Master of Science Dissertation

Data Deduplication: Breaking the Speed Limit of Digital Forensic Investigations

Stephen Kohlmann

A report submitted in part fulfilment of the degree of

MSc in Computer Science

Supervisor: Dr Mark Scanlon

Moderator: Dr Mark Scanlon

UCD School of Computer Science

College of Science

University College Dublin

August 23, 2016

Table of Contents

Abstract

Digital forensics is a relatively new field in both industry and academia. Many digital forensic tools have been created in this short time but the tools are never sufficient enough to deal with the exponential backlog of cases digital investigators face. The backlog of cases exists for a combination of reasons but two of the most prevalent are, a lack of standardisation of digital forensic investigations and the ease of public access to massive volumes of storage.

The research question proposed is, can the digital forensic investigation process be quicker? The data deduplication tool aims to speed up digital forensic investigations and offer digital forensics as a service (DFaaS) to digital investigators. The concept behind the tool is for a database in the cloud to store known acquired files where digital investigators can compare files for their ongoing investigations. The tool aims to speed up the process of disk acquisition by comparing a file previously acquired to a newly acquired file by the hash values from each file. If the hash values match it means the database already has the file but if the hash values do not match the file is then sent to the known files collection in the database for future investigations.

The literature review chapter gives a detailed overview of some of the key areas related to the data deduplication tool. The areas reviewed include remote evidence acquisition, the current backlog, hard drive forensics, information retrieval, DFaaS and other related work. The literature review lays the foundation for what the data deduplication tool is built upon. The design and flow of events of the tool are shown in Chapter 3 with an overview of why Python and MongoDB were chosen as the technologies to build the data deduplication tool.

The experimentation and results chapter consists of six tests carried out with the data deduplication tool. The initial test is carried out on Window 7,8,10 and Ubuntu to show the speed of reading the disk and acquiring file metadata only. The following five tests run the data deduplication tool from start to finish and are carried out on Windows 7, 8 and 10. The tests aim to prove how the implementation of the data deduplication tool can speed up a digital forensic acquisition which in turn will speed up a digital forensic investigation.

The body of work in this thesis outlines the real problem of digital evidence backlogs worldwide. The data deduplication tool has been built and rigorously tested with the primary aim of tackling this problem. The literature review and design chapter explain the data deduplication tool in detail and the tests carried out in Chapter 5 show the potential advantages of using the data deduplication tool in industry.

Chapter 1: **Introduction**

1.1 Research Question

1.1.1 Volume

When it comes to Digital Forensic Investigations one of the primary obstacles in pertaining evidence from a data source is the issue of the growth of volume and scale. As Lillis et al. states with the complexity and increase of forensic data, forensic tools must have the ability to adjust to the current climate. The infamous Moore's law as referenced by many academics puts the issue of increasing volume into perspective. Lillis et al. describes Moore's Law as the observation over the history of computer hardware where the number of transistors in an integrated circuit has doubled approximately every two years. Moreover, hard drive size is about twice the pace of Moore's Law [18].

Taking the approximate scale of Moore's law and the increase in hard drive sizes into consideration it is clear the field of Digital Forensics suffers to keep on top of the ever increasing amount of data. All of the above does not take into consideration the advent of crimes taking place within cloud storage mediums. As Quick and Choo point out, with easy access to cloud storage individuals can store and share illicit data to multiple locations. This adds multiple complications and massive volumes of scalable data for forensic analysis.

1.1.2 Lack of Experts

Digital Forensics is a technical field that requires years of training and experience for an individual to become an expert. Organizations report that it typically takes between one and two years of on-the-job training before a new forensics examiner is proficient in the skills to lead an investigation. Garfinkel, Quick and Choo point out that the current forensic lab involves forensic practitioners working on low level tasks like network management and system administration. Coupled with the lack of experts is the lack of average computer users to carry out the low level tasks. As Goodison et al. explains, departments do not have enough personnel to process the volume of digital evidence and the variety of tools available makes no difference to the large backlogs of data. Both Garfinkel and Goodison et al. suggest that law enforcement professionals have a lack of training in pertaining digital evidence. This is particularly prevalent when first responders to a crime scene do not know how to "bag" the digital evidence for lab analysis.

In 2010 Garfinkel stated that organizations encountered data that could not be analysed with many digital forensic tools because of format incompatibilities and encryption. Five years later this problem still exists as departments lack the tools to represent data in clear and understandable ways for an investigation and presentation [14]. In the academic field researchers lack a systematic approach to reverse engineering and developers emphasise complete forensic analysis over speed [13]. The lack of a defined system for digital forensic analysis causes each new case to become a stand alone project. As new cases pile up on the digital forensic specialists work bench a continuous backlog s inevitable. As Garfinkel states there is a lack of complex and realistic training data available for teaching digital forensics and dealing the

data volume issues.

The complications and volume challenges mentioned above all pertain to local hard disk drives. Another major contributor to the increased volume is the freely available and easily accessible storage in the cloud. As Morioka and Sharbaf states the lack of forensics investigation readiness of cloud providers is a major issue. Moreover cloud service providers are not required to take measures to provide detailed support in digital forensic investigations [19]. As Ruan et al. states cloud providers are unforthcoming to help in Digital Forensic Investigations. Cloud storage creates new challenges for digital investigations, coupled with the almost unlimited storage is the challenge of getting the data from the cloud providers to analyse.

1.1.3 Complexity of Problem

As Quick and Choo state the complex challenge in digital forensics investigations is finding the relevant information for cases especially when dealing with large volumes of information. Further adding to the complexity of the problem is the exponential growth in data in the cloud which impacts on timely analysis in investigations [24]. As Quick and Choo state the acquisition time for a readily available 3 terabyte hard drive can take over 11 hours to initially image. The imaging and processing times are increasing exponentially and the problem is compounded further for forensic departments who have constant difficulty keeping up with the speed of technological changes for collecting and analysing digital evidence [14].

As Quick and Choo state, in 2014 it was possible to buy a 4 terabyte hard drive for less than US150. Today it is possible to purchase an 8 terabyte Mac-ready external hard drive, compatible with Apple Time Machine that offers password protection and hardware encryption with the fast transfer rates of USB 3.0 for US270.60 [1]. This is twice the storage in comparison to 2014 with additional encryption and data transfer speeds for less than double the price. As Quick and Choo state a hard drive with 200 gigabytes of storage can store in the region of 4 million pictures. Take today's readily available 8 terabyte hard drive with the same numbers and there is a possibility of storing 160 million pictures.

Moreover it is possible on an 8 terabyte hard drive to store 240,000 minutes of compressed video. This equates to 4000 hours, or 166 days of video footage. For a digital forensic examiner who works 9 am to 5 pm five days a week it would take just under 2 years to play every video file from start to finish. Hard disk drives were introduced in 1956 and took thirty five years to reach 1 gigabyte. Fourteen years on hard drives reached 500 GB and it took an additional two years to reach 1 terabyte. The increase in hard disk sizes and complexity that this causes digital investigators is here to stay. [24].

As Morioka and Sharbaf states at present the prosecution and conviction of cyber crime and those associated is completely dependent on the collection of digital evidence from the suspect storage mediums. The hurdle of collecting digital evidence is only the start of the complexity for digital forensic investigations. As Morioka and Sharbaf and Ruan et al. state the digital evidence collected must satisfy all of the same legal requirements as standard evidence to make it admissible in court.

In addition to digital forensic evidence meeting strict standards to be admissible in court the providers of cloud services are reluctant and can even deny the access to their hardware for digital investigations [10]. As Chen et al. states this is particularly prevalent where a client could gain access to another clients data. It is time-consuming to analyse potential digital evidence in a distributed environment such as the Cloud. As Chen et al. states the actual effect of the evidence retrieved from the suspect device is largely dependent on the source of

the data. For example, the attributes of encrypted files may be discovered but may not be fully accessible causing the evidence to be non admissible evidence.

When the cloud is a data source for a digital forensic investigation there are additional complex problems for an investigator such as having no physical control and no physical location of the data. As Ruan et al. states the complexity of the cloud is compounded further with legal issues when multiple jurisdictions and multiple users are involved. This segregates evidence across multiple locations and multiple users making the initial recovery of data extremely time consuming.

Adding to the complexity of digital forensic investigations is the rapid growth of the Internet of Things (I.O.T). The I.O.T brings a multitude of devices including watches, smart TV's and smart shoes [4] that can connect to the internet and potentially contain data for a digital forensic investigation. As Sutherland et al. states there are many hacking forums aimed at modifying the firmware contained in smart TV's.

As Sutherland et al. states there are forums to support Peer to Peer software and instruction on torrent client installation. Smart TV's are known targets for malware and one very unpleasant example is a link that sends users to a child porn page. The page takes over the system and then asks for payment to clean the system [28]. There are many cases of hackers accessing the in built camera and microphone of computers and the same is true with Smart TV's [28].

As Garfinkel and Scanlon and Kechadi state there is currently no universal standard for the format for digital evidence, there has been some work in relation to create common file formats and processes but little universal standardization. This is the case because no international government policy is in place for formatting digital evidence [27]. As Garfinkel states very few digital forensic tools are built upon previous tools which adds to the lack of standard processes in the field.

As Scanlon and Kechadi state collecting evidence can pose legal issues such as accidentally capturing information with privacy or security implications such as passwords. Captured evidence is then subject to a hearing were a judge determines that the process used to capture the evidence was acceptable and reliable [27]. The judges decision is influenced heavily on if the digital forensic process is published or waiting a peer review. As Scanlon and Kechadi state the area of digital forensics has only one major peer-reviewed journal, the International Journal of Digital Evidence.

1.1.4 Amount of Cases

As Morioka and Sharbaf state the explosive growth of Internet-based information and technology increases the potential of cyber crimes occurring. The increasing volume leads to case backlogs in forensic labs and in 2012 the volume of data for open cases was 5.8 Petabytes [24]

Morioka and Sharbaf state that digital forensic cases are growing at a rate of about 35 new cases per year. The amount of cases are increasing as cloud services are becoming widespread and easily accessible [19]. Given the current digital society digital evidence is expansive in scope and the potential amount of cases therefore increases, additionally the wide range of devices that can contain digital evidence is increasing meaning digital forensic examiners must retrieve information from a plethora of devices [14]

1.1.5 Process

As Graves states the current basic stages of a digital investigation involves four main steps that are Assessment, Acquisition, Analysis and Reporting. These four steps are broken down further into sub sections as follows. The assessment path can be criminal, civil or internal depending on the investigation. The acquisition stage involves using the correct tools to collect the data and make forensic copies. The analysis stage involves analytical study of live, static and network data and the final reporting stage contains the collection, preparation and presentation of the data.

The collection and preparation of the data must documented in detail to be assessed on a legal level. As Scanlon and Kechadi states it is imperative all types of evidence and procedures used are reliable and verifiable. The lack of international and national standards in digital evidence means the evidence must pass legal criteria for the particular location that the court case will take place. As Garfinkel points out much of the last decades progress is quickly becoming irrelevant and the field of Digital Forensics is behind according to the current state of the art.

As Morioka and Sharbaf and Garfinkel state there is an urgent need for new technology and re-imagined digital forensic tools to speed up the digital investigation process. Moreover a major issue is the traditional digital forensic model is not effective when investigations involve Cloud services [10]. Regulations, laws and technology for cyber crimes needs an overhaul and updated process to tackle the current cases and future exponential cloud storage were cyber crime can take place [10].

As Garfinkel states digital investigators need a defined process for reverse engineering. Moreover, extra automation is needed to improve digital forensic investigation procedures and tools. Each case becomes a stand-alone process meaning the results of one case does not help speed up a new case [13]. Illicit or known incriminating files are not stored in a central repository for new investigators to cross analyse. A tool like this would greatly improve todays digital forensic process.

1.1.6 International Issues (Case priorities dictated by Government)

As Scanlon and Kechadi state one aspect of search and seizure warrants within the Internet environment pertains to the the geographical scope of the warrant issued by a judge. The main problem with International cases is the scope and process for each jurisdiction is different. For example in a case of child pornography some countries have lower legal age limits for consent meaning something deemed illegal in one country may not be in another. As Garfinkel states legal issues in regard to jurisdictions increasingly limit the scope for successful digital forensic investigations. Legal issues become more complex when data is uploaded to the cloud. As Morioka and Sharbaf and Chen et al. discuss evidence can be distributed over multiple servers located in different countries slowing investigations down due to legal issues and case priorities in each country.

As Chen et al. points out any user in any part of the world can use a variety of Cloud services with great ease to carry out cyber crimes leaving little trace of their actual identity. Ruan et al. lists the top challenges for cloud forensics with the top three all pertaining to International issues. Jurisdiction issues are the most prevalent followed by lack of international collaboration and legislative mechanisms. The third issue is the Lack of law and regulation standards on an international stage for digital forensic investigations.

The digital investigator must jump through multiple hoops to get the access to the physical

hardware for a case. As Morioka and Sharbaf mentions on many occasions access to data for a case is impossible. With no international regulation in place the effectiveness of cloud forensic investigations and retrieval are severely compromised [19]. Evidence recovery without compromising other users rights is also a major issue that compounds the international legal complication even further [19].

Chapter 2: **Literature Review**

2.1 Remote Evidence Acquisition

Remote evidence acquisition is the process of identifying and acquiring data from a suspect machine. An accurate copy of the data being replicated to or from the suspect machine must be kept to keep the evidence forensically sound. The following outlines some of the research in this area.

