is done either by real live network monitoring or by analysing captured network traffic which is of interest. Specialised tools called 'Network Forensic Tools' (NFT) and 'Network Forensic Process' (NFP) are extremely important in the process of collecting network data, analysing and examining these data. The NFP will assist in incident analysis by distinguishing between normal and abnormal data, which results in effective incident detection and response [3], [16].

Thus, network forensics can be described as an intersection between digital forensics, incident response, and network security. It can address both internal and external network attacks, while also providing an investigation of devices connected to a network. Challenges arise due to the sheer amount of data that is generated and expert resources need to investigate said data, which can be false positives. Artificial intelligence (AI), machine learning, and deep learning are increasingly becoming employed to detect attacks and classify traffic with accuracy. A variety of data sets are available to the public, which are relevant to network forensics and are used by these algorithms. However, some issues

remain because companies may not receive information about latest security breaches and reputational purposes [17], [18].

Database forensics (DBF) is another branch of digital forensics which is significant since data extracted from these databases can aid digital investigators and solve crimes. Essentially, the focus is on the study and analysis of databases, focussing on metadata, cached information, server RAM, transactions, and queries. Metadata is data about data, which can provide information on aspects of the data such as but not limited to authors, date created, date modified, and file size. Part of the examination process can involve timestamps, the updating of rows, and validating user actions, which can identify transactions indicating wrong doings. Thus, DBF will attempt to reconstruct the incident, by creating a chronological timeline of the activities conducted by intruder or malicious personnel [19].

As with the other branches, there are several challenges within the DBF, with several researchers focussing on the lack of a standardised process for an investigation. Another issue is that the literature often addresses case-specific scenarios, which can include outdated works, limited reviews, and a diverse database infrastructure. A database management system (DBMS) is software that will provide an interface to enable interaction with the database itself. However, different DBMSs will have an internal structure such as data storage mechanism, indexing methods, and data compression techniques. Another factor is the logical structure such as design, schemas, and relationships between entities, and thus the investigator will need to understand the conceptual model. The digital investigator must also understand the different views, queries, and reports that can be generated by a user [20], [21], [22].

The Internet of Things (IoT) forensics is another branch that focusses on the investigation of devices that are part of the IoT. The Internet of Things is essentially any device that is capable of connecting and exchanging data with other devices over the internet[23]. The ecosystem of these interconnected devices grows and IoT forensics is gaining importance to handle security incidents and investigate cybercrimes. IoT devices come in different shapes and forms such as smart kitchen appliances, home appliances, and even wearables such as fitness trackers. These IoT devices generate a large amount of data that can be extracted from billions of connected devices through the Internet [24]. This branch will also intersect with network and cloud forensics, since the data generated are transferred via the internet

and could be stored on a cloud environment. Thus, this ecosystem presents several benefits for our daily lives, but it can provide several challenges from a digital forensic point of view.

There is a lack of standardisation between vendors on data formats, which makes it challenging for digital investigators to develop forensic tools and procedures that apply across all IoT devices. These IoT devices also come with different functions, operating systems, and communication protocols, which means that an investigator must be very versatile [25]. Encryption is a major difficulty across all the branches, IoT devices use encryption to securely communicate and store data. Data can also be corrupted during transmission between devices, this can be due to transmission errors such as flaws in the process. The huge amount of data generated also has a flaw which is that it will be time consuming and resource intensive for experts, which can delay the identification and mitigation of cyber threats and criminal activities [26], [27].

Cloud Computing offers various services such as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). The IaaS model provides on-demand access to resources such as servers, storage, and networking. PaaS offers platforms for developing, running, and managing applications, while SaaS offers ready-to-use application software [28]. Cloud computing is becoming employed by businesses of all sizes due to its benefits such as cost savings, security, flexibility, insights, and business continuity among others. With the widespread usage of these systems, the cloud computing forensic branch has caught the interest of experts. Similarly, to the other branches, this specialisation focusses on the investigation of these specific mediums [29], [30]. These investigations can be from both the victims and the perpetrators perspectives, since attackers could also make use of such systems to conduct their attacks. One difference to note is that cloud forensic often does not have a physical component as with the other branches mentioned above due to the nature of cloud computing.

Cloud Computing Forensics has several challenges, which can be both legal and technical. These cloud services are hosted in different states or countries from the actual location of the user and jurisdiction complications will arise. For example, if someone based in Europe commits a crime through a cloud system hosted in America, there will be complications in determining which law enforcement agencies have authority over the crime[31]. The ability of a DFI to freeze the systems to carry out the investigation is impossible in the public cloud

since these will serve other customers. This means that the environment remains live and can be changed and thus evidence potentially lost. As mentioned above, physical limitations are applied to the DFP as regulations could prevent physical access to the premises or the cloud server could be located throughout the world from the DFI [32].

2.3 Operating Systems

A key part of the functionality of every device, being either portable or not, virtually or physically, is the operating system (OS). The OS can be described as the collection of software, and its main function is to manage the resources of a computer. These can be hardware and software resources, and thus it is in charge of the services for all computer programmes. Different use cases will need different operating systems, as it controls what tasks are to be achieved [33]. The primary use of a mainframe OS is to optimise the utilisation of hardware, while on the other hand, personal computers are designed to support a different number of applications and thus the OS is adjusted accordingly. Some OS are designed to be user friendly and thus prioritise an easy to use interface, while others are focused on efficiency. The operating system is generally divided into four components that are the hardware, the operating system itself, the applications, and finally the users that will run the OS [34]. Figure 2, which can be found below illustrates the connection between the Operating System components.

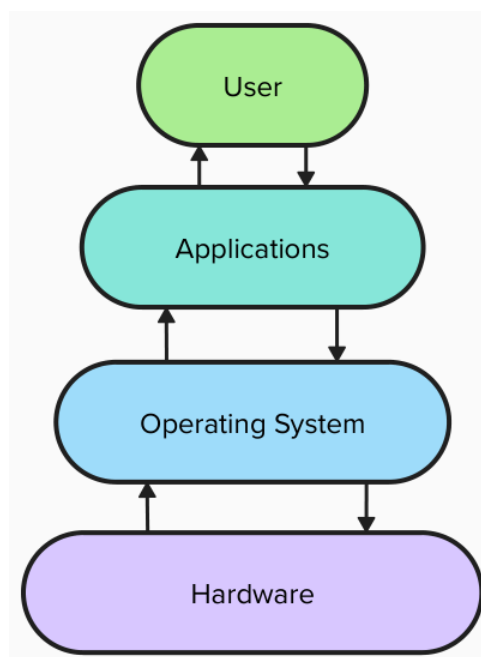


Figure 2 Operating System Components.

The hardware is mainly composed of the Central Processing Unit (CPU), Random Access Memory (RAM), storage hard disk drives (HDD) or solid-state drives (SSD), and Input/Output devices (I/O) such as keyboard, mouses, and monitors. As mentioned earlier, the OS acts as an intermediary between the hardware and the applications so that the system can perform its task. Applications are software programmes that perform a specific task for the user, such as browsing the Internet through a web browser [35]. Finally, the user is simply the individual who made the interaction with the machine, either via a graphical user interface or through the command line interface. The quality of an operating system can be assessed based on various attributes such as memory management, process management, storage management, protection, security, and software features. Different operating systems, such as Windows, Linux, Mac, Android, iOS, and others, have their own unique features and usability, which can influence user satisfaction and adoption [36].

2.3.1 Different Types of Operating Systems

As seen previously, different use cases will require different operating systems, and this is reflected in the market share. Like previous years, as of January 2024, Microsoft Windows holds had a signification portion of the market shares in the personal computer domain, which was 72.99%. A significant 16.13% of the market share is secured by macOS by Apple Inc. While the rest of the share is composed of the varieties of Linux, including ChromeOS [37]. Figure 3 demonstrates the market share for desktop operating systems as of 2024.



Figure 3 Market share of the desktop operating system 2024.

In the domain of mobile devices, which includes smartphones and tablets, we see a change of events. In January 2024, Android's share was 69.09%, followed by Apple's iOS and iPadOS with 30.18%, and other operating systems with 0.73% [38]. Figure 4 demonstrates the market share for mobile operating systems as of 2024.



Figure 4 Mobile & Tablet Operating System Market share of mobile and tablet operating systems 2024.

Several factors affect why a user may opt for one OS rather than the other but as a general overview these can be the cost, licencing, user interface, software compatibility, customizability, and flexibility. Linux distributions are usually open source and free, which means that anyone can download and use them, which is different for both Windows and macOS. The graphical user interface while being based on personal preferences, however, all OS offer a different amount of themes that can be applied to fit ones needs. Regarding software compatibility, Windows gains the edge in some instances due to the history of supported software vendors and its popularity throughout the years. However, some alternatives to these software are fond of other OS and thus mitigating the gap, while other software is proprietary and thus specific to the particular OS [36], [39], [40].

A kernel is the heart of the OS since it is the bridge between the hardware and the software and is responsible for managing the resources. This allocates CPU time for processing for different software and memory resource management, while also moving data between RAM and disk storage, which is called memory swapping. File system management is also handled by the kernel, and this stores data on storage devices by creating, reading, modifying, and deleting files and directories. The kernel of a Linux OS is developed by a large community because it is open source, while Windows is based on the proprietary Windows NT kernel, which is maintained by Microsoft [41], [42].

The way the interaction between the kernel, hardware and software is made is classified into five types, which are Monolithic kernel, Microkernel, Hybrid kernel, Exokernel, and Nanokernel. In a monolithic kernel, all the OS services reside in the kernel space, and it typically provides high performance since the system call will lead to minimal overhead. Microkernel, as the name suggests, is aimed towards keeping the kernel as small as possible, and typically, components run in separate address spaces. The hybrid kernel combines both micro- and monolithic kernels by providing process management in the kernel space and file systems in the user space. Both Exokernel and Nanokernel take a

minimalist approach. Exokernels expose hardware resources directly to applications by allowing applications to make decisions on the resources, while Nanokernels provide only essential functions and minimal task scheduling and are usually used in embedded systems where efficiency and predictability are critical [43], [44]. Table 1 shows various operating systems along to which kernel type they belong.

Table 1 Kernel Types of Operating Systems.

Kernel Type	Operating System
Monolithic	Linux, Android (modified Linux kernel), FreeBSD, OpenBSD
Hybrid	Windows NT family, macOS, iOS
Microkernel	MINIX, QNX
Nanokernel	L4 family
Exokernel	ExOS

2.4 Non-volatile and Volatile Data

The end goal of digital forensics is to collect data artefacts which can be found in two different types of memory classification, and these can be either volatile or nonvolatile memory. Depending on which type of data the investigator is attempting to extract, different techniques will be applied.

Volatile memory is the most delicate between the two as this type of information requires power to maintain the information stored. This means that it is critical to extract this type of data during the time that the device is powered on, since once turned off, these data can be potentially lost. Volatile memory is used for temporary storage of data that need to be accessed quickly by the CPU. This tends to be faster due to this requirement, and is stored in the RAM or in cache memory [45]. Both these locations are limited in terms of storage amount and thus adds to the complexity of data that can be lost due to data being cycled. RAM is the primary type of volatile memory found in computers and is used by the CPU to store data about a programme that it is currently using. Cache memory is used to store frequently accessed data and instructions. It also acts as a buffer between the CPU and the RAM, which improves the performance of the data to be accessed [46].

The main four differences between RAM and cache memory can be described as size, function, hierarchy, and persistence. RAM has a significantly larger capacity than cache, and modern devices range from a few gigabytes (GB) to several tens of GB. Cache memory is much smaller and is typically measured in kilobytes (KB) or megabytes (MB). A simple conversion would be that 1GB is equal to 1024MB and thus the different size between RAM and cache could be in the thousands. The function of RAM and Cache was already described, although both offer data at high speed, Cache is much faster, and thus expensive, which explains the difference in size. Memory speed is based on a hierarchy, and the closer the memory to the CPU, the faster it is [5]. Cache built directly into the CPU to give it the fastest possible access to memory, but it is divided into multiple levels, L1, L2, and L3. L1 is faster than L2, and L2 faster than L3, but all the levels are faster than RAM, which in terms of hierarchy, is between the cache and the HDD or SSD in terms of speed. In terms of persistence, both RAM and cache generally lose all data stored when the machine is completely off [47].

Data can be stored into nonvolatile memory, which opposes volatile memory and retains the data even when the power is turned off. This makes nonvolatile memory suitable for long-term storage of data that is needed even when the machine is turned off. Data using this type of memory can be stored in two primary types of Read-Only Memory (ROM) or in various secondary storage types [48]. The purpose of read-only memory (ROM) is to store data which can only be read and thus not modified or overwritten. ROM is thus used to store firmware or system software, which are essential for the boot up of the machine such as the BIOS. BIOS stands for Basic Input/Output System and is used by the processor to start the machine after it is powered on. Secondary storage devices are used to store data that need to survive reboot, and these include HDD, SSD and flash memory. The operating system is stored on this type of permanent storage that can range from hundreds of GB to terabytes (TB) [49].

Although both ROM and secondary storage are nonvolatile, there are distinct differences between the two. The first difference is the nature of Data Access, while ROM is designed to store critical software components used during boot up, secondary storage, has various use cases. Secondary storage allows the user to write data and is used to store the OS, user data such as applications, and personal files. Another difference is the accessibility of the

data by these two storage types. ROM data are accessed directly by the hardware during boot up without user intervention, while data located in secondary storage can be accessed by the OS, applications, or by the user. Both non-volatile and volatile data have their specific function during daily operations of a machine but are very different in how they operate [50], [51].

2.4.1 Volatile Data Operations

It is important to explain how the data is written to either volatile or non-volatile memory, figure 5 below is a simplification of communication paths.

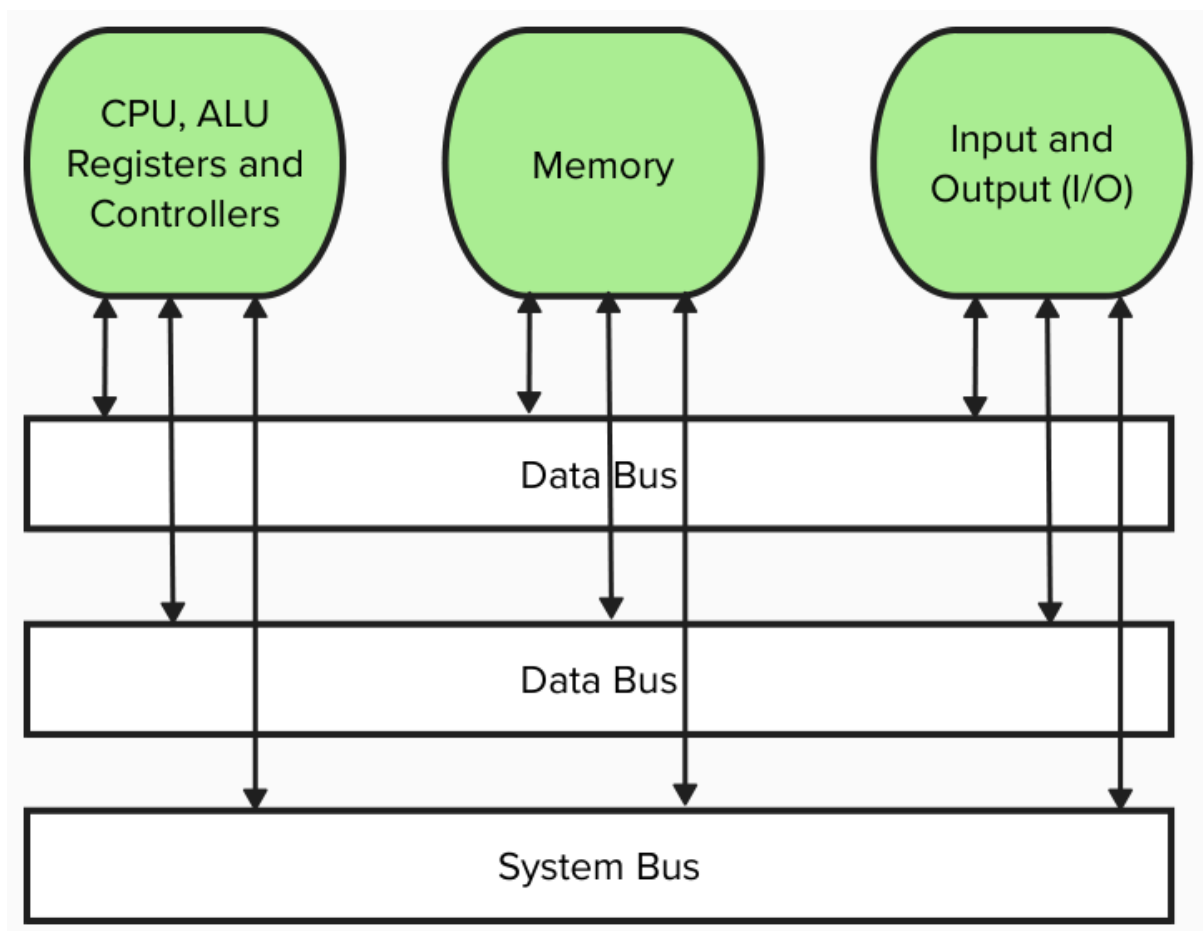


Figure 5 System Bus.

During processing, volatile data is transferred from secondary storage to RAM, which involves several steps and components within the computer architecture. This data transfer must be facilitated by the memory bus, which acts as a communicator between the CPU, RAM, and second storage devices. A critical component of the memory subsystem is called a 'memory controller', which is the manager of the data transfer operation that occurs

between the RAM and the secondary storage device to ensure optimised data transfer. To achieve this, the memory controller uses buffering, prioritising, and optimising data transfer. Buffering stores temporary data storage before writing it to the RAM, which reduces latency. The memory controller prioritises data transfer, based on the demands of the CPU [4].

The write operations are then organised to minimise latency and increase throughput, and this is influenced by memory allocation algorithms. These algorithms determine the allocation and management of memory within a RAM by making use of contiguous allocation, paging, and segmentation. Contiguous allocation is the allocation of contiguous blocks of memory to store data, which reduces fragmentation and thus minimises overhead [52]. Paging divides the RAM into fixed-size blocks called pages, and by making use of the page table, it manages the data retrieval. Finally, segmentation makes use of logical segments to divide the RAM and manages data storage based on the boundaries of these segments.

Memory spaces are allocated through dynamic, stack, or heap allocation techniques, and once the data are no longer needed, it is deallocated. Data transaction can be done through explicit transaction or automatic garbage collection, thus the memory space is reclaimed and can be re-used [53]. From a forensic analysis point of view, it is important to understand these processes due to what is called “Data Persistence”. When data are deleted from the RAM by either of the deallocation techniques mentioned, the data may persist in memory until it is overwritten, which can be of interest to a forensic investigator. However, due to automatic memory allocation, valuable data may be deleted, which complicates the forensic process [54].

