Improving Forensic Techniques for Acquiring Solid State Drives

By

Cecilia Pohlar

Supervisor

Dr. Ali Hadi

Abstract

Solid State Drives (SSDs) present many problems to digital forensic investigators due to the unpredictability of deleted evidence and the drive's inner processes. These inner processes, such as wear leveling and garbage collection, are deployed by the drive's controller chip and cannot be disabled by a user. Due to this, SSDs behave much differently than Hard Disk Drives (HDDs), and can result in varying forensic hash values and recoverability of files. It was identified that new guidelines and procedures should be developed for the forensic acquisition of SSDs, specifically to block these inner processes. Through testing different proposed solutions on various SSDs, a guide for law enforcement was created which outlines the best practices for acquiring SSD forensic images and preserving deleted evidence.

Acknowledgment

This research would not have been possible without the support of many people: my classmates from the Champlain College digital forensics program, who struggled through this past year with me; my mentor, Nick Butner, who helped me explore the initial topic sparked my interest in storage devices; my fiance, Kincaid, who supported me through long days testing and researching; and most importantly, Professor Ali Hadi, who always had kind and encouraging words when discussing my work, and pushed me to always strive for more.

Table of contents

Abstract **2**

Acknowledgment **3**

Table of contents **4**

Chapter 1: Introduction **1**

Motivation 1

Problem Statement 1

Objectives of This Work 2

Contribution 3

Research Limitations and Scope 3

Research Implications (put your "Constraints and Implications" writeup here) 4

Health and Safety/Environmental Concerns and Implications 4

Social and Cultural Concerns and Implications (These could include political concerns/implications.) 4

Legal Concerns and Implications (These might be similar to the ethical concerns/implications, or might be very different!) 5

Economic Concerns and Implications (These could be anything from manufacturing concerns to disruptive macroeconomic effects from a new technology.) 6

Ethical Concerns and Implications 7

Organization of the Thesis 7

Chapter 2: Background and Related Work **9**

2.1 Background 9

2.2 Technology of SSDs vs HDDs 12

2.2.1 Data Storage on SSDs 13

2.2.2 SSD Controller 16

2.2.3 Wear Leveling 17

2.2.4 Bad Block Management 18

2.2.5 TRIM 18

2.2.6 Garbage Collection 20

2.3 Previous Experiments 21

2.4 Existing solutions 23
2.5 Literature Review 26

Chapter 3: Methodology 37
3.1 Overview 37
3.2 Process Overview 38
3.2.1 Phase 1: Prepare the drive 39
3.2.2 Phase 2: Perform proposed solution 40
3.2.3 Phase 3: Analyze the drive 41
3.2.4 Analyze & Compare results 41

Chapter 4: Experiments and Observations 43
4.1 Overview 43
4.2 Materials 43
4.3 Setup 45
4.3.1 Secure Erase 46
4.3.2 Add test images, deletion 46
4.4 Perform proposed solution 47
4.4.1 No Write Blocker 47
4.4.2 Disable TRIM 48
4.4.3 Disable Automount 50
4.4.4 Factory Mode or Corrupt the Drive 51
4.5 Imaging and Analysis 52
4.5.1 Create Image 53
4.5.2 Analysis 1 53
4.6 Test Actions and Results 54
4.6.1 *Control* 55
4.6.2 Write blocker, TRIM 55
4.6.3 No write blocker, TRIM 57
4.6.4 Write blocker, no TRIM 58
4.6.5 No write blocker, no TRIM 59
4.6.6 Disable Automount 59
4.6.7 Factory Mode or Corrupt the Drive 60

Chapter 5: Results and Evaluations 63

5.1 Evaluation of Results 63
5.2 Recommendations 64

Chapter 6: Conclusion and Future Work 67
6.1 Conclusion 67
6.2 Recommendations and Future Work 68

References 70

List of Figures

Figure 2.1: HDD side-by-side with SSD 10

Figure 2.2: Floating Gate Illustration 13

Figure 2.3: SSD data storage types 15

Figure 2.4: SSD write and delete cycles 16

Figure 3.1: Methodology Flow Chart 39

Figure 3.2: Test .JPG file 40

Figure 4.1: SSD connected via disabled write blocker 46

Figure 4.2: Disabling Automount 51

Figure 4.3: FTK hash results 54

Figure 4.4: Autopsy view 57

List of Tables & Codes

Table 4.2.1: Drives 44

Table 4.2.2: Hardware used 44

Table 4.2.3: Software used 45

Table 4.4.1: Performed tests 47

Code 4: TRIM Commands 50

Table 4.6.1: HDD, Yes Write Blocker 55

Table 4.6.2: SSD TRIM enabled, Yes Write Blocker 56

Table 4.6.3: SSD TRIM enabled, No Write Blocker 58

Table 4.6.4: SSD TRIM disabled, Yes Write Blocker 58

Table 4.6.5: SSD TRIM disabled, Yes Write Blocker 59

Table 4.6.6: SSD TRIM enabled, Yes Write Blocker, Automount Disabled 59

Abbreviation

Abbreviation	Meaning
HDD	Hard Disk Drive
SSD	Solid State Drive
NAND	“Not And” Flash Memory Type
PCB	Printed Circuit Board
SLC	Single-Level Cell
MLC	Multi-Level Cell
TLC	Triple-Level Cell
QLC	Quad-Level Cell
EEPROM	Electrically Erasable Programmable Read-only Memory
FG	Floating Gate
SMART	Self-monitoring, analysis, and reporting technology
DRAT	Deterministic read after TRIM
DZAT	Deterministic zeros after TRIM
LBA	Logical Block Addressing

Chapter 1: Introduction

As Solid State Drives (SSDs) become more prevalent in devices because they are lighter in weight, faster, and more durable than Hard Disk Drives (HDDs), the forensics community must address the many issues that they present to investigators. SSDs have various inner processes which HDDs do not, that take place without user input whenever power is applied to the drive. These processes occur in order to improve the longevity and processing speed of the drive. Inner processes like wear leveling, garbage collection, and TRIM result in issues such as inconsistent hashes and an inability to recover deleted files. These inner optimization processes are triggered by the drive's controller chip, and they can't be easily disabled. Because these processes do not occur with HDDs, new protocols and technology must be developed in order to ensure the validity of SSD data in investigations.

1.1 Motivation

SSDs optimization processes can hinder a forensic investigation and the integrity of the data, so there is a need to prevent these processes from running. Blocking these processes will result in a stronger forensic image, specifically by preserving the same hash of the drive before and after creating a forensic image. This project was chosen in order to create improved forensic techniques for the acquisition of SSDs, to then share these techniques online.

