국제학석사 학위논문

A Study on Digital Forensic Data Acquisition Tools

디지털 포렌식 데이터 수집 도구에 관한 연구

함 지 윤(Ham, Jiyoon)

국제학과(Department of International Studies)
정보법과학전공(Major in Legal Informatics & Forensic Science)

한림대학교 대학원
(Graduate School, Hallym University)

국제학석사 학위논문

A study on digital forensic data acquisition tools

디지털 포렌식 데이터 수집에 관한 연구

함 지 윤(Ham, Jiyoon)

국제학과(Department of International Studies)
정보법과학전공(Major in Legal Informatics & Forensic Science)

한림대학교 대학원
(Graduate School, Hallym University)

Joshua I James 교수지도

국제학 석사 학위논문

함지윤의 석사 학위논문을 합격으로 판정함

2019년 12월 19일

심사위원장 박노섭

심사위원 장윤식

심사위원 조슈아제임스

목차(TABLE OF CONTENTS)

표 목차

그림 목차

제 1 장 INTRODUCTION

Digital investigators rely on tools to quickly and efficiently conduct investigations. Many tools, however, are once-off development projects to solve a single problem. In most cases, there is no long-term development or support plan for the tool. The same is valid for digital forensic imaging software.

An Internet search for 'Digital Forensic software tool' returns many different names and manufacturers. Most forensic software developers spend their time and effort to improve forensic investigation tools. Investigators rely heavily on acquisition tools. There are not enough developers working on the problem over the long term. Because of this, much of the forensic acquisition software is not updated. Still, tools are not updated as much as needed, even though acquiring the image file from digital devices is an essential part of the digital forensic investigation procedure.

In this work, we survey digital forensic acquisition software to understand the current state of the available tools. By doing a survey, we double-check that forensic imaging software was not updated. Based on the study, this paper will also argue the importance of improving forensic acquisition tools.

Chapter 2 will explain the research question that we started to face when we are trying to build the forensic acquisition tool, such as what the basic standard of the forensic acquisition tool is. Based on the hypothesis, is it possible to improve the digital forensic imaging tool, will derive many research questions, and these questions will lead to the answering of the hypothesis.

Chapter 3 will be the background research. We will look through previous work related to imaging tools, features of imaging tools with the brief explanation of each feature, and the importance of testing tools to figure out what is improving the forensic acquisition tool and is it necessary to fix it and how to figure whether it is improved or not.

Chapter4 will talk about the status of forensic acquisition tools. All this information dealt in Chapter 4 is from the survey of current existing forensic imager tools. It will point out the features that current forensic imaging tool support and the features that they do not support. Through the survey, it will give the concept of what forensic imager should have as a standard feature and try to get the concept of an improved version of the forensic imager.

Through the research on the NIST'S CFTT tool list, this paper studies thirteen forensic acquisition software tools in Chapter 4. Based on the features that lookup in Chapter 3, we divided the survey section into the type of data, file format, digest hash algorithm, encryption, concurrency, speed. All the thirteen-imaging tool information in the survey gather from the official website or the blog. In this chapter, we also did a speed test to figure the average speed of disk acquisition.

Chapter 5 will talk about forensic tool design. Combining the contents of Chapters 3 and 4, we will decide which forensic acquisition tool is close to our idea to compare. Then, we will also set a basic structure of an acquisition tool and set a goal speed. Furthermore, this paper will write its forensic acquisition tool using go language and compare it to the tool that we selected.

Among the possible place to improve the forensic acquisition tool, speed is the most significant way to improve current imaging tools. Chapter 6 will be discussing the go language features, including the goImager in speed. Through some of the code in golang, it will make forensic imager faster than the compared one with the result of the improved version of goImager test. Lastly, we will discuss future work and how we will continuously update and improve the goImager tool.

Through this paper, we will try to answer the question, "*can we improve the forensic acquisition tool.*" Through the chapter mentioned above, we will figure out what is the minimum features to be a forensic acquisition tool to sketch the basic forensic acquisition tool structure. With the designed structure, we will write the forensic acquisition tool then start to improve it from there.

제 2 장 RESEARCH QUESTION

In the digital forensic investigation, the procedure can be divided into evidence identification, acquisition and preservation, examination, analysis, documentation and presentation. Forensic acquisition is quite vital among the forensic procedure. During the forensic investigation, acquiring a forensic image of evidence takes some time. If it is just acquiring the one hard disk as evidence, it might not take that long. To find out the answer, we will make a sub-question to answer first. Based on this, we will improve the forensic acquisition tool.

This paper made a hypothesis about where we can improve the digital forensic tool. Based on this question, the paper came up with the research question such as Q1) what we have to improve the forensic acquisition? Q2) what the challenges for developing and updating digital forensic tool? Q3) what are a feature in the forensic acquisition tool have? what are the basic acquisition tool looks like? Q4) what can be improve? Q5) How can we improve the forensic acquisition tool?

Q1 why do we have to improve the forensic acquisition tool? Data acquisition itself is the primary step for the forensic investigation. When the time forensic practitioners go to the case, number of digital devices which can be the case-related data source is increasing. Even more, the spec of the devices that user possesses who are case-related gets higher. However, it seems as imaging methods and procedures have primarily stayed the same. Through this, this paper will be found out in the background research in chapter3 and survey in chapter 4 to find when each forensic acquisition tool has been updated their tool and get deep in to find the evidence about Q1.

Q2 what are the challenges to develop and update the digital forensic acquisition tool? Data acquisition tool had been updated but a framework related to imaging forensic disk imaging did not change much compare to the previous. In Q2, we will figure out what is the actual reason why forensic practitioners do not write and improve their forensic acquisition tool. By knowing what are the challenges, it will help to understand and decide whether can we write the tool? What we will do when we are facing the same challenges.

Q3 what are the features that other forensic acquisition on the market? Furthermore, Q4 what is the basic standard of disk imaging tool? goes together. This Q3 is quite essential to know to make a disk imaging tool. We will try to find the answer through the survey about the tools that are released in the forensic acquisition market. Based on this, we will make the forensic acquisition tool standard.

Q5 What can we improve? Speed seems like the most important feature to be improved. In fact, many people who are working in the digital forensic field have much trouble for acquisition time. When it is acquiring the one electronic device, it would not be a big problem to do an investigation, but the number of devices increases, the investigator will suffer for the time. If we can reduce the acquisition time through the speed feature, it can save other forensic practitioners from focusing more on the other forensic investigation procedure such as analysis, and this can be expected to improve the quality of the forensic investigation analysis slightly.

Q6 how can we improve? This paper is considered to write a tool with the go language. Develop the basic forensic acquisition tool based on Q4 then start to improve the imager one by one.

To have answers about the research question above, it is important to study previous research related to the forensic acquisition tool. In chapter 3, background research, we will get the definition of forensic acquisition-related term to understand and goes deep into the practical question one at the time.

제 3 장 BACKGROUND RESEARCH

People are often reminded of DNA analysis when they hear the term forensics. When the case happens and acquiring the DNA sample is necessary, investigators ask and get a DNA sample from the suspect. Similarly, Digital forensic investigators ask and acquire the image of suspect's digital devices to do a scientific investigation. The forensic acquisition is one of the procedure steps of doing the digital forensic investigation.

The forensic acquisition method is mainly divided into three methods, copy, clone, and imaging. Coping is the simplest way among three methods. Copying can only copy the logical data, so it cannot recover the data that the user deleted. Cloning is acquiring the original device's all the physical sector. The cloning method reads and clones all the bitstream of the digital evidence. Unlike copying, cloning can recover the deleted data[1]. However, if it is 256 GB SSD is the source disk, it clones exactly 256 GB to the more significant storage medium.

Imaging is merely saving all the physical sector of the device to a file format. While doing the disk imaging, it will read the first sectors to the end sector of the device then save it as a file format including the file, directory structure, slack space, and unallocated space[1]. The reason why digital forensic practitioner is acquiring the image is to maintain the originality of storing medium. In the digital forensics field, original data should not be changed while processing the forensic investigation procedure. In the early version of imaging tool used the dd(disk dump) method often, but many forensic imagers started to put specialized forensic image format to protect the acquired image file from the damage such as EWF(Expert Witness Compression Format) and AFF(Advanced Forensic Format). Using the forensic image format can compress the original bitstream data to reduce the forensic image size or protect the data through the encryption.

The Imaging method can be divided into hardware and the software. Imaging through software tool is copying all the physical sectors of the original target device to the destination disk[1]. While imaging through the software tool, it is crucial to connect the write blocker or mount the original target device as read-only. Typical forensic acquisition software tools are FTK imager, Guymager, EWFacquire, and Magnet Axiom.

Forensic imaging hardware tool is imaging using the hardware device through customized hardware devices. If forensic software we run by the host operating system to mount the original target device and transfer the data, hardware uses their independent hardware device [1]. Hardware runs the software inside, but it does not use the host system. Typical forensic acquisition hardware tools are Falcon, Super Imager, and TD3.

In the background research, we will focus on the forensic imaging software tool with the research questions. Based on the research question, we want to answer one by one and ultimately to answer whether we can improve the forensic acquisition tool.

3.1) WHY DO WE HAVE TO IMPROVE THE FORENSIC ACQUISITION TOOL?

According to Graeme Horsman[2], forensic software tools are essential in the field. Digital Forensic practitioners are entirely relying on tools to get valid results enough to bring it to the court. There is not much tool testing going on, except NIST's CFTT. Horseman asserted that people should test the forensic software tool to increase the reliability of the result that the tool generated. Horsman insists on the importance of tool testing. By keep testing the forensic software, it makes forensic practitioners think about what to improve and make this forensic field stronger. However, forensic practitioners rely on the current forensic software tool and consider the challenge more prominent than the testing repeatedly.

A horseman is not only one raised voice on tool testing issues. In this paper 'Evaluating Digital Forensic Tools,' Flandrin[3] gathered lots of voices from forensic practitioners, such as they rely too much on the manufacturer's tool about the validity. He also argues that forensic labs should not slowly trust other organization's tests such as Carrier's Model of Abstraction Layers, NIST Standardized Approach of Tool Evaluation[4]. The author summarized some of the methodologies that exist in the legal field to improve the validate digital forensic function. He asserted that Wilsdon’s methodology is suitable. His focusing way is quite useful to do forensic tools. Upgrading and implementing the forensic tool is the closest way to solve the problem directly.

3.2) WHAT ARE THE CHALLENGES FOR DEVELOP AND UPDATE THE FORENSIC ACQUISITION TOOL?