A key study in 2009 was carried out with the aim of only transmitting necessary data when sending forensically sound drive images from a remote client to a central site [31]. Necessary data does not include operating-system files, common media files and common applications. As Watkins et al. states these common files exist on all machines so there is no need to acquire them on every new case. To obtain a bit for bit forensic copy it is important for the transfer tool to work directly with disk blocks. Currently forensic tools pre-process hard drives and build a hash database of all known files. The files hash vales are not maintained therefore the processing is processed by the analysis tool [31]. This allows the hard drive image to be rebuilt when the analysis is completed.

As Watkins et al. states the Teleporter tool does the pre-processing on the client side and is able to do this as it maintains a copy of all known files on the server side. As it is commonly known and stated by Watkins et al. the definition of forensically sound in Digital Forensics is a bit by bit copy of a drive to a read only hard drive. The teleporter tool improves on the current process by obtaining a forensically sound copy from a hard drive image. The design of the teleporter enables work flow that was only possible on the server-side to be processed on the client side.

Teleporter works by building a unique file storage instance from the current known files. The hash databases of each unique instance are then sent to the clients. On the the client side the hash databases compare the data stored on the server side to the drives on the client side [31]. Moreover, the Teleporter tool has the ability to identify files and partial files that the tool has already picked up. As Watkins et al. states the tool utilises the SHA-256 hash algorithm due to the algorithms resilience against file collisions. On the server side the tool uses two tables constructed in a MySQL database, one table is for known disk clusters and the other table is for known files from previous investigations [31].

As Watkins et al. explains the server side must also keep a repository to host all known files to rebuild the original disk image. The repositories directory structure is important as the setup improves the access time to the known files. The lookup step normally required to locate a file is bypassed as the directory structure is arranged in a hash tree configuration. As Watkins et al. states the files are named according to their hash values and placed in directories that correspond to those values. Moreover, the tool allows for scalability as each branch of the tree has the option of distribution across several physical volumes.

As Watkins et al. and Roussev et al. explain the preservation and forensic integrity of the hard drive image is paramount for evidence to be admissible in court. One important requirement is that unallocated space is accounted for in the hard disk image copy. As Watkins et al. explains the Teleporter tool records the entire contents of the image including all unused disk clusters. This process is to ensure that each block of data has been accounted for on the disk

[31].

After processing all of the files, Teleporter begins scanning the free space sections of the disk which the file system considers to be empty. This area often contains partial files and residual copies of deleted files. The Teleporter tool is built in Java but it does rely on an external application that is file system specific. To accommodate for various file system storage implementations a parser needs to be designed for each file system [31]. The plus side to this design is that a developer only need to design a parser for the file system they wish to work on. Implementing a new parser does not affect the tools source code and no additional design modifications need to be used to run the tool [31].

As Watkins et al. states the results of the Teleporter tool can reduce the amount of data sent across networks by up to 70 percent. The Teleporter tool can create an analytically sound drive to investigate much faster then any other current tools in the Digital forensics industry [31]. Moreover, the most important element of a digital investigation of creating a forensically sound image of a suspect disk drive is possible with the Teleporter tool.

As Scanlon and Kechadi state when acquired digital evidence is transmitted over the Internet the bag of digital evidence contained should split the evidence into smaller packets. The reason for this is to minimise transmission times that dropped connections create. Moreover, any evidence that a digital forensic tool splits during the acquisition stage for storage or transmission must be able to compile the blocks back to the original disk for analysis [27]. As Scanlon and Kechadi state to ensure forensic integrity any tool used to split and compile evidence should verify the recompiled disk image to the original.

As Thethi and Keane state the acquisition and forensic analysis of digital evidence is extremely time consuming. The time is increased for remote acquisitions due to the complexity of tools required and transmission times from remote sources. Cloud providers reluctance to use forensic services within their design mean that the current remote evidence acquisition tools are the go to for acquiring remote evidence [29]. As Thethi and Keane state exploiting the computational resources available from the cloud for acquisition and using the remote capability of the FTK(Forensic Tool Kit) for partial and memory data acquisition, the time is reduced significantly for acquiring the remote evidence.

Digital crimes can often involve the combined use of local storage, cloud storage and virtual machines. As Almulla et al. states there is a possibility to create a snapshot of a virtual machine that provides a virtual machine image. Problems for digital investigations can occur when a snapshot is created if the virtual machine is working on a platform as a service or software as a service model. The issue is that access to virtual machines on these services is severely limited for digital investigations [6]. Roussev et al. asks the question, How do we formulate digital forensic analysis into a streamlined real-time process? The underlying issue is lack of speed within each stage of the investigation and as Roussev et al. states many other stages of a digital investigation are also very slow such as carving and indexing. For tools to achieve the necessary performance for remote acquisition developers today need to take into account the exponential volume of data and as Roussev et al. suggests the complete overhaul of the digital forensic architectural model is needed to implement a real-time sufficient process. Roussev et al. discusses the common open source forensic architecture known as the Sleuthkit. As stated by Roussev et al. there is no concern for latency within the Sleuthkit architecture meaning that it is impossible to achieve processing deadlines for digital investigations.

2.2 Backlog

2.2.1 The Current Reality

The backlog of digital forensic cases is due to two main factors, the first is the ever increasing storage on local devices and ease of access to cloud services that offer scalability of storage. The second issue is the process that must be followed to make the evidence acquired forensically sound for court. As Watkins et al. states the community accepted definition of a forensically sound drive is a method of copying a disk which does not alter any data on the drive that is duplicated. The copied disk drive must also be a bit by bit copy to be considered forensically sound.

As Lillis et al. states there have been delays on digital forensic investigations of up to four years. Moreover, the delays in cases have resulted in prosecutions being dismissed in courts [18]. Lillis et al. points out that in recent years progress has been made in the digital forensic process but only a small amount of this work is in relation to tackling the evidence backlog. The backlog often means investigations remain open looking for new leads even though the potential evidence is already in the hands of the police. As Lillis et al. and Roussev et al. both show the amount of data per case had grown exponentially at the FBIs regional computer forensic labs. In 2003 the amount of data was 84GB and by 2011 it had increased by 475GB to 559GB.

The volume of data to be acquired, stored, analysed in combination with the lack of experts in the field is a recipe that causes a significant backlog in cases waiting on forensic analysis [16]. As Scanlon and Kechadi and Lillis et al. state the digital evidence backlog is in the order of years for law enforcement around the globe.Hitchcock et al. carried out an informal review that resulted in digital evidence backlogs up to four years.

As Hitchcock et al. states with the current backlog a mechanism is needed to filter which investigations should be given priority. The standard and most logical mechanism used is based on the type of offence committed. In most countries crimes against a person are given priority and these cases usually are related to child pornography. Although this mechanism is important it does not decrease the overall backlog as other cases such as digital fraud are not investigated in a timely manner [16].

As Quick and Choo state the growing volume of data for digital analysis can often consist of terabytes of data for each investigation. Contributing to this backlog is the ever increasing number of devices to be investigated for each case and the size of data on each individual device is exponentially increasing. The results of the current backlogs mean suspects are denied access to their families while the forensic investigation is in place and the harsh reality that criminals are released without charge due to lack of evidence Quick and Choo

As Thethi and Keane and Almulla et al. state the ever increasing and almost infinite storage available in cloud computing services is a major problem for digital forensic investigators. As Thethi and Keane points out, in the near future the amount of data stored on a suspect device will become too large for digital forensic investigators to complete a full investigation. Moreover, extracting evidence from a multi-tenant cloud architecture is impossible with current digital forensic methods meaning that evidence based in the cloud may not be attainable for an ongoing investigation [6]. The exponential increase in the storage available in cloud computing services means that forensic analysis of the client and cloud service provider will create significant difficulties for standard forensic tools [6].

As Almulla et al. states there are two main reasons why the current speed of digital forensic tools contribute to the backlog of cases. Firstly, many digital investigators lack the training

for the tools and do not know how to set them up for best performance per investigation. Secondly, the developers have failed to address latency, reliability and correctness within the forensic tools they build [6].

2.2.2 The Current Challenges

Lillis et al. discuss the current challenges in the digital forensic field in relation to the exponential growth of data to be analysed. The challenges are broken down into five sections described below.

The Complexity Problem

The data acquired in a digital forensic investigation is at the lowest format possible on the device, in most cases this is binary data. The complexity problem relates to the exponential increase in the amount of data per case and the complexity problem occurs when the data set is too large for standard analysis. When this occurs the digital forensic investigator must use data reduction techniques before acquiring the disk for an investigation therefore increasing the time spent on each investigation [18].

The Diversity Problem

As Scanlon and Kechadi state the lack of clear standardisation of digital evidence storage and the formatting adds to the complex and diverse nature of data acquisition. Digital investigators have no go to tool or standard for dealing with multiple file formats, operating systems and cloud service provides.

The Consistency and Correlation Problem

As Lillis et al. states the consistency and correlation problem results from existing tools that are constructed to search for fragments of evidence. Although these fragments of evidence are important the tools do not help a digital forensic investigator prepare or formulate the data for a case.

The Volume Problem

The running theme of this project and major issue resulting in the backlog is increased storage capacities and the sheer number of devices that can store pertinent information. Increasing storage and multiple devices creates the volume problem and a lack of automation for analysis contributes to the backlog even further [18]. As Lillis et al. states mobile and smart technologies are becoming prevalent among the general population. Smart devices have reached a point where they can function on the same scale as an average household computer [18].

The Unified Time-Lining Problem

Developing a unified and coherent time line for a digital forensic case is essential for the evidence acquired to be admissible in court. Complications arise when multiple data sources displaying different time stamp interpretations or time zone references [18]. Coupled with this is the possibility of clock skew issues which can make the digital forensic investigation process more complex and time consuming when creating a time line in relation to the acquired evidence.

Cloud Storage

As Lillis et al. discusses the use of cloud services are commonplace and easily accessible among nearly all internet users. Cloud services such as iCloud, Google Drive, BitTorrent Sync and Dropbox offer free storage and affordable paid plans to increase cloud storage. Recently Apple lowered their iCloud pricing and it is now possible to get one terabyte of cloud storage for under ten pounds. This is the current reality and shows easy and affordable access to large amounts of storage for internet users.

The reality is the cloud brings some difficult challenges to a digital investigation. As Lillis et al. states the data uploaded and stored in the cloud is distributed on distinct nodes. There can be many distinct nodes to investigate and this is unlike the traditional model where data is stored on a single hard disk. The distributed nodes within cloud services also bring the possibility of data residing in multiple countries. Each country comes with its own legal jurisdictions further complicating the retrieval of data for an investigation [18]. Cloud services also offer IP anonymity to users that can allow minimal personal information during sign up and have led to cases were identity of a suspect is impossible to confirm [18] [26]. Crimes that utilize the cloud often use added encryption and various anti-forensic techniques to hide the users and content of the crime [18]. As Lillis et al. and Ruan et al. state data in the cloud can be deleted or overwritten at any time by the cloud provider or the user making the investigation time sensitive.

As Morioka and Sharbaf state data acquisition is a difficult challenge in cloud forensics. The acquisition can be split into two parts, firstly knowing what part of the cloud the data is hosted and secondly actually acquiring the data. As Morioka and Sharbaf and Ruan et al. states the data for the investigation can be distributed over multiple servers located in different countries. Many cloud service providers store copies of data on multiple servers to avoid data loss [19]. The replication can be helpful to an investigator but also cumbersome as they need to trace back the original upload of data to the cloud. The process of finding the original upload can open the investigation further as multiple users may have received or shared the data also. As Morioka and Sharbaf states the hyper-visor in the cloud architecture is a prime target for cyber attacks and with a lack of policies and techniques for investigating hyper-visor attacks leaves a window open for a tech savvy criminal to remove traces of their usage.

As Thethi and Keane and Almulla et al. state the cloud is an excellent platform for cyber criminals and with the addition of creation and deletion of virtual machines in a short space of time gives a criminal the ability to transfer and delete evidence in a very short time frame. Moreover, even if a virtual machine is present, most cloud service providers will not allow imaging of the hard disk that the virtual machine resides on [29]. The disconcerting reality is that digital forensic investigations involving cloud services may never be able to image the entire evidence meaning a case will be built on incomplete evidence [29].

As Thethi and Keane state the cloud computing architecture offers a suite of different ser-

vices and deployment models. Moreover, complex virtualization methods and distributed computing are important to identify within current digital forensics procedures [6]. Digital investigators must be trained in the complex attributes that data residing in the cloud creates for a digital investigation. Currently researches are faced with a lack of standardization, Service Level Agreements (SLAs) and data security when an investigation involves the cloud [6].

The Internet of Things(IoT) is becoming a massive contributor to the additional devices that potentially need investigation. Moreover, the issue is compounded further as the IoT devices use varying file formats, operating systems and communication protocols [18]. Additionally, the problem of removing the local storage from these devices can de difficult and time consuming. As Lillis et al. states in some cases IoT devices lack any form of persistent storage which brings the investigation down the path of getting warrants to access remote servers. As Lillis et al. states the current reality of the problem is growing and here to stay with 13.4billion IoT devices in existence. This figure is expected to reach 38.5 billion in the next five years [18].

As Baggili et al. states smart technology comes with unique challenges in relation to acquisition, handling and data visualization. Most smart watch models can interact with smart phones and until recently both devices were needed to gain the full benefit of the watch features. Apple recently announced that all apps for the Apple Watch must be native apps built on the watchOS 2 SDK [2]. As Baggili et al. points out smart watches can also create, process and transmit data like health and fitness information without been tied to a smart phone. The smart watch moving to an independent development platform will add more complexity to digital investigations and should be researched to understand the best methods of forensic analysis.

As van Baar et al. states the initial detective on the scene of case rarely has any knowledge of a digital investigation. Moreover, digital investigators have no detailed knowledge of the case as they are not involved from the beginning [30]. The current reality disconnects detectives and digital investigators from all of the case details causing the an investigation to be delayed.