1.2 Problem Statement

The admissibility of forensic evidence like Hard Disk Drives (HDDs) is decided largely by the preservation of data integrity, often determined by the data's hash value. In unprecedented times, Solid State Drives (SSDs) have inner optimization processes which run anytime power is applied to the drive, and the forensic hash value changes if they are run. As of right now, there is no way to easily disable these processes, as they run through the drive's controller chip and can't be disabled by the host computer. Existing solutions are expensive software, which are inaccessible to most local law enforcement agencies. The primary goal of this Capstone project is to create a better solution to these problems, specifically a guide to how to image SSDs while preserving the most evidence, as most existing solutions involve expensive software that is often unattainable for local law enforcement or forensic analysts without the support of large company budgets. In addition, this software is meant for data recovery, not digital forensic analysis. New guidelines and technology must be adopted to improve the validity of SSD images throughout a forensic investigation.

1.3 Objectives of This Work

The main objectives of this work can be summarized as the following:

(1) Create a solution for imaging SSDs and write a guide for forensic analysts who do not have access to expensive data recovery software,

(2) Develop deeper understanding of file systems, and

(3) Differentiate between forensic procedures for HDDs and SSDs.

1.4 Contribution

The most important concept when presenting evidence during a criminal investigation is that the evidence is legitimate and hasn't been tampered with. Due to the nature of SSD investigations right now, data integrity is understood to be impossible to the levels of integrity for HDDs. Investigators rely on detailed notes of their processes to submit to courts for admissibility, and deleted evidence is not always able to be recovered.

This research will greatly benefit the forensics field by establishing new possible standards for analysis and investigations. Making this research available to people online, specifically to local law enforcement, will impact forensic investigators' ability to solve investigations involving SSDs while preserving data integrity.

1.5 Research Limitations and Scope

This project specifically investigates SSD controller chips, but research and results are limited to only a select few types of controller chips due to the time and financial constraints. Specifically, Samsung and Kioxia (Toshiba) will be investigated because they make up such a large portion of the percentage of NAND flash memory controller chips. The findings from these popular controller chips can possibly be used to deduce similar tactics for other, less common controller chips, but emphasis will be placed on the most commonly used chips.

In addition, this research is limited to finding a solution through software manipulation, such as altering flags in the controller chip and trying different imaging techniques, and basic hardware manipulation, such as soldering and breaking

connections. More in-depth solutions such as combined hardware/software manipulation will not be accessible throughout this project due to financial costs and time constraints.

1.6 Research Implications (put your "Constraints and Implications" writeup here)

1.6.1 Health and Safety/Environmental Concerns and Implications

This Capstone project involves various proposed solutions to image an SSD while maintaining the hash value, some of which may involve hardware manipulation. These manipulations can involve removing certain components or connections through high heat applications involving a soldering iron. There are associated fumes with soldering these metal components, but safety equipment and a smoke absorber device can mitigate these concerns

1.6.2 Social and Cultural Concerns and Implications (These could include political concerns/implications.)

Traditionally, forensic investigators image HDDs using a write blocker, which is able to preserve the drive's state at the time of imaging. This doesn't work with SSDs because the inner processes occur despite the efforts of a write blocker. As more and more Solid State Drives are installed in devices, it is concerning that a solution as widely accepted as acquisition with HDDs doesn't exist yet. Whether they are educated in SSD technology or not, people can get away with computer crimes more easily due to the way evidence can be permanently deleted or altered while the SSD is connected to power. Oleg Afonin, an international digital forensics expert who has published many articles about SSDs, demonstrates this by saying that SSDs "are different in handling deleted

data, wiping evidence irreversibly in the background like they were criminals' best friends" (2019). Forensics experts should not permit the technological challenges of imaging SSDs to allow criminals to evade the law. Rather than accepting current limitations, the forensics field should pursue new standards for SSD investigation.

1.6.3 Legal Concerns and Implications (These might be similar to the ethical concerns/implications, or might be very different!)

The standard in the forensics community is that evidence must maintain a hash value throughout the legal investigation to ensure the evidence hasn't been tampered with. Evidence admissibility is what our legal system is built around, but this standard has been somewhat loosened recently due to SSD technology. All of the researchers reviewed for this project concur that there are no established standards for accessing information on an SSD. With HDDs, the standard has always been to use a write-blocker, which will prevent any user alterations to the drive while connected to the investigator's computer. This maintains the data's hash value. However, Gubanovis and Afonin described in their 2012 article that

> It is essential to realize that an SSD drive connected via a write blocking device will continue performing background garbage collection, possibly destroying the last remnants of deleted information from the disk [...] SSDs self-destroy court evidence, making it difficult to extract deleted files and destroyed information (e.g. from formatted disks) close to impossible.

With SSDs, most investigators currently use detailed notes to document the steps they took throughout analysis in order to show that they didn't tamper with the evidence, but the forensics community should be seeking a higher level of integrity in their work.

It is currently difficult for anyone except those with expert skills and custom hardware to extract all of the information from SSDs or maintain the drive's hash value (Gubanovis & Afonin, 2012). This Capstone project seeks to enable investigators to image a drive, including deleted information, and maintain the data's hash value.

This project will greatly impact the legal community, as it could impact the regulations surrounding evidence admissibility in the future. Making the research from this project publicly available online, specifically to local law enforcement, will impact forensic investigators' ability to solve investigations involving SSDs while preserving data integrity.

1.6.4 Economic Concerns and Implications (These could be anything from manufacturing concerns to disruptive macroeconomic effects from a new technology.)

The present solutions for blocking the inner processes on SSDs are extremely expensive and time-consuming, so a cheaper solution needs to be found for the future. Afonin described in his 2019 research article that "until very recently your only way of accessing deleted evidence on an SSD would be taking the chips off and performing a labour-intensive, time-consuming (let alone extremely expensive) chip-off analysis." The basis of this Capstone project is to find a way to make a forensic image of an SSD without involving expensive software or chip-off techniques so that these methods can be more widely available to local law enforcement and other people investigating SSDs.

There are many financial constraints involved in conducting this research, mainly the high cost of SSDs to use for testing. A few drives have been acquired from Professor Hadi, but additional small drives will be purchased if a solution can't be found with

these drives. In addition, the most popular SSD controller manufacturers will be considered and focused on, as imaging solutions may need to vary from manufacturer to manufacturer.