2.4.2 Non-Volatile Data Operations

As in the previous section, this section will focus on the write operations conducted when data are transferred to these secondary storage devices such as HDDs. These techniques will impact the operations and the implications for data that will impact the forensic analysis. HDDs make use of magnetic storage to write data onto spinning platters that are coated with magnetic materials. By making use of the read/write heads, the platter surface is transformed into magnetised surface which represents the binary data in terms of 1s and 0s. These write operations are influenced by a mechanical aspect since the performance is

based on the disk rotational speed, seek time, and data density [55]. This means that the higher the rotations per minute the HDD achieves, the faster the data is written to the disk.

The integrity of non-volatile data located on an HDD can be affected by several factors such as magnetic interference, head crashes, and sector errors. Since the data are represented by the magnetised surface, if magnetic interference occurs, it can corrupt the data stored on the sectors. This can occur due to nearby electronic devices or powerful magnets, but the extent of the damage will depend on the strength and duration of the interference [56]. Thus, a forensics investigator can make use of shielding environments such as Faraday bags to mitigate external magnetic fields. Head crashes occur when the read/write head of the HDD makes physical contact with the magnetic platters. This happens due to sudden shocks to the disk or mechanical failures and will cause physical damage to the platters, and the data will be lost or corrupted. Sector errors occur when individual sectors of the HDD develop errors and become unreadable, and nonvolatile data may be lost [57]. Figure 6 [57] describes the components mentioned along with others that are found in an HDD.

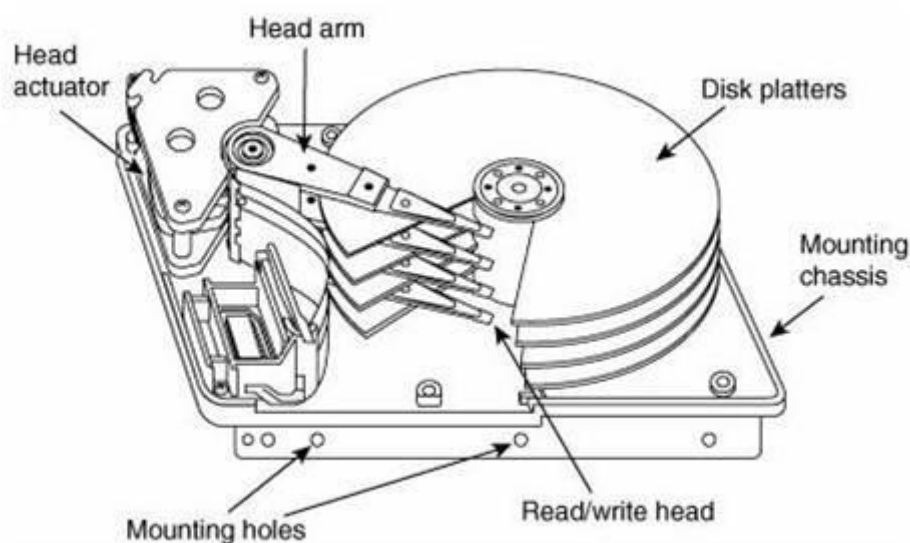


Figure 6 HDD components.

Mounting holes are simply standardised openings that allow the HDD to be securely attached to the host system. These are typically located on the sides of the chassis and prevent movement of the HDD, and thus the possibility of head crashes. The mounting chassis is sometimes referred to as an enclosure or housing and is the external casing that houses the internal components of the HDD [58]. This is made of metal or plastic and is

designed to host internal components while also providing protection against external forces. The head actuator is responsible for positioning the read/write head over the platters where data are stored. This movement is controlled by a servo mechanism to precisely position the heads over the platters and is connected to the head arm. The head arm, known as the actuator arm, holds the read/write heads and moves them to the position on the platters as needed [59].

SSDs are an alternative to HDD and are commonly found in machines, but they use a different write operation because they do not have a mechanical component [60]. Data is written by making use of NAND flash memory cells. When writing data to the SSD, electrical signals are sent to the NAND cells, which alter their charge levels and store the data. The controller is responsible for the management of these operations, data storage, and retrieval. This is faster compared to HDD since no mechanical parts are involved; however, write performance can degrade over time due to NAND wear [61].

The removal of data from nonvolatile memory devices is different from volatile memory and different techniques are deployed between HDD and SSD. In HDD, when a user wants to delete data, file system deletion is used. The file system marks the space occupied by the data that needs to be deleted as available for new data. The OS will remove the file from the file allocation table or the master file table, making it appear as deleted [62]. However, these data are not deleted at this point and remain intact in the physical sector until new data are written at the location of the deleted data. This means that deleted data can be recovered during digital forensics if new data has not been written yet. To prevent this, some specialised tools are developed to overwrite the sector with random or meaningless data multiple times, which is referred to as data wiping [63].

Like in HDD, when a user deletes a file from an SSD, the location of the data is marked as available for new data. The actual data remains stored in NAND flash memory until the controller erases them. Wear levelling algorithms are used to prolong the lifespan of NAND flash memory cells, and as a result, deleted data may be retrieved from the drive for a prolonged period. The TRIM command is used by the controller to collect garbage and reclaim unused blocks; however, this does not erase immediate blocks but only informs the controller that the blocks are available [64]. Secure erase is a feature that allows the user to securely delete data by resetting all NAND flash memory cells to their defaults. This can be

done through the SSD firmware or specialised software tools, and it deletes all data, making it unrecoverable even with forensic techniques[65].

2.5 Machine State

During daily operations, a machine goes through different states, which can be referred to as the 'machine state'. There are various states and can vary depending on the operating system and type of hardware. There could be different machine state names, but if the indicators are equivalent these can be understood as correct definition. Below is a general overview which tries to encompass a wide range of machine states.

- Running State (Active State / On)
- Shutdown state (Turned off)
- Restart state (rebooted)
- Sleep state (standby state)
- Hibernate State

2.5.1 Running, Shutdown, and Restart States.

When a machine is in running state it signifies that the machine can perform active operations such as execution of tasks, data processing and perform interaction with the user. The computer architecture is essentially able to perform computational and I/O operations by running various applications and services as requested by the user. In this state, dynamic behaviour is demonstrated, such as response to user interaction, execution of scheduled tasks, handling of requests, and maintenance of throughput under varying workload [66]. The resource management are optimal, and hardware resources such as CPU, memory and storage are allocated by the OS.

The shutdown state represents the process of powering off the machine, which terminates the complete cessation of processes that are running. This state is triggered by the shutdown command, which can be triggered by user or system events. The OS when receiving such a command will start the termination of the current processes which are running. By doing so, you are safeguarding the data saved on the system and the resources[67]. Essentially, there are two methods to achieve this state, graceful shutdown or forced shutdown. Graceful shutdown is the better approach as the system prompts the running processes to save their state before powering off the machine. Forced shutdown

can be needed if the system is unresponsive and there are no other ways to shutdown the system. This will terminate all processes without allowing them to properly shutdown and thus can result in data loss [68].

The next state in which a machine can be in is the restart state, which, at a high-level glance, does both shutdown and automatically turns on the machine again. This refreshes the state of machine to an operational state. The restart state is started by the user or by an automated process that triggers the restart command [69]. This state uses two phases, the shutdown phase, where it uses a graceful shutdown, and the reboot phase, where it triggers the initialisation of hardware components and the startup procedure. This has a specific purpose which can include system updates, troubleshooting, and refreshing the machine state. System updates may require a restart to be fully installed, while troubleshooting may include restarting the system and refreshing the machine to a fresh boot state [70].

2.5.2 Sleep and Hibernate States

The sleep state is essentially a low powered operational state where the machine reduces energy consumption while preserving the current state to quickly resume the previous state once required. The CPU activity is suspended, and the operations are paused, but essential system data and configurations are retained in the volatile memory. This state can be initiated by the user or by the system settings, such as battery-managed mode. When the sleep state is initiated, a sequence of actions occurs which are save system and power down components. The save system will preserve current system state, processes, application data, and configurations in volatile memory[71]. Power down non-essential hardware such as the CPU, graphics cards, displays, and peripherals occurs to reduce power consumption. The wake-up mechanism can then be initiated by user input such as keyboard or mouse response to activity, which resumes tasks quicker when compared to turning on the machine from start. Some other features can also be used, such as Wake-on-Lan, which remotely wakes up the system via a network signal or scheduled wake-up based on predefined scheduled tasks [72].

The hibernate state is like sleep mode in the sense that it represents a power saving mode where the current state is saved, allowing the machine to power off complete while preserving its operational state. This operational state can be resumed late but it saves the content of volatile memory, the RAM, to nonvolatile storage which is either the HDD or SSD.

This enables the machine to resume the exact state before hibernation even if the power is completely lost [73]. This is also triggered by the user or by predefined scheduled tasks and has two main sequences, the save system and power off. The process involves writing the contents of the RAM to a hibernation file stored on secondary storage, which may include processes, data, and user configurations, then the power is turned off. When the system resumes from the hibernation state, the machine will read the hibernation file from secondary storage. This is done during the boot process and restores the state of the system saved from secondary storage back to the RAM, allowing the user to continue the tasks as before entering hibernation [74].

2.6 Web Browser Data Artefacts

Web browsers can leave several traces of what was browsed, viewed, interacted with, or done on the internet, which could potentially lead to the establishment of the events in a digital investigation. Volatile and nonvolatile memory can both store valuable data artefacts related to this activity. It is very hard to complete a list of all the data artefacts that can be found in these media, but the following table identifies ten possible data artefacts for each type of memory that can be of significant relevance in a digital investigation. The following were chosen due to the potential significance in a criminal investigation, however, further data may be found during digital forensics. Table 2 illustrates these chosen data artefacts.

Table 2 Data Artefacts.

Data Artefacts in Non-Volatile Memory	Data Artefacts in Volatile Memory
Internet Browser cache	Web page content
Browsing History	Browser tabs and Windows
Cookies	Cookies
Bookmarks	Cache Entries
Download history	Network Requests and Responses
Form Autofill Data	Form Data
Session data	Search terms
Browser Preference	Session History
Browser Extensions and Add-ons	Email fragments
DNS cache	URLs

2.6.1 Data Artefacts in Non-Volatile Memory

The Internet browser cache is used by web browsers to store copies of web pages resources such as images and scripts that have been accessed by the user. This boosts performance since instead of redownloading the resources from the Internet each time the web page is visited, it is retrieved from cache. The browsing history is the records of web pages visited by a user that includes the URLs visited and the timestamps when it was accessed [75]. This can be very fruitful in an investigation, as it is evidence that the machine accessed the web page during the time marked by the timestamp. Cookies are stored on the user device by websites to remember user preferences, track user interactions, and maintain user sessions [2]. These are sent through each request and response, which offers a personalised experience, but also tracks user behaviour. Bookmarks are essentially shortcuts that a user creates to quickly visit web pages. These can also be organised in a folder structure and allow the user to visit websites without having to type the URL each visit. Download history is a list of all files that have been downloaded by the user, which can include the file name, source URL, timestamps, and download location. This is very convenient for the user, but if a crime is committed, this can be very valuable for a digital investigator to gather evidence [76].

Form autofill data are information that is automatically entered into web forms by the browser which saves user time when faced with commonly used fields. These fields can be personal information such as email addresses, name, surname, addresses, phone numbers, and credit card details. Session data is the data with the user browsing session that can include authentication tokens and temporary session storage [77]. This data is usually deleted once the session ends or the browser is closed. Browser preference is a specific configuration that is customised by the user, such as the homepage, preferred language, and search engines. These are modified through the settings tab of the browser and are stored locally on the user device. This information, like the language, can be very useful in an investigation since it can aid in confirming the identity of who uses the machine. Browser extensions and add-ons are small software that extends the features of the web browser that are installed by the user and stored in the browser configuration files [78]. The Domain Name Service (DNS) cache stores records of recently resolved domain names and IP

addresses. This can result in revealing which website the user accessed recently, which can be very fruitful in an investigation [54].

2.6.2 Data Artifact in Volatile memory data artefact

The web page content, which includes all elements to render a web page could be found during digital investigation, still located in the volatile memory. The web content may be found if the forensic image is taken before or after the web browser has been closed and will lead to strong evidence [75]. Similarly, browser tabs and windows may be extracted, which will in turn have multiple web page running. This could lead to extraction of chats, images, videos, and other resources. Cookies and cache entries could also be found in volatile memory and the same concepts as with nonvolatile memory apply [79]. Network requests and responses can be found in volatile memory, since some browsers may log network activity for either debugging or performance purposes.

Form data are similar to the form autofill data that can be found in the secondary storage; however, this can be found in volatile memory too. If found in volatile memory, it can be potentially more useful as data that was used in the actual form will be extracted rather than possible autofill data. This means that the data were inputted in the form of the web page visited by the user and is termed transient [80]. Search terms are phrases that are entered by the user into the search engine and that may be temporarily stored on volatile memory. Session history is the browsing activity during a browsing session and could be used to quickly navigate through pages, for example, with the back and forward buttons. This can then be transferred to nonvolatile memory as described in the previous section. Email fragments could be extracted, which may be able to recover portions of an email since email clients may make use of volatile memory as a temporary storage during processing. URLs are addresses that identify web pages, images, or files over the Internet, and since they are not stored persistently on the device, they could be temporarily stored in volatile memory [77] [78].

2.7 Private Web Browsers

Private Web Browsers is a term used for a browser that allows you to browse the Internet without saving data that normal browsers would. Normally, data artefacts such as browsing history, search history, cookies, bookmarks, cached images and files, file download history,

and other temporary data are saved by the browser, but Private Web Browsers are designed to eliminate these. Private web browsers address privacy concerns by promising to reduce the data mined from internet usage to a certain extent. The extent to which this occurs is somewhat ambiguous, as there is no clear answer to what data artefacts can still be extracted when using these Private Web Browsers when using Digital Forensics techniques [81], [82]. Figure 7 shows the well known icons of the Private Web Browsers in scope.



Figure 7 Browsers in scope.

Microsoft Edge was released in 2015 to replace Internet Explorer and offers significant improvements to enhance performance, security, and usability. Edge integrates easily with Microsoft services, such as OneDrive and Office 365 which enhances productivity. Robust privacy features have been incorporated into Edge, such as the browser tracking prevention feature and the InPrivate browsing mode[83]. These aim to prevent third-party trackers and prevent the storage of browsing data. The tracking prevention feature works by analysing the site components the user is visiting such as scripts and cookies, and tries to identify potential trackers based on known tracking domains or behaviour. Depending on the user's privacy preferences, the tracking prevention feature can be adjusted between strict, balanced, or basic. When using InPrivate mode, the browser will create a temporary browsing session, and this prevents data storage [84], [85].

Google Chrome was released in 2008 and offered fast performance, robust security features, and was offered across different operating systems. Features such as tabbed browsing, extensions, developer tools, and incognito mode were all offered on Chrome. Tabbed browsing is the concept of using tabs to navigate across different web pages, extensions enabled users to customise their browsing experience, while developer tools facilitated web development and debugging [82]. The incognito mode is similar to other

browsers and claims to allow the user to browse the Web without storing data. Several options are also available which preserve privacy are available to chrome, such as safe browsing, privacy settings, and the possibility to clear browsing data. Due to Google's ecosystem and data collection methods, Chrome has concerns remains regarding data collection of such browser[85], [79].

Mozilla Firefox was released in 2002, and as an open-source project, it was built to prioritise privacy and security. As with Edge, Firefox has an enhanced tracking protection feature that automatically blocks third-party trackers. Extensions, developer tools, and private browsing are also available in Firefox, which allows for a personalised experience, development features, and the ability to navigate web pages without data collection [86]. Due to this browser being open source, it enables community review, which ensures transparency in the development process. This transparency also allows users to know information about what data is collected and how it is protected [82].

Brave Browser is developed by brave software, and since its release in 2016 it has been gaining popularity. Brave is built on a chromium engine and was developed with a focus on privacy and security. Brave Shields is a built-in ad and protects against trackers and malicious scripts [87]. Tor is also integrated in brave, which enables users to browse the web anonymously by making use of this feature. Like the other browsers, Brave can make use of the private browsing mode, which is to prevent storage of browsing data. Brave tries to ensure encryption and will automatically upgrade your connection to HTTPS if possible, when your connection is made through HTTP [79].

2.8 Digital Forensics Tools

To perform these Digital Forensics techniques, tools are required, Various tools can be found, but some of the very best include for these scenarios are FTK Imager, Autopsy, and volatility [88]. All these variables will also be affected by the states of the machine running the operating system and the private web browser. These can occur after the Private Web Browser has been closed after a browsing session, while it is still open, the machine has been restarted while the Private Web Browser is still running, or the machine has been restarted after the Private Web browser was closed [89].

Once a digital forensic investigation is initiated, a tool to extract forensic images is needed, and FTK Imager is very popular among investigators. The FTK Imager was developed by AccessData and is designed primary to acquire forensic images from both volatile and nonvolatile sources [90]. The image, which is an exact copy of the memory to be examined, is used instead of the original copy to preserve the integrity of the original memory. In terms of volatile storage, this is done via live RAM imaging, while for nonvolatile memory, a feature called disk-to-disk imaging is used [91]. The captured RAM snapshot enables the investigator to investigate data such as processes, network connections, and registry keys, while the disk-to-disk feature will create an exact copy of the disk. A built-in tool to verify the integrity of the image is also provided by FTK Imager, and this checks the hash of the Image with that of the original. The FTK Imager makes use of a user-friendly interface and offers the possibility to extract forensic images in different formats [92].

Autopsy is an open-source forensic tool developed by Basis Technology and is widely used by investigators. This tool allows the investigator to navigate the contents of the disk images to extract valuable data using a graphical user interface (GUI) [77]. A feature that proves to be extremely useful is the ability to make use of keyword searching and data carving. This gives the possibility to search for specific keywords that may have been used during the crime. Email analysis is also built into Autopsy, and this can examine email messages, attachments, and meta data, which can lead to crucial data [93], [94]. An important feature is web artefact analysis which extracts data from web browsers such as history, cached files, cookies, and downloaded files. Autopsy also offers the possibility of integrating custom plugins that can provide a specific solution to investigations [93]

Another open source tool that has gained popularity among investigators targeting RAM forensics is volatility. This tool is written in the popular programming language Python and is cross-platform, meaning that it can be used on all operating systems that can run Python [95]. Volatility enables an investigator to make use of plugins to extract valuable data such as process information and network activity. Being open source, it enables customisation of plugins to target one needs during an investigation. Finally, volatility also enables the possibility to image volatile memory during live forensics [76], [93], [95].

2.9 Preceding Literature Reviews

In this subsection, the literature reviews regarding the topic of digital forensics and more specifically private web browsing are evaluated and summarised. This is to extract possible knowledge gaps in the forensic sector, where further investigation is needed. These preceding literatures are used as a foundation on which this thesis I built on. In all three reviews, a research gap was identified that served as motivation to investigate these gaps during this thesis.

The research conducted by [82] titled 'Digital forensic analysis methodology for private browsing: Firefox and Chrome on Linux as a case study' was published in 2022. This was one of the few papers that made use of a Linux environment when investigating different private web browsers. In fact, they mention that to the best of their knowledge, there was only another study conducted by [96] that was specific to Linux. The researchers stated the need for privacy due to raising awareness around the world, which was developed by several events such as revelations and the adoption of GDPR. They mention that there are tools that can show network traffic during a browsing session, but they focused on FTK Imager, LiME, and Autopsy.