Digital investigators are currently required to carry out a number of tasks that are unrelated to their expertise. Digital Investigators are required to perform mundane tasks such as response jobs, creating disk images, memory dumps, and network captures [30]. Although these tasks are necessary they can be carried out by individuals with moderate computer knowledge which would free up the digital investigator to carry out their main role. As van Baar et al. states the first couple of steps of a digital investigation are generally the same. The reasons why digital investigators are busy is due to the amount of time spent on mundane tasks and there is always another case waiting to be investigated. As van Baar et al. states there is never any time left for a digital investigator to innovate or share new knowledge in the community.

As Garfinkel states digital forensic tools are now used on a daily basis by examiners and analysts within local, state and Federal law enforcement. The current reality is cases involving digital evidence are increasing exponentially and there is an urgent need for new digital forensic tools to be designed with the backlog and current smart device growth in mind.

2.3 Hard Drive Forensics

The following section describes some of the current research in hard drive forensics. This area of research is important as analysing digital evidence is extremely time consuming [6]. As Almulla et al. states top of the line digital forensic work stations take up to 85 hours to analyse a 2 terabyte hard drive. Garfinkel describes some alternative methods for forensically analysing a hard drive. The methods explained below are stream based forensics, stochastic analysis and prioritized analysis.

2.3.1 Stream-based disk forensics

As Garfinkel explains stream based forensics is the process of analysing an entire disk image as a stream of bytes. The stream starts at the beginning of the disk and reads until the end. The benefit of this approach is that it eliminates drive head seek time and assures that all data on the disk will accumulated [13]. The downside to this approach is the amount of RAM needed to reconstruct the file system hierarchy [13]. As Garfinkel points out it may still be possible to recover useful information for an investigation from a drive without building the entire file system hierarchy. The method of stream based disk forensics is more beneficial for standard hard drives rather than solid state drives. As Garfinkel states solid state drives have no moving head eliminating the need to seek but it still maybe easier to use this method to scan from the start of the disk to the end. The standard method is to scan the disk making a file-by-file recovery followed by a second scan to acquire the details of unallocated sectors on the disk.

2.3.2 Stochastic analysis

As Garfinkel states another model for forensic processing is stochastic analysis. This method samples and processes a hard drive by choosing random sections on the drive instead of analysing the disk form the beginning to the end. As Garfinkel states this method can yield forensic results from a hard drive very quickly. The major disadvantage of this method is that small pieces data have the potential to be missed [13]. The data missed could be key to an investigation and although the method is fast the risk of missing data out weighs the advantage of speed in this case.

2.3.3 Prioritized analysis

As Garfinkel states prioritized analysis is a triage based method of analysis. With this method there is a sequence of forensic tasks that are designed to give the digital investigator critical information as quickly as possible. As Garfinkel states currently there is one commercial system using this approach by the name of IDEAL [3]. Unfortunately not many commercial or academic researches utilise this method but there is one system known as the System for Triaging Key Evidence(STRIKE) that IDEAL have developed. The technology is a hand-held forensics device that is designed to forensically analyse digital media [13]. The technology uses a touch screen display to show new information allocated to the user but the technology is not utilised in the digital forensic community [13].

The prioritized analysis mentioned above is based off the digital forensic triage model. Digital forensic triage is the acquisition process of partial forensic information from a device that needs to be acquired under severe time based or resource constraints [25]. The idea is that when given a certain time limit and or computational resource constraint that the user can acquire the most amount of pertinent data possible. As Roussev et al. states there is a finite amount of analysis that can be carried out on the target device but the goal of triage is to get to the most relevant information within these constraints. As Roussev et al. states the triage model is almost identical to the steps carried out in an early forensic investigation. The process aligns closely with what forensic analysts carry out on a device at the beginning of an investigation [25].

As Roussev et al. states it is well known in the field that the standard forensic workbench does not offer enough processing power to analyse a standard SATA hard disk drive. This issue causes problems for the triage model of analysis and coupled with the silent stumbling block of dealing with decompression across multiple file formats the triage method can fall short in a digital investigation [25]. As Roussev et al. points out the reality is that for a 23 terabyte hard drive it is only possible to access 10 to 15 percent of the data from the drive in under an hour. Roussev et al. conclude that after analysing multiple tools and methods of triage that there is a definite need for a systematic and automated approach to the triage model of digital investigation.

2.4 Information Retrieval

As Lillis et al. states information retrieval from a traditional standpoint is associated with identifying information to satisfy a users information query or need. Both academic and commercial information retrieval researchers face a trade-off between precision and recall when retrieving data [18]. Precision refers to retrieving only specific documents while recall refers to retrieving all relevant documents. As Lillis et al. states getting the balance between precision and recall can be difficult as improving on one metric usually results in the other having a performance reduction. As it stands for evidence to be forensically sound and court admissible recall is the most important metric for an investigation as missing a file or document can lead to serious legality issues for a case in motion [18].

Hitchcock et al. proposes a process model that expands on the triage method combined with offloading some basic forensic analyst functions to trained personnel. The idea is for front line investigators to receive basic training in forensic analysis to help maintain the forensic integrity of the digital evidence. As Hitchcock et al. states the trained members do not need to be dedicated digital forensic analysts but would have additional skills and training for the initial stages of a digital forensic investigation. The practice of a tiered model is already in practice in the Royal Canadian Mounted Police [16]. For example, lifting fingerprints from a crime scene is carried out by specially trained front line investigator and the fingerprint acquired is then analysed by a forensic identification specialist [16]. As Hitchcock et al. states the end result of information retrieval within a Digital Field Triage model is to increase current investigative efficiency and reduce the current backlog of digital evidence.

The current definition for the field triage process is defined by the investigative processes that take place in the first few hours of a case that leads to information for use in a suspect interview [16]. The process model usually involves on site analysis of a suspect device [16]. The model is broken into the 5 steps listed below.

- Acquire evidence immediately.
- Identify victim or victims at the most risk.
- Relay information for the ongoing investigation.
- Assess and identify potential charges.
- Address the information and assess the criminals danger to the public.

As Hitchcock et al. states digital evidence needs to be governed by the following three fundamental principles. Firstly, It is important for the evidence to be relevant to the investigation especially when the evidence is used for proving a specific point within a case [16]. Secondly, the evidence must be reliable to ensure the acquired digital evidence references the particular element of the case [16]. Thirdly, that the evidence is sufficient whereby the forensic collection holds enough digital evidence to allow the case to be examined efficiently [16]. As Hitchcock et al. states understanding this process is extremely important when time or cost is a concern for the digital investigation.

As Hitchcock et al. states, in the Triage model investigators and interviewers who are in contact with the suspect or witnesses need to provide the direct input to the digital forensic examiner. This is to ensure that the correct assumptions are made as this is usually the first time the digital investigator has been involved and has little or no case knowledge [16]. As Hitchcock et al. states the ISO 27037 international standard provides two defined positions to help in an investigation. These are defined as Digital Evidence First Responders who are

responsible for initial identification of suspect digital devices and Digital Evidence Specialists who are responsible for collecting the identified evidence. Although these two positions are defined clearly they do not help with the speed and acquisition of the evidence from the suspect device.

As Hitchcock et al. states the current Digital Field Triage model (DFT) is designed to provide non digital forensic specialists with the skills and abilities to conduct limited forensic activities. A DFT member identifies which items of digital evidence contain artefacts related to the offence under investigation [16]. For example, when an investigation relates to child pornography the DFT assessment would show if further analysis of the machine was needed if illicit images were located on the computer [16]. As Hitchcock et al. states the DFT member is able to state that they discovered illicit images on the machine but they do not have the skill set to provide information related to how the images got onto the machine or other important pieces of information only a trained forensic analyst could provide.

The proposed Digital Field Triage model from Hitchcock et al. follows four main stages. Firstly, in the initial stages of the forensic investigation the DFT member will provide assistance to the experienced digital investigator in the area of digital evidence [16]. When the investigation reaches the search warrant stage the DFT member then provides details of the search including details on the following.

Is it a mission-critical digital device that cannot tolerate any downtime? This question refers to if the digital evidence on the device is of critical importance for a case to proceed.

Is it within the DFT member's comfort zone? This question refers to the scope and skill set that the DFT has acquired. If the DFT is out of his or her comfort zone the experienced digital forensic investigator will take over the task.

What are the suspects abilities? This refers to the scope of the suspects crime and computer skills. If a computer is heavily encrypted or an investigation from a DFT shows that the suspect is the head of a large digital crime network the experienced digital forensic investigator would need take over the task.

What is the crime type being investigated? It is important to identify the type of crime early to know if the acquisition of evidence is time critical. For example, a current kidnapping or potential terrorist attack would be considered time critical as the digital evidence could stop a negative outcome.

As Hitchcock et al. states the DFT model provides a way to increase the efficiency of a digital investigation by giving digital investigators intelligence they can act upon when it is most needed. Actionable intelligence can range from identifying further areas within an investigation to providing digital artefacts that can be presented to a suspect [16]. Moreover, as the DFT member has been part of the digital investigation from the start the process becomes more streamlined as the DFT will know all important case knowledge [16].

As Hitchcock et al. states the first version of the DFT model was implemented in 2010 by a group of 20 digital forensic analysts. The digital forensic analysts supported approximately 8,500 employees that combined made up 127 police stations [16]. The DFT model implemented showed positive results and helped speed up the information retrieval digital investigations.

Information retrieval comprises of additional challenges when the data source is for an investigation is from the cloud. As Chen et al. states, regardless of the Cloud provider or the location of the ISP the laws and regulations of the country where the ISP located must be adhered to [10]. Moreover, if the data is encrypted this will add extra challenges as investigating encrypted information is a complex procedure. As Chen et al. states digital investigators

first have to gain access to the encryption keys before any information can be forensically analysed. It is possible for digital investigators to request access to the stored encryption keys from a cloud service provider or possibly obtain it manually by investigating the NetFlow and access logs [6]. As Almulla et al. and Garfinkel state information retrieval can also be affected negatively by the availability of massive storage for consumers which slows the indexing process and keyword searches dramatically.

As Garfinkel states various operating systems and file format options increases the requirements and complexity of information retrieval. Digital forensic cases in the past were commonly limited to the analysis of a single device, today cases require analysis of multiple devices and storage mediums [13]. The remote processing of data and storage means that only partial data can be found and sometimes no data can be found at all. Frequently, the local memory of RAID controllers, GPUs and network interfaces are routinely dismissed during digital forensic investigations [13]. As Garfinkel states these storage mediums can be utilised by users for illegal activity and ignoring them is not a good option.

Collange et al. explains that using existing data carving techniques and tools for information retrieval creates difficulty in recovering fragments of deleted illicit files. The main problem occurs when the file system metadata and file headers have been overwritten [11]. As Roussev et al. states robust data reduction techniques are needed to speed up processing rates and tackle the digital evidence backlog. This process is currently carried out by the digital investigator in an unsustainable way meaning the results are heavily dependent on the digital investigators experience [25].

2.4.1 Data Deduplication

The method of data deduplication is designed to improve data transfers over a network by reducing the amount of bytes that need to be sent. The method identifies byte or file chunk patterns that are accumulated during analysis. These file chunks are then compared to the known stored file chunks and only the unknown files are then captured and stored. With this method large amounts of data can be analysed quickly. As Roussev et al. states even with advanced methods of data deduplication the processing speed of current generation digital forensic tools is inadequate. The failure for developers and researchers to address the reliability and performance requirements have added to the ever growing backlog of digital evidence [25]. As Roussev et al. states digital investigators are often required to manually search through data that has little relevance to the investigation. A web searcher utilises methods similar to data deduplication as they sift out only the important information for an investigation rather than non-relevant material.

2.5 Digital Forensics as a Service(DFaaS)

A modern extension of the traditional digital forensic process is known as Digital Forensics as a Service (DFaaS) [18]. As Lillis et al. states DFaaS has been implemented in the Netherlands Forensic Institute (NFI) to tackle the volume of backlogged cases. DFaaS offers storage and automation solutions for the digital investigator. The advantages of DFaaS include resource management where general detectives can directly query the data [18]. As Lillis et al. states the improved resource management decreases the turnaround time between an investigator forming a hypothesis to achieving a confirmed time line of events based on the digital evidence. DFaaS makes collaboration on cases easier as detectives have easy access to shared knowledge.

Lillis et al. suggests that the current DFaaS system is a positive step but many improvements to the current model could be implemented. Improving the functionality available to the case detectives such as improving the indexing capabilities and incriminating evidence discovery during the acquisition process would offer a quicker result set for each case [18]. As the DFaaS model uses cloud storage some disadvantages can potentially occur [18]. As Lillis et al. states one such disadvantage is latency as the on-line platform is dependent on the upload bandwidth available during the physical acquisition of the investigation.

Watkins et al. and Lillis et al. describe a DFsaaS system that de-duplicates data which facilitates faster acquisition times as each unique file across a number of investigations only needs to be acquired once on the system. As Lillis et al. states eliminating non-pertinent data during the acquisition stage would greatly reduce the acquisition time. Examples of non-pertinent data include the operating system or local applications that come with the operating system [18]. As the non-pertinent information is removed from the initial acquisition the acquired pertinent information is available to detectives working on the case as early as possible [18].

Lillis et al. suggests adding the functionality to create a forensically sound disk image to the Teleporter tool. As Lillis et al. states for evidence to be court admissible a forensically sound disk image is needed. The idea would involve reconstructing a forensic copy of the drive from the de-duplicated data store [18]. Moreover, this system could then facilitate a cloud-to-cloud based monitoring system where only the changes acquired would be stored between each acquisition [18].