1.6.5 Ethical Concerns and Implications

Just as it's important to consider the legality of SSD evidence, it is also important to consider the ethics of allowing SSD evidence to be admitted if it doesn't have a verified hash value. Changes could have been made to the forensic image over the course of an investigation, but under current standards, differing hash values would be forensically acceptable because it is known that the inner processes of SSDs easily alter the hash value without user input. "Most importantly," Geier, who performed extensive research on how SSDs retain deleted evidence and change hash values, explains, "while restoring data from disks and gathering evidence, the original data must stay untouched and altered as little as possible" (2015, p. 27). Geier goes on to say that it is "impossible to prove integrity for a forensic examiner" while investigating a SSD (2015, p. 51). When such evidence is used unethically, it presents an opportunity to falsely incriminate a suspect. This research seeks to find a solution to this problem and to ensure that the digital forensics and law fields remain ethical as we move away from HDDs and towards SSDs.

1.7 Organization of the Thesis

This thesis is divided into six chapters. Chapter 1 introduces the topic. Chapter 2 interpretes detailed background information relating to SSDs, how they operate, and

possible solutions. Chapter 3 illustrates the methodology and basic procedures for the experimentation. Chapter 4 outlines the specific experiments and observations from them. Chapter 5 evaluates the results from the experiments. Lastly, Chapter 6 concludes the information found from this Capstone project and recommends future work for this topic.

Chapter 2: Background and Related Work

This section outlines the problem in more detail while providing additional information on the forces preventing SSDs from being investigated like traditional HDDs. A literature review is also included, as well as information on similar research done.

2.1 Background

Up until the 2000s, HDDs were the standard for most computers and devices. The first patent for SSD technology was in 1989 by Sandisk founders Eli Harari and Sanjay Mehrotra, along with Robert D. Norman. It was described at that time as “a system of flash EEprom (Electrically Erasable Programmable Read-only Memory) memory chips with controlling circuits serves as non-volatile memory such as that provided by magnetic disk drives” (Flash eeprom system, 1989). EEPROM memory chips do not lose data when the power supply is lost, so it is non volatile (Fa1c0n, 2020). This was a revolutionary technology at the time that is now present in most devices. Figure 2.1 illustrates the difference between the mechanical storage of HDDs and the electrical storage of SSDs.

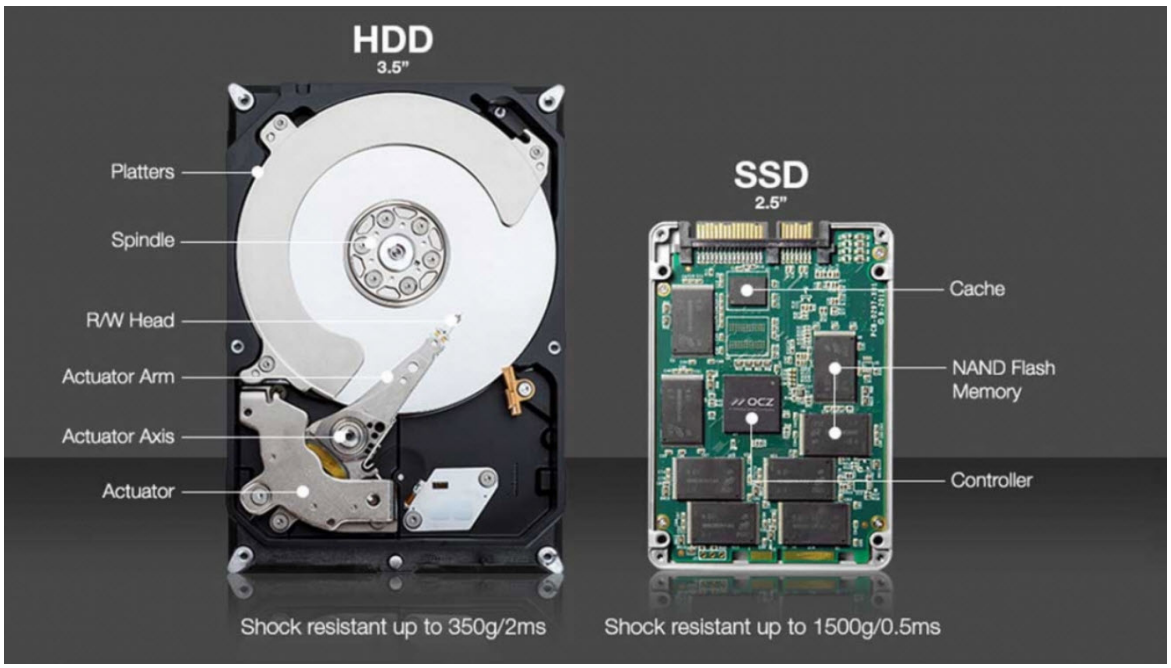

Figure 2.1: HDD side-by-side with SSD

While HDDs rely on mechanical moving parts, SSDs use electrical currents to store data. SSDs upgrade numerous shortcomings of HDDs by using NAND flash memory, but the two types of drives are vastly different in the ways that they operate. SSDs are substantially faster, use less energy, and are more resilient to physical damage than HDDs because of the lack of moving parts. That said, SSDs can only write and erase data a limited amount of times compared to HDDs, even though they can read an unlimited number of times (Uchiyama, 2014, p. 29). This is due to the electrical applications of the drive, which will be discussed in depth in the next section.

Optimization processes were created to prolong the life of SSDs and spread out use across the whole drive to prevent reaching this write limit. These processes, which are controlled by the SSDs controller and include garbage collection, wear leveling, and TRIM, “are active as soon as power is supplied to the device, and there is no way of

stopping them" (Uchiyama, 2014, p. 2). They also make it incredibly difficult to recover permanently deleted data and to ensure the integrity of any data on the drive. This poses a problem to investigators seeking forensic integrity in investigations. These processes will run as soon as power is applied to the drive, and one of the only ways to stop the processes, once started, is to disconnect the drive from power (Afonin, 2019). Commercial tools like Ace Labs PC30000 SSD Utility can put the drive essentially into factory or techno mode to stop the inner processes and read the NAND chips directly (Morozov, 2018). Factory mode is used by manufacturers to diagnose and repair SSDs and is not available without specialized expensive software, which most local law enforcement does not have access to. This research seeks to find an affordable and accessible solution to imaging drives without losing forensic integrity. For the purposes of this research, the goal is to ensure that the integrity measure equates to the data's hash value, maintaining deleted evidence, and is admissible in court.