Horsman[2] understands that there are several reasons why tool testing is not often happening, such as its hard to seek someone who knows test data set experience, hard to generate the dataset repeated continuously. It is not only the problem of doing the test, but also it can be the challenges for developing and updating the imaging tool. It is hard to find some people who understand the previous working and who can write the forensic tool.

Although there were many voices to point out forensic acquisition tool is outdated and should improve the tools. Nevertheless, Garfinkel's paper, written through his experience[5], points out why digital forensic software is barely updated from time. According to him, a digital forensic tool is hard to write because of data diversity, data scale, severely affected by another manufacturer's upgrade. The range of data should be analyzed quite hard. The digital forensic tool is used in the criminal investigation, civil lawsuit, or other kinds of investigation. Moreover, when the tool got an error, it not only should keep working but also record the reason why it got an error. Secondly, data scale and performance bottleneck make digital forensic practitioners write forensic tools. Since the data scale of digital evidence gets larger and larger, and while doing processing the data, it will cause the bottleneck over and over. Even more, the digital forensic practitioner must continuously update their tool based on another manufacturer's update. They should continuously support the previous version but include the new features to support the new things such as Microsoft's new version of the window system, Google's new JSON format.

3.3) WHAT ARE FORENSIC ACQUISITION FEATURE?

It is required to know what the current forensic imaging tool stands to figure out where to improve. What are the features that they have, and what are they currently support as an imager? This paper narrowed down to four features, which are hash function, file support type, types of forensic imaging, and countermeasures for encryption.

3.3.1 hash function

In the forensic field, the hash function is necessary to maintain the integrity of the image. It is a form of checksum. It contains numbers and letters to check whether data have errors. It calculates whether the source and target data have the same value. Any changes affect the hash value. It uses MD5(Message-Digest

Algorithm 5), SHA1(Secures Hash Algorithm1), SHA-256, SHA3-256, SHA3-512, and CRC-32. Almost the forensic imaging tool has both MD5 and SHA-1 as a hash function. The difference between MD5 and Sha1 is the length of the bit [6]. MD5 makes 128 bits, and SHA-1 makes 160 bits of the checksum.

3.3.2 file support type

File support type is also a requirement in forensic imaging tools. This is also another way to give integrity to the imaging file itself. Since the size of digital evidence gets bigger intensively, there is a need to compress the image file but still maintain integrity. Because of this, many manufacturers and forensic practitioners developed different kinds of file support types such as RAW, Expert Witness Formats, Advanced Forensic format [7]. According to Garfinkel [8], there are three main features of the file format. It should be extensible, non-proprietary, and compressed.

3.3.2.1 Raw Image

A RAW image is the bitstream image of the source. It duplicates the source to target as an image [1][7]. It will be the same size. It does not have metadata as a header of the image file. They save the case information and the hash value into .txt file in the same folder where the RAW image is created. All the forensic imager allows the raw image as a file support type such as dd, dc3dd, and ddcfldd.

3.3.2.2 Expert Witness Compression File Formats

Expert Witness Compression Format files were constructed to manage data efficiently through compression and able to be segmented with checksum[9]. EWF can be divided into physical and logical evidence file. Physical image is reading sector by sector of the target with hash. Logical evidence file is fit for the analysis when specific file is interested. There are many EWF exist, but here will deal famous and significant one which is E01,L01,EX01,LX01 and SMART to understand the file format[9].

3.3.2.2.1 E01 format

E01 format (EnCase Evidence File Format) is created from the company named Guidance Software. It is the most famous file format in EWF group, especially as a physical evidence file. E01 file contains the metadata in the header and footer[10]. The header contains information about the image, and the footer contains the hash function. Unlike the Raw image, E01 compresses the block by block using the deflate method[10]. Even more, E01 offers Cyclical Redundancy Check (CRC) between each block of data for integrity[10]. Because of this feature, it is the most popular file support type among the forensic acquisition tool, and most of the imagers support E01 formats such as EnCase, FTK imager, and axiom.

3.3.2.2.2 Ex01 (Encase 7 evidence file image)

Ex01 is also developed by Guidance. Unlike E01 format, Ex01 uses bzip2 compression method. In E01 format, there was CRC between the format. But Ex01 puts CRC at the footer with the hash values[11].

3.3.2.2.3 L01 format

L01 is the logical evidence file image format. It is a mixture of E01 and SMART format. It has header, volume, table, disk, sector, data, error, session, digest and four more sections[12]. It has three types of compression. It has L01 for the file name extension.

3.3.2.2.4 Lx01

Lx01 is the next version of L01 format. In Lx01 format does not have option for not compression. Lx01 made to use open source compression program either Bzip2 or LZ[13]. It contains device info, case data, sector data, sector table, error table, session table, increment data, md5 hash, sha1 hash, restart data, encryption keys and other fields. Just like L01, this one is also recommended to find when there is specific targets to look for.

3.3.2.2.5 SMART

SMART format 's extension is s01. Unlike other EWF files, it only has four categories, header, volume, table and 'next and done section'[14].

3.3.2.3 AFF (Advanced Forensic Format)

AFF is categorized into two, disk-representation layer and the data-storage layer. Disk-represent layer summarize the disk image information. Each AFF segment gets a segment name and metadata gather this information into pages[15]. Data-storage layer saves the segment as binary format. AFF can options for compression. It uses an open source program named zlib. It can also put metadata at the forensic image and put it separately. The biggest advantage of having AFF format is self-consistency checking. By doing self-consistency checking, it knows which part is corrupted or missing when disk image got error and it can recover the damaged part.

3.3.3 types of forensic imaging

This section is also essential to forensic imaging. Types of forensic imaging can be divided into three, physical, logical, and user's selection[4]. All the forensic imaging tools support the physical image, but logical and user selection is sometimes needed to do a quick search. Based on the different scenario, forensic practitioners can choose which type of forensic imaging fit in.

The physical image will get from the first LBA to the last, including the drive itself[16]. Consider the digital evidence as a clam. The physical image does disk image, including the shell of a clam. It also collects data that is empty or deleted. All the forensic imager supports physical image.

The logical image is only clam meat. If the physical image reads the disk sector to sector, the logical image reads bit by bit. Logical image is faster than imaging as physical, but it does not recover the deleted data. Some of the forensic imagers support logical image.

User-selects the sector to an image is efficient if forensic practitioners know where to look at it to do an investigation.

3.3.4 encryption

Support for an encrypted disk is also an important feature in the imaging. Since the operating system could make full disk encryption itself, it pushed the digital forensics field into a hard situation to recover the data fully by stopping the forensic duplicate of the source[17]. While it tries to duplicate the source drive, the disk with full disk encryption takes it over and the ciphertext. If it is fully encrypted, it is hard to gain the data at all. Since NTFS systems introduced to the world, the possibility of a target disk is fully encrypted goes higher, but there is not much imaging tool that supports the encryptions.

3.3.5 data recovery

Data recovery is one of the important points of forensic acquisition. For the forensic investigation, there is a possibility that suspect deletes the data related to the crime. Because of this, many forensic acquisition tools put data recovery as the main feature. There is physical and logical data recovery. But mainly, data recovery here is logical. Logical recovery means to recover the deleted data from the

Operating system. Logical recovery can be divided into three methodologies[18]. Recovery using metadata is reading the metadata to recover the data. In general, the user deletes the file; OS only changes the flag in the metadata. Because of this, it can recover the deleted data. Many of the forensic imaging tools offer this feature. Furthermore, it is possible to recover the deleted data through metadata in SSD's case.

Data carving is for the time when metadata overwrote. This time cannot recover the metadata. Using the data carving methodology, recover the data in the unallocated space[19]. In this case, it can recover the SSD. This is because the unallocated space of SSD in the filesystem is the same as the HDD. Overwritten data recovery was the case when metadata and file data got overwritten[1]. For the SSD and HDD, it is hard to recover the overwritten data.

3.4) WHAT CAN WE IMPROVE

This paper '**A systematic evaluation of disk imaging in EnCase 6.8 and LinEn6.1**'[20] focuses explicitly on the forensic imaging software tool. They set an own evaluation process to figure the encase product, which is EnCase6.8 and LinEn6.1. There is already CTFF Digital Data Acquisition Tool Specification enumerate the 8 mandatories, and 18 recommend features. However, Byers set their mandatory list for 12 to test two tools to prove the tool that they use has full ability to read all sectors, even hidden one. The test result turned out that both tools did not meet all the required list that Byers set but mostly did. For example, Encase 6.8 could not acquire the hidden sectors, could not acquire all partitions (only acquired first 25 partitions), and there was an error while acquiring single partition when there are no partition table entries. Just like this paper, we can improve the features of the forensic acquisition tools such as adding the file format, putting the function to do encryption for the image file, add more function to data recovery and put more hash function to reduce the collision.

3.5) HOW SHOULD WE IMPROVE THE FORENSIC ACQUISITION TOOL?

3.5.1 unsupported features

Acquire the forensic image is a battle against time. For one source disk, it might take a couple of hours to image. But if it is acquiring the image of one entire division to find one suspect in there. It might take more than a couple of days to image. Because of this kind of problem, forensic imagers should always seek the way to do image efficiently while maintaining the integrity of the image.

3.5.1.1 Imaging efficiency

Acquire the forensic image is a battle against time. For one source disk, it might take a couple of hours to image. But if it is acquiring the image of one entire division to find one suspect in there. It might take more than a couple of days to image. Because of this kind of problem, forensic imagers should always seek the way to do image efficiently while maintaining the integrity of the image.

3.5.1.1.1 HDD and SSD

The forensic filed started to have the burden of doing imaging when the size of evidence gets bigger and bigger. It got even harder when Solid States Drive (SSD) came out. Its same storage device, but the storage medium, is a semiconductor memory, which is flash memory. HDD uses an embedded magnetic disk to storage. SSD became a problem in the forensic field. Not only physically storage medium changed, but also how SSD read the data, write the data, and process the data is quite different compared to the HDD[21].

HDD is still most frequent in the world. It uses magnetic to read and write the data. Through the magnetic field, it records the data on the platter. HDD's platter today spins around 5400 ~15000 RPM[22]. It has very different kinds of the interface such as ATA, SATA, SCSI. Currently, forensic practitioners use SATA-3 the most, and the transmitter speed is 600 Mbps[21].

SSD is faster than the HDD. It uses a NAND flash memory[21]. SSD does not have arm and platter. This will make read, write, and access time is much faster than HDD. It read and writes data through the electronic signal.

As mentioned above, SSD and HDD use a different method to read and write the data[21]. Even though the forensic acquisition tool offers SSD to create a forensic image, it does not mean it is efficient. FTL, wear-leveling, and the trim makes the modern imager not efficient enough [23].