As Quick and Choo and van Baar et al. state an approach to dealing with data volume has been developed that includes feature extraction and analysis using various tools that creates an XML based output. The prototype system is called XML Information Retrieval Approach to digital forensics (XIARF) and is used to extract the metadata from forensic images [24]. This metadata is then stored in a database and can be accessed via a web interface. The XIRAF model has been implemented as part of a Digital Forensics as a Service (DFaaS) model in the Netherlands [24].

Quick and Choo and van Baar et al. point out features of the DFaaS implementation of XIARF. In the DFaaS model a team of support personnel take responsibility for administration of applications, databases and other necessary systems. The support personal are system administrators who are not trained in digital forensics [30]. The support personal have the ability to upload images to central a storage point, index files, and perform other tasks related to system administration [30]. All metadata including the timestamps are indexed and extracted from a suspect device and this information is stored in a keyword index. There are also separate administrators for application , database , storage and infrastructure tasks. Although these administrators are not digital forensic specialists they do have specific knowledge and skills to optimize systems and prevent data loss van Baar et al..

This model works as it frees up digital forensic specialists from administration to carry out specialised forensic tasks. As Quick and Choo states the DFaaS model in a real world scenario is beneficial to gather digital evidence early in an investigation. As van Baar et al. states the approach of DFaaS in the Netherlands has become a standard for hundreds of criminal investigations. The approach has is utilised by more than a thousand detectives on a regular basis [30].

As van Baar et al. states with the DFaaS model there is a centralized database of information that can be shared among all investigations. The centralizing of data comes with benefits such as backups and security settings that only need to be implemented once on the central server. The aforementioned is a benefit but more importantly when more than one department needs to investigate the same data it is possible to grant a digital investigator access to the case [30]. This process can be done in a matter of minutes rather than the old model where copies of the data needed to be created and distributed to each digital investigator which could take weeks [30].

As van Baar et al. and Quick and Choo discuss there has been a lot of digital forensic departments that have decreased backlogs with the DFaaS model. The performance in a digital investigation is more efficient and digital forensic investigators have time to perform in-depth research [30]. As van Baar et al. states traditional digital forensic procedures make it difficult for digital investigators and detectives to collaborate. Digital investigators report large result sets to multiple detectives, with a service model it is possible to analyse the relevant data for each case saving time in the investigation van Baar et al.. Moreover, the DFaaS model allows collaboration between detectives and digital investigators. This step is crucial for the correct digital evidence acquisition for a case and for the evidence to be understandable for a regular detective.

Although the DFaaS model has some key advantages it does also have disadvantages that should be addressed. The disadvantages are similar to those that occur in Software as a Service (SaaS) models [30]. One key disadvantage of a the DFaaS model is latency, as the model is dependent on utilising an internet connection [30]. As van Baar et al. states any DFaaS implementation needs to address this problem to ensure a working service.

2.6 Related Work

The following section gives an overview of some of the related work that this project is built upon. The importance of a data deduplication tool to speed up digital investigations is reiterated by Lillis et al. and Baggili et al. as they state the number of investigations requiring digital forensic expertise has increased. This is due to easy access to large storage mediums and smart devices that are available on the market [7] [6]. The huge digital evidence backlog encountered by law enforcement agencies shows the exponential increase in the number of cases requiring digital forensic analysis [18].

2.6.1 HPC and Parallel Processing

As Lillis et al. states high performance computing (HPC) has distinct advantages for digital forensic investigations and should be utilised wherever possible to reduce computation time when acquiring digital evidence. Implementation of HPC and parallel processing reduces the time required by digital investigators to work hands on with a device.

As noted by Lillis et al. the main technical bottleneck in many digital forensic investigations is the the disk read-speed. HPC and parallel processing does not address this issue but there are steps in the digital investigation process that are not limited by hard disk read-speed [18]. For example, the analysis phase has the potential to be extremely time consuming by both computers running the process and digital investigators, HPC can speed this process up. As Lillis et al. states traditional HPC techniques usually utilise parallelism but to date the digital forensic community have under utilised this option [18]. HPC techniques can be used for many applications and can expedite each part of a digital forensic investigation after the preprocessing, storage and reporting has taken place on a device [18].

2.6.2 Field Programmable Gate Arrays

As Lillis et al. states FPGAs are integrated circuits that are generally configured after the manufacturing stage. Moreover, FPGAs can utilize any function that application-specific integrated circuits (ASIC) can which offers some advantages when compared to a traditional CPU model. FPGAs can exploit low level algorithmic parallelism and often achieve the desired results with a smaller amount of logic operations compared a general purpose CPU [18]. The results of this inherent parallelism is faster processing times that can aid in speeding up digital forensic investigations [18]. Digital investigators have recently utilised FPGAs in many applications and positive results have been seen in imaging, video applications and and cryptography [18]. As Lillis et al. states desirable traits have been shown in the digital forensics field with FPGAs but digital forensic researchers have not exploited this technology for input/output facets of digital forensics. Moreover, as the common local storage medium moves to solid state drives over hard disk drives the input/output bottleneck will ease, FPGAs will therefore become more applicable in the digital forensics field.

2.6.3 GPU and Multi-threading

As Lillis et al. states graphics processing units (GPUs) offer the best results when executing single instruction and multiple data computations. Moreover, large numbers of stream processors can execute large threaded algorithms on multiple applications and in theory will work with many of the digital forensic investigation requirements [18].

As Lillis et al. states programming models with new heterogeneous architectures such as these offer powerful computer systems with added efficiency to digital forensic workstations. The workstations have transparent access to CPU virtual addresses and offer the digital investigator very low overhead for computation offloading [18]. As Lillis et al. shows the architectures utilising GPUs show distinct benefits with analytic processing. The reduction in I/O bottleneck and are beneficial for many digital forensic applications including solid state drive analysis [18]. The disk read speed does limit the benefit of using GPUs, although as Lillis et al. points out this conclusion assumes the digital investigator is following the traditional digital forensic model and only using one disk [18]. As Lillis et al. states the new era of cloud forensics and technological evolutions mean the I/O bottleneck will be much less restrictive meaning GPUs will be extremely beneficial to digital investigations.

Some interesting research was carried out in 2015 by Iacob et al. on GPUs and bloom filters. The research aimed to speed up the Bloom filter algorithm by using a Graphics Processing Units (GPU) based implementation [17]. As Iacob et al. states the research began by starting with regular CPU implementation of the Bloom filter algorithm. The two common Bloom filter operations of querying and mapping were tested using various optimization techniques [17].

Although it is algorithmically fairly simple a large percentage of computational load is taken up by the process of mapping and searching [17]. As Iacob et al. states increasing the speed of the mapping and searching operations would greatly improve computing performance and the speed of digital forensic investigations. As Iacob et al. states the GPU is similar to the CPU and as part of the kernel the GPU can launch in parallel with a very large number of threads. The threads on each kernel are organized on three levels, the first and lowest level of thread groups consist of 32 threads grouped into warps [17]. The second and middle level the threads are the same but the warps of threads are grouped into thread blocks [17]. As Iacob et al. states the third and highest level consists of the thread blocks organized into a 1D, 2D or 3D structure.

Applications that use Bloom filters consist of two main operations which are the insertion of each element into a set and the querying operation test performed for an elements membership [17]. The two main operations of the Bloom filter can run in parallel and as Iacob et al. states this parallelism can be exploited on a GPU.

Iacob et al. explain how a test was carried out with a CPU-GPU based hybrid Bloom filter implementation. Firstly, text files are read that are then used to generate Bloom filters where preprocessing techniques are applied. The text stream is split into words removing white space and punctuation marks [17]. This is followed by the removal of predefined stop words from the data to reduce the amount of data processing [17]. As Iacob et al. states stop words tend to have low semantic value as they typically are included in documents on a frequent basis. Moreover, the use of pre-defined stop words will not produce any loss of information [17].

As Iacob et al. states GPU based implementations can significantly outperform CPU based implementation. The GPU based implementation significantly speeds up the execution time of both the mapping and querying operations [17]. As Iacob et al. states the querying operation is the most important metric as the process is carried out online. Both Bloom

filter operations for GPUs show an increase in speed with mapping showing 300 times faster than CPUs and querying showing 20 times faster for GPUs over CPUs [17].

Power et al. explore the use of integrated GPUs and show a significant reduction in overheads when integrated GPUs are utilised. Power et al. specifically show that each integrated GPU is up to three times faster and up to three times lower in energy consumption than performance of a discrete GPU. As Collange et al. states the data cache of a CPU uses additional cycles to gain access to the data located in main memory of a bus. Moreover, the hash table of the GPU implementation is directly accessed without any intervening cache and a threaded GPU implementation shows improvements with data latency [11].

Power et al. show results that discrete GPUs are not a good choice for most workloads because of their limited memory capacity. Although GPUs are not mainstream in digital forensics or data analytics today, Power et al. argues that the situation will change in the future due to GPU hardware trends. Moreover, the general GPU architecture can be more energy-efficient than CPU architecture for certain workloads which will be of interest to the digital forensic field [20].

2.6.4 Hash Functions and Bytewise Approximate Matching

As Breitinger and Roussev states hash functions have a long tradition of been applied to many areas of computer science including digital forensics. In 2006 bytewise approximate matching came to the forefront of the digital forensic research community with an algorithm called context triggered piecewise hashing (CTPH) [9]. As Breitinger and Roussev states from the developments in 2006 a small community of computer scientists came up with similarity hashing which has been been published, researched and tested in various implementations showing some positive and negative outcomes.

The three most prominent approximate matching algorithms implementations are ssdeep, sdhash and mrsh-v2 [9]. As Breitinger and Roussev states a some further approximate matching algorithms algorithms have been designed but the majority of these implementations are very limited. For instance, MinHash and SimHash can only detect changes on a couple of bytes and mvHash-B is file dependent [9].

As Breitinger and Roussev states the ssdeep is widely accepted in the digital forensic community with the first implementation introduced as a proof of concept for context triggered piecewise hashing (CTPH). Breitinger and Roussev explains that ssdeep based on a spam detection algorithm, with the basic function is to split an input into smaller chunks. The chunks are then independently hashed and the individual chunks are then reassembled into a digital fingerprint[9]. As Breitinger and Roussev states incremental improvements have been attempted of the algorithm the implementations have not been made available to the public.

Breitinger and Roussev explains that the sdhash algorithm was introduced in 2010 to address some of the failures of ssdeep. The sdhash algorithm uses improbable features for each object instead of splitting the data into chunks like the ssdeep algorithm. The byte sequence is hashed using SHA-1 and then inserted into a Bloom filter [9]. As Breitinger and Roussev states sdhash also supports block mode where inputs are split into fixed-size chunks where the best features can be chosen from each block.

The mrsh-v2 algorithm introduced by Breitinger and Baier in 2012 is built upon the design ideas from the ssdeep and sdhash implementations. As Breitinger and Roussev states the objective of mrsh-v2 is to divide an input into chunks and hash each chunk based on ssdeep. These chunks are then combined with Bloom filters like sdhash for the similarity digest [9]. Breitinger and Roussev has introduced an open source, extensible framework called FRASH for systematic and reproducible testing of bytewise approximate matching. The current version provides facilities for evaluating three different aspects of an approximate matching algorithm's performance. [9]

2.6.5 The Sleuth Kit

The Sleuth Kit is a suite of analysis tools that run on all major platforms allowing access to common data types and a C library that offers functions to recover files and analyse those recovered files. The Sleuth Kit application is used in many commercial forensic tools and open source platforms. The official website [5] for The Sleuth Kit tool hosts the relevant documentation, version history and downloads for the latest versions of the software. The tool is user friendly as it is open source with an open format which allows users from academia and industry to learn from the tool.

The Sleuth Kit is designed to find as much information as possible. The information gathered may not be of use to a particular application but the user is required to filter out the data that is important for each case.

Chapter 3: **Design and Architecture**

3.1 Python

3.1.1 About

Python is a programming language that is developed under an open source license. The language is freely accessible to both academia and industry. Python is a programming language that offers multiple paradigms to the user. Some of the paradigms supported are object-oriented programming, structured programming and functional programming. The list is not extensive and many other paradigms can be implemented by using extensions. Features in Python consist of dynamic typing and a garbage collector for automatic memory management [12]. Functions and variable names are bound during program execution with dynamic name resolution and thousands of third-party modules are hosted in the Python Package Index (PyPI). The standard library, active community and third-party modules offer scope for creating robust digital forensic tools.

3.1.2 Benefits

There are many technical benefits of using python as the programming language for this project. As Prechelt states python is more productive than programming languages like C and Java especially when the program involves string manipulation and searching through a dictionary. Moreover, memory consumption is more streamlined in Python over Java [21]. As Garfinkel states it is simple to build automated forensic programs using Python and the python modules make it easy to access complex functions used in disk analysis.

3.1.3 Libraries Used

The program consists of two files which are server.py and client.py. Below is an overview of the libraries imported for the program.

-The following libraries are imported on both the server and the client.

import datetime: This library is imported to track the date and time of when the program is running on a suspect device.

import socket: This library is imported as both the client and server communicate via sockets.

import threading: This library is imported as the program works with multiple threads and functions within this library are required for each thread.

import json: This import converts data into JavaScript Object Notation for use with the MongoDB database.

from threading import Thread: This module is imported to access light weight process functions for multiple threads.

from Queue import Queue: As the program is multi-threaded this import allows the information from the client to be exchanged safely with the server for each thread.