This problem is important to consider as the technology behind SSD flash storage does not allow a high rate of recovery for deleted files, as HDDs do. This is because, "in contrast to hard disk drives, flash memory and in particular SSDs," explains Geier, "have internal routines that cannot be influenced from outside for example with a write blocker" (2015, p. 28). In traditional HDD analysis, a copy is made of the original drive and all of the forensic investigation is performed on that copy rather than the original. This process is difficult to perform with SSDs because the optimization processes could still be occurring on the original drive as it is copying data to the secondary drive. This can alter the data hash, making it impossible to verify the copy's integrity to the same level of accuracy as HDDs (Uchiyama, 2014, p. 2). In addition, SSD inner processes

“purge” residual deleted data in these background processes, which presents a huge issue to forensic data recovery analysts, who rely on this residual data to carve and recover files (Uchiyama, 2014, p. 2). SSDs can also purge data when the drive is reformatted, even in quick format mode (Fa1c0n, 2020). Compared to HDDs, which can’t even ensure fully permanent erasure after 1 overwrite pass, this is a new frontier in forensics (Fa1c0n, 2020). SSDs operate differently than HDDs, which this section seeks to provide a deeper insight into.

2.2 Technology of SSDs vs HDDs

The technology behind the two diverse types of drives is important to consider when analyzing how to approach a forensic investigation. HDDs are mechanical drives that use spinning platters and mechanical arms to read data through magnetic material. SSDs, on the other hand, use electric charges on NAND flash chips connected to a PCB to store information as bits for each charge. NAND is a type of flash memory, which means it stores information based on Floating Gate (FG) technology. FG entails “two overlapping gates, one completely surrounded by oxide and the other forming the gate terminal. If voltage is applied to the control gate, electrons can pass from the source through the dielectrics and settle on the floating gate” (Geier, 2015, p. 14). These electrons represent the data stored on each cell, so the amount of charge in each cell will represent either a 0 or 1 in binary. 0 is when electrons are present, and 1 is when the cell is empty or erased (Yohannes, 2011, p. 12). In order to delete the cell, these electrons are drained into the substrate for erasure.

This model guarantees years of charge retention, according to Uchiyama, but is still known to be shorter retention than HDDs, which retain data for decades depending on

the storage conditions (Uchiyama, 2014, p. 28, 39). The more times that an electrical current passes through the insulation of each cell on an SSD, the thinner that insulation becomes, and thus, "the fewer cycles of rewriting any particular chip can bear" (Morozov, 2018).

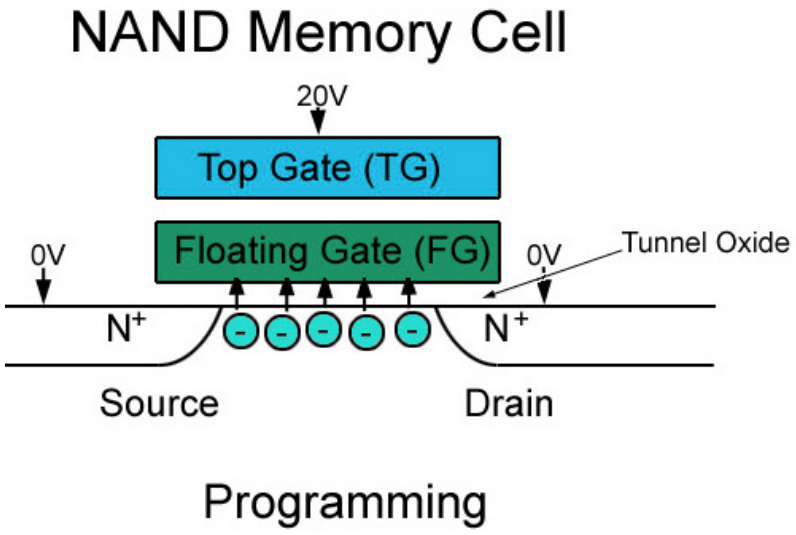

Figure 2.2: Floating Gate Illustration (Kronos, 2011)

2.2.1 Data Storage on SSDs

SSDs store information in contrasting ways from HDDs. Afonin describes that there is no linear writing for SSDs. Instead, data is written in small blocks across all chips on the drive. These chips contain a type of grid of electrical cells, which use the FG technology (Fa1c0n, 2020). Even if there is only one chip, data most likely won't be written in a linear fashion (2019). This makes SSDs much faster than HDDs, because there is no need for a spinning arm to move locations to read data. The SSD controller can just read each location quickly and continuously.

There are different types of storage methods for SSDS, which vary based on price and manufacturer. SSDs have four main types of capacities, which are measured by bits per cell: SLC (single-level cell - one bit), MLC (multi-level cell - two bits), TLC (triple-level cell - three bits), and QLC (quad-level cell - four bits) (Afonin, 2019). These NAND flash types have differing speeds and reliability levels, where SLC is the fastest and most reliable. For example, while SLC only needs to check if the cell is storing a 0 or a 1, MLC will need to check if the cell is storing 11, 10, 01, 00 because it contains two bits (Yohannes, 2011, p. 13).

Because of this, most drives employ SLC for the drive's cache, while making the actual storage MLC or TLC to allow more available storage space for a lower cost (Afonin, 2019). TLC storage can degrade to a critical level between 30 and 90 rewrite cycles, whereas SLC can take 30,000 to 100,000 (Morozov, 2018).