Just like flash memory, SSD also limits the number that can be erased. Because of this feature, it disperses data to make sure the OS uses shell evenly. SSD uses Flash Translation Layer to do wear-leveling[23]. Even though current forensic imagers support SSD to acquire an image, but it did not consider the FTL and the wear-leveling.

SSD supports the command name trim. SSD does garbage collection. Garbage collection happens often happens by firmware[24]. When the page is filled up with data, it collects the valid data altogether to the empty block and erases the page.

Trim is the method to reduce the number of overheads from garbage collection. Because of this function, it might be getting harder to recover the deleted file.

3.5.1.1.2. Speed

Speed is another measurement to decide whether current disk imaging is efficient or not. In a speedway, there are two essential things to improve. One is Speed, and the other is Ram.

This paper mentioned about interface before. In this paper, focus on SATA-3. In current pc, SATA 3.0 Gbit/s already jump over hard disk's maximum transmission speed. SATA 6.0Gbit/s is suitable for one SATA for many drives[25]. While the Speed of SATA-3 goes up, forensic imagers are stopped at the past version of the interface. It supports SATA-3, but it does mean data transfer got faster through. If SATA-3 is fully working with SSD, it should be acquiring the SSD image close to 600 MB/s[1]. The current version of forensic imagers does not support this part. In this paper, we did speed tests for acquiring an image of SSD and HDD. In this paper, we did a speed test for SSD and HDD. For SSD, much forensic software transferred data around 150MB/s, such as Belkasoft Acquisition Tool.

While cleaning the data for 30 forensic imaging tools, we were very excited about the Atola Insight Forensic tool. On the official homepage[26], they wrote that their tool is the most efficient system to image HDD and even SSD. It can image three targets at once. Moreover, its maximum imaging speed is 520MB/s. For the active imaging part, it is the closest tool that we want to create. However, using the tool itself was not easy. We were not able to image the SSD to see the efficiency.

Random Access Memory (RAM) is another point that changed for decades. People started to put more RAM into their desktop and computer. A large amount of RAM helps to process the data faster. Faster processing data reduce the amount of acquisition time. With this fact, a large amount of the RAM helps the forensic acquisition.

3.5.1.1.3 concurrent

While talking about the Atola insight tool, this paper mentioned about concurrency. There is some forensic imaging tool that supports the many source disk to the target disks. This is quite effective for forensic practitioners to save the acquisition. This makes forensic practitioners also more efficiently use their time. If one practitioner receives 20 forensic evidence with a certain amount of time. If an improved forensic acquisition tool saves time in imaging, a practitioner can spare time more into the analysis. This makes forensic analysis more accurate and better. Because of this, the forensic practitioner should continuously find a way to try more source to disk imaging at the same time while maintaining the integrity of forensic data.

3.5.1.1.4 Accuracy

Improving the accuracy rate is significantly essential. It is the closest way to maintain the integrity of the forensic evidence and reduce the error rate at the same time. This can be done by testing the forensic imager over and over to find out the error and fix it. However, as background research, forensic practitioners rely on the current forensic tool. Currently, NIST's CTFF exists, but most of them are quite outdated compared to this paper's data collection. Some of the papers mentioned their own tool testing, but there is a need to make one integrated and transparent test framework to make forensic field reliant.

Based on this, this paper was able to figure what current forensic imaging software have and not. In the features that are fully supported. Based on each forensic imaging software manufactures aim, they supported some. This part will be dealt deeply into chapter 4 which is survey of the many forensic imaging tool.

3.5.2 CODING-LANGUAGE

While developing a tool, choosing platform and language, high-performance computing, whether tools should be all in one or single-use, evidence container file format are the things to consider writing forensic software tools. The recommendation from Garfinkel[5] is quite essential. In his commendation in choosing a language to write a forensic tool, he said C# for Linux and Mac and C++ for Window OS[5]. However, since he wrote this paper, there are many languages came out, and one of them might be better making forensic acquisition tool than C# or C++[5]. He said it is hard, but he did not mean to write it. But focus on developing forensic tools goes to the other side, such as analyzing the image file.

Matter of fact, some practitioners focused on changing language will help to improve the forensic acquisition tools. This paper 'Nugget: A digital forensics language'[27] introduces to the forensic field to change the language of the forensic tools to nugget, which is a domain-specific language(DSL).The most significant point of the DSLs compiler is to catch the intent of the forensic tools and organize the data intuitively[27]. Even though it is a protocol type to introduce forensic society, but they expect to optimize the time duration while making a forensic acquisition and deal with the volume diversity. Nugget or the other language can be used to replace Garfinkel's recommendation. Based on both two paper, choosing the coding-language can be very important element for writing the forensic acquisition tool. But we should not forget the main purpose of both is making better forensic tools from what forensic practitioners are facing as a problem.

3.5.3 FILE FORMAT

Language can be one solution to update the acquisition tool but some paper raises their voice to chose more file format to improve the forensic acquisition tool. Another paper named 'Wirespeed: Extending the AFF4 forensic container format for scalable acquisition and live analysis'[28] explain not only why we should focus on implementing the forensic acquisition tool more than others but also what to do to implement the tool. Since Solid States Drive (SSD) came out to the world, forensic society's worry towards data scale to acquire data got worse. This is because forensic imaging tool does not keep pace with the growth in volume and I/O rate. Because of this growth caused the restriction of spinning disk and bottleneck related to bandwidth[28]. Schatz[28] pointed out those two reasons as the main reason why the speed of acquisition takes a longer time to image. He came up with some variables to optimize the acquisition rate. He argued that by reorganizing the

Advanced Forensic Format(AFF)4, it would improve the imaging speed and reduce bottleneck. Based on his test with set variables, this paper was able to fasten acquiring the data. However, it is not supported by most forensic acquisition tools. Just like AFF4 , it seems like it is important to find out what file format we will like to support and adding more forensic file format is important.

3.5.4 CASE

There is another language name Cyber-investigation Analysis Standard Expression (CASE). According to Harm van Beek[29],CASE is a community-developed ontology to support Reporting of digital traces, exchanging of digital traces, and tool validation. The CASE community[30]'s the primary purpose of the CASE is interoperability. Digital Forensic investigation can be used for many purposes, such as digital forensic itself, incident response, criminal justice. Each place uses different tools and different platform, and optimizing these tools, need an official language to do data processing[2].

Through the background research, we got the concept of what is forensic acquisition, what kind of forensic acquisition method exist, what is forensic imaging is. Then we understand there is two types of forensic imaging; hardware and software. When we deeply get inside of forensic software tool, we found the voices that forensic software tool should be tested and practitioner should not rely on the forensic software tool. Because of this we understood why do we have to improve the forensic acquisition tool. Then checked on what are the challenge of developing and improving the forensic acquisition tool. But at the same time, we found out that it is hard to do so, but there are needs exist to develop and improve their forensic tool. Next step was find out what are the features of the forensic imager. Through the research it turned out hash function, file support type, encryption, data recovery are the features of forensic acquisition tool. Then we consider as by implementing the forensic features can be improving the forensic acquisition tool. Finally, we found some papers about the voice that how forensic practitioners wants to implement their tool. we decided to do the survey of current forensic acquisition tools to earn samples for each forensic acquisition tool's features and find out what are we going to develop as a forensic acquisition tool and how can we assert our forensic imager is improved.

제 4 장 The survey of current forensic imaging software

Previous chapter, we figure what is forensic imaging software features, but we realize which forensic acquisition tool have what features and not. Survey of the current forensic imaging software is quite important to to get the concept of what is the standard of forensic imager, Because of this reason, we decide to make survey based on NIST's CTFF and improve the chart first to improve the forensic acquisition tool.

4.1) The purpose of the survey

In this chapter, we are going to talk about the survey and the speed test. The purpose of the survey is to collect information about the forensic imaging tools to understand what kind of features they support and not. By doing this survey, it is also making people understand the current circumstances of current imaging software.

4.1.1 Survey method

Based on the NIST's, CTFF searched 30 imager tools. This paper gathered CTFF forensic imager information and updated the information through each website of the imager.

4.1.2 Survey target

Atola Insight Forensic, Belkasoft Acquisition Tool, Magnet Axiom, Data Recovery System, DC3DD, OSForensics, Guymager ,X-WAY(FORENSIC, IMAGER, WINHEX), EnCase, FTK imager, FTK IMAGER CLI version, DDcfldd, EWF acquire, MacQuisition, Magnet Aquire, MiniDAS, PC-300 Data Extractor, CFID V3, Detago Ultimate Suite, Fast Disk Acquisition System, Forensic Falcon, Forensic Replicator, Solo-101 Forensics, Solo-4 Forensics, SuperImager 7" Field Imaging and Triage Platform unit, SuperImager 8" Field Unit, SuperImager Rugged 12" Field Computer Forensic Imaging and Filed Platform, TD2 and TD3.

4.1.3 Main Survey target

On that list above, this paper found several acquisition software tools that can access, such as freeware, open source tool, and trial. For the thirteen forensic software tool (Atola Insight Forensic, Belkasoft acquisition tool, Magnet Axiom, Data Recovery System, DC3DD, OSForensics, Guymager, X-Way imager, EnCase, FTK imager CLI version, ddcfldd, dd, EWF acquire), we decide to try each tool and based on the survey, doing the comparison of each tools.

Survey target 1) Atola Insight Forensic

Atola Insight Forensic released the tool version 4.13.2 on September 11th, 2019. Based on their website, it does physical, logical, and a user can choose the sector to do disk imaging. For the file format, it only allows RAW, EWF(E01). It does do hash calculation before imaging starts, during the acquisition, after imaging finishes[26]. Atola offers MD5, SHA1, SHA224, SHA256, SHA384, SHA512 for the hash calculation. Through the hash calculation, it compares whether the source and target data are appropriately acquired. If there is a bad sector, it filed with 00 as a default[26]. They can image three target sources at the same time. They do not support data encryption. Eye-catching information on the website, they asserted that their imaging speed is up to 520 MB/s.

Survey target 2) Belkasoft acquisition tool

Belkasoft acquisition tool released in September 2018 as version 9.3. It can do physical and logical disk imaging. Moreover, it allows mobile data, cloud data, and Random Access Memory to make an acquisition[31]. This tool only offers RAW and EWF(which is E01) as a file format. For the checksum, they allow MD5, SHA1, and SHA2-256. They do not support data encryption.

Survey target 3) Magnet axiom

Magnet axiom released on October 2nd, 2019. Magnet Axiom can do a physical, logical image. It can acquire the image from a mobile phone, RAM, cloud, IoT devices[32]. For the hash, they support MD5 and SHA1. Their evidence source is RAW, EWF(which is E01, Ex01, L01, Lx01), AD1, Virtual Machine images, DMG, and Archives. Magnet Axiom is not a single product. Axiom process does acquire an image and examine does analysis at the same time. They do not support data encryption.