During their methodology, they mention that several steps were taken in their browsing phase such as accessing a web page, open new tab, enter URL, download PDF files, and login. Then, the data acquisition phase would commence where data were extracted on both the hard disk and the memory. They made use of four tests for both volatile and nonvolatile memory. These tests varied from taking memory dumps while the browser is running, after the browsing session, on reboot, and 10 seconds after shutdown. They mentioned that some BIOS can wipe the contents of the RAM on reboot, and in such cases the results may be different. It resulted that by using digital forensic techniques it could be possible to extract some keywords, and files, but the level varied between Firefox and Chrome and the test environment. They also concluded that different hypervisors may return different data and that this could be examined in future research, along with other operating systems, private web browsers, and not using hypervisors at all.

Research conducted by [81] titled 'A forensic examination of web browser privacy modes' was published in 2019 and has similar, but different, results to those of [82]. The

researchers stated that by highlighting the importance of privacy in the United States and that 33% of users reported that they use private web browsers. The researchers state that volatile memory can hold valuable information even when extracting data from private web browsers. An evaluation of 30 web browsers was conducted, and the testing machine was running a Windows 10 OS. The tools chosen were X-Ways Forensics 19.7, Griffeye Analyze DI Pro 18.5 and LACE Carver v.12.8.56.

The methodology is different from that found in [82], they focused on retrieving search terms from the web browsers after the machine has undergone a standard shutdown. To limit the amount of data that can be extracted to only those relevant to the investigation, the browsers were downloaded on an external machine. Once the browsers were installed, the same methodology was used to navigate to five URLs. During their results, they mentioned that four web browsers encountered an error, which were Lynx, Links, Falkon, and Konqueror. Also, three web browsers did not have the possibility to run in private mode which were GreenBrowser, Netsurf, and Sleipnir. This resulted in that in practice twenty-two private web browsers were tested. The researchers stated that image cached data was also present during extraction. Finally, they concluded that some factors, such as the amount of volatile memory, can affect the amount of data.

The research conducted by [79] titled 'A Critical Comparison of Brave Browser and Google Chrome Forensic Artefacts' was published in 2022 and investigated non-private web browsers. The researchers chose to investigate Brave and Google Chrome running on Windows 10 operating system. The chosen OS was determined due to its popularity, and they state that the OS controlled 76.26% of the market share in 2021. The choice of the web browsers was due to popularity of Google Chrome and on contrast the less popular Brave Browser. VMWare Workstation Pro 15 was used to create the virtualized environment where the OS would reside.

The methodology was similar to both [81], [82], yet again it included important variations. The virtual machine was used for one hour, during which time periods, browsing sessions were conducted. Several URLs were visited, files were downloaded, searches were affected, browsing data was deleted, and finally the browser was closed. The researchers chose to use FTK Imager to extract the image and X-Ways WinHex and NirSoft tools to extract the data. During the examination, several artefacts such as Cache, Cookies, Media and Search

History, and File download history were extracted. Another location that holds valuable data was the user local AppData which had stored typed URLs, keyword searches, and graphics. We note that a significant amount of data were extracted, and this is possibly due to not using the private web browser but rather the standard browser.

2.10 Knowledge Gap

The preceding literature leads to the question of questions that were partly addressed in future studies of such a literature. Researchers in [82], [81], and [79] that more research would be needed to determine the role that the operating system plays in terms of data artefacts extracted during the investigation. This leaves a gap since the researchers only investigated one operating system and did not compare what data artefacts the same private web browser would leave when running on another OS. Both [82] and [81] used two private web browsers and compared each of them on a Linux or Windows machine, but as seen by [79], several web browsers have the ability to operate in private mode.

The researchers highlighted the possibility that some of results may vary if the tests are recreated in a non-virtualized environment, which brings the question, what are those differences, if any. Another gap identified is the test that the researchers put into their scope. Neither of the researchers in [82], [81], and [79] investigated all possible 'test' scenarios due to time limitations. The scope the researchers conducted was a combination of scenarios in the domain of running, shutdown, or restart states. This leaves at least two states to investigate, sleep and hibernation. When combining these variables which are, different operating systems, private web browsers, and states, an interesting research area is extracted.

Chapter 3 – Research Methodology

Experimental Research Design was used during this research due to the main concept being a linear relationship between variables. These variables were a combination of a Private Web Browser, Operating System, and States. The motivation for choosing this method comes from previous researchers that deployed this method during similar research.

During the research conducted by [77] this method was deployed during a comparison of two different Operating Systems, “Kali Linux” and “Parrot OS” forensic tools. Their scope for was investigating data artifacts found by digital investigators which is generated by the web browsers and stored on the devices used during the browsing session. The researchers utilized a virtual environment to simulate a browsing session and subsequent digital investigation.

Experimental research design served to identify and measure the relationship between the variables. With this design we can measure the effect of an independent variable (Private Web Browser/States/Operating System/Search Engine) on the dependent variable (Data Artifacts).

Thus, the chosen Experimental research will then quantitatively measure the impact of variables like states, Operating Systems and Search Engines on Private Web Browsers, allowing for a comprehensive understanding of their relationship in terms of data artifacts found on the device used during the browsing session [97], [98].

3.1 Overview

In this chapter, we describe the variables the experiments pertaining to digital forensics and data artefact extraction from volatile and nonvolatile memory. There will be a total of 32 experiments which will consist of different operating systems, private web browsers, and machine states. The private web browsers used in the experiments were Microsoft Edge, Mozilla Firefox, Google Chrome, and Brave. The reason for these specific browsers was as follows:

- Microsoft Edge – Comes preinstalled on Windows Operating Systems.
- Firefox – Comes pre-installed in Linux operating system.
- Google Chrome, popular web browser.

- Brave browser – This browser is developed with a priority to privacy.

The private web browser will be tested on two different operating systems, a Windows- and Linux-based operating system. The Operating Systems were chosen due to the popularity of Windows which had a market share of 72.99% in terms of personal computer as of January 2024. Linux, on the other hand, was chosen due to two aspects, the first being the popularity of Linux-based OS in mobile devices, which had a market share of 69.09% as of January 2024. The other reason was due to the lack of investigation of such an operating system in the Digital Forensics domain as described by [82] in 2022. The researchers state that at the time, only one other study was conducted in terms of extracting web browsing artefacts from Linux based operating systems. The Operating Systems used in the experiment are as follows:

- Windows 11 version 23H2
- Ubuntu 23.10

The specific Windows and Linux operating systems were chosen because at the time of writing these were the latest versions. Both Operating Systems are specific to desktop PC and laptops so that it mimics as much as possible a scenario where a crime is committed, and Digital Forensic techniques are deployed to extract data artefacts from such devices. Other devices, media, and operating systems may be used in such investigations, but for the scope of this thesis, only data artefacts from personal computers are in scope. The system requirements for both operating systems are found in table 3 below:

Table 3 System Requirements.

Operating system	Windows 11 version 23H2	Ubuntu 23.10
Processor	1GHz + with 2 or more cores	2GHz + with 2 or more cores
System Memory	4GB+	4GB+
Hard Drive Space	64GB+	25GB+

As mentioned in the previous section, the machine can operate in different states, and this different state can be found during the Digital Investigation. By making use of the states in scope that are Running and Restarted, a holistic overview of what a Forensic Investigator will be faced with is captured. In addition to these states, there is a possibility that the

browser is either open or closed when the machine is found in the state. Table 4 below illustrates the different machine states which will be denoted by “SX” where “X” is a number between 1 and 2. This will facilitate the process of analysing the data artefacts extracted during each test.

Table 4 Machine States Explanation.

State Donation	Machine State Summary
S1	The machine is running/on
S2	The machine has been restarted and is now running

The combination of these variables produced a total of 32 experiments to be investigated. A high-level diagram of these experiments of the Windows based operating system is found in figure 8 below:

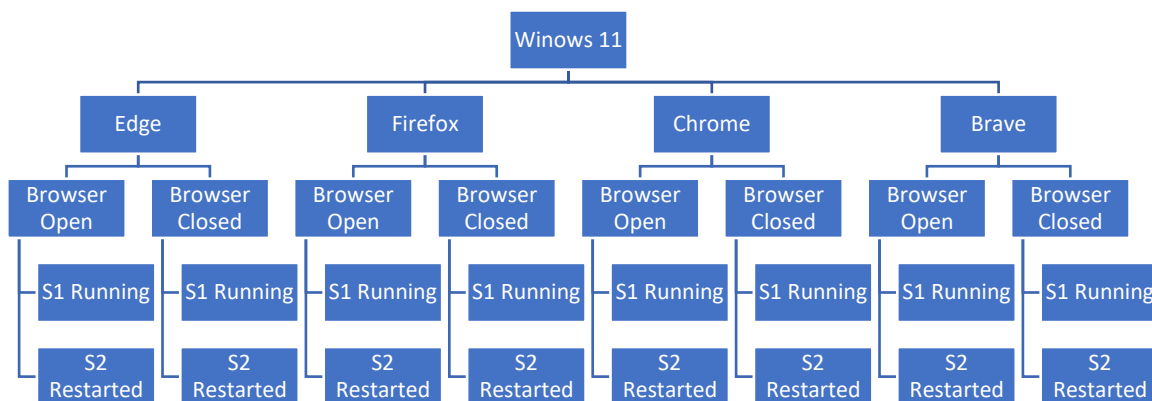


Figure 8 Windows 11 Experiment Overview.

A high-level diagram of these experiments for the Linux-based operating system is found in figure 9 below:

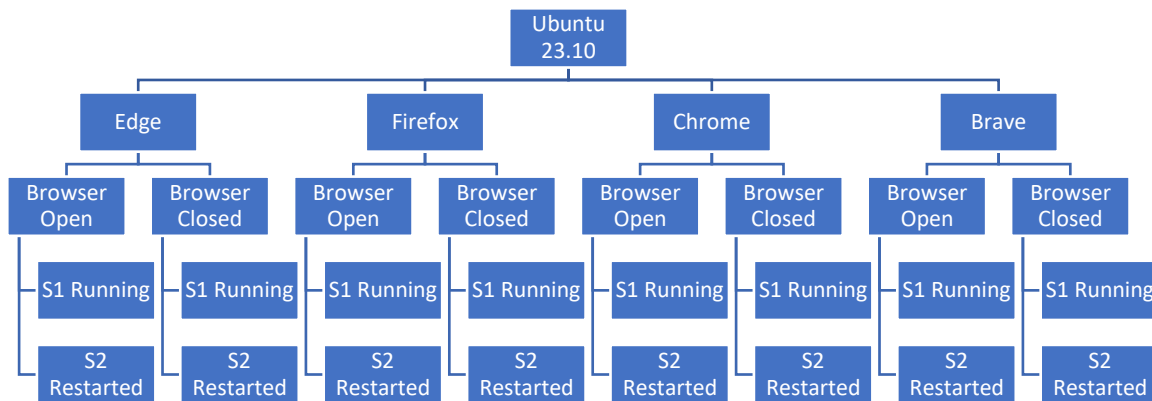


Figure 9 Ubuntu 23.10 Experiment Overview.

3.2 Experiment Explanation

In this Subsection an explanation of the different tests will be provided to further understand the scope and the dynamics of the experiments. Similarly to the states, the experiments will be denoted with an 'EX', where 'X' is a variable between 1 and 32. This will facilitate the process of data analysis. The list below gives a summary of the explanation of each test, the web browsing activity will be described in greater detail in the next subsection. To avoid repetition, experiments E1 to E16 were conducted using Windows 11 version 23H2 as the operating system. On the other hand, experiments E17 to E32 were conducted using Ubuntu 23.10 as the operating system. After the web browsing activity had been conducted, a forensic image of both the RAM and the HDD was taken to be examined.

- E1 – Microsoft Edge running in 'InPrivate mode' was used, the browser was left open and the machine was in state S1.
- E2 – Mozilla Firefox operating in private mode was used, the browser was still open, and the machine was in the S1 state.
- E3 – Google Chrome in Incognito mode was used, the browser was still open, and the machine was in S1 state.
- E4 – Brave Browser running in private mode was used, the browser was still open, and the machine was in the S1 state.

- E5 – Microsoft Edge running in 'InPrivate mode' was used, the browser was closed, and the machine was in state S1.
- E6 – Mozilla Firefox operating in private mode was used, the browser was closed, and the machine was in the S1 state.
- E7 – Google Chrome in Incognito mode was used, the browser was closed and the machine was in the S1 state.
- E8 – Brave Browser running in private mode was used, the browser was closed, and the machine was in S1 state.
- E9 – Microsoft Edge running in 'InPrivate mode' was used, the browser was left open and the machine was in state S2.
- E10 – Mozilla Firefox operating in private mode was used, the browser was still open, and the machine was in the S2 state.
- E11 – Google Chrome in Incognito mode was used, the browser was still open, and the machine was in S2 state.
- E12 – Brave Browser running in private mode was used, the browser was still open, and the machine was in the S2 state.
- E13 – Microsoft Edge running in 'InPrivate mode' was used, the browser was closed, and the machine was in state S2.
- E14 – Mozilla Firefox operating in private mode was used, the browser was closed, and the machine was in the S2 state.
- E15 – Google Chrome in Incognito mode was used, the browser was closed and the machine was in the S2 state.
- E16 – Brave Browser running in private mode was used, the browser was closed, and the machine was in S2 state.
- E17 – Microsoft Edge running in 'InPrivate mode' was used, the browser was left open and the machine was in state S1.
- E18 – Mozilla Firefox operating in private mode was used, the browser was still open, and the machine was in the S1 state.
- E19 – Google Chrome in Incognito mode was used, the browser was still open, and the machine was in S1 state.

- E20 – Brave Browser running in private mode was used, the browser was still open, and the machine was in the S1 state.
- E21 – Microsoft Edge running in 'InPrivate mode' was used, the browser was closed, and the machine was in state S1.
- E22 – Mozilla Firefox operating in private mode was used, the browser was closed, and the machine was in the S1 state.
- E23 – Google Chrome in Incognito mode was used, the browser was closed and the machine was in the S1 state.
- E24 – Brave Browser running in private mode was used, the browser was closed, and the machine was in S1 state.
- E25 – Microsoft Edge running in 'InPrivate mode' was used, the browser was left open and the machine was in state S2.
- E26 – Mozilla Firefox operating in private mode was used, the browser was still open, and the machine was in the S2 state.
- E27 – Google Chrome in Incognito mode was used, the browser was still open, and the machine was in S2 state.
- E28 – Brave Browser running in private mode was used, the browser was still open, and the machine was in the S2 state.
- E29 – Microsoft Edge running in 'InPrivate mode' was used, the browser was closed, and the machine was in state S2.
- E30 – Mozilla Firefox operating in private mode was used, the browser was closed, and the machine was in the S2 state.
- E31 – Google Chrome in Incognito mode was used, the browser was closed and the machine was in the S2 state.
- E32 – Brave Browser running in private mode was used, the browser was closed, and the machine was in S2 state.

3.3 Software and tools

In this section an overview of the software and the tools that were used to conduct the experiments will be described. Table 5 illustrates a list of software and tools that were used.

Table 5 Tools used Overview.

Software Tool	Overview
Main Workstation	Physical hardware
VMware Workstation	Hypervisor
Windows 11 version 23H2	Windows-Based OS
Ubuntu 23.10	Linux-Based OS
Microsoft Edge	Private Web Browser
Mozilla Firefox	Private Web Browser
Google Chrome	Private Web Browser
Brave Browser	Private Web Browser
Access Data FTK Imager	HDD and RAM Imager
Volatility	RAM Forensics
Autopsy	HDD Forensics

Main Workstation - This is the workstation that hosts the virtual machines that will be used to conduct the experiments. The processor was a 12th Gen Intel® Core™ i7-12700K at 3.6GHz, along with 32GB of RAM, and 2TB of storage. The operating system was Windows 11 PRO version 23H2, which was a 64-bit operating system.

VMware Workstation 17 Pro, version 17.0.0 was the hypervisor that was used to host the virtual machines to be examined. This allowed multiple machines to run on the main workstation, which enabled the functionality to simulate both Linux and Windows OS on the same physical hardware. The resources were easily allocated to each virtual machine to meet the minimum system requirements. To maintain the experiments as unbiased as possible, the same resources were allocated to each operating system, which were 2 processor cores, 6GB of RAM, and 80GB of storage. These met the minimum system requirements, enabled smooth operation of web browsing sessions, and decreased noise data. A function that VMware workstation offers is the ability to make use of snapshots, which are used to restore the machine at a particular point in time. A snapshot after the clean installation was taken on the machines, which served as a baseline for the experiments. After the experiment has been conducted, a snapshot was used to revert to the clean installation and thus the next experiment would commence.

As mentioned above, the operating systems that were installed on the virtual machines were Windows 11 version 23H2 and Ubuntu 23.10. Microsoft Edge, Mozilla Firefox, Google Chrome, and Brave Browser were the private Web browsers chosen for these experiments. These were installed individually on a clean install of the OS, and once the experiment was conducted, the machine was reverted to ensure that no cross contamination was present.

The Access Data FTK Imager was used to extract both the HDD image and the RAM capture to be examined after the experiment was conducted. This was used for both Windows and Linux based operating system, which ensured that both OS were examined on the same level. Autopsy and volatility were the tools used to examine the actual image and extract digital data artefacts that the private web browser left after a session was performed.

3.4 Browsing Session Activity

This sub-section of the thesis will explain what web browsing activities were conducted during the experiment to understand what data artefacts were in scope during the forensic investigation. During browsing sessions, any prompts to remember the information entered, cookies, and data autofill were saved or agreed to. The default search engine was used on all four different browsers and no further privacy and security features were used. All the web pages visited were also saved into the bookmarks section after they were found through the search engine via the queries. The scope was to simulate daily activities that could be performed by a user, table 6 is displayed below to highlight the search queries along with the website that was browsed.

Table 6 Search Query results.

Search Query	Website
LTU	https://www.ltu.se/en
University admission Sweden	https://www.universityadmissions.se/en/
Canvas LTU se	https://weblogon.ltu.se/
Maltapark	https://www.maltapark.com/
Armed Forces of Malta	https://afm.gov.mt/en/Pages/Default.aspx
Times of Malta	https://timesofmalta.com/
Facebook	https://www.facebook.com/
Youtube	https://www.youtube.com/

Gmail	https://mail.google.com/
Google Drive	https://www.google.com/drive/
Ecourts Malta	https://ecourts.gov.mt/onlineservices/

For testing purposes, an email account “brandonthesis24@gmail.com” was created and will be used as a base for all other accounts as will be described below.

LTU – once on the main web page, the drop-down menu was used to navigate to 'education' and then to 'Our programmes'. The section 'Master programmes' was visited and the string 'Information Security' was searched, resulting in 1 hit. The “Master Programme in Information Security” was visited and the “Programmee syllabus” which enabled the syllabus to be displayed and also the option to download the “PDF” version, which was clicked, and the file was downloaded to the local disk.