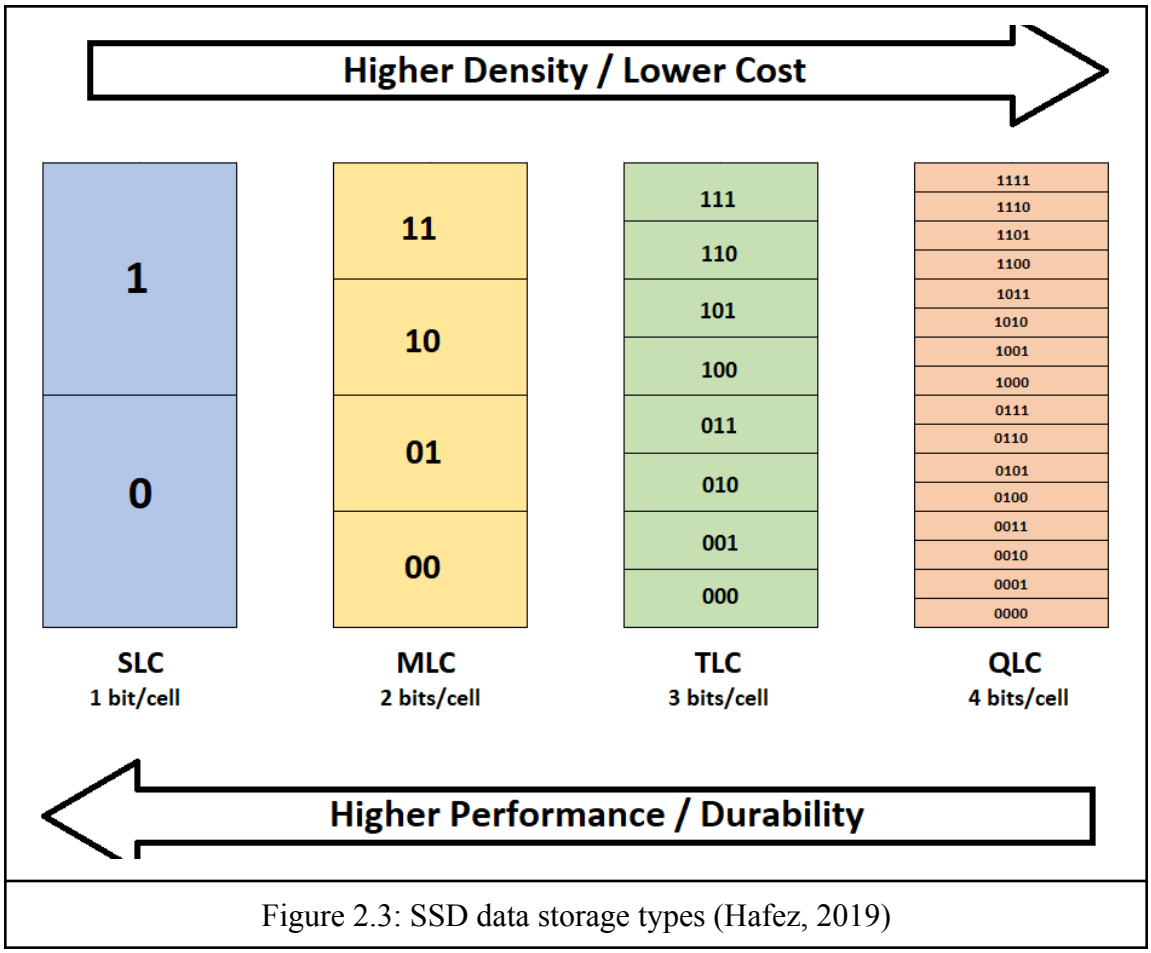

Figure 2.3: SSD data storage types (Hafez, 2019)

SSDs also have additional storage, which is considered non-addressable and known as the overprovisioned or reserve area, that are used by SSDs to temporarily store data while rewriting or moving data and can contain data. Essentially, the SSDs controller will use these extra data blocks when a fresh block is required for writing data, then completely erase the old block (Gubanov and Afonin, 2014). These portions of the drive are not visible to the average user, and specialized software to put the drive into factory mode is necessary (Afonin, 2019).

A page is the smallest unit of data which can be read from an SSD. Even if only one byte of data needs to be read, an entire page will be read. A page is normally 512, 2048,

4096, 8192, or 16534 bytes. Moreso, for data deletion, a specific amount of pages, also known as a block, is the minimum unit for erasure. SSDs must erase a block before writing new data to that block. This varies drive to drive, but is normally 64, 128, 256, or 512 pages (Afonin, 2019). So, to read data, a page is the smallest unit allowed, while a block is the smallest unit for deleting data. The example in Figure 2.3 illustrates how data is written by the page, which may be 4KB, but it can only be erased blocks, which may be 256KB.

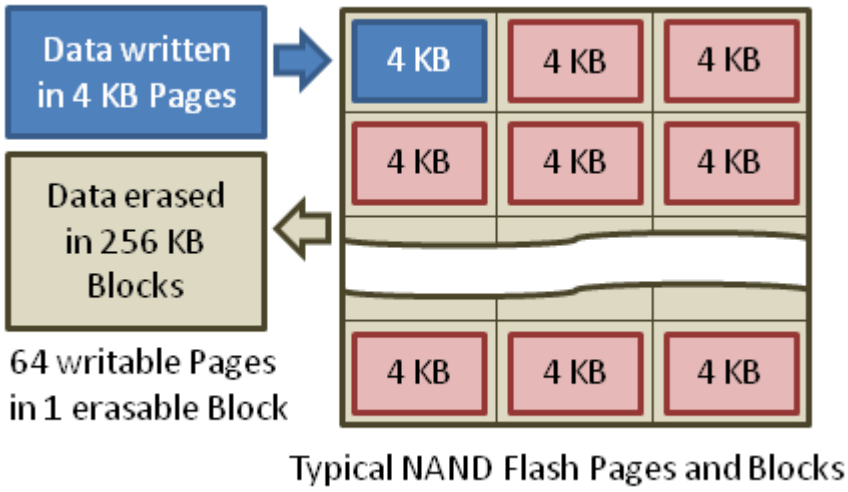

Figure 2.4: SSD write and delete cycles (Thatcher, 2009)

2.2.2 SSD Controller

The SSD controller "executes the codes that are provided by the SSDs firmware" (Fa1c0n, 2020). The controller is extremely important to the operation of the drive because it also controls all of the inner optimization processes except TRIM. This differs from HDDs, which only has some extra functionality, like S.M.A.R.T (Self-Monitoring,

Analysis and Reporting Technology) (Geier, 2015, p. 15). An SSDs NAND controller, on the other hand, is responsible for "SSD performance and reliability" with two main responsibilities: supplying the interfaces and protocols for the host and the NAND, and maximizing performance for handling data and distributing the processes (Uchiyama, 2015, p. 30). This includes the inner optimization processes like wear leveling. The controller acts as a "bridge between the NAND memory mechanisms and the computers using firmware level code," which allows the drive to perform optimization measures in the background during time the drive is idle. Essentially, "each time the computer needs to read or write data, it communicates with the controller and the controller translates that to the SSD, so the function can be carried out" (Lawson, 2018, p. 14-15). In addition, if encryption is enabled for a drive, most consumer SSDs hold the encryption key in the storage chip, unprotected. This chip also stores the logical addresses of all the data on the drive, so it is vital in descrambling all of the data and returning a readable format (Afonin, 2019). Morozov explains, "Almost all present-day SSDs have hardware data encryption. That's why the chip-off method becomes useless for accessing the digital evidence on a damaged device" (Morozov, 2018). The controller chip is necessary to descramble and decrypt any data on the memory chips.

2.2.3 Wear Leveling

Wear leveling is an inner process that essentially prevents data from being written to the same blocks over and over again. SSDs have very limited write operations available before data degradation sets in, so optimization processes were created to keep track of which blocks have been written to more than others. Wear leveling monitors each page and block to keep data distribution consistent and limit processes being centered on a

certain block too much (Uchiyama, 2015, p. 32). There is no way to stop this process from occurring. “When writing data,” Afonin explains, “the SSD controller will try to choose blocks with the least rewrite count” (2019). Information is spread out and written to these blocks which are then assigned logical addresses. The logical addresses are known by the drive’s controller, which in turn is able to descramble all of the data and return a readable format. The SSD controller is imperative to be able to piece together the information stored throughout the drive (Afonin, 2019).