Survey target 4) Data Recovery System (DRS)

DRS is from SalvationData technology. Their recent version updated on December 25th, 2018, as of version 18.7.3.292[33]. They only allow the physical acquisition, support MD5, acquire a RAW image. On the website, it does not mention about SSD anywhere for a disk acquisition. However, it does allow to do disk to acquire, copy, clone at the same time. They do not support data encryption.

Survey target 5) dc3dd

Dc3dd got updated on July 11th in 2018. However, when looking at the file modified time, it is April 29th,2016. It supports the physical, logical, user-selected sector to make the acquisition. It only supports the RAW file format. Dc3dd allows MD5, SHA1, SHA2-256 for the checksum. They do not support data encryption[34].

Survey target 6) OSForensics

OSForensics updated their tool as 7.0.1005 on October 10th, 2019. They support RAW, EWF(E01,ex01), AFF, Virtual Disk format as a file format. For the checksum, OSForensics only supports MD5. They do not support data encryption[35].

Survey target 7) guymager

Guymager is the open source tool to acquire. It is Linux based. Guymager webpage[36] did not mention when it got updated, but the latest version is 0.8.11. It allows physical and logical acquisition. For the file format, the tool offers RAW, EWF(E01), and AFF. For the checksum, it supports MD5, SHA1, SHA256. It supports encryption[37].

Survey target 8) X-ways

X-ways have two possible products that could be a target for the survey, which is X-Ways forensics, and X-Ways Imager[38]. However, they have another product to acquire to taste how X-Way acquisition works. Its call as WinHex. Based on WinHex, this paper will collect information. Physical and logical, user-defined sector ranges are available for the acquisition. Both products support RAW and EWF(E01). Nevertheless, in the WinHex, it only supports their file format, which is whx. X-way forensic tools do offer encryption.

Survey target 10) Encase forensic imager

Encase forensic imager is from Guidance Software. This tool runs on Windows operating system. It can acquire physical and logical drives. For the image supports type, it does RAW, E01, Ex01, Lx01, L01. It does offer AES 256-bit encryptions[39].

Survey target 11) FTK imager

FTK imager updated the tool on March 11th, 2019. It can acquire the physical, logical, and user-defined sector range as an image file. It offers RAW, EWF (E01 and SMART), and AFF as an image type. It does allow RAW and SHA-1 for hash calculation[40].

Survey target 12) FTK imager CLI version

FTK imager CLI version updated on September 19, 2012[41]. It does the same function as same as the survey target 11.

Survey target 13) dcfldd

Dcfldd updated version 1.3.4.1 on December 19th, 2006. It does physical and logical disk imaging. It works at Linux based and only creates the RAW image. It offers RAW and SHA-1 for hash calculation. It does allow encryption[42].

Survey target 14) EWF acquire

EWF acquire a Linux command-line tool to make disk acquisition. It can make physical and logical disk acquisition and set file formats like RAW, EWF(e01, and SMART). For the EWF acquire, digest type is MD5, SHA-1, and SHA-256[43].

Based on the collected information, this paper figured what forensic acquisition tool should contain.

4.1.4 Survey sections

Divided the section through types of data, file format, digest hash algorithms, encryptions, concurrent, speed.

4.1.5 survey result

4.1.5.1 Type of data

In the survey, all the forensic imagers at least support the physical acquisition. Half of the forensic imager allows the logical acquisition, and 2/5 of the imager offers users to select the sector range to make the acquisition.

Disk Imaging forensic Tool by features from NIST CFTT					
Name	Version	Release date	PHYSICAL	LOGICAL	USER_DEFINE
Atola Insight Forensic	4.13.2	2019-09-11	O	O	O
Belkasoft Acquisition Tool	9.3	2018-09	O	O	X
Magnet Axiom	3.6.0.15906	2019-10-02	O	X	X
Data Recovery System(DRS)	18.7.3.292	2018-12-25	O	X	X
DC3DD	7.2.646	2018-07-11	O	O	O
OSForensics	7.0.1005	2019-10-10	O	O	O
Guymager	0.8.11	X	O	O	X
X-ways Forensic	19.9	2019-11-24	O	O	O
Encase	X	X	O	O	O
FTK Imager	4.21	2019-03-11	O	X	X
FTK Imager cli version	3.1.1	2012-09-19	O	O	X

Ddcfldd	1.3.4.1	2006-12-19	O	O	X
EWF acquire	X	2019	O	O	X
ILookXimager	4	2012-10	O	O	O
MacQuisition	1.2	2019-05-30	O	O	X
Magnet Acquire	2.20.0.17984	2019-10-02	O	X	X
MiniDas	1	2013-11	O	X	X
PC-300 Data Extractor	5.5.2	2016-10	O	O	X
CFID V3	3	2016-09	O	X	O
Detago Ultimate Suite	3.4	2017- 09	O	X	X
FDAS	2.0.1	2007-06	O	X	O
Falcon	2.3	2013-05	O	X	X
Forensic Replicator	4.3	2012-09	O	O	X
Solo-101 Forensic	X	2011-01	O	O	X
Solo-4 Forensics	X	2008-09	O	X	O
SuperImager 7" Field Imaging and Triage Platform unit	1.4.2.3	2014-08	O	X	X
SuperImager 8" Field Unit	1.4.4.1	2014-02	O	X	O
SuperImager Rugged 12"	1.4.4.1	2014-01	O	X	O

Field Computer Forensic Imaging and Filed Platform					
TD2	1	2012-03	O	X	O
TD3	1	2011-12	O	X	X

Table 1 acquisition type survey

4.1.5.2 File format

100 percent of the forensic imagers acquired image create a RAW image. Eighty-three percent of the imagers support the E01 image – the rest of EWF rarely supported by forensic imagers. Sixteen percent of the acquisition tool makes AFF forensic tools. Many of updated tools were barely supported the image file format. Atola Insight Forensic, Belkasoft Acquisition Tool, Magnet Axiom, Data Recovery SYSTEM, DF3DD, X-WAY forensics updated their forensic tools in these two years, but they did not update the file format.

Disk Imaging forensic Tool by features from NIST CFTT									
Name	RAW(dd)	E01	L01	Ex01	AFF	SMART	Virtual disk format	Single file	dmg

Atola Insight Forensic	O	O	X	X	X	X	X	X	X
Belkasoft Acquisition Tool	O	O	X	X	X	X	X	X	X
Magnet Axiom	O	O	O	X	X	X	O	X	O
Data Recovery System(DRS)	O	X	X	X	X	X	X	X	X
DC3DD	O	X	X	X	X	X	X	X	X
OSForensics	O	O	X	O	O	O	O	X	X
Guymager	O	O	X	X	O	X	X	X	X
X-ways Forensic	O	O	X	X	X	X	X	X	X
Encase	O	O	X	X	X	X	X	X	X
FTK Imager	O	O	X	X	O	O	X	O	X
FTK Imager cli version	O	O	X	X	O	X	O	X	X

Ddcfldd	O	X	X	X	X	X	X	X	X
EWF acquire	O	O	X	X	X	O	X	X	X
ILookXimager	O	O	X	X	X	X	X	X	O
MacQuisition	O	O	X	X	X	X	X	X	O
Magnet Acquire	O	O	O	X	O	X	O	X	X
MiniDas	O	O	X	X	X	X	X	X	X
PC-300 Data Extractor	O	O	X	O	O	X	O	X	X
CFID V3	O	O	O	X	X	X	X	X	X
Detago Ultimate Suite	O	X	X	X	X	X	X	X	X
FDAS	O	O	X	X	X	X	X	X	X
Falcon	O	O	O	X	X	X	X	X	X
Forensic Replicator	O	X	X	X	O	X		X	X

Solo-101 Forensic	O	O	X	X	X	X	O	X	X
Solo-4 Forensics	O	O	X	X	O	X	X	X	X
SuperImager 7" Field Imaging and Triage Platform unitX	O	O	X	X	O	X	O	X	X
XSuperImager 8" Field Unit	O	O	X	X	X	X	X	X	X
SuperImager Rugged 12" Field Computer Forensic Imaging and Filed Platform	O	O	X	X	X	X	X	X	X
TD2	O	O	X	X	X	X	X	X	X
TD3	O	O	X	X	X	X	X	X	X

Table 2 file format survey

4.1.5.3 Digest hash algorithms

All the forensic acquisition tools support MD5. Only 10 percent of the forensic does not support SHA-1. Hash algorithms are the primary method to prove the integrity of the target file. Nevertheless, more than half of the forensic imager does not support more than SHA-1.

Disk Imaging forensic Tool by features from NIST CFTT							
Name	MD5	SHA1	SHA2-256	SHA2-512	SHA3-256	SHA3-512	CRC32
Atola Insight Forensic	O	O	O	O	X	X	X
Belkasoft Acquisition Tool	O	O	O	X	X	X	X
Magnet Axiom	O	O	X	X	X	X	X
Data Recovery System(DRS)	O	X	X	X	X	X	X

DC3DD	O	O	O	X	X	X	X
OSForensics	O	X	X	X	X	X	X
Guymager	O	O	O	X	X	X	X
X-ways Forensic	O	O	O	X	X	X	X
Encase	O	O	O	X	X	X	O
FTK Imager	O	O	X	X	X	X	O
FTK Imager cli version	O	O	X	X	X	X	X
Ddcfldd	O	O	X	X	X	X	X
EWF acquire	O	O	X	X	X	X	X
ILookXimager	O	O	O	O	O	O	X
MacQuisition	O	O	O	X	X	X	X

Magnet Acquire	O	O	X	X	X	X	X
MiniDas	O	O	X	X	X	X	X
PC-300 Data Extractor	O	X	X	X	X	X	X
CFID V3	O	O	O	X	X	X	X
Detago Ultimate Suite	O	O	X	X	X	X	X
FDAS	O	O	X	X	X	X	X
Falcon	O	O	O	X	X	X	X
Forensic Replicator	O	O	X	X	X	X	X
Solo-101 Forensic	O	O	O	X	X	X	X
Solo-4 Forensics	O	O	O	O	X	O	O
SuperImager 7" Field Imaging and	O	O	O	X	X	X	X

Triage Platform unit							
SuperImager 8" Field Unit	O	O	O	X	X	X	X
SuperImager Rugged 12" Field Computer Forensic Imaging and Filed Platform	O	O	O	X	X	X	X
TD2	O	O	X	X	X	X	X
TD3	O	O	X	X	X	X	X

Table 3 hash algorithm survey

4.1.5.4 Encryption

Many of the forensic tools do not support data encryption. The importance of personal privacy goes up, many electronic devices manufacturers started to focus on data encryption, and some of them offer the full encryptions. Because of this phenomenon, it will interrupt the forensic investigation. However, only half of the forensic acquisition focuses on this point.