University admission Sweden – an account was created for testing purposes and was used to login during all the 32 experiments. The drop-down menu was clicked, and the “search for courses” tab was used to navigate through the available courses. The My pages section was later visited and the Profile section, which holds information such as email address, mobile phone, and address, was displayed.

Canvas LTU se – this web page required a personal login and the student account provided was used. Once logged in, the account, dashboard, courses, groups, calendar, inbox, and history were visited. In the courses section, the course 'MSc Information Security Programmeme' was visited and navigation through assignments, grades, people and files was carried out. In the inbox section, a new message was composed and sent to 'Brandon Spiteri'.

Maltapark – two accounts were needed to conduct this stage of the experiment, one was a personal account, while the other was the newly created account. A car listing was added by making use of the personal account, and the testing account was used to contact the seller. Similarly, on the testing account, a new listing was added to sell a personal laptop, after all the details were filled, the listing was published.

Armed Forces of Malta – once on the website, “Force Structure” was the first area visited which led us to the organisation of the force. Operations and equipment and deployments were visited that included information about national operations, overseas operations, different weapon systems, and vehicles. Finally, the search bar was used with the query 'computer', which returned two results.

Times of Malta - The following areas were visited, latest, national, world, opinion, sport, motoring, business, and community. The search bar was also used to search for the query 'AFM'.

Facebook – the testing account was used to conduct the various actions on the platform. These included sending a friend request to a personal account, searching for 'cyber security', liking the page 'National Cybersecurity Coordination Centre MT', liking a post on said page. In addition, a message was sent to the personal account that included a text and an image named 'fbtest.jpg'.

Youtube – the testing account logged into the platform and used the search bar to look for “digital forensics”. A video called 'Digital Forensics Overview' by ISACA was returned and this was watched and added to a playlist called 'experiment playlist'.

Gmail – an email with the subject “secret message” was sent from the personal account that included text and a PDF file called “secret.pdf”. The testing account was used to replay the email by adding another PDF called 'reply.pdf'.

Google Drive – the testing account logged in and uploaded 3 files, 'laptop.jpg', 'fbtest.jpg' and 'reply.pdf'. A new Google Doc was created that was named 'experminetX'.

Ecourts Malta - When visiting the site, the language was first changed to English, after which various tabs such as civil cases, judgments, judicial sales, and services were visited.

According to the judgments, there was a possibility to search for results and a fake reference number of 8080 was used. Due to ethical considerations, in the parties section, the name 'DigitalForensicThesis' was used to ensure that no judgments match the query.

Chapter 4 – Findings

In this chapter, a result of the experiments described in the previous section will be explained and analysed. These are categorised into two sub-section which are live acquisition (Volatile Data) and dead acquisition (Non-Volatile data). Section 4.1 will describe dead acquisition, while Section 4.2 will focus on live acquisition. Due to having a total of 32 experiments, it was ideal to represent the findings by making use of tables, which are limited to the private web browser, operating system, and state, as described in the experiment details. Thus, a table was generated for each operating system and private web browser, and each table included the results of eight experiments in terms of live or dead acquisition. Figure 10 illustrates the differences between the data acquisition.

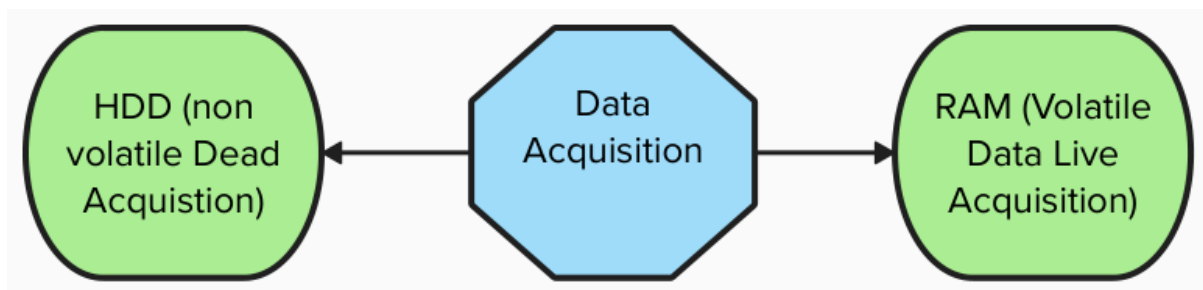


Figure 10 Data Acquisition Scope.

4.1 Dead acquisition

In terms of non-volatile data, the variables which will affect which data artefacts are the Operating System, and the Private Web Browser. This meant that eight tests were sufficient to determine any data artefacts after the web browsing session was conducted. A table to highlight the difference between private web browsers is shown in table 7 below. The symbol “✓” denotes that the residual data artefact was found in the experiment while “✗” denotes that the residual data artefact was not found. In terms of labelling, the first letter of the operating system was used, which is 'W' (Windows) or 'U' (Ubuntu). The second letter determines the private web browser, 'E' for Microsoft Edge, 'F' for Mozilla Firefox, 'C' for Google Chrome and 'B' for Brave Browser. Any residual data that was left by the browsers before running in private mode was not included as it was not in the scope of this thesis.

Table 7 Dead Acquisition Results.

Residual Data Artefact	W-E	U-E	W-F	U-F	W-C	U-C	W-B	U-B
Browsing History	✗	✗	✓	✓	✓	✓	✓	✓
Cookies	✗	✗	✗	✗	✗	✗	✗	✗
Bookmarks	✓	✓	✓	✓	✓	✓	✓	✓
Download history	✓	✓	✗	✗	✓	✗	✓	✗
Browser Extensions and Add-ons	✓	✓	✗	✗	✗	✗	✗	✗

W-E: The first investigation was conducted on Windows based operating system and Microsoft Edge. It resulted that Browser extensions, Download History, and bookmarks were still retrieved even when using private mode. By default, the operating system will store data in the following location:

“C:\Users\%USERNAME%\AppData\Local\Microsoft\Edge\UserData\Default\”

During the methodology section, we investigated 11 websites, and for each website, a bookmark was created. During the investigation, 12 bookmarks were recovered as seen in figure 11. By default, Internet Explorer generates a bookmark titled 'Bing.url', with this addition, the total is 12.

Source Name	S	C	O	URL	Title	Date Created	Program Name	Domain
Bookmarks			0	https://www.ltu.se/en	Luleå University of Technology Luleå tekniska universitet	2024-03-30 15:41:26 CET	Microsoft Edge	ltu.se
Bookmarks			0	https://www.universityadmissions.se/riti/start	Universityadmissions.se - Apply to Swedish universities	2024-03-30 12:31:10 CET	Microsoft Edge	universityadmissions.se
Bookmarks			0	https://webblog.ltu.se/cas/login?service=https%3A%2F...	Luleå tekniska universitet - Inloggning	2024-03-30 12:31:33 CET	Microsoft Edge	ltu.se
Bookmarks			0	https://www.makapark.com/	Home Makapark	2024-03-30 12:34:44 CET	Microsoft Edge	makapark.com
Bookmarks			1	https://afm.gov.mt/en/Pages/Default.aspx	Armed Forces of Malta	2024-03-30 12:40:46 CET	Microsoft Edge	gov.mt
Bookmarks			0	https://timesofmalta.com/	News, sport and opinion from Malta and the world @E"Time...	2024-03-30 12:43:22 CET	Microsoft Edge	timesofmalta.com
Bookmarks			0	https://www.facebook.com/	Facebook - log in or sign up	2024-03-30 12:44:23 CET	Microsoft Edge	facebook.com
Bookmarks			0	https://www.youtube.com/?app	YouTube	2024-03-30 12:46:52 CET	Microsoft Edge	youtube.com
Bookmarks			1	https://mail.google.com/mail/u/0/?ui=en#inbox	Inbox (6) - brandonhies24@gmail.com - Gmail	2024-03-30 12:52:35 CET	Microsoft Edge	google.com
Bookmarks			1	https://drive.google.com/drive/home	Home - Google Drive	2024-03-30 12:52:45 CET	Microsoft Edge	google.com
Bookmarks			1	https://ecourts.gov.mt/online/services/Judgements	Fitxex is-Senżenzi BĊ" eCourts.gov.mt	2024-03-30 12:54:20 CET	Microsoft Edge	gov.mt
Bing url			2	http://go.microsoft.com/fwlink/?LinkId=255142	Bing url	2024-03-30 12:22:54 CET	Internet Explorer Analyser	microsoft.com

Figure 11 Bookmarks Extracted from Microsoft Edge Running on Windows OS.

The bookmarks also indirectly gave us other relevant information such as the URL, the date created, and the domain, Figure 12 illustrates these findings for the 'Gmail' bookmark.

Bookmark Details	
Title:	Inbox (6) - brandonthesis24@gmail.com - Gmail
Date Created:	2024-03-30 12:52:35 CET
Domain:	google.com
URL:	https://mail.google.com/mail/u/0/?hl=en#inbox
Program Name:	Microsoft Edge
Other	
Comment:	bookmark_bar
Username:	Default
Source	
Host:	E1 Dead Acquisition.001_1 Host
Data Source:	E1 Dead Acquisition.001
File:	/img_E1 Dead Acquisition.001/vol_vol6/Users/brand/AppData/Local/Microsoft/Edge/User Data/Default/Bookmarks

Figure 12 Gmail bookmark.

The download history was found since the files will still need to be stored on a disk irrelevant if private or normal web browsing is used. Figure 13 illustrates the file downloaded from LTU, named "Syllabus_FMISA.pdf". The domain and URL were not available, but it still confirms that the file was downloaded from the Internet, which can help to trace the sequence of events the user has conducted.

Downloaded File	
Domain:	
URL:	about:internet
Path:	/Users/brand/Downloads/Syllabus_FMISA.pdf
Program Name:	
Other	
Comment:	Internet Zone
Path ID:	16389
Source	
Host:	E1 Dead Acquisition.001_1 Host
Data Source:	E1 Dead Acquisition.001
File:	/img_E1 Dead Acquisition.001/vol_vol6/Users/brand/Downloads/Syllabus_FMISA.pdf:Zone.Identifier

Figure 13 Web download file "Syllabus_FMISA.pdf".

U-E: the second drive to be analysed was using Ubuntu and Microsoft Edge as variables, which resulted that Browser extensions, Download History, and bookmarks were still retrieved even when using private mode. These were the same as results when compared to a Windows operating system running Microsoft Edge; however, the location where these data were stored is different. In Ubuntu, the default location was "/home/%Username%/.config/Microsoft-edge/Default/". Figure 14 illustrates a snapshot of the retrieved bookmarks.

```

bookmarks {
  "checksum": "df7a338cf4bbd3920577db936483d947",
  "roots": {
    "bookmark_bar": {
      "children": [ {
        "date_added": "13356544640336747",
        "date_last_used": "0",
        "guid": "65177ee6-2227-4841-8f81-265963aedfb9",
        "id": "7",
        "name": "Luleå University of Technology | Luleå tekniska universitet",
        "show_icon": false,
        "source": "unknown",
        "type": "url",
        "url": "https://www.ltu.se/en"
      }, {
        "date_added": "13356544737110941",
        "date_last_used": "0",
        "guid": "43e888b9-9c96-4c02-81df-051536684349",
        "id": "8",
        "name": "My applications - Universityadmissions.se",
        "show_icon": false,
        "source": "unknown",
        "type": "url",
        "url": "https://www.universityadmissions.se/intl/mypages"
      }, {

```

Figure 14 Bookmarks Extracted from Ubuntu Running Microsoft Edge.

W-F: During the digital forensic investigation of Mozilla Firefox running on Windows Operating System, several artefacts were still retrieved even when the browser is operated in Private mode. The bookmarks and browsing history were extracted as shown by Figure 15. The location of the web browsing history and bookmarks is found in “/Users/%Username%/AppData/Roaming/Mozilla/Firefox/Profiles/txqndzw7.default-release/places.sqlite”.

Source Name	S	C	O	URL	Title	Date Created	Program Name	Domain	Data Source
places.sqlite			3	https://www.mozilla.org/about/	About Us	2024-04-02 12:01:33 CEST	Firefox Analyzer	mozilla.org	WindowsFirefo
places.sqlite			3	https://www.mozilla.org/firefox/?utm_medium=firefox-des...	Getting Started	2024-04-02 12:01:35 CEST	Firefox Analyzer	mozilla.org	WindowsFirefo
places.sqlite			2	https://www.maltapark.com/login/success	Login Maltapark	2024-04-02 12:03:13 CEST	Firefox Analyzer	maltapark.com	WindowsFirefo
places.sqlite			2	https://www.ltu.se/en	Luleå University of Technology Luleå tekniska universitet	2024-04-02 12:05:06 CEST	Firefox Analyzer	ltu.se	WindowsFirefo
places.sqlite			2	https://www.universityadmissions.se/intl/mypages	My applications - Universityadmissions.se	2024-04-02 12:07:09 CEST	Firefox Analyzer	universityadmissions.se	WindowsFirefo
places.sqlite			2	https://canvas.ltu.se/	Dashboard	2024-04-02 12:10:42 CEST	Firefox Analyzer	ltu.se	WindowsFirefo
places.sqlite			3	https://afm.gov.mt/en/Pages/Default.aspx	Armed Forces of Malta	2024-04-02 12:11:19 CEST	Firefox Analyzer	gov.mt	WindowsFirefo
places.sqlite			2	https://timesofmalta.com/	News, sport and opinion from Malta and the world - Times ...	2024-04-02 12:12:54 CEST	Firefox Analyzer	timesofmalta.com	WindowsFirefo
places.sqlite			2	https://www.facebook.com/?sk=welcome	(1) Facebook	2024-04-02 12:14:45 CEST	Firefox Analyzer	facebook.com	WindowsFirefo
places.sqlite			2	https://www.youtube.com/	YouTube	2024-04-02 12:17:36 CEST	Firefox Analyzer	youtube.com	WindowsFirefo
places.sqlite			4	https://mail.google.com/mail/u/0/#inbox	Inbox (7) - brandonthesis24@gmail.com - Gmail	2024-04-02 12:19:13 CEST	Firefox Analyzer	google.com	WindowsFirefo
places.sqlite			4	https://drive.google.com/drive/home	Home - Google Drive	2024-04-02 12:20:25 CEST	Firefox Analyzer	google.com	WindowsFirefo
places.sqlite			3	https://ecourts.gov.mt/onlineservices	Sign In - eCourts.gov.mt	2024-04-02 12:23:23 CEST	Firefox Analyzer	gov.mt	WindowsFirefo

Figure 15 Bookmarks extracted from Windows OS and Firefox.

The web browsing history gives valuable information, which are the “host” and the “frequency” as shown in figure 16. These combinations are valuable insights since it can be deduced the amount of time the domain has been visited.

id	prefix	host	freccy	recalc_fr...	alt_frece...	recalc_al...
1	https://	www.mozilla.org	545	0		1
2	https://	support.mozilla.org	280	0		1
3	https://	www.maltapark.com	140	0		1
4	https://	www.ltu.se	140	0		1
5	https://	www.universityadmissions.se	140	0		1
6	https://	canvas.ltu.se	140	0		1
7	https://	afm.gov.mt	140	0		1
8	https://	timesofmalta.com	140	0		1
9	https://	www.facebook.com	140	0		1
10	https://	www.youtube.com	140	0		1
11	https://	mail.google.com	140	0		1
12	https://	drive.google.com	140	0		1
13	https://	ecourts.gov.mt	140	0		1

Figure 16 Web History in Windows and Firefox OS.

U-F: During the investigation, which made use of the Ubuntu operating system and Mozilla Firefox as variables, the web browsing history and bookmarks were also extracted. The location of these data was

“/home/%Username%/snap/firefox/common/.mozilla/firefox/ztz1c3be.default/places.sqlite”.

Figure 17 illustrates the browsing session, while Figure 18 illustrates the bookmarks.

id	url	title	rev_host	visit_count	hidden	typed	freccy
6	https://www.mozilla.org/about/		gro.allzom.www.	0	0	0	140
7	https://www.mozilla.org/firefox/?utm_medium=firefox-desktop&utm_source=bookmarks-toolbar&utm_campaign=firefox-desktop-bookmarks		gro.allzom.www.	0	0	0	140
8	https://www.ltu.se/en		es.utl.www.	0	0	0	140
9	https://www.universityadmissions.se/intl/mypages		es.snoissindaytisrevinu.www.	0	0	0	140
10	https://canvas.ltu.se/		es.utl.savnac.	0	0	0	140
11	https://www.maltapark.com/login/success		moc.krapatlam.www.	0	0	0	140
12	https://afm.gov.mt/en/Pages/Default.aspx		tm.vog.mfa.	0	0	0	140
13	https://timesofmalta.com/		moc.atlamfosemit.	0	0	0	140
14	https://www.facebook.com/?sk=welcome		moc.koobecaf.www.	0	0	0	140
15	https://www.youtube.com/		moc.ebutuoy.www.	0	0	0	140
16	https://mail.google.com/mail/u/0/#inbox		moc.elgoog.llam.	0	0	0	140

Figure 17 Web browsing history extracted from Ubuntu and Firefox.

Source Name	S	C	URL	Title	Date Created	Program Name	Domain	Data Source
places.sqlite		3	https://www.ltu.se/en	Luleå University of Technology Luleå tekniska universitet	2024-04-02 15:12:14 CEST	Firefox Analyzer	ltu.se	sdaUF_image
places.sqlite		3	https://www.universityadmissions.se/intl/mypages	My applications - Universityadmissions.se	2024-04-02 15:15:37 CEST	Firefox Analyzer	universityadmissions.se	sdaUF_image
places.sqlite		3	https://canvas.ltu.se/	Dashboard	2024-04-02 15:19:14 CEST	Firefox Analyzer	ltu.se	sdaUF_image
places.sqlite		3	https://www.maltapark.com/login/success	Login Maltapark	2024-04-02 15:21:58 CEST	Firefox Analyzer	maltapark.com	sdaUF_image
places.sqlite		3	https://timesofmalta.com/	News, sport and opinion from Malta and the world - Times of Malta	2024-04-02 15:26:11 CEST	Firefox Analyzer	timesofmalta.com	sdaUF_image
places.sqlite		3	https://www.facebook.com/?sk=welcome	(1) Facebook	2024-04-02 15:28:15 CEST	Firefox Analyzer	facebook.com	sdaUF_image
places.sqlite		3	https://www.youtube.com/	YouTube	2024-04-02 15:30:31 CEST	Firefox Analyzer	youtube.com	sdaUF_image
places.sqlite		4	https://support.mozilla.org/products/firefox	Get Help	2024-04-02 15:10:55 CEST	Firefox Analyzer	mozilla.org	sdaUF_image
places.sqlite		4	https://support.mozilla.org/kb/customize-firefox-controls-but-not-look-like-firefox	Customize Firefox	2024-04-02 15:10:55 CEST	Firefox Analyzer	mozilla.org	sdaUF_image
places.sqlite		4	https://www.mozilla.org/contribute/	Get Involved	2024-04-02 15:10:55 CEST	Firefox Analyzer	mozilla.org	sdaUF_image
places.sqlite		4	https://www.mozilla.org/about/	About Us	2024-04-02 15:10:55 CEST	Firefox Analyzer	mozilla.org	sdaUF_image
places.sqlite		4	https://www.mozilla.org/firefox/?utm_medium=firefox-desktop&utm_source=bookmarks-toolbar&utm_campaign=firefox-desktop-bookmarks	Getting Started	2024-04-02 15:10:56 CEST	Firefox Analyzer	mozilla.org	sdaUF_image
places.sqlite		4	https://afm.gov.mt/en/Pages/Default.aspx	Armed Forces of Malta	2024-04-02 15:24:42 CEST	Firefox Analyzer	gov.mt	sdaUF_image
places.sqlite		4	https://ecourts.gov.mt/online-services/	Sign In - eCourts.gov.mt	2024-04-02 15:33:36 CEST	Firefox Analyzer	gov.mt	sdaUF_image

Figure 18 Bookmarks extracted from Ubuntu and Firefox.