2.2.4 Bad Block Management

Wear leveling limits the number of retired cells, but it is still bound to happen. Bad block management keeps a map of obsolete cells, which provides a better path to open cells (Uchiyama, 2015, p. 32). Bad Block Management is another optimization process that is used for keeping track of which cells are nearing the limits for write maximums. When a cell reaches its write limit, which is calculated by the silicon size and bit density, the cell will be “retired” through Bad Block Management and kept track of by the drive’s controller chip (Uchiyama, 2015, p. 29).

2.2.5 TRIM

SSDs differ from HDDs also in that they must delete data before new data can be written to the cell. Geier explains that “the TRIM functionality erases blocks that have been marked to be deleted by the operating system” (Geier, 2015, p. 4). For HDDs, data can just be overwritten. “NAND flash delivers fast reads, slow writes and very slow erases” because of the process necessary for deleting (Afonin, 2019). The process in which a block is deleted involves marking the block for TRIM, queuing it for Garbage Collection, then Garbage Collection fully deleting it. This includes a period in which

data may be able to be read while queued for Garbage Collection. Gubanovis and Afonin explain that it is a common misconception “that discarded blocks of an SSD drive are immediately erased… instead, [TRIM] adds them to a queue of pending blocks for being cleared by the garbage collector” (2012).

This period results in varying data returns when trying to read the data from that block. It depends on the type of TRIM used by the controller. Non-deterministic TRIM can return the actual data, all zeroes, or something in between. Suhanov specifies that successful recovery of deleted data is most likely the result of non-deterministic TRIM on a drive (2020). Deterministic read after TRIM (DRAT), the most common type in SSDs, will return a predefined value which is the same no matter what the content is. Lastly, deterministic zeros after TRIM (DZAT) will return all zeroes. Lawson describes DRAT as “the most helpful form of TRIM” due to the period of time before new data overwrites the deleted section, where recovery is possible (2018, p. 16). That said, most experts researched for this project disagreed. Afonin explained that DRAT and DZAT essentially prevent the ability to read data after the TRIM command has been sent and before the garbage collection process has begun (Afonin, 2019). It is important to acknowledge the different possible returned values when investigating drives with deleted data in which TRIM is enabled. It is also vital to understand that processes like wear leveling and garbage collection can not be stopped by a user, but TRIM can be stopped by a standard writeblocker because it is executed by the operating system (Uchiyama, 2015, p. 39). TRIM is enabled by default on most SSDs except in the cases of RAID controllers, USB enclosures, and some Linux distributions (Suhanov, 2020).

2.2.6 Garbage Collection

"The moment data has been deleted," writes Uchiayama, "an artifact of deleted data on the memory cells are facing the risk of the automatic wiping algorithm known as garbage collection" (Uchiyama, 2015, p. 39). Garbage collection works as a background process in relation to TRIM. When a block is deleted by the operating system, it will be flagged for TRIM. Once the block is marked by TRIM, it will be queued for garbage collection, which will then completely erase the contents of that block. Because there is a queue, this may not be erased immediately. If there are a lot of blocks to process or if the drive is a slow speed of NAND flash, then it could take upwards of 15 minutes to finish cleaning the trimmed blocks (Afonin, 2019). Geier explains, "different SSDs are expected to show different behaviour when deleting data," so it can be unpredictable (2015, p. 4). Because garbage collection is an inner process controlled by the SSD controller, write blockers are known to have no effect in "protecting data from deletion". Additionally, because inner processes like garbage collection cannot be stopped by traditional forensic techniques, they can alter the drive in the time it takes to create a forensic image, which can alter the forensic hash value of the data (Lawson, 2018, p. 15).

With the process of wear leveling, the goal is to keep as many free blocks open as possible, given the amount of data written on the drive. SSDs are not able to overwrite data in the same way as HDDs. Instead, a process called Logical Block Addressing (LBA) assigns a new block to copy the entire block of new information, and marks the old block for trash (Uchiyama, 2014, 38). This employs Garbage Collection as well to combine leftover data from various cells to free up more space by defragmenting data

and reallocating it when the volume of free space drops below a specific point (Geier, 2015, p. 4). Neyaz, Shashidhar, and Karabiyik identified that "When a file is stored on a solid-state media, it stays there as long as it is allocated in the media's storage space. However, once it is deleted from the media, it starts to get corrupted, modified or can be overwritten, thereby rendering the original file to be completely lost. Hence, it renders forensic analysis difficult and error prone" (2018, p. 1706).

2.3 Previous Experiments

Beyond the research done by various experts in the field, some noteworthy experiments include that of Uchiyama and Geier. Uchiyama tested various forensic acquisition guidelines traditionally used for HDDs to see success levels when used for SSDs. Uchiyama tested several SSD models and found that complete recovery of deleted files can be very difficult, but not impossible. They also found that smaller files can be recovered more easily and that disabling TRIM "does not stop the firmware's destructive behavior" (Uchiyama, 2014, p. 109). Uchiyama also makes the distinction that the guidelines for SSD forensics should vary from that of HDDs, and "it is unknown why SSD was treated like HDD to start with" (2014, p. 109).

Geier's experiments tested recovery rates of deleted files on HDDs compared to different NAND technologies, like SSDs and USBs. In Geier's experiments, it was found that HDDs had recovery rates of deleted data of over 99%, while only 0-5% of deleted data from SSDs could be recovered, despite imaging taking place immediately after file deletion (p. 40-42). In addition, another experiment by Geier yielded results of different hashes on SSDs when analyzing a file immediately after deleting a file and again three hours later. HDDs were used as controls in this experiment and their hashes

remained the same, proving that the internal processes in SSDs do impact data on the drive over a period of time (2015, p. 48). “Another interesting finding,” Geier writes, “was the impact of the TRIM function on the recovery rate. If disabled before deletion the tested SSDs would act like HDDs and would have a recovery rate from 85-99%” (Geier, 2015, p. 52). Vierya et al concurs this, explaining, “In SSDs without TRIM, nothing is telling the SSD to clear data with garbage collection” (n.d., p. 14). That said, this does not pertain to most investigations as TRIM is enabled by default on SSDs and especially if evidence has been deleted purposely.

Lawson’s experiments are closely related to the goals of this capstone research project. Lawson performed experiments on SSDs involving testing the effectiveness of write blockers and the effect of disabling automount on recoverability of files. Interestingly, Lawson concluded that, for their tests, “the MD5 hash stayed the same, regardless of being connected with a write blocker for 4+ hours. This may suggest that the garbage collection doesn’t work while connected to a write blocker” (2018, p. 38). Preserving the MD5 hash through imaging is incredibly important for preserving the integrity of the drive as much as possible. Lawson also found that, while no solid evidence supported this, there were suggestive results of the SSD self-corroding *less quickly* when a write blocker was connected to the drive, i.e. more files were recoverable (2018, p. 61).