-The following libraries are imported on the server.

import subprocess: As multiple processes run on the server the import allows creation and work flow of additional processes.

import os: The program needs to run on various operating systems and the import gives the program access to operating system dependent functionality.

import time: The time import is used to measure the acquisition speed and log the time taken in a csv file.

from pymongo import MongoClient: This import is used to create a MongoClient to store the metadata acquired for each file in a MongoDB database. (Explained further in the flow of events on page:)

-The following libraries are imported on the client.

import hashlib: This import provides access to secure hash and message digest algorithms for use within the program.

import pytsk3: The pytsk3 import is a Python binding for the popular analysis tool Sleuth Kit. This allows the program direct access to the forensic functions openly available on the Sleuth kit website [5].

import csv: This import is used in conjunction with the time import to output test results to a csv file.

import pyvhdi: The pyvhdi import is used to enable the program to run on virtual hard disk (vhd) files.

3.2 MongoDB

3.2.1 About

MongoDB is a database application that utilises not only structured query language (NoSQL). MongoDB offers the best of what relational databases offer with additional flexibility, scalability and performance required for today's big data applications. MongoDB is not expensive to operate and with the scalability and reliability across the globe it is the go to option for many enterprise applications. MongoDB is open source software allowing industry and academics access to the technology for free.

NoSQL utilises database technologies developed in direct response to the storage and reliability needs of most modern applications. Applications developed today commonly create very large volumes of new data that can be structured, unstructured or polymorphic data. Applications are commonly delivered as a service (SaaS) that run continuously and globally to millions of users. Cost reduction and physical storage space is made available for organizations as there is no need to keep large servers or storage infrastructure. The cloud is now the go to solution and MongoDB is one of the leaders of the NoSQL database field.

3.2.2 Benefits

As mentioned above NoSQL databases offer more scalability and provide better performance all round especially when dealing with large quantities of different data such as structured and unstructured data. The MongoDB developers work in agile sprints using small and frequent code iterations to update the MongoDB technology. This means there is no downtime on the database while updates occur. For this project the data acquisition stage is very important as a case maybe time sensitive the tool cannot afford to have a bottleneck due to a server update.

3.2.3 Advantages for the Data Deduplication Tool

The long term objective of the program is to have a central repository of known files that can be accessed by multiple digital investigators in various global locations. When a file is not in the repository the file will be uploaded to the repository updating the known files for ongoing investigations. The current model of the project stores common files of an OS which speeds up the acquisition of the disk as these files do not need to be acquired for each test.

The database service that houses this repository must be reliable, scalable and cost effective. MongoDB offers all of these features and more so it makes a perfect solution to the back end of the project. Qi et al. describe some of the key features that MongoDB offers such as handling big data replication and distribution of data over multiple servers. Moreover, as both Qi et al. and Qi state MongoDB is extremely proficient at dealing with faults and server failures. This redundancy is key to the data deduplication tool as digital investigations can be time sensitive and the the server cannot afford to fail.

3.3 Data Deduplication Program and Flow of Events

This section describes the steps shown in the architecture and flow of events diagrams. The digital investigation begins in a typical style where an investigators workstation is used to run a program to analyse the suspect device. The suspect device will show the locations and artefacts of files on that disk that are then investigated further.

These artefacts are then sent to the MongoDB database where the SHA-512 hash for each file is compared to the known files on the database. If a match is found the file is not sent to the database as it is already stored there. During testing the files stored in the MongoDB database are mostly common system files that come from the initial acquisition of each operating system. For example, when a disk with Windows 8 is analysed for the first time the entire disk will be acquired including all system files and folders. When another disk with Windows 8 is analysed the MongoDB database will compare the files by their hash values and the common files will not be acquired again speeding up the acquisition of that disk.

If no match is found for a file, the metadata of that file is then sent to the MongoDB database and stored in a collection. The current iteration of the program is running on one machine which will be explained below. The final implementation of program would have the MongoDB database utilised as a DFaaS Cloud System for global access for all investigators to compare files.

The current flow of events starts with running MongoDB from the command line by issuing the command mongod. This command opens up connections for the information to be acquired to the database. Following on from this the server.py file is run from the command line using the command python server.py. When the server runs it attempts to connect to the client and shows the amount of attempts made. The client.py file is required to run for successful communication to be made between the server and client. When the client is running and no errors occur the client accepts the connection. The program has been tested on both dmg and vhd image files with positive results shown in the experimentation and results section. The primary function for this project is for the program to run on Windows vhd disk images.

The metadata of each file is acquired and this metadata is then sent to the MongoDB database. When the program runs for the first time the database is created and two collections are added to this database. MongoDB houses no settings or pre-defined databases until the program has completed running a test on a vhd image for the first time. The two collections are named metadata acquisition and server files. The metadata acquisition holds all files for each disk analysis and the server files contains only the de-duplicated files. In the experimentation and results section the benefits of this method will be explained further.

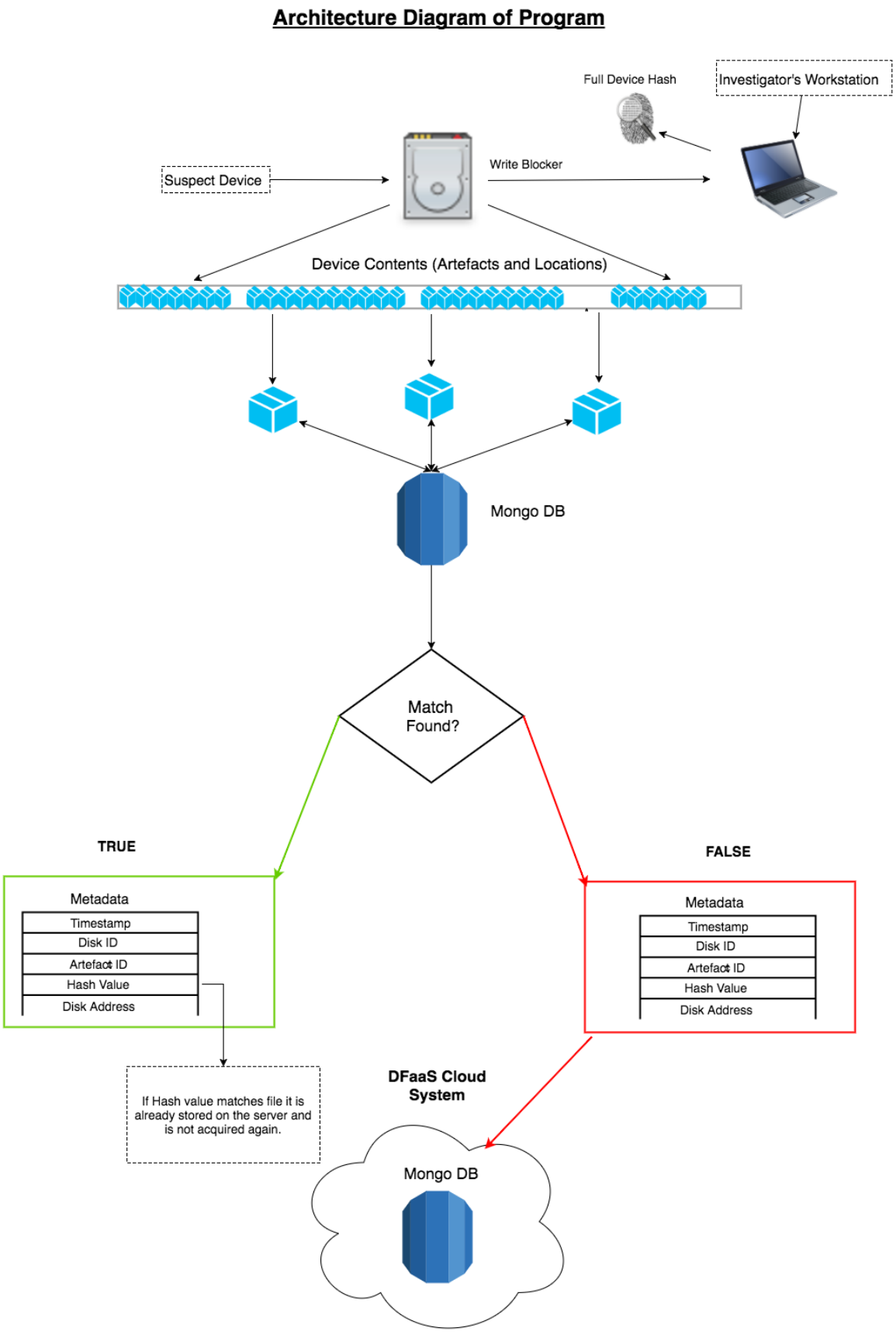

Figure 3.1: Architecture Diagram

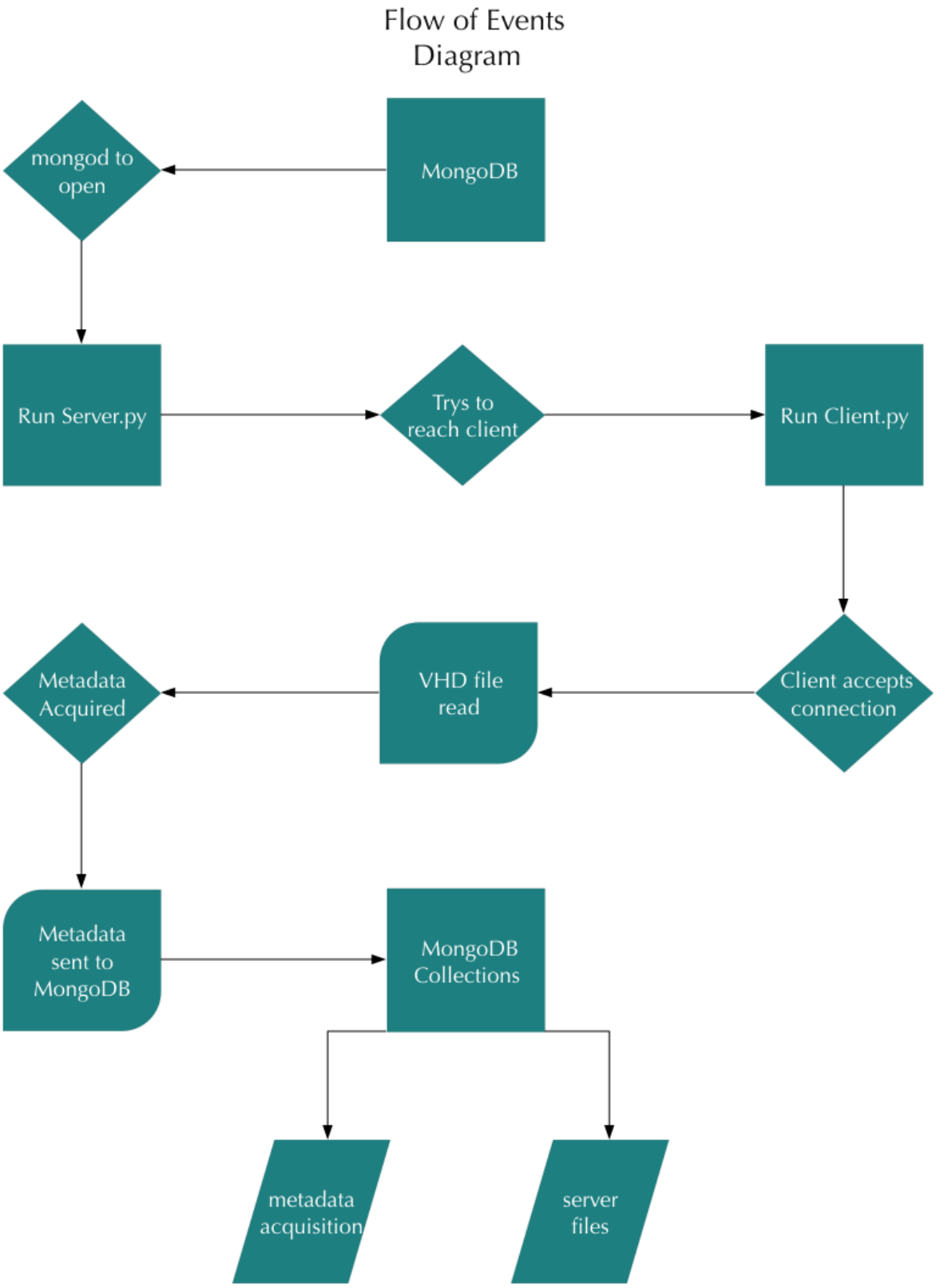

Figure 3.2: Flow of Events Diagram

Chapter 4: **Experimentation and Results**

4.1 Testing

The following chapter displays the results of the testing carried out on the data deduplication tool. The initial tests were carried out on base images including Windows 7, 8, 10 and Ubuntu. Additional changes were made to each machine for further tests outlined below. The server and client is on the same machine for all tests which may effect the throughput figures as the hard disk is been read from and written to at the same time. The specifications for the machine that all tests were carried out on is listed below.

Hardware Overview of machine for all tests

- Model Name: MacBook Pro
- Model Identifier: MacBookPro9,2
- Processor Name: Intel Core i5
- Processor Speed: 2.5 GHz
- Number of Processors: 1
- Total Number of Cores: 2
- L2 Cache (per Core): 256 KB
- L3 Cache: 3 MB
- Memory: 4 GB
- Boot ROM Version: MBP91.00D3.B0D
- SMC Version (system): 2.2f44
- Serial Number (system): C02HM28UDTY3
- Hardware UUID: 895798D0-E6FE-5B2C-8D68-68C7016C3752
- Sudden Motion Sensor:
- State: Enabled
- 1 Terabyte HDD

4.2 Initial Test - Metadata Only Acquisition

For the initial testing phases each disk image is a fresh install of an operating system with no additional logins or application downloads from a user. For each test the MongoDB database is cleared to ensure acquisition times and amount of files acquired in the database are correct. The initial test is carried out on Windows 7, 8, 10 and Ubuntu but all following tests are carried out only on the Windows operating systems. Listed below are the metadata acquisition times and amount of files acquired on disk.