Disk Imaging Forensic Tool by features from NIST CFTT			
Name	Version	Release date	Encryption
Atola Insight Forensic	4.13.2	2019-09-11	X
Belkasoft Acquisition Tool	9.3	2018-09	X
Magnet Axiom	3.6.0.15906	2019-10-02	X
Data Recovery System(DRS)	18.7.3.292	2018-12-25	X
DC3DD	7.2.646	2018-07-11	X
OSForensics	7.0.1005	2019-10-10	X
Guymager	0.8.11	X	O
X-ways Forensic	19.9	2019-11-24	O
Encase	X	X	O
FTK Imager	4.21	2019-03-11	O
FTK Imager cli version	3.1.1	2012-09-19	X
Ddcfldd	1.3.4.1	2006-12-19	O
EWF acquire	X	2019	X
ILookXimager	4	2012-10	X
MacQuisition	1.2	2019-05-30	X
Magnet Acquire	2.20.0.17984	2019-10-02	X
MiniDas	1	2013-11	X
PC-300 Data Extractor	5.5.2	2016-10	O
CFID V3	3	2016-09	X
Detago Ultimate Suite	3.4	2017- 09	O
FDAS	2.0.1	2007-06	O
Falcon	2.3	2013-05	O
Forensic Replicator	4.3	2012-09	O
Solo-101 Forensic	X	2011-01	O
Solo-4 Forensics	X	2008-09	O
SuperImager 7" Field Imaging and Triage Platform unit	1.4.2.3	2014-08	O
SuperImager 8" Field Unit	1.4.4.1	2014-02	O

SuperImager Rugged 12" Field Computer Forensic Imaging and Filed Platform	1.4.4.1	2014-01	O
TD2	1	2012-03	X
TD3	1	2011-12	X

Table 4 encryption

4.1.5.5 Concurrent

Atola Insight forensic can do three evidence tools at once[26]. Magnet Axiom does acquisition and analysis at the same time with two tools (which are Axiom process and Axiom Examine)[32]. However, many of the forensic acquisition tools that surveyed did not support the concurrency. Nevertheless, concurrency is the crucial thing in the forensic field. First, the size of the digital evidence gets larger. Second, people started to carry more than two electronic devices. Because of these kinds of circumstances, it requires many of the forensic acquisition tools to reduce the time of imaging.

4.1.5.6 Speed

Through the survey, it was hard to figure out the speed. Instead of that, downloaded the available thirteen imaging software tool to figure out what is the average of the current acquisition tool speed is. If there is a possible place to improve the imaging tool in the speedway, find out what it is.

4.1.6.6.1 Speed Test Methodology:

- Based on the Operation System (Linux and Windows), select the acquisition tool.
- Remove all the running applications, then run only the acquisition tool.
- The select personal computer as an SSD source device then plugs in the external hard disk to acquire.

Test type divided into acquiring SSD as source into RAW image while target disk plug into USB3.0, HDD as source into RAW image while target disk plug into USB3.0, SSD as source into EWF(E01) image while target disk plug into USB3.0, HDD as source into EWF(E01) image while target disk plug into USB3.0. These four different circumstances tests happen both Linux and Windows OS systems.

4.1.6.6.2 Speed Test environment

SSD: 256GB SSD SAMSUNG MZNLN256HCHP-000B1

HDD: 16 GB SMI USB DISK USB Device

Computer environment: DUAL MODE of Window 10 & Linux mint version (4GB RAM, Intel CORE i3)

4.1.6.6.3 Speed Test Result

Out of thirteen tools, seven are window OS based tool, four are Linux based, and ftk-imager cli version is dual.

In window based tool, belkasoft acquisition tool, winhex and ftk imager were success to do acquisition.

	W_PHY_SSD_DD	W_PHY_HDD_DD	W_PHY_SSD_E01	W_PHY_HDD_E01
Belkasoft Acquisition tool	0:53:26	0:04:52	4:17:01	0:11:37
X-ways Forensics & X-Ways Imager & Winhex	2:29:13	X	X	X
FTK imager	0:38:52	0:03:36	1:12:16	0:04:37

Table 5 window-based tool acquisition

Belkasoft acquisition tool took 2 hours 50 minutes and 14 seconds to make raw disk image when imaging the SSD to the HDD using the USB2.0 PORT. Using the

USB3.0 port, it took 53 minutes and 26 seconds to do the same acquisition. For the imaging HDD(USB stick) to HDD, creating raw disk image for USB 2.0 took 14 minutes and 9 seconds and USB 3.0 took 4 minutes and 52 seconds.

FTK imager tool was creating raw disk image in 1hour 15minute and 10 seconds when SSD is source and HDD is plugged in USB 2.0 port. For the USB 3.0 port, it took 38 minutes and 52 seconds to acquire the raw disk image. HDD to HDD for making raw disk image took 03minutes and 36 seconds in USB3.0 port and minutes and seconds for the USB2.0.

WinHex offers to make their own disk image. Their extension is .whx. Using the WinHex takes 2 hour 50 minutes and 14 seconds to acquire in USB 2.0.

Through the testing the forensic acquisition tool, we set the ftk imager as the model to catch up. From now on we are going to make the ftk imager and try to get the concept of the forensic acquisition speed.

제 5 장 Forensic Tool Design

Thinking about the forensic tool design is quite essential. To write a forensic acquisition tool, we have to understand how the forensic acquisition tool works. Based on the research and the survey, draw the structure of the forensic acquisition tool.

Through the background research and the survey, we understand that many forensic imagers should at least contain RAW, EWF (which is almost E01) supports checksum for MD5 and SHA1. All the acquisition tools at least acquire the physical disk. This is the standard and the average of the forensic software acquisition tool. We also realized that there is no forensic acquisition tool exists which supports acquisition for physical, logical, and sector that users select while it offers many different types of checksum for avoiding collision such as MD5, SHA-1, SHA-256, SHA-512, SHA3-256. Even more, it would be better if it supports not only RAW and

EWF but also AFF at the same time with the better speed. Based on this, we decide to improve the forensic acquisition tool from the standard.

Improved forensic acquisition software should be better in four ways. First, a more forensic practitioner can use this forensic acquisition tool to access it. Second, more acquisition time should be reduced. Third, more reliability in the forensic field should arise. Fourth, more support for different devices and features to cover more forensic evidence. To improve these points, it is essential to picture how the acquisition tool should look like.

5.1) Acquisition tool design

This section will talk about the design of the improved forensic tool. It is mainly focusing on software. Forensic software will be Linux-based coding first then move on to the window version. This is because there is not much forensic acquisition tool to support more than two operating systems. It should support more than two OS types to cover many forensic devices. It should support compression to reduce the size of the image file. It should support more than two file formats. They use a different compression method, so it is vital to know about it. Through the CRC, keeping the integrity of forensic evidence is essential. It is crucial to decide where to put the CRC and the information related to CRC. It should support the SSD also. Enhanced features towards the SSD will be the biggest strength of this forensic acquisition tool. The price of SSD goes down and more companies and the personal user will be purchasing the SSD more than the current spread of SSD. When there is a large amount of SSD, speed should not be the huddle for the forensic investigation. Furthermore, it should be better to have better hardware combine with the software. It will make a significant synergy to enhance the forensic acquisition tool.

5.1.1 support OS type

It will support two operation system ,which is Linux and Window. while doing the survey, there were not many forensic acquisition tools supports both operating systems at the same time. By supporting the two OS system, it will help many forensic practitioners to acquire the image based on their computing system. Each operating system will understand the individual system more than others. For example, Linux and Window use different languages to control they are on the system. In Linux, if a user wants to see the local disk information, they must type 'lsblk' into the terminal. In the window, it can be 'wimic DISKDRIVE get Interface Type, Name, Size, Status'. Based on language, it understands their operating system and how it structs to make disk acquisition. So, covering as much as the operating system can be substantial. For instance, there is a possibility that the forensic practitioner takes a source disk from the suspect's computer. Suspect's computer was Linux based OS, and source data is written through Linux kernel. If a forensic practitioner only has window version forensic acquisition tool, there is more possibility exist that those acquired data might be misread or not fully understand by the acquisition tool.

5.1.2 Compression and file format

Having various kinds of forensic format is very important. This is because, through the forensic image format, it can reduce the image size by compressing the original bitstream data. One of the reasons that we choose Linux based code is because it allows many different compressions. For the one hard disk drive, time might not be a big problem, but as keep mentioned over time that practitioners can improve the quality of forensic analysis if they can save the time for making disk acquisition. To make it better, we would like to design the forensic tool to gzip compression to support AFF and AFF4 format. Forensic software will read the bytes based on the sector and write the image file based on the setting. When it is a writing image, each file will contain the CRC for integrity.

5.1.3 Integrity

CRC plays a significant role in this acquisition software. Put CRC between the block of data or end of the entire file will be matters for the integrity. There are even more matters for this forensic software. This is because we decide to use multi-threading fully. Based on the developed technology, a large amount of RAM can make the OS system to make multi-thread to operate adequately. If there are no resting threads, it will keep running to transport the forensic data and find the resting thread to give the following forensic data. This constant working will affect to improve speed. Most of the forensic acquisition tool creates an image file one at a time. If using multi-threading, forensic acquisition tools can expect to write several image files at the same time, which can be sufficient enough to reduce image acquisition time. People might be concerned about the possibility of losing integrity as forensic evidence. But if CRC can check the integrity of every single sector of evidence file, it will not only improve the time but also maintain the integrity of the forensic field.

CRC is not the only one to check integrity. Checksum takes a large part in the forensic acquisition also. Many of the tools consider just offeringMD5 or supporting MD5 and SHA1 might be enough to check whether the source and target data are valid. There might be a chance that hash checksum itself can be a problem. MD5 or SHA-1 cannot cover the enormous amount of digital evidence. There might be a possibility that the forensic acquisition tool can calculate wrongly that source and target of the forensic evidence are whether right or wrong.

5.1.4 Encryption

There were not many forensic disk imagers that do not support encryption. This software imager will investigate the encryption part. Especially for the window version of the software, it will consider the NTFS system's full disk encryption system and figure out how to deal with it.

Encryption is essential in another way. During the disk acquisition, there is a possibility that an image file can be damaged. To protect the image file from the damage, there is a way to encrypt the image to protect the data inside. This forensic tool will support encryption to protect forensic evidence.