W-C: The combination of Windows operating system and Google Chrome web browser returned that browsing history, bookmarks, and web downloads were retrieved. The location was “/Users/%Username%/AppData/Local/Google/Chrome/User Data/Default/”,

and the directories were “bookmarks” and “History”. Figure 19 illustrates the “Youtube” bookmark, which gives us additional information such as the date and time created and accessed.

Hex	Text	Application	Source File Metadata	OS Account	Data Artifacts	Analysis Results	Context	Annotations	Other Occurrences
Metadata									
Name:	/img_Windows Chrome.001/vol_vol6/Users/brand/AppData/Local/Google/Chrome/User Data/Default/Bookmarks								
Type:	File System								
MIME Type:	text/plain								
Size:	5528								
File Name Allocation:	Allocated								
Metadata Allocation:	Allocated								
Modified:	2024-03-30 13:09:46 CET								
Accessed:	2024-03-30 13:09:46 CET								
Created:	2024-03-30 12:31:59 CET								
Changed:	2024-03-30 13:09:46 CET								
MD5:	a639b983760cef66b184da88ad3ef815								
SHA-256:	a9cc49944371eab6454ce7051be0f80cc7992d93f25a9f1dc689a835967fe095								
Hash Lookup Results:	UNKNOWN								
Internal ID:	3911								

Figure 19 YouTube bookmark extracted from Windows OS and Google Chrome.

Figure 20 illustrates the history of the browsing session found in the directory '/History'.

/img_Windows Chrome.001/vol_vol6/Users/brand/AppData/Local/Google/Chrome/User Data/Default/History									
Hex	Text	Application	File Metadata	OS Account	Data Artifacts	Analysis Results	Context	Annotations	Other Occurrences
Table Urls 11 entries Page 1 of 1 Export to CSV									
id	url	title	visit_count	typed_c...	last_visit_time	hidden			
1	https://www.ltu.se/en	Luleå University of Technology Luleå tekniska universitet	0	0	13356271916603416	1			
2	https://www.universityadmissions.se/int/start	Universityadmissions.se - Apply to Swedish universities	0	0	13356272029135974	1			
3	https://canvas.ltu.se/	Dashboard	0	0	13356272396886314	1			
4	https://maltapark.com/	Home Maltapark	0	0	13356272523410314	1			
5	https://afm.gov.mt/en/Pages/Default.aspx	Armed Forces of Malta	0	0	13356272874208694	1			
6	https://timesofmalta.com/	News, sport and opinion from Malta and the world – Times of Malta	0	0	13356273012335848	1			
7	https://www.facebook.com/	(1) Facebook	0	0	13356273498943030	1			
8	https://www.youtube.com/	YouTube	0	0	1335627363777977	1			
9	https://mail.google.com/mail/u/0/#inbox	Inbox (7) - brandonthesis24@gmail.com - Gmail	0	0	13356273825568698	1			
10	https://drive.google.com/drive/home	Home - Google Drive	0	0	13356273934254514	1			
11	https://ecourts.gov.mt/onlineservices	Sign In - eCourts.gov.mt	0	0	13356274182682905	1			

Figure 20 History extraction from Windows OS and Google Chrome.

The web downloads section gives information on which file was downloaded from the internet, a comment is added that says 'Internet Zone', but the URL is still missing as shown below in figure 21.

Source Name	S	C	O	Path	URL	Domain	Program Name	Comment	Data Source
Syllabus_FMISA.pdf;Zone.Identifier				/Users/brand/Downloads/Syllabus_FMISA.pdf	about:internet			Internet Zone	Windows Chrome.001

Figure 21 Web download from Windows OS and Google Chrome.

U-C: Ubuntu running Google Chrome resulted that the bookmarks and the web history were extractable but a direct reference to the downloaded files was not made. This data was

discovered under “home/%Username%/.config/google-chrome/Default/”. Figure 22 illustrates part of the bookmarks extracted, while Figure 23 illustrates the web history.

```

/img_sdaUC.image/vol_vol5/home/thesis/.config/google-chrome/Default/Bookmarks
Hex Text Application File Metadata OS Account Data Artifacts Analysis Results Context Annotations Other Occurrences
Strings Indexed Text Translation
Page: 1 of 1 Page Go to Page:
+++++
name: "Inbox (6) - brandonthesis24@gmail.com - Gmail",
  "type": "url",
+++++
url: "https://mail.google.com/mail/u/0/#inbox"
}, {
  "date_added": "13356540827437513",
+++++
date_last_used": "0",
  "guid": "fa2a58fe-e9b6-497b-b6ff-12efca1dde92",
  "id": "14",
  "meta_info": {
+++++
    "power_bookmark_meta": ""
  },
+++++
name: "Home - Google Drive",
  "type": "url",
+++++
url: "https://drive.google.com/drive/home"
}, {
  "date_added": "13356540856008521",
+++++
date_last_used": "0",
  "guid": "acc3e56f-6d30-4866-ad8f-cf18eb77dc16",
  "id": "15",
  "meta_info": {
+++++

```

Figure 22 Bookmarks Extraction from Ubuntu and Google Chrome.

id	url	title
1	https://www.ltu.se/en	Luleå University of Technology
2	https://www.universityadmissions.se/intl/mypages	My applications - University of Applied Sciences
3	https://canvas.ltu.se/	Dashboard
4	https://www.maltapark.com/login/success	Login Maltapark
5	https://afm.gov.mt/en/Pages/Default.aspx	Armed Forces of Malta
6	https://timesofmalta.com/	News, sport and entertainment
7	https://www.facebook.com/?sk=welcome	(1) Facebook
8	https://www.youtube.com/	YouTube
9	https://mail.google.com/mail/u/0/#inbox	Inbox (6) - brandonthesis24@gmail.com
10	https://drive.google.com/drive/home	Home - Google Drive
11	https://ecourts.gov.mt/online-services/	Sign In - eCourts

Figure 23 Web History extraction from Ubuntu and Google Chrome.

W-B: The results of the variables which were composed of Windows based operating system and Brave browser were that the bookmarks, download history and browsing history were retrieved. These data artifacts were located in “/Users/%Username%/AppData/Local/Brave-

Browser/User Data/Default/" directory. Figure 24 illustrates the 'courts' bookmark, Figure 25 illustrates the browsing history, and Figure 26 illustrates the web download history.

Bookmark Details

Title: Sign In - eCourts.gov.mt
 Date Created: 2024-04-02 10:20:58 CEST
 Domain: gov.mt
 URL: https://ecourts.gov.mt/onlineservices
 Program Name: Brave

Other

Comment: other
 Username: Default

Source

Host: Windows Brave.001_1 Host
 Data Source: Windows Brave.001
 File: /img_Windows Brave.001/vol_vol6/Users/brand/AppData/Local/BraveSoftware/Brave-Browser/User Data/Default/Bookmarks

Figure 24 e-Courts Bookmark Extracted from Windows OS and Brave Browser.

Table: urls 11 entries Page 1 of 1 Export to CSV

id	url	title
1	https://www.ltu.se/en	Luleå University of Technology Luleå tekniska universitet
2	https://www.universityadmissions.se/intl/start	Universityadmissions.se - Apply to Swedish universities
3	https://canvas.ltu.se/	Dashboard
4	https://www.maltapark.com/	Home Maltapark
5	https://afm.gov.mt/en/Pages/Default.aspx	Armed Forces of Malta
6	https://timesofmalta.com/	News, sport and opinion from Malta and the world – Times of Malta
7	https://www.facebook.com/?sk=welcome	(1) Facebook
8	https://www.youtube.com/	YouTube

Figure 25 Web History Extracted from Windows OS and Brave Browser.

Source Name	S	C	O	Path	URL	Domain	Program Name	Comment	Data Source
Syllabus_FMISA.pdf.Zone.Identifier				/Users/brand/Downloads/Syllabus_FMISA.pdf	about:internet			Internet Zone	Windows Brave.001

Figure 26 Download History Extracted from Windows OS and Brave Browser.

U-B: Ubuntu running the Brave browser resulted in the bookmarks and browsing web history data artefacts still recoverable. The data artifacts were retrieved from the location “home/%Username%/.config/BraveSoftware/Brave-Browser/Default”. Figure 27 illustrates the web browsing history, while Figure 28 illustrates the bookmarks retrieved.

id	url	title
1	https://www.ltu.se/en	Luleå University of Technology Luleå tekniska universitet
2	https://www.universityadmissions.se/intl/mypages	My applications - Universityadmissions.se
3	https://ltu.instructure.com/	Dashboard
4	https://www.maltapark.com/login/success	Login Maltapark
5	https://afm.gov.mt/en/Pages/Default.aspx	Armed Forces of Malta
6	https://timesofmalta.com/	News, sport and opinion from Malta and the world – Times of Malta
7	https://www.facebook.com/?sk=welcome	(1) Facebook
8	https://www.youtube.com/	YouTube
9	https://mail.google.com/mail/u/0/#inbox	Inbox (6) - brandonthesis24@gmail.com - Gmail
10	https://drive.google.com/drive/home	Home - Google Drive
11	https://ecourts.gov.mt/online-services/	Sign In - eCourts.gov.mt

Figure 27 Web browsing history retrieved from Ubuntu and Brave Browser.

```

}, {
  "name": "YouTube",
  "type": "url",
  "url": "https://www.youtube.com/"
}, {
  "date_added": "13356548340136946",
  "date_last_used": "0",
  "guid": "7b0fbb73-56fa-402c-98b5-690fe617dad3",
  "id": "13",
  "meta_info": {
    "power_bookmark_meta": ""
  },
  "name": "Inbox (6) - brandonthesis24@gmail.com - Gmail",
  "type": "url",
  "url": "https://mail.google.com/mail/u/0/#inbox"
}, {

```

Figure 28 Bookmarks retrieved from Ubuntu and Brave Browser.

During the 8 digital forensic investigations, it was determined that different web browsers will store a different amount of data artefacts on disk. The major difference between the operating system is the default location from which these data are retrieved. Furthermore, when using the software 'Autopsy', it is easier to extract data from Windows based operating system than Linux. To extract data artefacts from Linux, the digital investigator

should know where the default locations of the software are located. No cookies were extracted from the four private Web browsers that were tested, although cookies were unintentionally retrieved from the Web browsers before entering private mode. Bookmarks were extracted through all the Private Web Browsers, while some browsing web history was also retrieved.

4.2 Live Acquisition

In this subsection, Random Access Memory was examined to extract digital data artefacts from volatile media. In terms of volatile data, the variables that will affect the data artefacts are the Operating System, the state of the machine, and the Private Web Browser. This meant that 32 tests were needed to determine what data artefacts can still be recovered by a digital forensic investigator after the web browsing session was conducted. To give a high-level presentation of the findings, 4 tables that are specific to the private web browser were used. Each table included eight experiments, which were running on different states but the same operating system.

The symbol “✓” denotes that the residual data artefact was found in the experiment while “✗” denotes that the residual data artefact was not found. In terms of labelling, the experiment number will be used as explained in the methodology section. This will be composed of the operating systems, the Private Web Browser which can be either open or closed, and the states.

- S1 - Running
- S2 - Restart

Any residual data that the browsers left before running in private mode or data that cannot be accurately correlated with the user were not included in these findings. Due to the number of experiments, a sample of the evidence will be visually presented throughout each private web browser and operating system. This is done to prevent repetition and keep the thesis as concise as possible, otherwise, hundreds of evidence will need to be presented. These data artefacts were extracted using tools called “volatility” and “Autopsy”.

Table 8 Microsoft Edge Experiment Results.

Residual Data Artefact	E1	E5	E9	E13	E17	E21	E25	E29
Web page content	✓	✓	✗	✓	✓	✓	✓	✓
URL's	✓	✓	✗	✓	✓	✓	✓	✓
CDN attachments	✓	✗	✗	✓	✓	✓	✓	✓
Images	✗	✗	✗	✗	✗	✗	✗	✗
Email address	✓	✗	✗	✗	✓	✓	✓	✓
Passwords	✓	✗	✗	✗	✓	✓	✗	✗
Search queries	✓	✓	✗	✓	✓	✓	✓	✗

Table 8 above denotes a high-level explanation of Microsoft Edge running on Windows 11 and in different states. The experiment which is denoted by 'E1' was the first to be investigated, resulting in the extraction of a significant amount of data artefacts, even when the private browser was used. The web page content was extracted as demonstrated in Figure 29, which is an extract of the website "ltu.se" after the site was integrated as described in the methodology section.

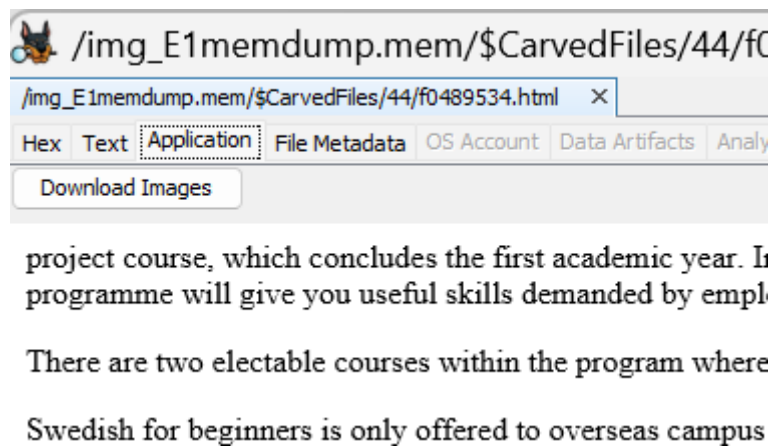


Figure 29 Web Content Extracted from Microsoft Edge Running on Windows 11.

The next artefact was 'sessions' which gave information such as what sessions interactions were made with the computer system. URLs were also extracted from the volatile memory location, which can give an indication of what sites were visited by the user. CDN attachments and search queries data artefacts were extracted, which increases the amount of artefacts found. Two important data evidences were the username and the password that the user used. These can be significantly important, as they can give access to

additional information to a digital investigator during a case. These were extracted in clear text, and no encryption was used by the private web browser.

Experiment 5 introduced a memory capture after the browser has been closed, but the machine was still running. Search queries, URLs, and page content were still recovered via “Autopsy” and the keywords “search!?q=”. Figure 30 illustrates two URLs that were retrieved from this experiment for both the website called 'Maltapark' and 'Armed Forces of Malta'. This proves that the user has interacted with the said website at a certain point in the web browsing session.

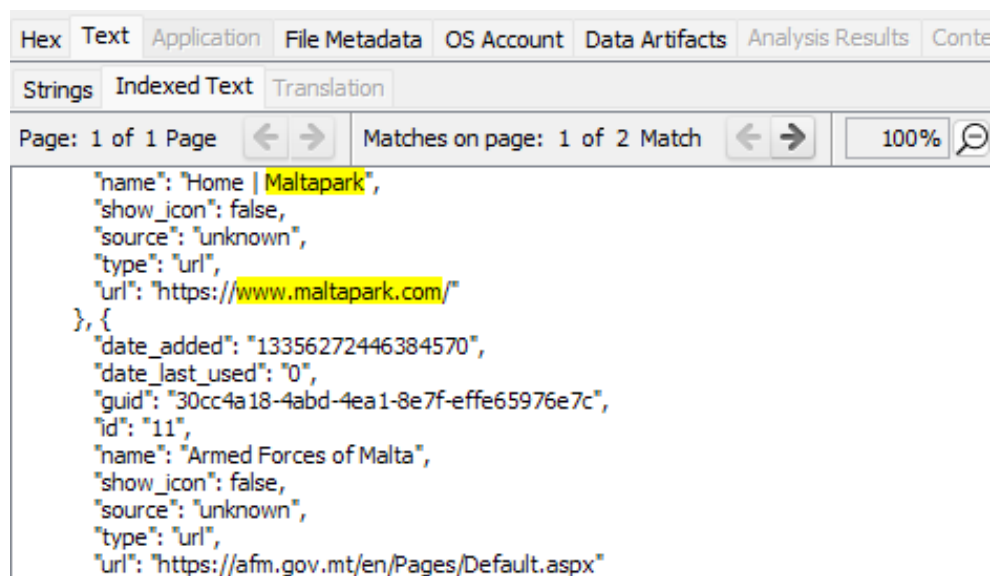


Figure 30 URL return for Experiment 5.

Experiment 9 was the first experiment that introduced a restart of the system as a variable, but the browser was left open when the restart was affected. As expected, several data were not retrieved, but the sessions and the process were still recovered. This was because the RAM is not completely wiped out yet. Some other data were also extractable, such as keywords such as “maltapark” and “ecourts”, however, they were not related to any activity and therefore an investigator could only determine that there were some data stored in RAM related to the keyword. For this reason, it was included as evidence since it has little value when presented to a court of law.

Experiment 13 was captured after the browser had been closed and the machine was restarted, surpassingly, URL’s, images, search queries, and web page content were still retrieved. This is like experiment 9, but more data was recovered because it was still not

recycled by the RAM during restart. Web pages make use of content delivery network (CDN) to speed up webpage loading, but these can be retrieved by an investigator as seen in Figure 31. The image was retrieved by the user when visiting the website called “Times Of Malta”, and once recovered by the digital investigator, it can be accessed to view the content viewed by the user. The CDN content was found under the image object along with the meta-data of the image.