Windows 7

- Time to Acquire Files: 8 minutes 40 seconds
- Number of Files Acquired: 48,405

Windows 8

- Time to Acquire Files: 17 minutes 14 seconds
- Number of Files Acquired: 81,150

Windows 10

- Time to Acquire Files: 20 minutes
- Number of Files Acquired: 106,569

Ubuntu

- Time to Acquire Files: 13 minutes
- Number of Files Acquired: 211,828

4.2.1 Time to Acquire Metadata

The bar chart below 4.1 shows amount of time in minutes and seconds for the program to acquire the file metadata for each operating system. The Windows operating systems follow a steady increase in acquisition time which corresponds to the amount of files shown in the diagram of section 4.2. The Ubuntu operating system does not follow the time or file patterns of the Windows operating systems. The specific times for each metadata acquisition are listed below.

- Win 7: 8 minutes 40 seconds
- Win 8: 17 minutes 40 seconds
- Win 10: 20 minutes
- Ubuntu: 13 minutes

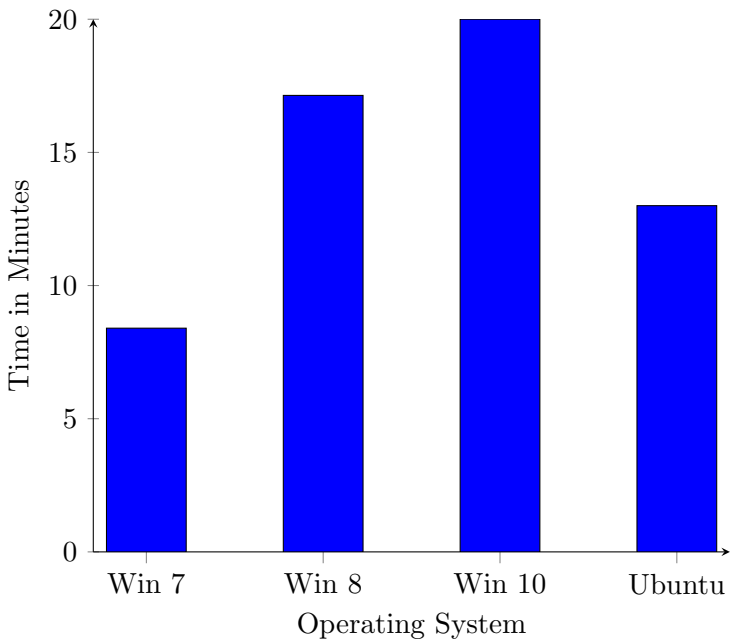

Figure 4.1: Initial Metadata Acquisition - Time

4.2.2 Amount of Files Acquired

The bar chart below 4.2 shows the amount of files acquired for each initial metadata acquisition on the tested operating systems. The Windows operating systems show a gradual and predictable increase in the amount of files per upgrade. The Ubuntu operating system shows a massive increase in files, almost twice the amount as acquired from the Windows 10 disk image. This is due to the Ubuntu operating system having more drive partitions than the Windows operating systems. Interestingly the metadata acquisition for Ubuntu is 7 minutes quicker than that of Windows 10 even though the Ubuntu acquisition contains 105,259 more files than Windows 10. The file count for each operating system is listed below.

- Win 7: 48,405 files
- Win 8: 81,150 files
- Win 10: 106,569 files
- Ubuntu: 211,828 files

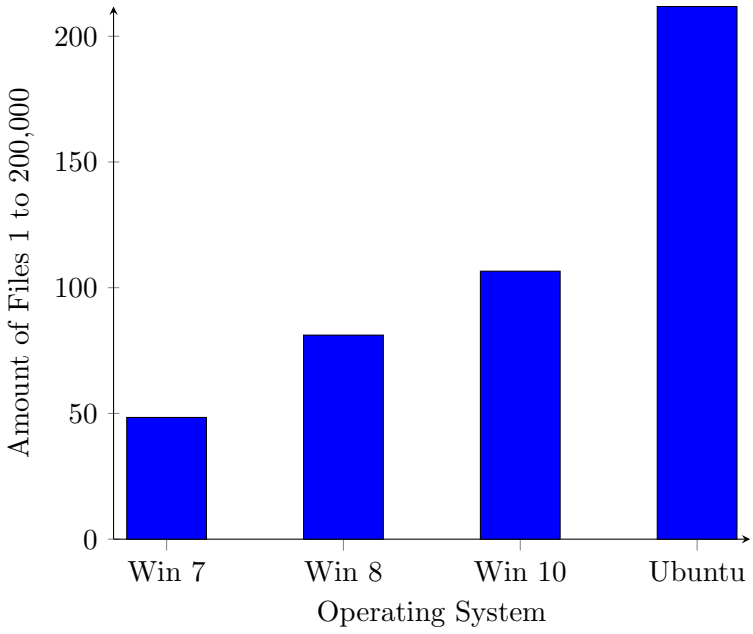

Figure 4.2: Initial Metadata Acquisition - Time

4.3 Test 2 - Base Images

Test 2 is the primary test that runs the program from start to finish for the first time. The test acquires all metadata for each file on each Windows operating systems and then sends the files to the MongoDB database. The files are stored in the server files collection in MongoDB and are not deleted for each new acquisition. The Windows 7 acquisition sends 41,887 files to the server, followed by Windows 8 sending an additional 73,837 files to the server and finally Windows 10 sends an additional 67,054 files. When the program runs on all three Windows operating systems the server then hosts 182,778 files. The statistics represent the same information for each test but the results vary for each test. Below is a brief overview of what each statistic refers to.

Statistics Overview

- Reading Time: refers to the amount of time to read the disk.
- Files on Disk: refers to the amount of files on the disk.
- File Size on Disk: refers to the file size in bytes on the disk.
- Files Sent to Server: refers to the amount of files sent to the server for storage.
- Total Bytes Sent to Server: refers to the total bytes sent to the server.
- Actual System Throughput (Mb/sec): refers to how many seconds each megabyte takes to send to the server.
- Effective System Throughput (Mb/sec): refers to how many seconds it would take if all files needed to be sent to the server in megabytes per second.

The results below show what is obtained from the csv log file after the program runs on each operating system.

Windows 7

- Reading Time: 2445.103141
- Files on Disk: 48,405
- File Size on Disk: 9060415863
- Files Sent to Server: 41,887
- Total Bytes Sent to Server: 7234126335
- Actual System Throughput (Mb/sec): 2.82
- Effective System Throughput (Mb/sec): 3.53

Windows 8

- Reading Time: 5038.627798
- Files on Disk: 81,138
- File Size on Disk: 13330778403
- Files Sent to Server: 73,837
- Total Bytes Sent to Server: 12225915232
- Actual System Throughput (Mb/sec): 2.313923645
- Effective System Throughput (Mb/sec): 2.523107582

Windows 10

- Reading Time: 3795.837482
- Files on Disk: 106,010
- File Size on Disk: 15401286861
- Files Sent to Server: 67,054
- Total Bytes Sent to Server: 9003133413
- Actual System Throughput (Mb/sec): 2.261951425
- Effective System Throughput (Mb/sec): 3.869238362

4.3.1 Files on Disk

The bar chart below 4.3 is showing the amount of files on each disk as output from the csv log file. The newer the operating system the more files that are on the disk image and this expected behaviour is displayed below 4.3. The Windows 8 image hosts 32,733 files more than Windows 7 and Windows 10 hosts an extra 24,782 files more than Windows 8. Listed below are the amount of files for each disk image.

- Win 7: 48,405 files
- Win 8: 81,138 files
- Win 10: 106,010 files

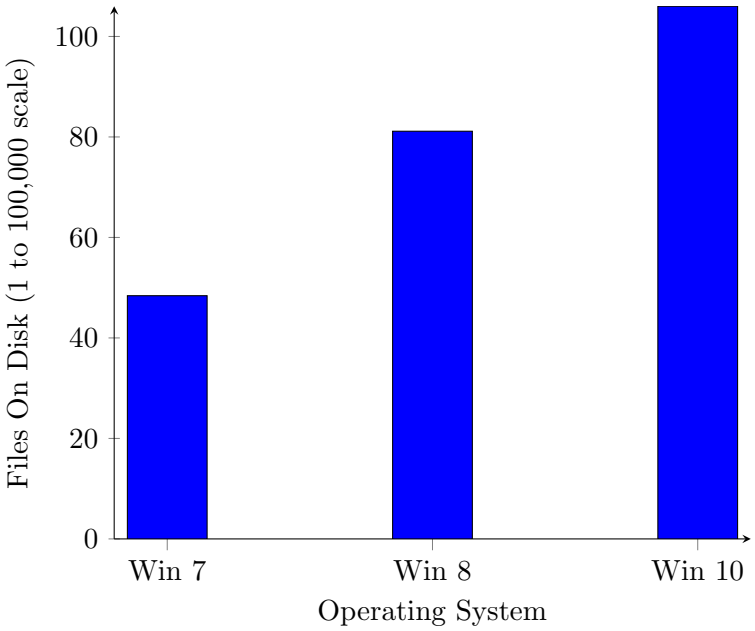

Figure 4.3: Files on Disk

4.3.2 Files Sent to Server

The bar chart below 4.4 compares the amount of files sent to the server from each operating system image acquisition. It is interesting to note that not all files that have been acquired are sent to the server. Moreover, some of the system files between operating systems overlap meaning those files will not be sent to the server. This can be seen in 4.4 as the Windows 8 image sends 6,783 more files to the server than Windows 10. The total files sent to the server for each disk image is listed below.

- Win 7: 41,887 files sent
- Win 8: 73,837 files sent
- Win 10: 67,054 files sent

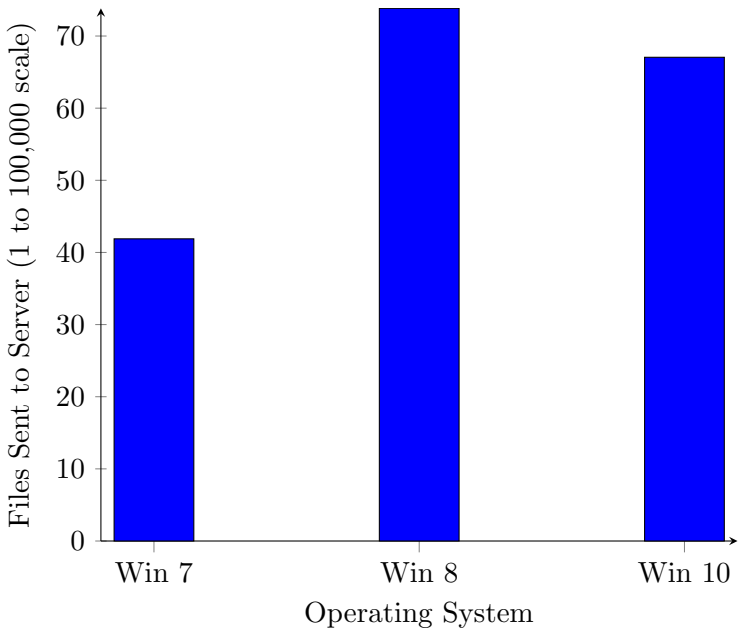

Figure 4.4: Files Sent to Server

4.4 Test 3 - Base Images Second Time No Changes

Test 3 is very important as it is the first test to show a clear result set of the data deduplication process. The test consists of acquiring the base images for each operating system with no changes for a second time. The test results below prove the deduplication program works with great improvements on all statistics. The most important statistics to note for this test are Files Sent to Server, Total Bytes Sent to Server and the actual/effective system throughput. Both Windows 8 and Windows 10 show no files were sent to the server which means by default no bytes were sent to the server and the actual system throughput was also zero. There is an anomaly with Windows 7 as 16 files were sent to the server. This means either 16 files had changed between test 2 and test 3 or the initial file sending from test 2 did not include those 16 files. Regardless of this anomaly only 16 files were sent to the server over the initial base image acquisition of Windows 7 which sent 41,887 files to the server. Listed below are the statistics for test 3.

Windows 7

- Reading Time: 1213.44966
- Files on Disk: 48,400
- File Size on Disk: 9060403752
- Files Sent to Server: 16
- Total Bytes Sent to Server: 15136008
- Actual System Throughput (Mb/sec): 0.011537355
- Effective System Throughput (Mb/sec): 7.120196481

Windows 8

- Reading Time: 1412.515974
- Files on Disk: 81,138
- File Size on Disk: 13330778403
- Files Sent to Server: 0
- Total Bytes Sent to Server: 0
- Actual System Throughput (Mb/sec): 0
- Effective System Throughput (Mb/sec): 9.000252198

Windows 10

- Reading Time: 1983.915414
- Files on Disk: 106,010
- File Size on Disk: 15401286861
- Files Sent to Server: 0
- Total Bytes Sent to Server: 0
- Actual System Throughput (Mb/sec): 0
- Effective System Throughput (Mb/sec): 7.403037396

4.4.1 Files on Disk compared to Files Sent to Server

The bar chart below 4.5 compares the amount of files on each disk to the files sent to the server from the second base image acquisition. Both Windows 8 and Windows 10 send no files to the server on the second acquisition. As mentioned above Windows 7 sends 16 files to the server for this test. F.S on the x axis of the bar chart 4.5 stands for Files Sent and the number following each F.S refers to the operating system.