5.1.5 speed

As mentioned above, many of the forensic acquisition does some of the features. If software tool focuses on multi-threading and compression, the possibility to reduce the acquisition time goes rapidly down. Previously most of this section talked about the software; it is because many of the forensic practitioners cannot afford to have forensic hardware. Forensic hardware is powerful but expensive. Compression and multi-threading can be the best way to improve the speed of the forensic acquisition tool. When there is a combination of forensic hardware and improved software will be the most potent combination for forensic acquisition.

5.1.6 Concurrency

Many of software only acquisition tool cannot support the concurrency. When there is forensic acquisition for different sources at one time, it will save lots of time. There is a big stress for the forensic hardware. If forensic acquisition hardware is strong, the size of the hardware gets bigger and expensive. There is necessary for compromised plan. This is not only for the developer, but also for the other forensic practitioners. If the acquisition hardware gets expensive, there will be less people can get to use the tool over time.

5.1.7 Keep updating

this is the software design for the forensic tool. We will use go language to write this forensic acquisition tool to improve it. there are many forensic acquisition tools do not update their tool often. But reality is quite harsh. Constant update of electronic devices and new types of data makes forensic field to feel the needs for forensic acquisition tool to be constantly develop. Because of this, it should be flexible to update the new features but at the same time maintain the previous one.

5.2) Design of tool

This paper made the basic tool shape to develop in golang. Then started to improve the forensic acquisition tool based on the acquisition tool designed that mentioned. This section will explain the basic structure first. This tool is an integrated version of the Linux version and window version. Most of the forensic acquisition tool only supports one operating system, but it does not mean all of them does. Some of them support more than one operating system with a different extension. This basic version of golang imager is considered to run at Linux only.

It will be extended to window OS later. The forensic acquisition tool will show the devices if the user wants. While the user runs the software, the user will also type the source and the destination with the file location and extension. Then the goImager starts to read the size of the source and the cluster to calculate how much time it should start to loop. When it is done, goImager starts to read the source as much as cluster size and write it till the calculation time. When it is over, it starts to calculate the checksum of the source and created a destination file to check whether their file appropriately imaged.

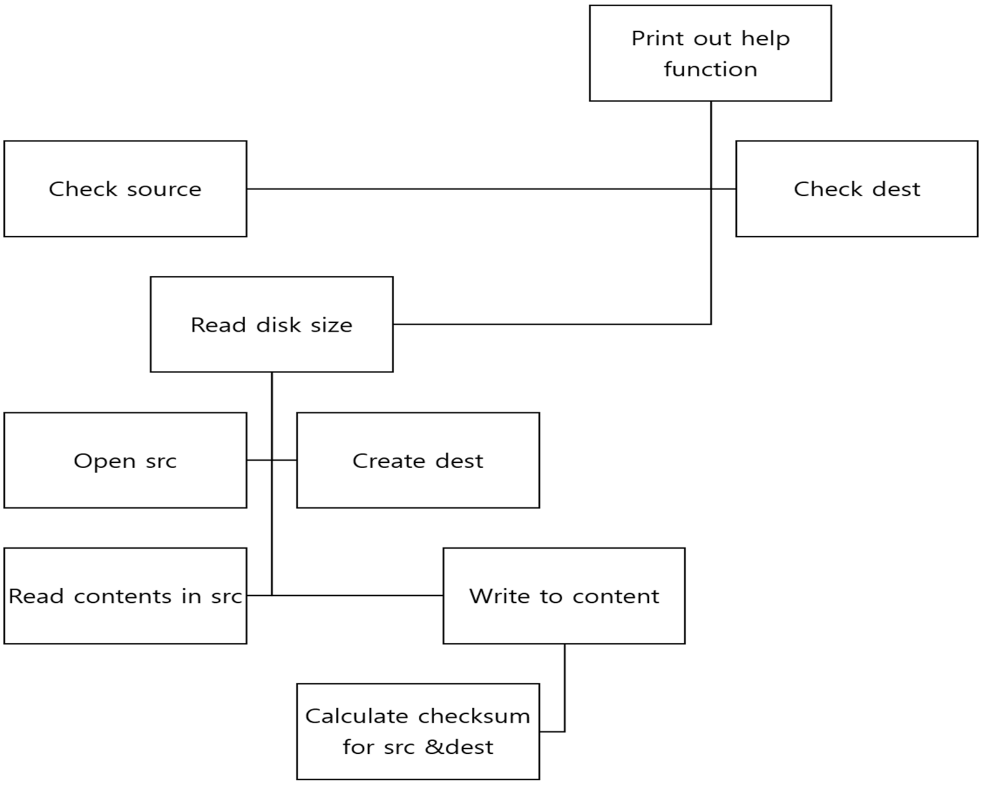

Figure 1 basic goImager structure

Basic tool based on the above diagram have similar speed to do disk image. How to improve the goImager will be dealt in chapter 6.

5.3) Basic Design of tool

Basic structure of goImager is written in go language.

5.3.1 Go language

Golang is developed at google, and it is an open-source language. Inventors of Go were decided to make programming language as optimized, fast and straightforward. Golang is developed for system programming. System programing requires steady states. Go language is compiled language. It is a language that translating the human language into machine language. Based on the different conversion for each platform, it must build a different execution file, but compile language, in general, quite fast[44]. Go language support FreeBSD, windows, mac, and Linux Operating system.

5.3.1.1 packages

Golang program is made with packages. Golang contains lots of built-in packages. It is simple to think golang packages as a library[44]. Golang has lots of built-in packages. By bringing the built-in package, it will save time to write a full code of it.

Instead of that, have to find the proper package structure and the function to write about it.

5.3.1.1.1 os package

Package os provides goImager act as operating system. os.Open will give goImager to access for reading the source disk[45]. Then os.Create the destination file to write function. Read reads up to the given size of bytes. In goImager reads source disk till the given size and it starts to fill the buffer. Buffer hands read contents to.

5.3.1.1.2 log package

Log is quite important to the imager in general. This is because it is important to know what imager do at a certain time. It is also good to know when the error happened. Especially, in the goImager, log package is used to figure when the full path of source disk is given and to find out when goImager fail to run[46]. It will give a guide to user to do proper acquisition.

5.3.1.1.3 io package

Go is the program language to work with bytes. Io package offers interface and helper to work with bytes stream. goImager uses reader interface to read bytes from the stream. Basic structure of reading bytes from stream is reader interface. Buf := make([]byte, 1073741824) will read one gigabytes for one buffer[47].

5.3.1.1.4 checksum package

Hash checksum is quite important. This is because of the integrity of the forensic image. By checking the source and destination disk's checksum matches, it became one of the way to figure whether destination file maintain the identity of the source contents. Fortunately, golang have built-in hash function from md5 to sha512. By calculating the multiple hash, it will reduce the possibility of collision. Just like other forensic imager, consider to give md5 as a default hash and if user selects certain kind of checksum, then goImager can give the checksum result. By using multi-writter, goImager can give all different kind of checksum[48].

Through the packages that mentioned above, this paper was able to write basic forensic acquisition tool written in golang. From now it will talk about the speed test of goImager to test and figure where and how to improve this goImager.

5.4) testing and validation

Testing and validation is a necessary procedure to develop and improve the forensic acquisition tool. Testing can be considered as a speed test, and validation can be counting the error rate for the forensic acquisition tool and hashing. Checking error rate is essential to confirm the validation for the forensic acquisition tool itself, and checking hash checksum is vital to confirm the validation for the acquired image.

In this paper, check the forensic acquisition speed test is done by running goImager in the Linux system.

1. open the terminal in Linux OS system.
2. type command ‘go run goImager.go [-s source] [-d destination] [-MD5]’

This goImager will calculate the time automatically through the deduction of time from when it starts to end. GoImager will also calculate the hash function if the user sets the hash function command in the 2). For the double-check our hash checksum is validated, we also did the checksum test for acquisition source disk and the destination file in the window OS system. All the hash result was matched to the goImager hash calculation value.

5.5) speed test of basic acquisition tool

Based on the speed test of basic acquisition tool, we compared ftk imager cli version to goImager. Both tests were run in the same environment. Both were run at the same condition to do raw disk image.

	FTK IMAGER CLI VER	GO IMAGER	GAP/WINNER
HDD to HDD	2:17:54	3:51:35	1:33:41/FTK
SSD to HDD	0:42:24	1:41:18	0:58:54/FTK

Table 6 basic goImager speed test without RAM cleaning

FTK imager turned out to be faster to acquire the image. We made another hypothesis, the reason why goImager is slower compare to the FTK imager is ram cleaning. FTK imager might clear the ram before doing the acquisition. This way, it will helps ram to give more access to using goImager.

	FTK IMAGER CLI VER	GO IMAGER	GAP/WINNER
HDD to HDD	2:17:54	3:14:56	0:57:02/FTK
SSD to HDD	0:42:24	0:43:35	0:01:14 /FTK

Table 7 basic goImager speed test with RAM cleaning

There was big difference between speed. We made hypothesis that it is because of the ram disk. Based on that hypothesis, we run the goImager with Ram clearing. We cleared the ram cache through sync; echo 3 > /proc/sys/vm/drop_caches in su root mode.

	GoImager without RAM cleaning	GoImager with RAM cleaning	Time difference
HDD to HDD	3:51:35	3:14:56	0:36:39
SSD to HDD	1:41:18	0:43:35	0:57:43

Table 8 with&without ram cleaning

In the chapter talked about own imager written in Go language. Based on the wish list of features, we decide the structure of the goImager. Based on drawing, we wrote code to do disk image and successes. To make basic version of goImager, we used to run the go Imager.

By cleaning the ram, goImager was able to reduce the time. At the same time, we also figured that larger storage medium delays goImager to finish acquiring the image. It is important to consider what we can improve the basic acquisition tool based on considering the features that mentioned above in go lang.

제 6 장 IMPROVED FORENSIC ACQUISITION TOOL

In the basic go Imager, paper described Basic goImager as able to do physical disk imaging, support raw disk image and support md5 hash checksum. Then also explained that improved version of forensic imaging tool can be define as having more features compare to the basic features.

In this chapter six, this paper narrowed down to the types of data, digest hash algorithm, file format and speed section and explained how we improved and what should be more included.

6.1) types of data

6.1.1 logical imaging

physical imaging is mainly done by reading this code 'getSize, _ := oSize.Seek(0, io.SeekEnd)'. When user types the source disk such as /dev/sda then it reads the size of physical disk from 0 to end to do physical imaging. Just like this, goImager is able to read the size of the logical disk by typing /dev/sda6. By doing the logical imaging, it will help the forensic practitioner to save the time later.