ImageObject

<https://cdn-attachments.timesofmalta.com/61fa655352202cbfe81208e691744c79960f6607-1711797064-31089927-1920x1280.jpg>
Firefighters work to extinguish flames at one of the two sites. Photo: CPD

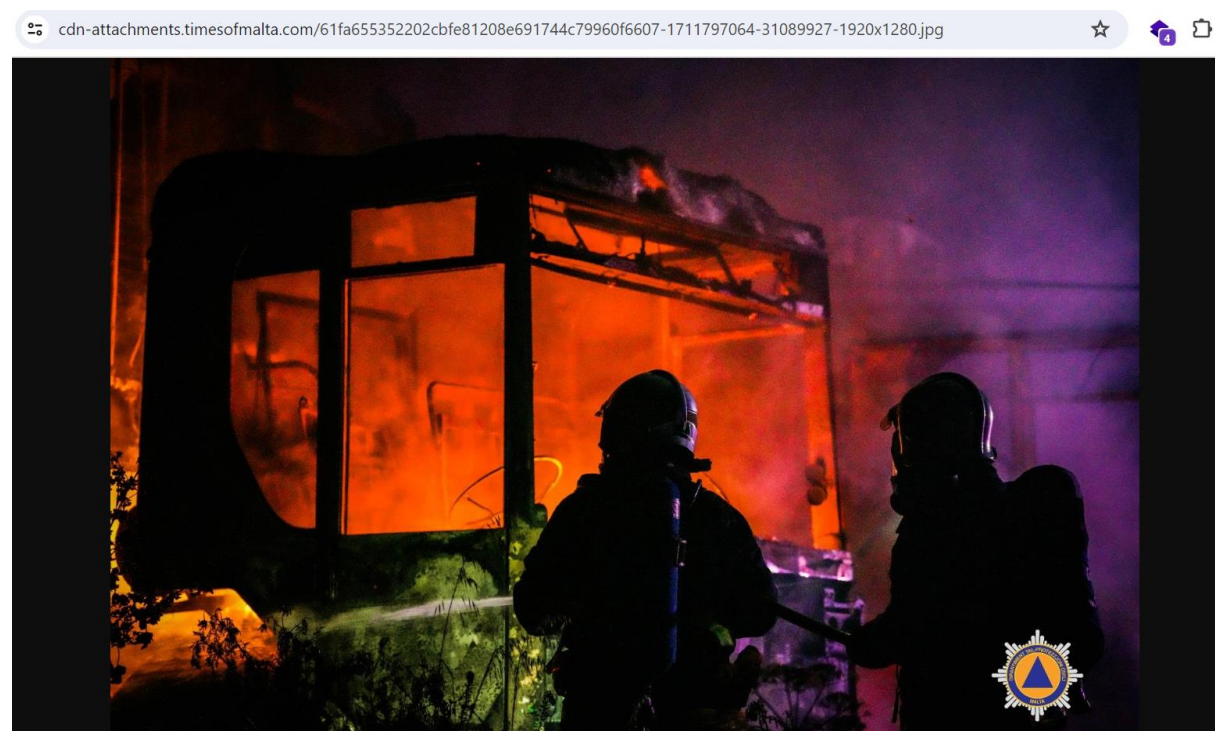


Figure 31 Image obtained from Experiment 13 CDN.

Experiments 17 and 21 were the first to introduce the Ubuntu operating system as a variable. Like E1 which is equivalent to E17 for windows, web page content, URL images, email addresses, passwords, and search queries were extracted. Experiment 21 resulted in the extraction of the same artefacts and there was no change compared to when the browser was open or closed. This means that the data were still residing in the RAM. However, E21 is compared to E5 and this results that E5, which ran on windows, retained less data once the browser was closed. Figure 32 shows part of the content on the website extracted from E17.

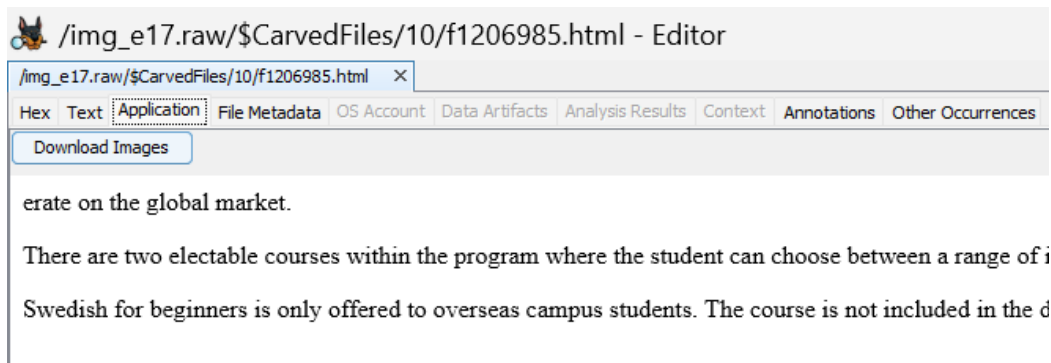


Figure 32 Web page content of E17.

Experiment 25 was conducted on the Ubuntu operating system and introduced a restart before memory acquisition. A significant amount of data artefacts were still extractable even after the restart has been affected. Web page content, URL's, images, email address, and search queries were extracted. Experiment 29 was conducted after the browser had been closed and the machine restarted. This also included a significant amount of data artefacts, with the only difference that search queries were not extracted compared to E25. Figures 33 and 34 illustrate the email addresses extracted from E25 and the URL from E29, respectively.

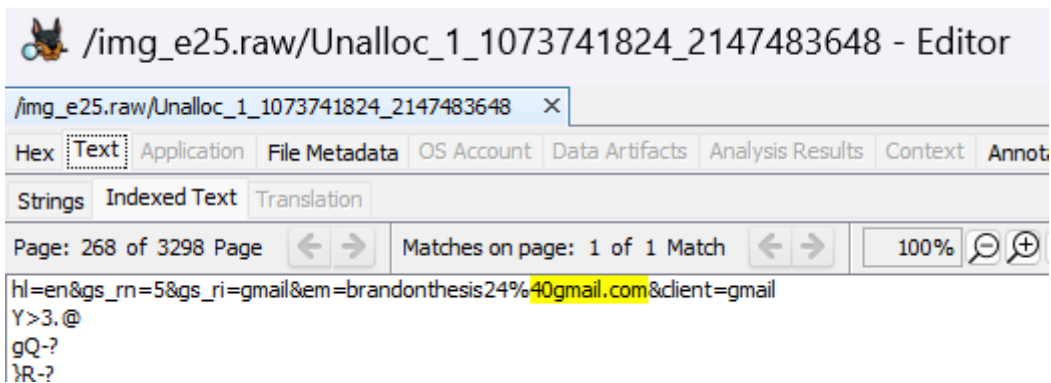


Figure 33 Email Address Extraction from E25.

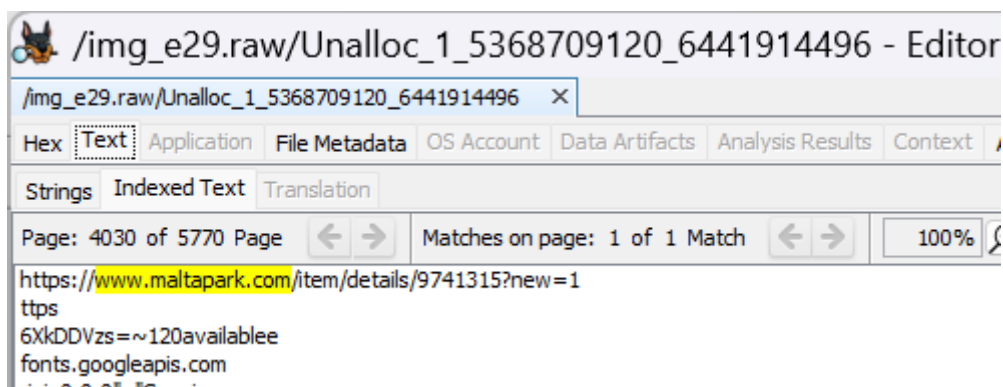


Figure 34 URL Extraction from E29.

The next experiment batch is based on Mozilla Firefox web browser and were taken with both operating systems, machine states, and browser state. Table 9 gives a high-level explanation of the results obtained during the 8 experiments conducted in this section.

Table 9 Mozilla Firefox Experiment Results.

Residual Data Artefact	E2	E6	E10	E14	E18	E22	E26	E30
Web page content	✓	✓	✗	✓	✓	✓	✓	✓
URL's	✓	✓	✗	✓	✓	✓	✓	✓
CDN attachments	✓	✓	✗	✓	✓	✓	✓	✓
Images	✓	✗	✗	✓	✗	✗	✗	✗
Email address	✓	✓	✓	✓	✓	✓	✓	✓
Passwords	✓	✗	✗	✗	✗	✗	✗	✗
Search queries	✓	✗	✗	✓	✓	✓	✓	✓

The first experiment in this section was E2 which was conducted on a Windows-based operating system with the browser still open when the memory dump was taken. It resulted that all the seven parameters that were tested were extracted from the dump. This meant that E2 was the experiment that produced most of the batch data artefacts and the only one that had images present, as shown in Figure 35.

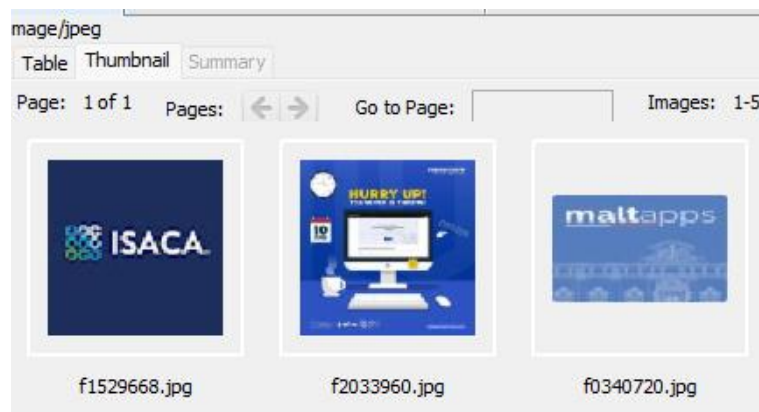


Figure 35 Image extraction from E2.

E18 was the equivalent to E2 but running on Ubuntu operating system. The result varied, and neither images nor passwords were extracted from the memory dump, while the other data artefacts were extracted, too. E22, E26, and E30 were a combination of machine states, and browser state; however, there was no difference in the data artefacts extracted when

compared to E18. E6 was running on the Windows operating system, but the browser was closed, and the memory dump was taken. This resulted in a decrease in data extraction and no images, passwords, and search queries were extracted. E10 has introduced a restart before the memory dump, but the browser was still open, resulting in that only an email address was found in memory. On the other hand, in E14, the browser was closed before restart and all data artefacts except the password were recovered.

The next batch of experiments was conducted on Google Chrome web browser and on both operating systems. Results of the eight experiments in the batch can be found in Table 10 below.

Table 10 Google Chrome Experiment Results

Residual Data Artefact	E3	E7	E11	E15	E19	E23	E27	E31
Web page content	✓	✓	✗	✓	✓	✓	✓	✓
URL's	✓	✓	✗	✗	✓	✓	✓	✓
CDN attachments	✓	✓	✗	✗	✓	✓	✓	✓
Images	✓	✗	✗	✗	✗	✗	✗	✗
Email Address	✓	✓	✗	✗	✓	✓	✓	✓
Passwords	✓	✓	✗	✗	✓	✓	✗	✗
Search queries	✓	✓	✗	✗	✓	✓	✓	✓

The first experiment in this batch was E3, which was conducted on a Windows-based operating system with the browser still open when the memory dump was taken. It resulted that all the seven parameters that were tested were extracted from the dump. This meant that E3 was the experiment that produced most of the data artefacts from the batch. Figure 36 illustrates the credentials extracted from the memory dump, which include both the email address and the password used to log on to the website called “Maltapark”.

`username=brandonthesis24%40gmail.com&Password=Thesispark1%21&`

Figure 36 Extraction of credentials from E3.

E19 was equivalent to E3 but running on Ubuntu, which returned similar results, with the exception that no image data artefacts were found, while E23 returned the same results as

E19. E11 was the only experiment that returned no data artefacts at all, while E15 returned only web page content. Both E27 and E31 returned the same results, allowing the extraction of web page content, URL's, CDN Attachments, email address, and search queries. During the investigation, important information was also extracted, such as email messages and Facebook chat messages. However, these were classified as web page content since the presence of such artefacts will vary depending on the interaction by the user. Figure 37 illustrates an email received by the user, while Figure 38 illustrates the extracted Facebook chat, along with the timestamps of the messages.

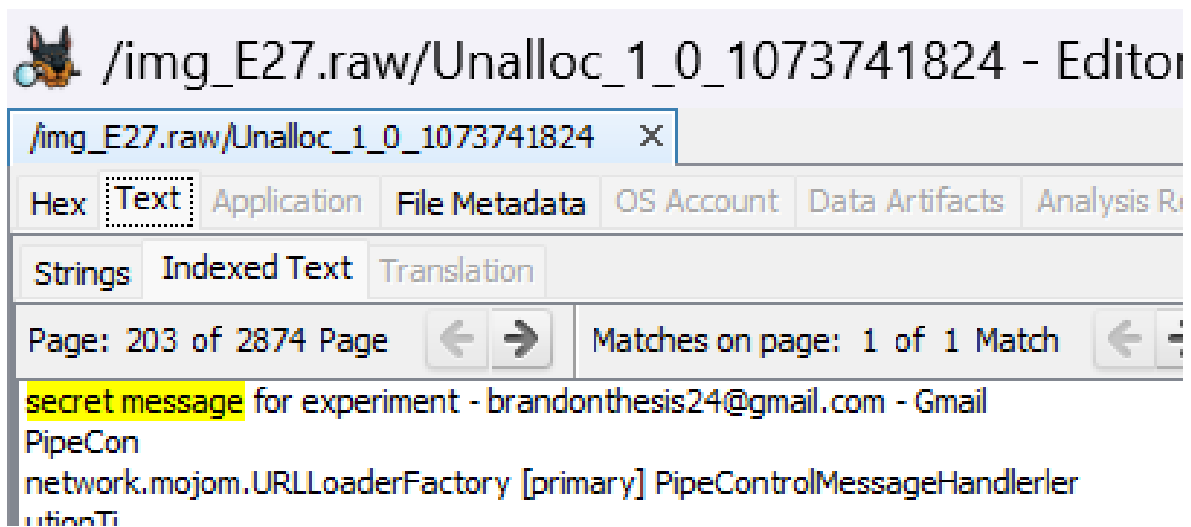


Figure 37 E-mail content extracted from E27.

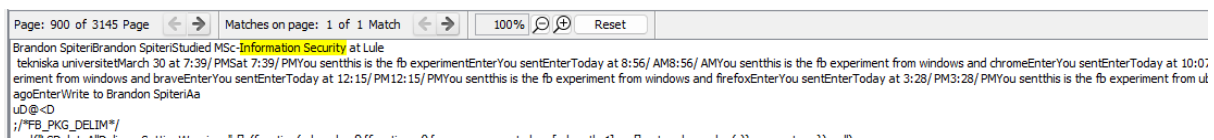


Figure 38 Facebook chat messages extracted from E31.

The final batch of experiments was conducted on Brave Browser, both operating systems, and different machine states. Table 11 illustrates a high-level explanation of the results obtained.

Table 11 Brave Browser Experiment Results.

Residual Data Artefact	E4	E8	E12	E16	E20	E24	E28	E32
Web page content	✓	✓	✓	✓	✓	✓	✓	✓
URL's	✓	✓	✗	✗	✓	✓	✓	✓
CDN attachments	✓	✓	✗	✗	✓	✓	✓	✓
Images	✓	✓	✗	✗	✗	✗	✗	✗
Email Address	✓	✓	✗	✗	✓	✓	✓	✓
Passwords	✓	✓	✗	✗	✗	✗	✗	✗
Search queries	✓	✓	✗	✗	✓	✓	✓	✓

E4 and E8 were both conducted on Windows based operating system and consisted of a memory dump while the browser was either open or closed. Both experiments retained all seven data artefacts along with a considerable number of images that could be extracted, as illustrated in Figure 39. These images were linked to different web pages, such as the 'Malta Police Force' and 'Facebook' profile pictures and pages.

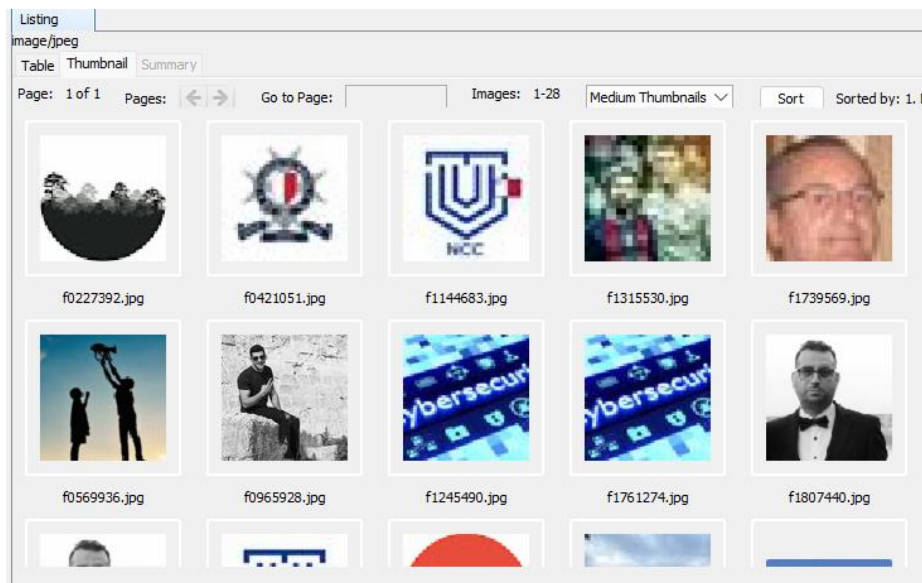


Figure 39 Images extracted from E4.

During the investigation of both E12 and E16, only web page content was found. E20, E24, E28, and E32 made use of a Linux operating system and the results of data retrieved vary when compared to the data artefacts extracted from a Windows based machine. Web page content, URLs, CDN attachments, email address and search queries were extracted but no images or passwords were recovered.

Chapter 5 – Discussion

The scope of this thesis was to determine what data artefacts can be recovered from both Linux and Windows based operating systems. The focus was on private web browsers and the state of the machine that the digital forensic investigator could find. The investigated states were when the machine was in 'running' state or 'restated' state. Another factor that was considered was the state of the private web browser itself, which could be 'open' or 'closed'. To generate data, a simulation of a general user browsing the web was conducted, which included 11 websites, as described in the methodology section.

These combinations resulted in the fact that a total of 32 experiments were needed to generate enough data. During the forensic investigation, data acquisition was divided into two, live and dead acquisition. Dead acquisition simulates the instance in which the digital forensic investigator will acquire data which resides on the disk. This data acquisition method was repeated eight times to extract data from four different private Web Browsers, running on both Linux and Windows based OS. Live acquisition simulates a scenario in which the digital forensic investigator is extracting valuable data from volatile memory, mainly RAM. Since this is highly volatile and data can vary during every process, data acquisition and investigation was repeated across all the experiments, which resulted that 32 investigations were needed.

5.1 Discussion of Dead Acquisition

Table 5 gives a high-level explanation of the results obtained from the investigation focused on Dead Acquisition. The Browsing History was the first data artefact to be extracted from disk, which is extremely valuable. The browsing history was extracted from Mozilla Firefox, Google Chrome, and Brave, while Microsoft Edge did not return any history. The main variable was the browser itself, as the operating system did not affect the data found. However, where the data artefact is found will vary depending on the OS and how the software is installed. Thus, it is important that the digital investigator knows when to research on the specific OS. During investigations, having the ability to trace which site the user visited can be evidence that the user had a particular interest in the area. For example, if a crime is committed which involved a particular firearm, and data about the said firearm are found on the users device, it could mean that the user is aware of what the firearm is.

Ideally, a date and time of when the web page was visited would serve as valuable information, but during data extraction, this was not found. However, the number of times the site was visited was extracted, which can further confirm interest in the search.