Additionally, Lawson’s tests on automount coincide with Ferraria’s article on the effectiveness of disabling automount in order to “render the TRIM function and garbage collection ineffective” (Ferraria, 2018; Lawson, 2018, p. 50). Disabling automount resulted in the best preservation and recoverability of deleted files for the tested SSDs (Lawson, 2018, p. 58).

2.4 Existing solutions

The basis of this project is to find a solution to creating a forensic image while maintaining the original hash value and collecting as much deleted data as possible. Geier explains that "while restoring data from disks and gathering evidence, the original data must stay untouched and altered as little as possible" (2015, p. 27). This currently isn't possible as write blockers are known to not have an effect on SSDs inner processes, as they are controlled by the drive's controller chip and can't be stopped by the host computer. That said, write blockers will disable TRIM commands from being sent from the host computer to the drive. Lawson describes the following in their report:

> Bell and Boddington (2010) tested the theory that the presence of a write blocker may be able to interrupt the implementation command for the garbage collection and TRIM functions. They suggested that some computers may send the triggering signals via the SATA channel to the SSD, which in the presence of a USB writeblocker may be interrupted and ultimately disabled… Unfortunately, the outcomes were inconclusive as they were unable to determine the reason for the data alterations which occurred with proof" (Lawson, 2018, p. 15)

In addition to using a write blocker, many investigations involve turning off TRIM to prevent further commands from being sent and investigating whatever information is available on the drive, but this doesn't include deleted evidence. Processes like wear leveling and garbage collection can not be stopped by a user, but TRIM can be stopped by a standard writeblocker because it is executed by the operating system (Uchiyama, 2014, p. 39).

In the same vein as disabling TRIM, disabling automount from the host computer is thought to have a positive effect on the recoverability and preservation of data for an SSD. As explained previously, Lawson and Ferraria both concluded the effectiveness of disabling automount. "When automount is disabled, it has the potential to bypass the garbage collection function as the computer should not be registered to the device" (Lawson, 2018, p. 50). Both of these changes to the host computer appear to be promising at least in potentially preventing further destruction of evidence from the drive once connected to a power source for imaging, at no cost to the user.

However, there are a couple existing solutions beyond this, though they are expensive or time consuming. One such solution is to perform chip-off on the drive. "Until very recently your only way of accessing deleted evidence on an SSD would be taking the chips off and performing a labour-intensive, time-consuming (let alone extremely expensive) chip-off analysis", Oleg Afonin explained in a 2019 blog post. Uchiyama described the best method for preserving data to be immediately removing the SSD from the power source, removing the NAND flash memory chips from the PCB, then using a chip reader to view and copy the data (2014, p. 40-41). That said, this presents many issues. An investigator or analyst must have deep knowledge of file systems to properly reconstruct the data, because SSDs allocate data much differently than HDDs. Unlike HDDs, SSDs do not store data sequentially, so it can be extremely difficult to match the fragments. Encryption only exacerbates this problem, even if the password is provided. The SSD controller chip stores all of the information for decryption and defragmentation, and "different SSD controllers employ different formats for such translation tables; the more NAND chips the SSD drive contains, the more difficult it will be to reconstruct the translation table" (Afonin, 2019). Another issue with

this process is that chip-off and soldering requires a lot of knowledge and experience because a lot of harm can be done to the chips if they are removed incorrectly. This is seen as a very difficult solution.

A more modern solution is the use of software such as Ace Laboratories PC-3000 Express. Activating the internal processes can be circumvented by reading single memory chips, similar to chip-off but without the physical extraction part. Only a few tools are capable of this, which are mainly from Russian developers: PC-3000 Flash SSD Edition, Dumppicker, Flash Extractor, and Flash Doctor (Geier, 2015, p. 28). These tools are able to preserve forensic integrity while investigating the drive by reading the contents of a single memory chip, then uses information from the chip manufacturer to recover the files as needed. That said, "SSD controller's manufacturers face very strong competition and are not willing to share the insight of the internal routines, encryption, wear leveling and garbage collection" with many software developers (Geier, 2015, p. 28).

Essentially, tools like the PC-3000 put the drive into factory mode, which is the setting used by manufacturers to make changes to a drive, which prevents background processes from starting (Afonin, 2015). Then, a forensic image can be taken. This software is extremely useful, but it is also extremely expensive and unavailable to most local law enforcement investigators. Geier describes the problem in their 2015 degree project:

> It appears that the key to the differences and unpredictability in data recovery lies in the firmware of the memory controller. There are only a handful of manufacturers producing SSD memory controllers and they are controlling the

whole SSD market and still each protects its algorithms as a secret black box creating a market without using any standards. Overriding this firmware could therefore be the key to more reliable and higher recovery rates as well as constant data on memory while in a read only mode which would lead back to well documented guidelines and standards for data recovery specialists and forensic examiners" (Geier, 2015, p. 52)

Without a viable affordable process for investigators to put the drive into factory mode, another option is to send the drive directly to the manufacturer to pull a forensic image through factory mode, which will allow the NAND chips to be directly read from, instead of using the controller chip to translate data. This is very rare, as described by Afonin, but would be the best option because manufacturers can put the drive into factory mode to do a full recovery and extraction.

All in all, most experts recommend that the highest-recovery of deleted files and the best stability results when the NAND chips are read directly, instead of enabling the controller chip. Bell and Boddington explained that SSDs generally should be viewed as a "grey area" for forensic data recovery and legal admissibility until better guidelines can be produced (Bell and Boddington, 2010, p. 12). The goal of this project is to be able to image an SSD while preserving as much deleted evidence as possible and maintaining the drive's hash value without the aid of expensive software.

2.5 Literature Review

Afonin, O. (2019). Life after trim: Using factory access mode for imaging SSD drives. Elcomsoft Blog. Retrieved from

https://blog.elcomsoft.com/2019/01/life-after-trim-using-factory-access-mode-for-imaging-ssd-drives/.

This blog post from Elcomsoft, a leading digital forensics tool developer, outlines the issues associated with SSDs. Afonin outlines how SSD technology differs from HDDs, and the future of SSD analysis, given these differences. This article is extremely helpful to my research because it identifies the exact issue which I identified for the digital forensics community. Specifically, Afonin explains how SSD inner processes work, how they store information, and how factory mode can provide the power to forensic analysts to image a drive without triggering the inner processes which permanently delete data and alter hash values. The biggest shortcoming of this article is that the solutions proposed utilize ACE Labs PC3000 software, which is what this project seeks to find an alternative to. That said, the information in this article goes more in depth into the actual problem than any other article found and can be used for finding more specific articles that may provide the basis for a solution.