6.1.2 user-define disk imaging

user define disk imaging is also available in goImager. By user define the specific file name with the file location, then goImager make user-defined imaging. Through this imaging method, it will support forensic investigator to save more acquisition time when they know where to look at it.

6.2) digested hash algorithm

6.2.1 Basic goImager only contains md5 hash algorithm.

In golang, it has package "crypto/md5[49]". Using crypto/md5, goImager can compare the source disk and imaging file of the destination. But this can be possible in all the forensic imager. golanguage does not only support md5, with the package, it will scan the user's command and do the hash comparision when target disk imaging is over. goImager supports not only md5 but also sha1,sha256,sha512.sha3-

256 and sha3-512 if user select it. If user does not select any of the hash function, then it will just do a disk imaging by a default.

6.3) File Format

Currently EWF file format and AFF file format is not putted in the goImager. If EWF and AFF file format is putted in the goImager, it will automatically do the compression and reduce the time to do acquisition.

6.4) SPEED : Golang features to improve the acquisition tool

Speed of the goImager can be faster through optimizing and adapting the concurrency. Even more, if buffer size can be automatically calculated based on the RAM, it might reduce the number of reading and speed of reading time.

6.4.1 Concurrency

Basic goImager works this way. After all calculation is over, goImager read certain amount of size and hand it to the buffer. In the content in the buffer goes to the destination. In the structure of basic goImager, while computer is reading, it cannot write to the destination file. If reading the source disk and the writing to the destination file at the same time, can expect the duration of imaging will be much shorter and it will make goImager much faster.

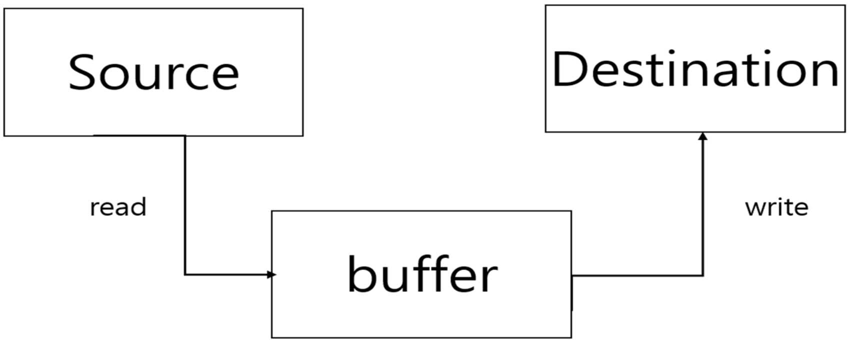

Figure 2 basic goImager concurrency1

To do concurrency, it is necessary to cut this loop. Make reading function and writing function independent first, then connect with the buffer channel which can help concurrency of goImager.

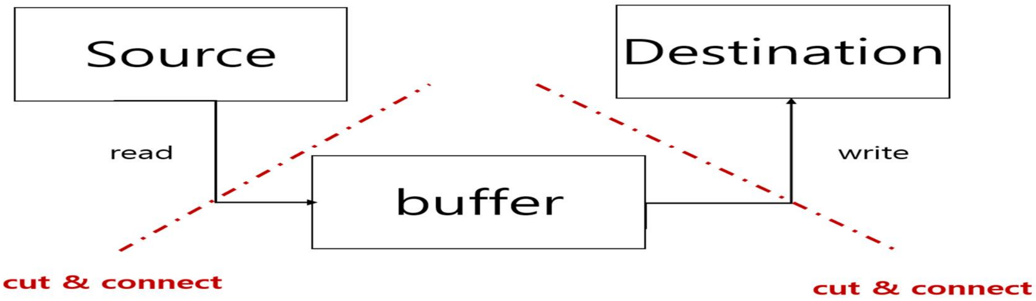

Figure 3 basic goImager concurrency2

Using the buffered channel, reading function will keep bring the context to the buffer in order, and the reading function will receive the context.

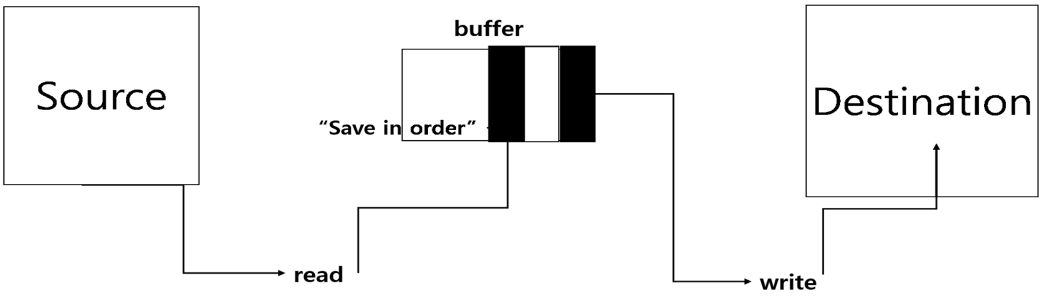

Figure 4 basic goImager concurrency3

6.4.2 Go routine

Go routine is light version of thread. In the code, put go in front of the function. Through the go routine, it will make reading function and write function run almost at the same time. By using Go routine, goImager have similar speed of reading and writing and this will enhance the concurrency of the goImager and effect the speed of imaging. If we fix this part, the speed of goImager will be much faster.

6.4.3 Result of improving speed through concurrency and Go routine

Based on the hypothesis, this paper considers the golang feature concurrency and Go routine will make goImager to improve their speed. For the concurrency, this paper used buffered channel for connecting reading function and the writing function. In this function used the buffer and buffered channel.

6.4.3.1 concurrency

	FTK IMAGER CLIVER	GO IMAGER	GO IMAGER with concurrency	GAP/WINNER
HDD to HDD	2:17:54	3:51:35	5:36:14	1:44:39/GO
SSD to HDD	0:42:24	1:41:18	2:49:32	1:08:17/GO

Table 9 Comparison for GoImager and GoImager with concurrency

This paper considered that concurrency would help based on the research question. However, it turned out that goImager with concurrency made goImager even slower. For example, table 5 is the result of goImager and goImager with concurrency without RAM cleaning. When running HDD to HDD on GoImager, it took 3 hours 51 minutes and 35 seconds, however, with the goImager, it took 5 hours 36 minutes and 14 seconds to make a RAW disk image. GoImager with concurrency consumed 1hour 44 minutes and 39 seconds more than goImager. For the SSD to HDD, there was 1 hour and 8 minutes and 17 seconds gap between goImager and goImager with concurrency. One of our research questions was false. GoImager with concurrency does not work, and there are three possible hypotheses based on this. GoImager with concurrency's buffer size was too small. However, there was no option because Go language says there is a limit in the size of the data we can send through the channel.

제 7 장 CONCLUSION

The forensic acquisition tool is quite essential in the digital forensic investigation procedure. By acquire the forensic image properly, it not only enhances the integrity of the forensic image but also the digital forensic itself. However, many forensic acquisition software developers and manufacturers do not update their tools. Base on the research and survey, this paper figured out what forensic acquisition has for the features. Using that information, we designed the basic forensic acquisition tool written in golang. Fortunately, a forensic acquisition tool did work to make raw disk images with log, checksum.

Nevertheless, there is more possibility to improve this forensic acquisition tool to the other acquisition tool that searched for the research and might beyond the features in the speedway. By adding more features, there is a possibility to have a forensic acquisition tool. Through the test over and over, this paper will come up with an improved version of the forensic acquisition tool, and it will be updated over and over.

Just like us, we recommend other forensic practitioners and developers to test and improve their forensic acquisition tools. Since they updated their tools, the size of the medium gets more prominent, and it directly affects the forensic acquisition time. Speed can be one of the measurements to judge whether the forensic acquisition tool is improved or not. The forensic investigation does depend on the time many of the time.

This paper had been talked continuously about the importance of improving the digital forensic acquisition tool and tried to answer whether can we improve it. To do so, we must understand what forensic acquisition is, is there a need to improve the forensic acquisition tool. Through the background research of previous work related to imaging tools, we found out that there is a need to improve the forensic acquisition tool, and the possible place to fix the forensic imaging tools is appending more features to the tool itself. In the background research, we found a possible place to improve can be put more hash function, add more file support type and forensic imaging type, put an encryption function, and enhance the data recovery.

Another central hypothesis is, can we make one forensic imager and improve it. Because of this, we need a sample of the other forensic imager. By surveying the forensic imager, pointed the FTK imager as a compete for imaging tool to compare the features.

While studying and developing the acquisition tool, we draw the structure of our forensic acquisition tool and planned the afterward of done developing. Based on that picture, we wrote the tool to go language and got success in making one. We called this as an goImager. The speed test result of goImager and FTK imager is close when the size of medium storage is small. As the medium of the source targets gets bigger and bigger, goImagers speed starts to get behind of FTK imager.

Based on these results, this paper came up with the method to improve the Basic goImager in chapter6. By adding the more hash algorithm through the go language package 'crypto,' goImager can have md5, sha1, sha256, sha512, sha3-256, and sha3-512 as options.

For the speed part, goImager tried to use golang features, concurrency to improve the speed. However, acquisition time took longer than basic goImager.

Furthermore, we provide logical and user-select disk imaging in the type of forensic imaging section. For the speed part, goImager tried to use golang features, concurrency to improve the speed. However, acquisition time took longer than basic goImager. Future work of goImager is simple. Improving and updating goImager will continuously happen. We considered putting more file format into the goImager, mainly the next target is EWF file format. We also consider enhancing the speed of GoImager. The reason why concurrency did not work seems to be the size of the buffer was too small to send data because of golang channel's limited data size.

Sending the bytes as bytes array through the channel, converting bytes again from reading to writing function seems another reason why golang with concurrency makes slow. In the future, we will remove the concurrency by removing the buffer channel. Instead of dividing the reading and writing function, goImager will maintain the goroutine for the GoRoutine to improve the speed.

제 8 장 REFERENCE

[1] PRONEER, “disk imaging,” *http://forensic-proof.com*, 2012. [Online]. Available: http://forensic-proof.com/archives/3613. [Accessed: 09-Dec-2019].

[2] G. Horsman, “Tool testing and reliability issues in the field of digital forensics,” *Digit. Investig.*, vol. 28, pp. 163–175, 2019.

[3] F. Flandrin, W. J. Buchanan, R. Macfarlane, B. Ramsay, and A. Smales, “Evaluating Digital Forensic Tools (DFTs),” *7th Int Conf Cybercrime Forensics Educ. Train.*, no. January 2015, pp. 1–16, 2014.

[4] “No Title.” [Online]. Available: https://toolcatalog.nist.gov/. [Accessed: 27-Sep-2019].