Cookies was the next data artefact that was attempted to be extracted, but there was no success across both operating systems and different browsers. On the other hand, the bookmarks were extracted through all browsers and both OS. This is because there is no difference in how between private and non-private browsers, and the bookmark will be saved on the disk regardless. Like browsing history, the location from which this will be extracted will vary depending on the operating system. The URL, date, and time of the bookmark were also extracted, along with the domain of the website. Browser extensions and extensions were extracted only from Firefox, but there was a difference between operating systems. The final data artefact extracted was the download history, which returned variable result across the operating systems and private web browsers. Both OSs running Microsoft Edge returned the download history, while they did not when running Mozilla Firefox. For Google Chrome and Brave browsers, the data artefact was only found when the OS was Windows-based.

5.2 Discussion of Live Acquisition

The second part of the investigation, Live acquisition, was carried out in efforts to extract information from the RAM. The 32 experiments were divided into 4 so that tables were generated to give a high-level explanation of the results. These were tables 6 to 9, which included operating systems, machine, and browser as variables. Data artefacts that were of interest during the investigation were the following.

- Web page content
- URL's
- CDN attachments
- Images
- Email Addresses
- Passwords
- Search Queries

Other data artefacts such as email content and messages were also recovered, however, these are essentially the content of the website. This means that it is a variable of what data is displayed during the memory acquisition and thus was not included as a specific data artefact.

An overview of the results obtained from Microsoft Edge can be found in Table 6. Both website content and URL's were found across all the 8 experiments in the batch except for when a restart was conducted on a Windows-based OS. CDN attachments were extracted through all machine and web browser states when running on Linux OS. However, on Windows, there were only two instances in which these were extractable. These were when the Browser was left open and the machine was still running, and when the browser was closed and the machine restarted. Once the browser is closed, the data is not deleted immediately from the RAM, but the small capacitors will start to discharge slowly. Since the machine was shutdown, no memory refresh was conducted, and that is possibly why the data was still recovered.

The images were the only data artefacts that could not be recovered from either of the experiments. The email address used was retrieved from all instances of Linux-based operating system while it was only retrieved from Windows, when the browser was still open and the machine running. Passwords were also extracted from both operating systems, and for Linux based, there was no difference if the browser was still open or closed. Windows, on the other hand, only enabled the extraction from an open web browser. Search queries were recovered from most of the experiments, but there was a difference which was generated by the machine state. On Windows, the data artefact was not recovered if the browser was left open during a restart, while on Linux it was. On the other hand, the data artefact did not survive the restart on Ubuntu as it did on Windows. When comparing these batches of experiments, it resulted that Windows retained the least data artefact in general.

The results obtained from Firefox running in private mode are represented in Table 7, which included a batch of 8 experiments. Valuable data was extracted from these experiments which can potentially lead Digital Investigators to insights about the case they would be working on. An important data artefact found was the email address used by the user, and this was found across all the 8 experiments in the batch. This is valuable because this email

address could be used on other platforms, thus resulting in additional evidence. Search queries were extracted from all instances of Linux, but a mix between open browser and closed browser on Windows.

Passwords were only extracted from a Windows based OS when the browser was open and the machine was still running. The images were also extracted from Windows, but apart from when the machine was running, the data artefact was extracted from RAM even when the browser was closed and the machine restarted. The content of the Web page, the URLs, and the CDN attachments were extracted from all instances, except when the browser was open and the machine was restarted. In general, during this batch of experiments, the Linux-based OS performed the best between the two. The data artefacts retrieved varied across both OS, however, some critical factors such as passwords were only retrieved from Windows.

During the Google Chrome batch of experiments, the results were extremely varied in all the data artefacts. The only instance that returned images as data artefacts was running Windows with the browser still open. The content of the Web page was extracted from all instances, with the exception of the instance in which the browser was open and the machine was restarted. URL's, search queries, email addresses, and CDN attachments were recovered on all Linux instances, but when the Windows machine was restarted, these data artefacts were not retrievable. The password was extractable from both Windows and Linux when the machine was in the running state. Extracting the password means that the investigator can have access to additional data, especially if the user has only one password for each account. During this batch of experiments, the Windows operating system performed better than Linux, as it retained less data artefacts.

The final batch of experiments was conducted on Brave Browser, and a high-level explanation of the result is found in Table 9. The Web page content was the only data artefact that was recovered from all the experiments. Passwords and images were found on Windows when the machine is in running state, but once the machine was restarted, these data artefacts were not retrievable. Search queries, URLs, email addresses and CDN attachments were recovered from all Linux instances, but only when the machine was turned on in terms of Windows. Search queries are an important discovery because they can further strengthen the particular interest of the user. For example, visiting a website similar

to Facebook does not mean that the user has a particular interest. This means that if search queries are found, these can be linked to the user activity. During the final batch of experiments, the Linux base OS retained more data artefacts than Windows. However, the passwords and images were extractable only from Windows, and thus it was deemed that neither if the OS performed better than the other.

After the experiments were conducted and examined, it was determined that some data artefacts will still be recoverable even if private web browsers are used during the browsing session. These artefacts will vary depending on the machine state, the browser state, and the operating system. Different OS will store data artefacts in different locations and thus it is important that a Digital Investigator is familiar with the different paths. Determining which operating system performed the best would really depend on the variables that add up the whole experiment.

Chapter 6 - Conclusions

This final chapter serves as a conclusion of the thesis, which included the answers, limitations, and future research areas. In the Answers section the deductions of the 3- research question will be presented, while in the Limitations section the limitations of the thesis will be highlighted. Finally, future research areas will be mentioned that can be investigated by other researchers.

The scope of this thesis was to answer the following main research questions:

- Are there any data artefacts left on the device when different popular private Web Browsers are used, and if so, what are they?
- What are the differences between operating systems in terms of the data artefacts found when using Private Web Browsers?
- How do variations in experimental scenarios or states impact the types of data artefacts discovered on digital devices?

Question 1 – The 32 experiments conducted on the 4 private web browsers resulted in that all the browsers left data artefacts even in private mode. This means that a digital investigator will be able to extract valuable information from both RAM and HDD. Defining which browser performed better is difficult, as this depends on the data artefacts in scope. However, when the total number of artefacts extracted from both Live and Dead acquisition is added, Microsoft Edge performed slightly better. Table 12 illustrates the total number of data artefacts per browser.

Table 12 Total Data Artefacts Extracted.

Private Web Browser	Live Acquisition	Dead Acquisition	Total Data Artefacts
Microsoft Edge	34	6	40
Mozilla Firefox	38	4	42
Google Chrome	36	5	41
Brave Browser	36	5	41

Question 2 – The results obtained suggest that there is a difference in terms of the number of artefacts extracted. The Windows based operating system performed better and less data

artefacts were extracted. Table 13 shows the number of artefacts extracted during both investigations, along with a total for each OS. The number of artefacts extracted is extremely important as this can determine the likelihood of relevant data artefacts found during a case.

Table 13 Total Data Artefacts per OS.

Operating system	Live Acquisition	Dead Acquisition	Total Data Artefacts
Windows 11	61	11	72
Ubuntu 23.10	83	9	92

Question 3 – The states investigated lead to the conclusion that there are variations in which data artefacts can still be extracted. Windows 11 lost most of the data artefacts during S2, while Ubuntu retained almost the same number of artefacts. The results in Table 14 include the Live acquisition segment of the experiments.

Table 14 Effects of the machine state on data artefacts.

Operating system	S1 Running	S2 Restarts	Total Data Artefacts
Windows 11	47	14	61
Ubuntu 23.10	44	39	83

6.1 Limitations

One of the biggest limitations was 'time' in the sense that the experiments took a significant amount of time to conduct. 32 experiments were conducted and a browsing web session would take around 60 minutes to generate the data that we were interested in. This was based on 11 web pages and simple interactions, but if more websites and interactions were to be conducted, the results may vary. Another factor was the amount of time it takes to acquire digital forensic images, which falls around the 60 minute bracket when considering both RAM and HDD. The software used will also take around 120 minutes to read the data. This is highly dependent on the size of the HDD and the RAM, which for our experiments were 80GB and 6GB, respectively. Finally, the data analysis itself will take around 60 minutes due to the repetition of the methodology. This amount to at least 5 hours for each experiment, which amounts to 160 hours to achieve the results from 32 experiments.

The tools used were also a limitation due to the price tag of more professional software options. Autopsy is extremely popular among digital investigators; however, specialised software is available on the market. Having the possibility to test these software options could yield benefits such as faster data extraction or even more data artefacts. Autopsy was very useful and worked well, but the question if specialized, premium, software would achieve better results remains. Hardware was also a limitation, since the tests were performed on virtual machines rather than actual hardware.

6.3 Future Work

This research has provided several insights, but future research can still be conducted in the area. The strength of the thesis is based on the number of experiments, thus further experiments could be tested. The results are based on four private web browsers, two operating systems, and two machine states. One can test more private web browsers, more operating systems such as 'Macintosh', and include other machine states such as hibernate and sleep.

As mentioned in the limitations section, further tools can be examined to determine if the same results are obtained. This can be done by making use of trial versions or purchasing premium software. A comparison of the tools can be conducted, which would lead to a different dynamic. Another research area could be if there are any differences when conducting the experiments on a virtual machine, and on physical hardware. This will require several computers that have similar specs across different operating systems, but it is still achievable.

Appendix

Disk enc password -> Thesispassword

Windows password PIN -> 191817

Hex Text Application File Metadata OS Account Data Artifacts Analysis Results Context Annotations Other Occurrences

Strings Indexed Text Translation

Page: 363 of 2874 Page Matches on page: 1 of 7 Match 100% Reset Text Source: Search Results

You sent this is the fb experiment
You sent 39m ago Enter
esJ@
q this is the fb experiment from ubuntu
this is the fb experiment from ubuntu
now
Today at 3:28/PM
Apr
i.c.i
brandonthesis24@gmail.com
brandonthesis24@gmail.co
this is the fb experiment from ubuntu
brandonthesis24@gmail.com
men
1218fb4-1b1f4b4-bb07-ebcf390e6a72
ps-A
x-On
/src:phov3vZr/j5SFE3KJ1_K.png
this is the fb experiment from ubu
lead
webv
this is the fb experiment from ubuntu
scantentLxx.fbcdn.netut
3n2n0GwfrW5k+4FydGATQ==0 GMT

Page: 2171 of 2874 Page Matches on page: 1 of 1 Match 100% Reset

gmail experiment with windows and brave
<CAEF4v6+=CL2UK69_yokJ2HtvW6bpusuRQyqgUdJsq339UafbZQ@mail.gmail.com>
<CABjtFUaiHhTb=cDBYkg7v1xhq_PssajrWJq1m+7Nz6YNV3JXLA@mail.gmail.com>
18e9ddea29a78fa7
msg-a:r-572921412620798291
Re: secret message for experiment
<CAEF4v6JJj8awsovGQh=T=eQgFP++PVp6LCxs2s9baQQxYn5JaQ@mail.gmail.com>
<CABjtFUaiHhTb=cDBYkg7v1xhq_PssajrWJq1m+7Nz6YNV3JXLA@mail.gmail.com>
18e9e529a6d34958
msg-a:r-8619088230918106918
Re: secret message for experiment
gmail reply experiment from ubuntu and firefox
<CAEF4v6KNp8BT-OwAVH8VQDOynpAPP+ksBK6h2uRk6ZrJYd7v1w@mail.gmail.com>
<CABjtFUaiHhTb=cDBYkg7v1xhq_PssajrWJq1m+7Nz6YNV3JXLA@mail.gmail.com>
18e9f024f3a0f432
msg-f:1794963255491278962
Your Password Reset Link
Dear brandon thesis, You have recently requested to reset your password. To reset your password click on the link below. <https://www.maltapark.com/resetpass/?to>
<66082733.170a0220.2cc27.3dd0SMTPIN_ADDED_MISSING@mx.google.com>
18e8fd921c6fb872
?-Ug
?-Ug
~

References

- [1] K. Ruth *et al.*, "A World Wide View of Browsing the World Wide Web," in *Proceedings of the ACM SIGCOMM Internet Measurement Conference, IMC*, Association for Computing Machinery, Oct. 2022, pp. 317–334. doi: 10.1145/3517745.3561418.
- [2] K. Hughes, P. Papadopoulos, N. Pitropakis, A. Smales, J. Ahmad, and W. J. Buchanan, "Browsers' private mode: Is it what we were promised?," *Computers*, vol. 10, no. 12, Dec. 2021, doi: 10.3390/computers10120165.
- [3] F. M. Ghabban, I. M. Alfadli, O. Ameerbakhsh, A. N. Abuali, A. Al-Dhaqm, and M. A. Al-Khasawneh, "Comparative analysis of network forensic tools and network forensics processes," in *2021 2nd International Conference on Smart Computing and Electronic Enterprise: Ubiquitous, Adaptive, and Sustainable Computing Solutions for New Normal, ICSCCE 2021*, Institute of Electrical and Electronics Engineers Inc., Jun. 2021, pp. 78–83. doi: 10.1109/ICSCCE50312.2021.9498226.
- [4] S. D. Viglas, "Data Management in Non-Volatile Memory," *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, vol. 2015-May, pp. 1707–1711, May 2015, doi: 10.1145/2723372.2731082.
- [5] H. Liu *et al.*, "A Survey of Non-Volatile Main Memory Technologies: State-of-the-Arts, Practices, and Future Directions," Oct. 2020, [Online]. Available: <http://arxiv.org/abs/2010.04406>
- [6] G. Bansal and F. F. H. Nah, "Internet Privacy Concerns Revisited: Oversight from Surveillance and Right To Be Forgotten as New Dimensions," *Information and Management*, vol. 59, no. 3, Apr. 2022, doi: 10.1016/j.im.2022.103618.
- [7] C. C. Murphy, "State Surveillance & Social Democracy: Lessons after the Investigatory Powers Act 2016," 2019. [Online]. Available: <https://ssrn.com/abstract=3494880>
- [8] A. Dimitriadis, N. Ivezic, B. Kulvatunyou, and I. Mavridis, "D4I - Digital forensics framework for reviewing and investigating cyber attacks," *Array*, vol. 5, p. 100015, Mar. 2020, doi: 10.1016/j.array.2019.100015.
- [9] S. Kumar, S. K. Pathak, and J. Singh, "A Comprehensive Study of XSS Attack and the Digital Forensic Models to Gather the Evidence," *ECS Trans*, vol. 107, no. 1, pp. 7153–7163, Apr. 2022, doi: 10.1149/10701.7153ecst.
- [10] M. Khanafseh, M. Qatawneh, and W. Almobaideen, "A Survey of Various Frameworks and Solutions in all Branches of Digital Forensics with a Focus on Cloud Forensics," 2019. [Online]. Available: www.ijacsa.thesai.org
- [11] D. Paul Joseph and J. Norman, "An analysis of digital forensics in cyber security," in *Advances in Intelligent Systems and Computing*, Springer Verlag, 2019, pp. 701–708. doi: 10.1007/978-981-13-1580-0_67.
- [12] R. S. Bisht, S. Jain, and N. Tewari, "Study of Wearable IoT devices in 2021: Analysis Future Prospects," in *Proceedings of 2021 2nd International Conference on Intelligent Engineering and Management, ICIEM 2021*, Institute of Electrical and Electronics Engineers Inc., Apr. 2021, pp. 577–581. doi: 10.1109/ICIEM51511.2021.9445334.

- [13] F. J. G. Peñalvo *et al.*, “Mobile Cloud Computing and Sustainable Development: Opportunities, Challenges, and Future Directions,” *Int. J. Cloud Appl. Comput.*, vol. 12, pp. 1–20, 2022, [Online]. Available: <https://api.semanticscholar.org/CorpusID:253165616>
- [14] A. Fukami, R. Stoykova, and Z. Geradts, “A new model for forensic data extraction from encrypted mobile devices,” *Forensic Science International: Digital Investigation*, vol. 38, Sep. 2021, doi: 10.1016/j.fsidi.2021.301169.
- [15] I. Goni and M. Mohammad, “Machine Learning Approach to Mobile Forensics Framework for Cyber Crime Detection in Nigeria,” *Journal of Computer Science Research*, vol. 2, no. 4, pp. 1–6, Sep. 2020, doi: 10.30564/jcsr.v2i4.2147.
- [16] A. Mishra, C. Singh, A. Dwivedi, D. Singh, and A. K. Biswal, “Network Forensics: An approach towards detecting Cyber Crime,” in *2021 International Conference in Advances in Power, Signal, and Information Technology (APSIT)*, 2021, pp. 1–6. doi: 10.1109/APSIT52773.2021.9641399.
- [17] S. Rizvi, M. Scanlon, J. McGibney, and J. Sheppard, “Application of Artificial Intelligence to Network Forensics: Survey, Challenges and Future Directions,” *IEEE Access*, vol. 10, pp. 110362–110384, 2022, doi: 10.1109/ACCESS.2022.3214506.
- [18] R. Alnafrani and D. Wijesekera, “AIFIS: Artificial Intelligence (AI)-Based Forensic Investigative System,” in *2022 10th International Symposium on Digital Forensics and Security (ISDFS)*, 2022, pp. 1–6. doi: 10.1109/ISDFS55398.2022.9800801.
- [19] H. Choi, S. Lee, and D. Jeong, “Forensic Recovery of SQL Server Database: Practical Approach,” *IEEE Access*, vol. 9, pp. 14564–14575, 2021, doi: 10.1109/ACCESS.2021.3052505.
- [20] I. Alfadli, A. N. Abuali, F. M. Ghabban, A. Al-Dhaqm, O. Ameerbakhsh, and M. A. Al-Khasawneh, “CIPM: Common Identification Process Model for Database Forensics Field.”
- [21] A. Al-Dhaqm *et al.*, “Database forensic investigation process models: A review,” *IEEE Access*, vol. 8, pp. 48477–48490, 2020, doi: 10.1109/ACCESS.2020.2976885.
- [22] A. Al-Dhaqm *et al.*, “Categorization and Organization of Database Forensic Investigation Processes,” *IEEE Access*, vol. 8, pp. 112846–112858, 2020, doi: 10.1109/ACCESS.2020.3000747.
- [23] S. R. J. Ramson, S. Vishnu, and M. Shanmugam, “Applications of Internet of Things (IoT) – An Overview,” in *2020 5th International Conference on Devices, Circuits and Systems (ICDCS)*, 2020, pp. 92–95. doi: 10.1109/ICDCS48716.2020.243556.
- [24] J. Hou, Y. Li, J. Yu, and W. Shi, “A Survey on Digital Forensics in Internet of Things,” *IEEE Internet of Things Journal*, vol. 7, no. 1. Institute of Electrical and Electronics Engineers Inc., pp. 1–15, Jan. 01, 2020. doi: 10.1109/JIOT.2019.2940713.
- [25] I. Yaqoob, I. A. T. Hashem, A. Ahmed, S. M. A. Kazmi, and C. S. Hong, “Internet of things forensics: Recent advances, taxonomy, requirements, and open challenges,” *Future Generation Computer Systems*, vol. 92, pp. 265–275, Mar. 2019, doi: 10.1016/j.future.2018.09.058.
- [26] W. Yang, M. N. Johnstone, L. F. Sikos, and S. Wang, “Security and Forensics in the Internet of Things: Research Advances and Challenges,” in *2020 Workshop on Emerging Technologies for Security in IoT (ETSecIoT)*, 2020, pp. 12–17. doi: 10.1109/ETSecIoT50046.2020.00007.