- Win 7: 48,400 files on disk
- Win 7: 16 files sent to server
- Win 8: 81,138 files on disk
- Win 8: 0 files sent to server
- Win 10: 106,010 files on disk
- Win 10: 0 files sent to server

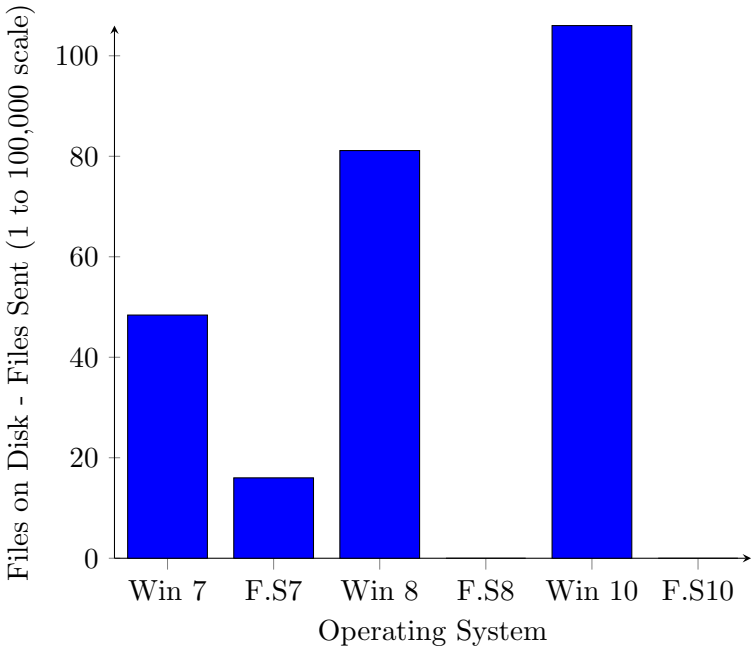

Figure 4.5: Files on Disk compared to Files Sent to Server

4.5 Test 4 - Logins

Test 4 shows the additional files that are created on each operating system after a simple login and shutdown. The result set is important as it shows that only the new files acquired from the login and shutdown are sent to the server. Windows 7 shows that 151 new files were acquired and sent to the server while Windows 8 only shows 12 new files and Windows 10 shows 1,597 files. Listed below are the statistics for test 4.

Windows 7

- Reading Time: 1009.117125
- Files on Disk: 48,388
- File Size on Disk: 8654759505
- Files Sent to Server: 151
- Total Bytes Sent to Server: 268871660
- Actual System Throughput (Mb/sec): 0.253687103
- Effective System Throughput (Mb/sec): 8.178436177

Windows 8

- Reading Time: 1320.336839
- Files on Disk: 81,239
- File Size on Disk: 13446132381
- Files Sent to Server: 12
- Total Bytes Sent to Server: 150609956
- Actual System Throughput (Mb/sec): 0.108305696
- Effective System Throughput (Mb/sec): 9.711915643

Windows 10

- Reading Time: 2739.772599
- Files on Disk: 12,6764
- File Size on Disk: 15896700213
- Files Sent to Server: 1,597
- Total Bytes Sent to Server: 1145755942
- Actual System Throughput (Mb/sec): 0.398573225
- Effective System Throughput (Mb/sec): 5.533305941

4.5.1 Files Sent to Server After Initial Login

The bar chart below 4.6 shows the amount of files sent to the server from the login disk acquisition. The bar chart 4.6 shows that Windows 10 contains more files than both Windows 7 and Windows 8 from an initial login and shutdown process. More importantly 4.6 shows that only the changed files have been sent to the database further proving the functionality of the data deduplication tool.

- Win 7: 151 files sent to server
- Win 8: 12 files sent to server
- Win 10: 1,597 files sent to server

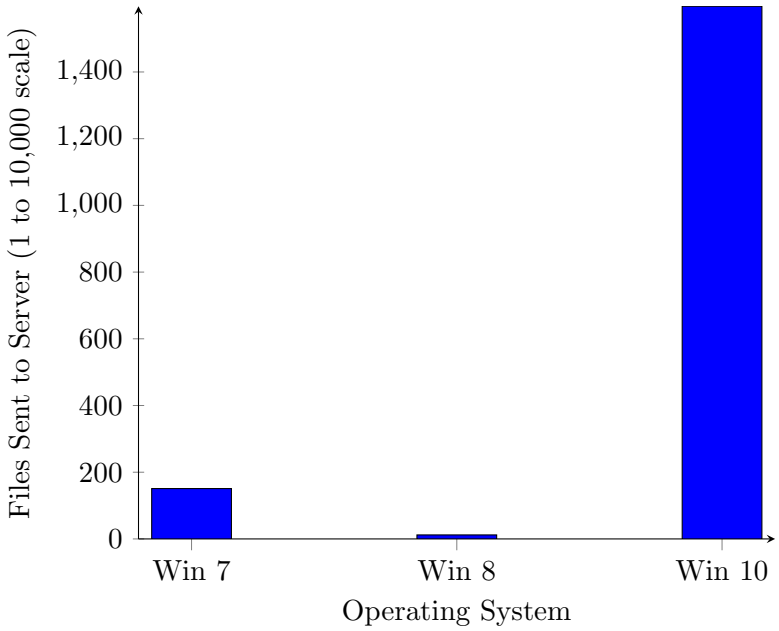

Figure 4.6: Files Sent to Server After Login and Shutdown

4.6 Test 5 - Text Document on Desktop

Test 5 consists of a real world scenario whereby a text document is created on a users machine. The text document is created from the desktop on each operating system by right clicking the mouse on the desktop and selecting to create a new text document from the drop down menu. The text file on each operating system is populated with the text "test, testing, tested". The file is saved to the desktop and the operating system is then shutdown. The test shows the deduplication program working in a similar manner to test 3 and 4. The results show savings in throughput of just under 9Mb/sec for Windows 7, 10Mb/sec for Windows 8 and just over 8Mb/sec for Windows 10. Listed below are the statistics for test 5.

Windows 7

- Reading Time: 931.135983
- Files on Disk: 48,464
- File Size on Disk: 8963829542
- Files Sent to Server: 120
- Total Bytes Sent to Server: 253876520
- Actual System Throughput (Mb/sec): 0.259897592
- Effective System Throughput (Mb/sec): 9.180184373

Windows 8

- Reading Time: 1280.591116
- Files on Disk: 81,192
- File Size on Disk: 13584826864
- Files Sent to Server: 10
- Total Bytes Sent to Server: 150590564
- Actual System Throughput (Mb/sec): 0.111667181
- Effective System Throughput (Mb/sec): 10.11642189

Windows 10

- Reading Time: 1782.182517
- Files on Disk: 121,625
- File Size on Disk: 16125451400
- Files Sent to Server: 102
- Total Bytes Sent to Server: 311639938
- Actual System Throughput (Mb/sec): 0.166649598
- Effective System Throughput (Mb/sec): 8.628745851

4.6.1 Files Sent to Server after Text File Creation

The bar chart below shows the amount of files sent to the server after a text document was created for each disk acquisition. The bar chart 4.7 is similar to 4.6 as it shows that Windows 10 contains more files than both Windows 7 and Windows 8. This similarity suggests that Windows 7 and Windows 8 share common files after a login and text document is created. More importantly 4.7 shows that only the changed files have been sent to the database further proving the functionality of the data deduplication tool.

- Win 7: 120 files sent to server
- Win 8: 10 files sent to server
- Win 10: 102 files sent to server

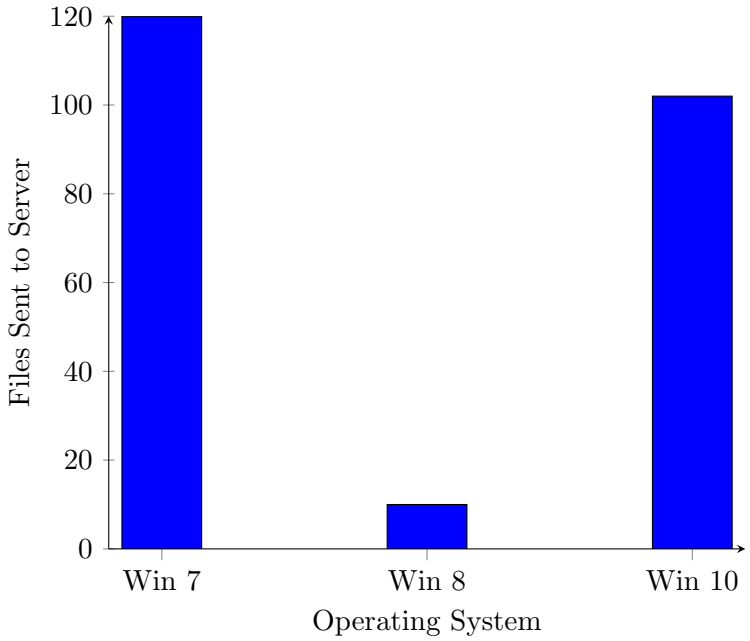

Figure 4.7: Files Sent to Server after Text File Creation

4.7 Test 6 - Multiple Application Installs

For the final test multiple application installs have been made on each operating system. All tests before test 6 involved incremental changes to show the program functionality. This test aims to show a more realistic overview of how the tool would work in a real world digital investigation. Each operating system consists of some applications that are unique to that machine while others are installed on all machines. The test maps the real world process as some machines will have the same applications while others will not. The tables below 4.8 and 4.9 show the list of applications and crossovers installed on each operating system. The letter Y in the tables means the application is installed.

Application List			
Application	Windows 7	Windows 8	Windows 10
Chrome	Y		
Firefox	Y	Y	Y
Skype	Y		
Yahoo	Y		Y
VLC	Y		
Silverlight	Y	Y	
qBittorrent	Y		
Malwarebytes	Y		
Dropbox	Y		
Google Drive	Y		
One Drive	Y		
Evernote	Y		
7-Zip	Y		
Python	Y		
Thunderbird	Y		
Java 8	Y		
PDF Creator	Y		
Team Viewer	Y		
Opera		Y	
AIM		Y	
Trillian		Y	
Air		Y	
Sugar Sync		Y	
BiTTorrent Sync		Y	Y
Filezilla			Y
Imgburn			Y
Spybot 2			Y
Mozy			Y
Peazip			Y
Application Total	18	8	7

Figure 4.8: Application List

Application Crossover			
Application	Windows 7	Windows 8	Windows 10
Firefox	Y	Y	Y
Silverlight	Y	Y	
BitTorrent Sync		Y	Y
Yahoo	Y		Y

Figure 4.9: Application Crossover

4.7.1 Application Crossover

The table 4.9 shows that Firefox is installed on all three operating systems, Silverlight is installed on Windows 7 and Windows 8, BiTTorrent Sync is installed on Windows 8 and Windows 10 and finally Yahoo is installed on Windows 7 and Windows 10. The data deduplication program ran on Windows 7 first followed by Windows 8 and then Windows 10.

Therefore any application acquired and sent to the server on Windows 7 did not need to be sent to the server again. For example, the metadata for the Firefox install on Windows 7 was found on the server when Firefox metadata from Windows 8 and Windows 10 was compared to it. The server has the file already and does not need it again. Listed below are the statistics for test 6.

Windows 7

- Reading Time: 3443.24244
- Files on Disk: 57,553
- File Size on Disk: 11038523166
- Files Sent to Server: 9,877
- Total Bytes Sent to Server: 2701828754
- Actual System Throughput (Mb/sec): 0.7481320427
- Effective System Throughput (Mb/sec): 3.057292707

Windows 8

- Reading Time: 3084.066313
- Files on Disk: 82,558
- File Size on Disk: 14539151992
- Files Sent to Server: 1,341
- Total Bytes Sent to Server: 1383024251
- Actual System Throughput (Mb/sec): 0.4273578666
- Effective System Throughput (Mb/sec): 4.495688028

Windows 10

- Reading Time: 2826.924997
- Files on Disk: 109,139
- File Size on Disk: 15901510424
- Files Sent to Server: 1,566
- Total Bytes Sent to Server: 616191534
- Actual System Throughput (Mb/sec): 0.2076461175
- Effective System Throughput (Mb/sec): 5.364132411

4.7.2 Files Sent to Server - Multiple Applications

The bar chart below shows the amount of files sent to the server after multiple application installs on each operating system. Figure 4.10 shows that only the changed files have been sent to the database further proving the functionality of the data deduplication tool.

- Win 7: 9,877 files sent to server
- Win 8: 1,341 files sent to server
- Win 10: 1,566 files sent to server

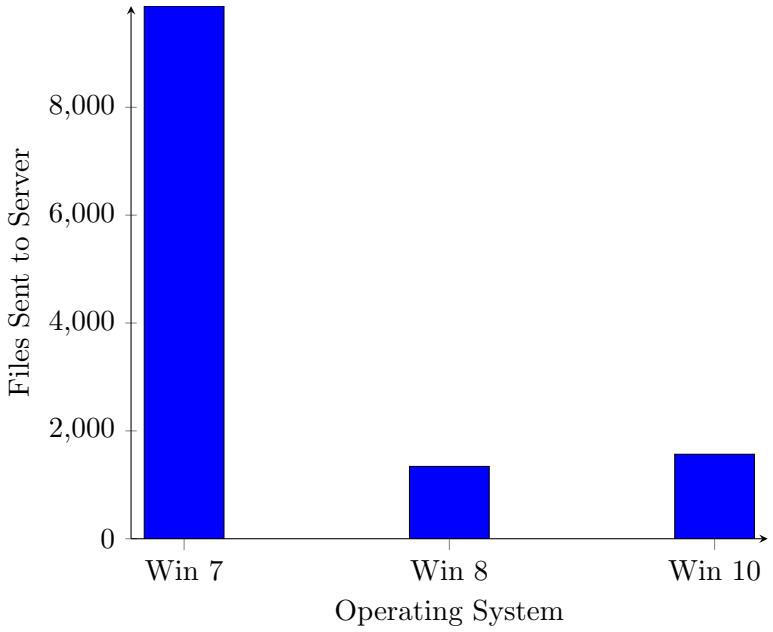

Figure 4.10: Files Sent to Server after Text File Creation

4.7.3 Actual and Effective System Throughput

The bar chart below 4.11 compares the actual and effective throughput for file sending. The actual throughput is always less than the effective throughput proving that the data deduplication tool speeds up the acquisition time for each operating system. The x axis of the bar chart 4.11 consists of each operating system represented by W7, W8 or W10. AST stands for actual sytem throughput and EST stands for effective system throughput.