[5] S. Garfinkel, “Lessons learned writing digital forensics tools and managing a 30TB digital evidence corpus,” *Proc. Digit. Forensic Res. Conf. DFRWS 2012 USA*, vol. 9, pp. S80–S89, 2012.

[6] V. Roussev, “Hashing and data fingerprinting in digital forensics,” *IEEE Secur. Priv.*, vol. 7, no. 2, pp. 49–55, 2009.

[7] S. Vandeven, “Forensic Images: For your Viewing Pleasure,” 2019.

[8] S. Garfinkel, D. Malan, K. Dubec, C. Stevens, and C. Pham, “Advanced forensic format: An open extensible format for disk imaging,” *IFIP Int. Fed. Inf. Process.*, vol. 222, pp. 13–27, 2006.

[9] Library of Congress, “Expert witness disk image,” 2017. [Online]. Available: https://www.loc.gov/preservation/digital/formats/fdd/fdd000407.shtml. [Accessed: 16-Sep-2019].

[10] Library of Congress, “Expert Witness Disk Image, EnCase E01 Bitstream,” 2015. [Online]. Available: https://www.loc.gov/preservation/digital/formats/fdd/fdd000408.shtml. [Accessed: 11-Nov-2019].

[11] Library of Congress, “Expert Witness Disk Image, EnCase Ex01 Bitstream,” 2015. [Online]. Available: https://www.loc.gov/preservation/digital/formats/fdd/fdd000410.shtml. [Accessed: 08-Nov-2019].

[12] Joachimmetz, “libewf,” 2017. [Online]. Available: https://github.com/libyal/libewf/blob/master/documentation/Expert Witness Compression Format (EWF).asciidoc. [Accessed: 07-Nov-2019].

[13] L. of Congress, “Expert Witness Compression Format, EnCase Lx01 Logical,” 2015. [Online]. Available: https://www.loc.gov/preservation/digital/formats/fdd/fdd000411.shtml. [Accessed: 14-Nov-2019].

[14] Library of Congress, “Expert Witness Disk Image, ASR SMART,” 2015. [Online]. Available: https://www.loc.gov/preservation/digital/formats/fdd/fdd000407.shtml. [Accessed: 18-Nov-2019].

[15] Library of Congress, “Advanced Forensic Format Disk Image, AFF Version 1.0,” 2015. [Online]. Available: https://www.loc.gov/preservation/digital/formats/fdd/fdd000412.shtml. [Accessed: 18-Nov-2019].

[16] Copycat, “Understanding the har disk,” 2013. [Online]. Available: https://webdir.tistory.com/160. [Accessed: 13-Oct-2019].

[17] E. Casey, G. Fellows, M. Geiger, and G. Stellatos, “The growing impact of full disk encryption on digital forensics,” *Digit. Investig.*, vol. 8, no. 2, pp. 129–134, 2011.

[18] V. Roussev, “Chapter 15 DATA FINGERPRINTING,” *Ifip Int. Fed. Inf. Process.*, vol. 337, no. January 2010, pp. 207–226, 2010.

[19] D. Povar and V. K. Bhadran, “Forensic Data Carving,” 2011, pp. 137–148.

[20] D. Byers and N. Shahmehri, “A systematic evaluation of disk imaging in EnCase®6.8 and LinEn 6.1,” *Digit. Investig.*, vol. 6, no. 1–2, pp. 61–70, 2009.

[21] PRONEER, “Computer Hardware (2) - storage medium,” 2009. [Online]. Available: http://forensic-proof.com/archives/306. [Accessed: 11-Nov-2019].

[22] A. Forensics, “An Overview of the Hard Disk,” 2019. .

[23] F. Geier, “The differences between SSD and HDD technology regarding forensic investigations,” p. 67, 2015.

[24] PRONEER, “SSD Forensics:Data Recovery,” *Forensic-Proof*, 2012. [Online]. Available: http://forensic-proof.com/archives/3221. [Accessed: 10-Nov-2019].

[25] S. S. R. Marupudi, “Solid State Drive : New Challenge for Forensic Investigation,” p. 100, 2017.

[26] A. Technology, “disk imaging.” [Online]. Available: https://atola.com/products/insight/disk-duplication.html. [Accessed: 23-Sep-2019].

[27] C. Stelly and V. Roussev, “Nugget: A digital forensics language,” *DFRWS 2018 EU - Proc. 5th Annu. DFRWS Eur.*, vol. 24, pp. S38–S47, 2018.

[28] B. L. Schatz, “Wirespeed: Extending the AFF4 forensic container format for scalable acquisition and live analysis,” *Proc. Digit. Forensic Res. Conf. DFRWS 2015 USA*, vol. 14, pp. S45–S54, 2015.

[29] H. van Beek, “CASE Cyber-investigation Analysis Standard Expression Community update,” 2019. [Online]. Available: https://evidence2e-codex.eu/p/d/1/d1-02-e2eworkshop20190326caseoverview-419.pdf. [Accessed: 20-Nov-2019].

[30] “CASE,” 2019. [Online]. Available: https://caseontology.org/. [Accessed: 24-Nov-2019].

[31] Belkasoft, “Belkasoft Acquisition Tool.” [Online]. Available: https://belkasoft.com/bat. [Accessed: 22-Sep-2019].

[32] MAGNET, “MAGNET AXIOM.” [Online]. Available: https://www.magnetforensics.com/products/magnet-axiom/. [Accessed: 21-Oct-2019].

[33] S. Technology, “DATA RECOVERY SYSTEM.” .

[34] D. C. C. Center, “dc3dd package description,” *KALI TOOLS*. [Online]. Available: https://tools.kali.org/forensics/dc3dd. [Accessed: 19-Oct-2019].

[35] P. Software, “osForensic.” [Online]. Available: https://www.osforensics.com/. [Accessed: 22-Sep-2019].

[36] “guymager homepage.” [Online]. Available: https://guymager.sourceforge.io/. [Accessed: 22-Sep-2019].

[37] “guymager Package Description,” *KALI TOOLS*. [Online]. Available: guymager Package Description. [Accessed: 25-Sep-2019].

[38] X-Ways, “X-Ways Forensics: Integrated Computer Forensics Software.” [Online]. Available: http://www.x-ways.net/forensics/. [Accessed: 22-Sep-2019].

[39] O. Security, “EnCase Forensic Imager.” [Online]. Available: https://www.guidancesoftware.com/document/product-brief/encase-forensic-imager. [Accessed: 17-Sep-2019].

[40] A. Data, “FTK Imager.” [Online]. Available: https://accessdata.com/product-download/ftk-imager-version-4-2-1.

[41] A. Data, “Command Line Versions of FTK Imager.” [Online]. Available: https://accessdata.com/product-download/debian-and-ubuntu-x64-3-1-1. [Accessed: 19-Sep-2019].

[42] “dcfldd.” [Online]. Available: http://dcfldd.sourceforge.net/. [Accessed: 18-Sep-2019].

[43] Hoffmann, “ewfacquire(1) - Linux man page,” 2010. [Online]. Available: https://linux.die.net/man/1/ewfacquire.

[44] D. Guide, “Golang: the simple programming language from Google,” 2014. [Online]. Available: https://www.ionos.co.uk/digitalguide/server/know-how/golang/. [Accessed: 19-Oct-2019].

[45] “Package os.” [Online]. Available: https://golang.org/pkg/os/?m=all. [Accessed: 06-Dec-2019].

[46] “package log.” [Online]. Available: https://golang.org/pkg/log/. [Accessed: 15-Nov-2019].

[47] B. Johnson, “Go Walkthrough: io package,” 2016. [Online]. Available: https://medium.com/go-walkthrough/go-walkthrough-io-package-8ac5e95a9fbd#.41qxgc1zr. [Accessed: 06-Dec-2019].

[48] M. Castilho, “Calculating Multiple File Hashes in a Single Pass,” 2015. [Online]. Available: http://marcio.io/2015/07/calculating-multiple-file-hashes-in-a-single-pass/. [Accessed: 06-Dec-2019].

[49] “Package crypto.” [Online]. Available: https://golang.org/pkg/crypto/. [Accessed: 18-Dec-2019].

Appendix

<Appendix 1 > concurrency in goImager

```
var bufChan = make(chan buf, 20000)

Func read() {

          reader  := bufio.NewReader(src)

          var buffer [1 * KB]byte

          for {     bytesReadm err := reader.Read(buffer[1:*KB])

                    if err ! = nil {

                              if err == io.EOF {

                                        fmt.Println("closing buffchan")

                                        close(bufchan)

                                        break

          }else{

                    panic(err)

}

}

  sending :=buf{bytesRead,buffer}
  bufChan <- sending
Func write() {
```

디지털 포렌식 데이터 수집 도구에 관한 연구

2019.12.19

국제석사학위논문

함지윤

국제학과

지도교수: 조슈아 아이작 제임스

디지털 포렌식 이미징은 디지털 포렌식 절차 중 가장 기본적인 절차이다. 점점 디지털 포렌식 조사에 착수할 시 한 케이스 당 이미징 하는 기기들의 양이 점차 늘고, 사람들이 구매하는 기기들의 사양이 점점 높아지지만 이미징 방법과 절차는 기본적으로 초기와 비슷하다는 점에서 디지털 포렌식 이미징 툴을 개선할 필요성이 있다. 디지털 포렌식 이미징 툴을 개선하는데 있어서 중요하지만 어려운 점은 이미징의 속도가 빠르면서 포렌식의 무결성을 잃지 않아야 한다. 무엇보다 다른 포렌식 이미징 툴이 가지고 있는 특성들을 지원해야 한다는 점이 중요하다. 이러한 점을 지키면서 본 논문은 디지털 포렌식 이미징 툴을 개발 및 개선한다.

주제어 : 디지털 포렌식, 디지털 포렌식 이미징, 이미징 도구, 디스크 이미징

A study on Digital Forensic Data Acquisition Tools

2019. 12.19

Master’s Degree

Ham, Jiyoon

Department of International Studies

Advisor Prof. Joshua I. James

Digital forensic Imaging is vital in digital forensic investigation process. As time changes, the number of digital devices increase in the forensic imaging filed and the spec of digital devices gets advance. However, the imaging methods and procedure have largely stayed the same. The most hard and important point of improving the digital forensic acquisition tool is that it should be fast and forensically sound. Moreover, it should support the features that other forensic acquisition on market. While maintain this point, this paper will develop and improve the digital forensic tool by adding more features through survey.

Keywords: Disk Imaging, forensic disk imaging, imaging features

국제학석사학위논문

A Study on Digital Forensic data acquisition tools

2019

함지윤