- [27] A. Quist, "Digital Forensic Challenges in Internet of Things (IoT)," *Advances in Multidisciplinary and scientific Research Journal Publication*, vol. 1, no. 1, pp. 119–124, Jul. 2022, doi: 10.22624/AIMS/CRP-BK3-P20.
- [28] T. Kaur, "Cloud Computing: A Study of the Cloud Computing Services," 2019. [Online]. Available: www.ijraset.com
- [29] D. Dzulhikam and M. E. Rana, "A Critical Review of Cloud Computing Environment for Big Data Analytics," in *2022 International Conference on Decision Aid Sciences and Applications (DASA)*, 2022, pp. 76–81. doi: 10.1109/DASA54658.2022.9765168.
- [30] N. H. N. Ahmad, A. S. S. A. Hamid, N. S. S. Shahidan, and K. A. Z. Ariffin, "Cloud Forensic Analysis on pCloud: From Volatile Memory Perspectives," 2020. [Online]. Available: <https://api.semanticscholar.org/CorpusID:226692047>
- [31] A. A. Kazaure, A. Bin Jantan, M. N. Yusoff, A. Maigari, M. K. Ishak, and N. R. M. Noor, "Evidence Collection and Forensic Challenges in Cloud Environment," 2020. [Online]. Available: <https://api.semanticscholar.org/CorpusID:250084933>
- [32] T. Raja Sree and S. Mary Saira Bhanu, "Data Collection Techniques for Forensic Investigation in Cloud," in *Digital Forensic Science*, IntechOpen, 2020. doi: 10.5772/intechopen.82013.
- [33] A. Pon Bharathi, M. Ramachandran, S. Chinnasamy, and M. Mani, "Analysis of Operating System Using TOPSIS MCDM Method," *Electrical and Automation Engineering*, vol. 1, no. 2, pp. 114–122, Aug. 2022, doi: 10.46632/eae/1/2/7.
- [34] K. Tsvetkov Bishop and K. Preslavski, "Списание за наука 'Ново знание' OPERATING SYSTEMS: PAST, PRESENT AND FUTURE", [Online]. Available: <http://science.uard.bg>
- [35] M. Ayyavaraiah, *OPERATING SYSTEM*. Horizon Books (A Division of Ignited Minds Edutech P Ltd), 2021.
- [36] I. Odun-Ayo *et al.*, "Comparative Study of Operating System Quality Attributes," *IOP Conf Ser Mater Sci Eng*, vol. 1107, no. 1, p. 012061, Apr. 2021, doi: 10.1088/1757-899x/1107/1/012061.
- [37] "Stat counter Global Stats Desktop." Accessed: Feb. 24, 2024. [Online]. Available: <https://gs.statcounter.com/os-market-share/desktop/worldwide/#monthly-202401-202402-bar>
- [38] "Stat Counter Global Stats Mobile." Accessed: Feb. 24, 2024. [Online]. Available: <https://gs.statcounter.com/os-market-share/mobile-tablet/worldwide/#monthly-202401-202402-bar>
- [39] M. Mucha, J. Lato, and T. Szymczyk, "Comparison of the most popular operating systems in terms of functionalities Porównanie najpopularniejszych systemów operacyjnych pod względem funkcjonalności," 2022.
- [40] A. Jaiswal, "Linux-the Operating System : Journal of Advances in Shell Programming Journal of Advances in Shell Programming Linux-the Operating System," *JoASP*, pp. 1–5, 2020, doi: 10.37591/JoASP.

- [41] H. Malallah *et al.*, "A Comprehensive Study of Kernel (Issues and Concepts) in Different Operating Systems," *Asian Journal of Research in Computer Science*, pp. 16–31, May 2021, doi: 10.9734/ajrcos/2021/v8i330201.
- [42] A. Barbadekar, N. S. Gawande, and A. Gitte, "Comparison of monolithic, Micro, and Hybrid kernel based on performance and architectural parameters," *Journal of Scientific Development and Research*, p. 2104, 2023, [Online]. Available: www.ijdsr.org
- [43] G. Heiser, "The seL4 MicroKernel An Introduction," Jun. 2020.
- [44] O.-A. Isaac, K. Okokpujie, H. Akinwumi, J. Juwe, H. Otunuya, and O. Alagbe, "An Overview of Microkernel Based Operating Systems," *IOP Conf Ser Mater Sci Eng*, vol. 1107, no. 1, p. 012052, Apr. 2021, doi: 10.1088/1757-899x/1107/1/012052.
- [45] L. Wu, H. Liu, J. Lin, and S. Wang, "Volatile and Nonvolatile Memory Operations Implemented in a Pt/HfO₂/Ti Memristor," *IEEE Trans Electron Devices*, vol. 68, no. 4, pp. 1622–1626, 2021, doi: 10.1109/TED.2021.3061033.
- [46] A. H. Lashkari, B. Li, T. L. Carrier, and G. Kaur, "VolMemLyzer: Volatile Memory Analyzer for Malware Classification using Feature Engineering," in *2021 Reconciling Data Analytics, Automation, Privacy, and Security: A Big Data Challenge (RDAAPS)*, 2021, pp. 1–8. doi: 10.1109/RDAAPS48126.2021.9452028.
- [47] L. Chang *et al.*, "Trend of Emerging Non-Volatile Memory for AI Processor," in *2021 18th International SoC Design Conference (ISOCC)*, 2021, pp. 223–224. doi: 10.1109/ISOCC53507.2021.9613905.
- [48] Y. Qiu, P. Xiong, and T. Zhu, "Memory Management," *Professional C++*, 2018, [Online]. Available: <https://api.semanticscholar.org/CorpusID:45555720>
- [49] M. Hepisuthar and Priyankasharma, "Comparative Analysis Study on SSD, HDD, and SSHD," 2021.
- [50] L. Gao, Q. Zhang, R. A. Evans, and M. Gu, "4D Ultra-High-Density Long Data Storage Supported by a Solid-State Optically Active Polymeric Material with High Thermal Stability," *Adv Opt Mater*, vol. 9, no. 17, Sep. 2021, doi: 10.1002/adom.202100487.
- [51] Q. Cao *et al.*, "Nonvolatile Multistates Memories for High-Density Data Storage," *ACS Appl Mater Interfaces*, vol. 12, no. 38, pp. 42449–42471, Sep. 2020, doi: 10.1021/acsami.0c10184.
- [52] H. Nyholm *et al.*, "The Evolution of Volatile Memory Forensics," *Journal of Cybersecurity and Privacy*, vol. 2, no. 3. Multidisciplinary Digital Publishing Institute (MDPI), pp. 556–572, Sep. 01, 2022. doi: 10.3390/jcp2030028.
- [53] C. Meyers, A. R. Ikuesan, and H. S. Venter, "Automated RAM analysis mechanism for windows operating system for digital investigation," *2017 IEEE Conference on Applications, Information and Network Security, AINS 2017*, vol. 2018-January, pp. 85–90, Jul. 2017, doi: 10.1109/AINS.2017.8270430.
- [54] F. Pagani, O. Fedorov, and D. Balzarotti, "Introducing the temporal dimension to memory forensics," *ACM Transactions on Privacy and Security*, vol. 22, no. 2, Mar. 2019, doi: 10.1145/3310355.

- [55] R. Chaudhary and A. Kansal, "A PERSPECTIVE ON THE FUTURE OF THE MAGNETIC HARD DISK DRIVE (HDD) TECHNOLOGY." [Online]. Available: www.ijtra.com
- [56] F. D. S. Lima, F. L. F. Pereira, I. C. Chaves, J. C. Machado, and J. P. P. Gomes, "Predicting the Health Degree of Hard Disk Drives with Asymmetric and Ordinal Deep Neural Models," *IEEE Transactions on Computers*, vol. 70, no. 2, pp. 188–198, Feb. 2021, doi: 10.1109/TC.2020.2987018.
- [57] V. Tomer, V. Sharma, S. Gupta, and D. P. Singh, "Hard disk drive failure prediction using SMART attribute," in *Materials Today: Proceedings*, Elsevier Ltd, 2021, pp. 11258–11262. doi: 10.1016/j.matpr.2021.03.229.
- [58] K. Kaur and K. Kaur, "Failure prediction, lead time estimation and health degree assessment for hard disk drives using voting based decision trees," *Computers, Materials and Continua*, vol. 60, no. 3, pp. 913–946, 2019, doi: 10.32604/cmc.2019.07675.
- [59] P. Karunakaran, M. Shahril Osman, S. Chee Cheng, and M. Djun Lee, "Optimization of a Hard Disk Factory," *International Journal of Mechanical Engineering and Technology (IJMET)*, vol. 10, no. 10, pp. 217–228, 2019, [Online]. Available: <http://www.iaeme.com/ijmet/issues.asp?JType=IJMET&VType=10&IType=10>
<http://www.iaeme.com/IJMET/issues.asp?JType=IJMET&VType=10&IType=10>
- [60] R. Pinciroli, L. Yang, J. Alter, and E. Smirni, "Lifespan and Failures of SSDs and HDDs: Similarities, Differences, and Prediction Models," *IEEE Trans Dependable Secure Comput*, vol. 20, no. 1, pp. 256–272, Jan. 2023, doi: 10.1109/TDSC.2021.3131571.
- [61] R. Micheloni and L. Crippa, "Solid state drives (SSDs)," *Springer Series in Advanced Microelectronics*, vol. 58, pp. 1–17, Apr. 2017, doi: 10.1007/978-3-319-51735-3_1/FIGURES/12.
- [62] Binaya Raj Joshi and R. Hubbard, "Forensics Analysis of Solid State Drive(SSD)," 2016. [Online]. Available: www.sdiwc.net
- [63] D. M. G. A. , M. Y. U. Dr Abdulaziz Aldaej, "Solid State Drive Data Recovery In Open Source Environment," 2017.
- [64] M. Kumar, "Solid state drive forensics analysis—Challenges and recommendations," *Concurr Comput*, vol. 33, no. 24, Dec. 2021, doi: 10.1002/cpe.6442.
- [65] S. Sai and R. Marupudi, "Solid State Drive: New Challenge for Forensic Investigation," 2017. [Online]. Available: https://repository.stcloudstate.edu/msia_etdshttps://repository.stcloudstate.edu/msia_etds/30
- [66] G. Lukosevicius, A. Rodriguez Arreola, and A. S. Weddell, "Using Sleep States to Maximize the Active Time of Transient Computing Systems," in *ENSys 2017 - Proceedings of the 5th International Workshop on Energy Harvesting and Energy-Neutral Sensing Systems, Part of SenSys 2017*, Association for Computing Machinery, Inc, Nov. 2017, pp. 31–36. doi: 10.1145/3142992.3142998.
- [67] 丛卫东, "The computer system shutdown," 2011. [Online]. Available: <https://api.semanticscholar.org/CorpusID:196052368>

- [68] N. Chumuang, M. Ketcham, T. Ganokratanaa, P. Pramkeaw, W. Yimyam, and Chuamoo, "Automatic Computer Shutdown with Image Processing via Webcam to Save Energy," *2023 IEEE International Conference on Cybernetics and Innovations (ICCI)*, pp. 1–7, 2023, [Online]. Available: <https://api.semanticscholar.org/CorpusID:258465895>
- [69] D. Starkov and S. Belan, "Universal performance bounds of restart.," *Phys Rev E*, vol. 107 6, p. L062101, 2022, [Online]. Available: <https://api.semanticscholar.org/CorpusID:252222281>
- [70] J.-H. Lorenz, "On the Complexity of Restarting," in *Computer Science Symposium in Russia*, 2019. [Online]. Available: <https://api.semanticscholar.org/CorpusID:195353506>
- [71] Y. Agarwal, S. Hodges, J. Scott, R. Chandra, P. Bahl, and R. Gupta, "Somniloquy: Maintaining Network Connectivity While Your Computer Sleeps."
- [72] B. Setz, F. Nizamic, A. Lazovik, and M. Aiello, "Power management of personal computers based on user behaviour," in *2016 5th International Conference on Smart Cities and Green ICT Systems (SMARTGREENS)*, 2016, pp. 1–8.
- [73] K. Kurisu, J. Miura, J. Nakatani, and Y. Moriguchi, "Hibernating behavior for household personal computers," *Resour Conserv Recycl*, vol. 162, Nov. 2020, doi: 10.1016/j.resconrec.2020.105015.
- [74] D. Balsamo, A. S. Weddell, G. V. Merrett, B. M. Al-Hashimi, D. Brunelli, and L. Benini, "Hibernus: Sustaining Computation During Intermittent Supply for Energy-Harvesting Systems," *IEEE Embed Syst Lett*, vol. 7, no. 1, pp. 15–18, Mar. 2015, doi: 10.1109/LES.2014.2371494.
- [75] L. B. Younis, S. Sweda, and A. Alzu'Bi, "Forensics Analysis of Private Web Browsing Using Android Memory Acquisition," in *2021 12th International Conference on Information and Communication Systems, ICICS 2021*, Institute of Electrical and Electronics Engineers Inc., May 2021, pp. 273–278. doi: 10.1109/ICICS52457.2021.9464591.
- [76] M. Khan, "Malware Analysis Using Volatility," 2020. [Online]. Available: <https://api.semanticscholar.org/CorpusID:219042050>
- [77] S. Qureshi *et al.*, "Browser Forensics: Extracting Evidence from Browser Using Kali Linux and Parrot OS Forensics Tools," *International Journal of Network Security*, vol. 24, no. 3, pp. 557–572, 2022, doi: 10.6633/IJNS.20225_24(3).19.
- [78] N. R. Mistry and M. S. Dahiya, "Signature based volatile memory forensics: a detection based approach for analyzing sophisticated cyber attacks," *International Journal of Information Technology (Singapore)*, vol. 11, no. 3, pp. 583–589, Sep. 2019, doi: 10.1007/s41870-018-0263-4.
- [79] S. Berham and S. Morris, "A CRITICAL COMPARISON OF BRAVE BROWSER AND GOOGLE CHROME FORENSIC ARTEFACTS," *Journal of Digital Forensics, Security and Law*, 2022, doi: 10.15394/jdfsl.2022.1752.
- [80] J. Raja Sri and D. V. V. Kumari, "Issue 11 www.jetir.org (ISSN-2349-5162) JETIR2011328 Journal of Emerging Technologies and Innovative Research (JETIR) www.jetir," 2020. [Online]. Available: www.jetir.org
- [81] G. Horsman *et al.*, "A forensic examination of web browser privacy-modes," *Forensic Science International: Reports*, vol. 1, p. 100036, Nov. 2019, doi: 10.1016/j.fsir.2019.100036.

- [82] X. Fernández-Fuentes, T. F. Pena, and J. C. Cabaleiro, "Digital forensic analysis methodology for private browsing: Firefox and Chrome on Linux as a case study," *Comput Secur*, vol. 115, Apr. 2022, doi: 10.1016/j.cose.2022.102626.
- [83] S. Ursell and T. Hayajneh, "Desktop Browser Extension Security and Privacy Issues," *Lecture Notes in Networks and Systems*, 2019, [Online]. Available: <https://api.semanticscholar.org/CorpusID:181818511>
- [84] N. Tsalis, A. Mylonas, A. Nisioti, D. Gritzalis, and V. Katos, "Exploring the protection of private browsing in desktop browsers."
- [85] D. J. Leith, "Web Browser Privacy: What Do Browsers Say When They Phone Home?," *IEEE Access*, vol. 9, pp. 41615–41627, 2021, [Online]. Available: <https://api.semanticscholar.org/CorpusID:211545567>
- [86] M. Piekarska, Y. Zhou, D. Strohmeier, and A. Raake, "Because we care: Privacy Dashboard on Firefox OS," *ArXiv*, vol. abs/1506.04105, 2015, [Online]. Available: <https://api.semanticscholar.org/CorpusID:14593832>
- [87] A. R. Mahlous and H. Mahlous, "Private Browsing Forensic Analysis: A Case Study of Privacy Preservation in the Brave Browser," *International Journal of Intelligent Engineering and Systems*, vol. 13, no. 6, pp. 294–306, 2020, doi: 10.22266/ijies2020.1231.26.
- [88] H. Adamu, A. Adamu Ahmad, A. Hassan, and ad Barau Gambasha, "IJRSI | Volume VIII, Issue V," 2021. [Online]. Available: www.rsisinternational.org
- [89] J. Kim, S. Lee, and D. Jeong, "Digital forensic investigation methodology for Storage Space: Based on the <scp>NIST</scp> digital forensic process," *J Forensic Sci*, vol. 67, no. 3, pp. 989–1001, May 2022, doi: 10.1111/1556-4029.14992.
- [90] Fernando. Carbone, *Computer forensics with FTK : enhance your computer forensics knowledge through illustrations, tips, tricks, and practical real-world scenarios*. Packt Pub, 2014.
- [91] E. Akbal and S. Dogan, "Forensics Image Acquisition Process of Digital Evidence," *International Journal of Computer Network and Information Security*, vol. 10, no. 5, pp. 1–8, May 2018, doi: 10.5815/ijcnis.2018.05.01.
- [92] Ş. Şentürk, T. Apaydın, and H. Yaşar, "Image and File System Support Framework for a Digital Mobile Forensics Software," in *2020 Turkish National Software Engineering Symposium (UYMS)*, 2020, pp. 1–3. doi: 10.1109/UYMS50627.2020.9247055.
- [93] A. Almutairi *et al.*, "Evaluation of autopsy and volatility for cybercrime investigation a forensic lucid case study," *International Journal of Digital Crime and Forensics*, vol. 12, no. 1, pp. 58–89, Jan. 2020, doi: 10.4018/IJDCF.2020010104.
- [94] F. Alief, Y. Suryanto, L. Rosselina, and T. Hermawan, "Analysis of Autopsy Mobile Forensic Tools against Unsent Messages on WhatsApp Messaging Application," in *2020 7th International Conference on Electrical Engineering, Computer Sciences and Informatics (EECSI)*, 2020, pp. 26–30. doi: 10.23919/EECSI50503.2020.9251876.
- [95] Volatility, ! " ! 2.4!Edition!" Accessed: November 12, 2023 [Online]. Available: www.memoryanalysis.net!!

- [96] Anuradha P, Raj Kumar T, and Sobhana N. V, *Recovering Deleted Browsing Artifacts from Web Browser Log Files in Linux Environment*.
- [97] J. Bloomfield and M. J. Fisher, "Quantitative research design," *Journal of the Australasian Rehabilitation Nurses Association*, vol. 22, no. 2, pp. 27–30, 2019.
- [98] A. Mehrad and M. T. Zangeneh, "Comparison between Qualitative and Quantitative Research Approaches: Social Sciences." [Online]. Available: <https://orcid.org/0000-0003-4364-5709>