- Win 7: Actual System Throughput (Mb/sec): 0.7481320427
- Win 7: Effective System Throughput (Mb/sec): 3.057292707
- Win 8: Actual System Throughput (Mb/sec): 0.4273578666
- Win 8: Effective System Throughput (Mb/sec): 4.495688028
- Win 10: Actual System Throughput (Mb/sec): 0.2076461175
- Win 10: Effective System Throughput (Mb/sec): 5.364132411

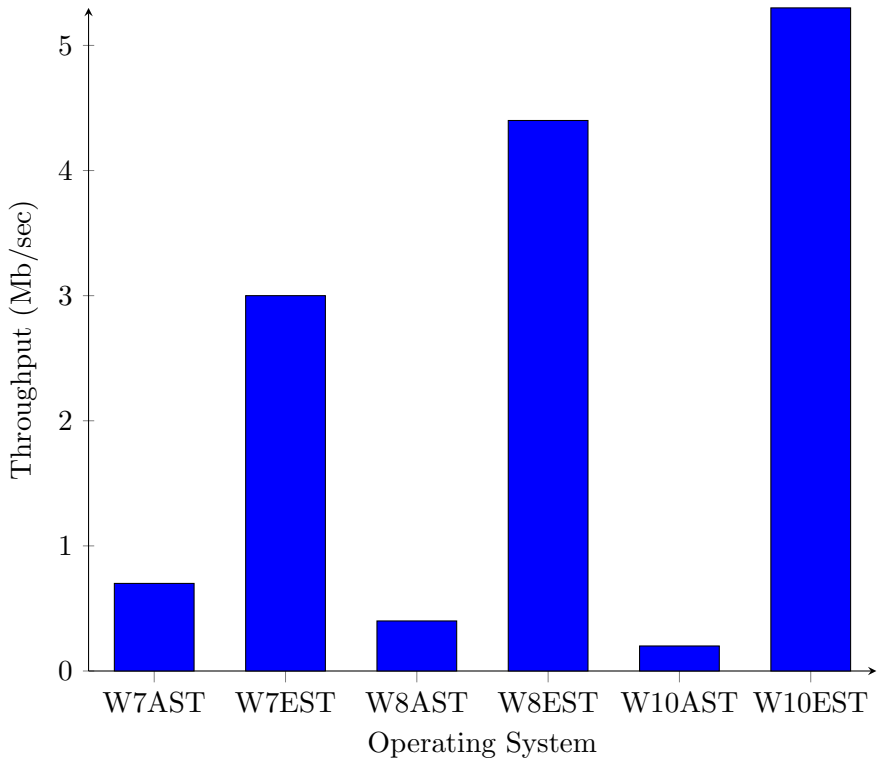

Figure 4.11: Throughput

Chapter 5: **Conclusion and Future Work**

5.1 Future Work

The first steps for future work should be to test the data deduplication tool on Mac OS and Ubuntu. Following these tests more robust real world testing should be carried out similar to test 6 4.10 4.11. The data deduplication tool is designed to run multiple clients at the same time so it is important to test this scenario before releasing the tool to industry. To model real world usage a test should be carried out using a remote server and multiple client machines to test for real world performance.

A GUI and and documentation of the data deduplication tool should be developed to assist digital investigators in implementing the tool in their investigations. Following on from this industry partners should be found to help role out large scale testing. Recently, both the FBI and National Institute of Standards and Technology (NIST) have shown a keen interest in real world testing of the data deduplication tool. Moreover, NIST has offered resources and remote storage solutions for real world testing.

Currently, the tool runs with a free version of MongoDB and while very efficient with low level testing further robustness should be implemented. MongoDB Atlas is a premium service that offers scalability, security and disaster recovery to a database. These features are important as the database will exponentially grow from each disk acquisition. The test results of the tool show great promise in tackling the digital forensic backlog and future work could include testing the tool to see how quickly it can decrease the digital forensic backlog of an individual police department.

5.2 Conclusion

The research question proposed, "how can the digital forensic investigation process be quicker?" was tackled in this body of work. Specifically, the thesis tackles the problem of digital forensic investigators acquiring all data for each new digital investigation they perform. To answer this question and offer a solution to the problem detailed research was carried out on the current digital forensic tools and methods used for digital investigations. From this research a data deduplication tool was developed to test on multiple operating systems. To speed up the digital investigation process the tool was designed to acquire a disk image and search for duplicated data. If duplicated data was found the tool stopped this data from been sent to the database.

The test results of the tool prove that the data deduplication tool speeds up the digital forensic process. The Files Sent to Server and Actual/Effective System throughput metrics in the testing chapter show that the time and amount of files both decrease when an acquisition is performed with the data deduplication tool. The data deduplication tool has shown to improve the digital forensic process in an academic setting. Moreover, the data deduplication tool has the potential to make a positive impact in the digital forensic community and offers a real world solution to the problems faced by digital forensic experts in industry.

Bibliography

[1] External Hard Drive walmart prices. http://www.walmart.com/search/?refineresult=true&search_query=external+hard+drive&ic=48_0&search_constraint=0&cat_id=3944_3951_1073804_514537&facet=hard_drive_capacity%3A3.0TB+%26+Above&sort=price_high, . Accessed: 2016-08-10.

[2] Applesdk apple. https://developer.apple.com/news/?id=04222016a, . Accessed: 2016-08-10.

[3] IDEAL strike tool. http://www.idealcorp.com/products/index.php?product=STRIKE. Accessed: 2016-08-10.

[4] Digisole smart shoes. http://www.digitsole.com/warm-series-product-presentation/. Accessed: 2016-08-10.

[5] Sleuthkit sleuthkit. http://www.sleuthkit.org/. Accessed: 2016-08-10.

[6] S. Almulla, Y. Iraqi, and A. Jones. Cloud forensics: A research perspective. In *Innovations in Information Technology (IIT), 2013 9th International Conference on*, pages 66–71, March 2013. doi: 10.1109/Innovations.2013.6544395.

[7] I. Baggili, J. Oduro, K. Anthony, F. Breitinger, and G. McGee. Watch what you wear: Preliminary forensic analysis of smart watches. In *Availability, Reliability and Security (ARES), 2015 10th International Conference on*, pages 303–311, Aug 2015. doi: 10.1109/ARES.2015.39.

[8] Frank Breitinger and Harald Baier. Similarity preserving hashing: Eligible properties and a new algorithm mrsh-v2. pages 167–182, 2013. doi: 10.1007/978-3-642-39891-9_11. URL http://dx.doi.org/10.1007/978-3-642-39891-9_11.

[9] Frank Breitinger and Vassil Roussev. Automated evaluation of approximate matching algorithms on real data. *Digital Investigation*, 11, Supplement 1:S10 – S17, 2014. ISSN 1742-2876. doi: http://dx.doi.org/10.1016/j.diin.2014.03.002. URL http://www.sciencedirect.com/science/article/pii/S1742287614000073. Proceedings of the First Annual {DFRWS} Europe.

[10] G. Chen, Y. Du, P. Qin, and J. Du. Suggestions to digital forensics in cloud computing era. In *2012 3rd IEEE International Conference on Network Infrastructure and Digital Content*, pages 540–544, Sept 2012. doi: 10.1109/ICNIDC.2012.6418812.

[11] S. Collange, Y. S. Dandass, M. Daumas, and D. Defour. Using graphics processors for parallelizing hash-based data carving. In *System Sciences, 2009. HICSS '09. 42nd Hawaii International Conference on*, pages 1–10, Jan 2009. doi: 10.1109/HICSS.2009.494.

[12] S. L. Garfinkel. Automating disk forensic processing with sleuthkit, xml and python. In *Systematic Approaches to Digital Forensic Engineering, 2009. SADFE '09. Fourth International IEEE Workshop on*, pages 73–84, May 2009. doi: 10.1109/SADFE.2009.12.

[13] Simson L. Garfinkel. Digital forensics research: The next 10 years. *Digital Investigation*, 7, Supplement:S64 – S73, 2010. ISSN 1742-2876. doi: http://dx.doi.org/10.1016/j.diin.2010.05.009. URL http://www.sciencedirect.com/science/article/pii/S1742287610000368. The Proceedings of the Tenth Annual {DFRWS} Conference.

[14] Sean E Goodison, Robert C Davis, and Brian A Jackson. Digital evidence and the us criminal justice system, 2015.

[15] Graves. Digital archaeology the art and science of digital forensics, 2013.

[16] Ben Hitchcock, Nhien-An Le-Khac, and Mark Scanlon. Tiered forensic methodology model for digital field triage by non-digital evidence specialists. *Digital Investigation*, 16:S75–S85, 2016.

[17] A. Iacob, L. Itu, L. Sasu, F. Moldoveanu, and C. Suciu. Gpu accelerated information retrieval using bloom filters. In *System Theory, Control and Computing (ICSTCC), 2015 19th International Conference on*, pages 872–876, Oct 2015. doi: 10.1109/ICSTCC.2015.7321404.

[18] David Lillis, Brett Becker, Tadhg O'Sullivan, and Mark Scanlon. Current Challenges and Future Research Areas for Digital Forensic Investigation. 05 2016.

[19] E. Morioka and M. S. Sharbaf. Cloud computing: Digital forensic solutions. In *Information Technology - New Generations (ITNG), 2015 12th International Conference on*, pages 589–594, April 2015. doi: 10.1109/ITNG.2015.99.

[20] Jason Power, Yinan Li, Mark D. Hill, Jignesh M. Patel, and David A. Wood. Toward gpus being mainstream in analytic processing: An initial argument using simple scan-aggregate queries. In *Proceedings of the 11th International Workshop on Data Management on New Hardware*, DaMoN'15, pages 11:1–11:8, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3638-3. doi: 10.1145/2771937.2771941. URL http://doi.acm.org.ucd.idm.oclc.org/10.1145/2771937.2771941.

[21] L. Prechelt. An empirical comparison of seven programming languages. *Computer*, 33 (10):23–29, Oct 2000. ISSN 0018-9162. doi: 10.1109/2.876288.

[22] M. Qi. Digital forensics and nosql databases. In *Fuzzy Systems and Knowledge Discovery (FSKD), 2014 11th International Conference on*, pages 734–739, Aug 2014. doi: 10.1109/FSKD.2014.6980927.

[23] M. Qi, Y. Liu, L. Lu, J. Liu, and M. Li. Big data management in digital forensics. In *Computational Science and Engineering (CSE), 2014 IEEE 17th International Conference on*, pages 238–243, Dec 2014. doi: 10.1109/CSE.2014.74.

[24] Darren Quick and Kim-Kwang Raymond Choo. Impacts of increasing volume of digital forensic data: A survey and future research challenges. *Digital Investigation*, 11(4):273 – 294, 2014. ISSN 1742-2876. doi: http://dx.doi.org/10.1016/j.diin.2014.09.002. URL http://www.sciencedirect.com/science/article/pii/S1742287614001066.

[25] Vassil Roussev, Candice Quates, and Robert Martell. Real-time digital forensics and triage. *Digital Investigation*, 10(2):158 – 167, 2013. ISSN 1742-2876. doi: http://dx.doi.org/10.1016/j.diin.2013.02.001. URL http://www.sciencedirect.com/science/article/pii/S1742287613000091. Triage in Digital Forensics.

[26] Keyun Ruan, Joe Carthy, Tahar Kechadi, and Ibrahim Baggili. Cloud forensics definitions and critical criteria for cloud forensic capability: An overview of survey results. *Digital Investigation*, 10(1):34 – 43, 2013. ISSN 1742-2876. doi:

http://dx.doi.org/10.1016/j.diin.2013.02.004. URL http://www.sciencedirect.com/science/article/pii/S1742287613000121.

[27] Mark Scanlon and M. Tahar Kechadi. Digital evidence bag selection for P2P network investigation. *CoRR*, abs/1409.8493, 2014. URL http://arxiv.org/abs/1409.8493.

[28] Iain Sutherland, Huw Read, and Konstantinos Xynos. Forensic analysis of smart tv: A current issue and call to arms. *Digital Investigation*, 11(3):175 – 178, 2014. ISSN 1742-2876. doi: http://dx.doi.org/10.1016/j.diin.2014.05.019. URL http://www.sciencedirect.com/science/article/pii/S1742287614000620. Special Issue: Embedded Forensics.

[29] N. Thethi and A. Keane. Digital forensics investigations in the cloud. In *Advance Computing Conference (IACC), 2014 IEEE International*, pages 1475–1480, Feb 2014. doi: 10.1109/IAdCC.2014.6779543.

[30] R.B. van Baar, H.M.A. van Beek, and E.J. van Eijk. Digital forensics as a service: A game changer. *Digital Investigation*, 11, Supplement 1:S54 – S62, 2014. ISSN 1742-2876. doi: http://dx.doi.org/10.1016/j.diin.2014.03.007. URL http://www.sciencedirect.com/science/article/pii/S1742287614000127. Proceedings of the First Annual {DFRWS} Europe.

[31] Kathryn Watkins, Mike McWhorte, Jeff Long, and Bill Hill. Teleporter: An analytically and forensically sound duplicate transfer system. *Digital Investigation*, 6, Supplement:S43 – S47, 2009. ISSN 1742-2876. doi: http://dx.doi.org/10.1016/j.diin.2009.06.012. URL http://www.sciencedirect.com/science/article/pii/S1742287609000383. The Proceedings of the Ninth Annual {DFRWS} Conference.