Bell, G. and Boddington, R. (2010, December). Solid State Drive Forensics: Where Do We Stand?. Journal of Digital Forensics, Secture and Law, Vol. 5(3). Retrieved April 26, 2020 from https://www.researchgate.net/publication/228662565_Solid_State_Drives_The_Beginning_of_the_End_for_Current_Practice_in_Digital_Forensic_Recovery.

Bell and Boddington explore this topic and how it relates to legal proceedings. They explore how existing investigation and acquisition techniques do not present the same security and expected behavior as that of traditional HDDs for SSDs. Their experiments

investigate how data is destroyed on SSDs and how forensic guidelines should be updated to match the volatile nature of SSDs. This report provides a deep dive into the topic, including the technology behind SSDs, and the authors also provide recommendations for how to image and handle SSDs in forensic investigations.

Fa1c0n. (2020, April 11). How to perform SSD Forensics: Part-I. Retrieved October 23, 2020, from https://medium.com/@songchai.d01/how-to-perform-ssd-forensics-part-i-1158dd3975e8.

This blog post explores the same question posed in the research of this project. Specifically, the author investigates what the current practices are for investigating HDDs compared to SSDs and provides a good overview of how the two differ in technology and practice in the forensics field. This article is part one in a two part series, but was only published in April 2020 and the second part had not been released yet at the time of publishing this capstone report. The information provided in this article is extremely beneficial to the background of this project as it provides in-depth information about topics which are not discussed much in other articles, but does not offer a direct solution to the problem.

Ferreira, J. (2018, March 13). Forensic acquisition of solid state drives with open source tools. Forensic Focus. Retrieved April 4, 2021 from

https://www.forensicfocus.com/articles/forensic-acquisition-of-solid-state-drives-with-open-source-tools/.

This article provides an beneficial overview to the research topic, specifically in the effects of disabling automount on recoverability and stability of the forensic image. Ferreira and Lawson are the only researchers that provided this overview and research on automount, which may indicate that it is not as widely accepted as a solution as other possible solutions. Ferreira also includes great recommendations for forensics acquisition and investigation of SSDs.

Geier, F. (2015). The differences between SSD and HDD technology regarding forensic investigations (Doctoral dissertation, Linnaeus University). Retrieved October 23, 2020 from http://www.gti.bh/Library/assets/fulltext01-gshhsy652.pdf.

This report is very beneficial to the foundational knowledge needed to understand how SSDs work and the impacts of TRIM and other inner processes on the persistence of files and hash values. The author provides accurate explanations of the issues facing SSD forensics, and why the technology differs from HDDs. Specifically, the author explained why these optimization processes exist and possible solutions to blocking them. Lastly, Geier performs experiments on SSDs and other flash technology, using HDDs as a control. These experiments are hugely beneficial to this project because it provides a basis for what to expect when disabling TRIM, comparing hash values, and the imaging process for SSDs.

Gubanovis, Y., & Afonin, O. (2012, October). Why SSD drives destroy court evidence, and what can be done about it. Retrieved November 2, 2020, from https://belkasoft.com/why-ssd-destroy-court-evidence.

This article introduces the topic by two authors who are greatly knowledgeable of the topic of SSD forensics. Their 2014 article covering SSD forensics was the basis for most of the research in this project. In this article, the authors cover the operation of SSDs in an easy-to-understand language, while outlining the forensic problems associated with that technology. That said, because this article is from 2012, some of the information is outdated in relation to SSD forensics today, but it provides a great base knowledge of the topic. The article is extremely informative, but slightly too hopeful of the state of SSD forensics for the future, as most of the possible solutions discussed in the article have not been clarified to date.

Gubanovis, Y., & Afonin, O. (2014, September 23). Recovering evidence from SSD drives in 2014: Understanding TRIM, garbage collection and exclusions. Retrieved October 23, 2020, from https://www.forensicfocus.com/articles/recovering-evidence-from-ssd-drives-in-2014-understanding-trim-garbage-collection-and-exclusions/.

This article is the follow-up to the authors' research performed in 2012. While this article is from 2014, it provides specific information on how SSD forensics is affected by the controller manufacturer and TRIM. The authors of this article have provided invaluable information on this topic, specifically on the journey to finding a solution to

the problem of how to investigate SSDs in a forensically-sound manner. Afonin also wrote a detailed article in 2019 (referenced above) about using Factory Mode to analyze an SSD without triggering inner optimization processes. The authors publish new information on SSDs every couple of years, which helps illustrate the development of the technology and how it is impacting the forensic field. This article is different from others in that it provides more technical information on TRIM and the background processes, and the future of SSD forensics from a hardware level. This article also goes into detail on the exact various solutions available and how different alterations and bugs could aid gaining access to data on an SSD that otherwise would be unavailable. Additionally, corruption of certain areas, such as the partition table, was introduced as a method for easier recovery. Information like this was very beneficial to this project because it introduced additional ways which the most data could be preserved.

Kwak, J., Kim, H. C., Park, I. H., & Song, Y. H. (2016). Anti-forensic deletion scheme for flash storage systems. Department of Elections and Computer Engineering, Hanyang University, South Korea. Retrieved from https://drive.google.com/file/d/0BzeNSwG4LXLIMVNtNXN4c3FzR3c/view.

While the research performed in this article is not directly related to this project, it provides deep insight into the specific technology behind how SSDs operate, from a technical level. For example, the authors explore “an additional layer called the flash translation layer (FTL) between the file system and storage devices” (2016, p. 1). The authors specifically seek out anti-forensic techniques to ensure data is deleted promptly by the optimization processes, but this research is still widely beneficial to this project

because it can provide a deeper understanding into the technology so a solution for delaying the deletion could also be found.

Lawson, M. (2018). Investigating the effects of garbage collection on potentially volatile data during the process of forensic extraction of SSDs. Retrieved from https://pats.cs.cf.ac.uk/@archive_file?p=962&n=final&SIG=574b37d36ac0ae5eb127d541c8628f551848025e6655da45126277ae8ece0bd4.

Lawson's research greatly aided this capstone project, mainly due to the contributions in research on the effects of disabling automount for SSDs. Lawson tested many similar experiments to this capstone research and found similar results, which emphasizes the importance and credibility of this research. Lawson found that write blockers help "stabilize" an SSD, disabling automount results in the best recoverability rates, and TRIM types result in varied recoverability of files. The author also appropriately documents the benefits and shortcomings of SSDs while exploring those impacts on the forensics and legal fields.

Lee, S. W., & Kim, J. S. (2011). Understanding SSDs with the OpenSSD platform. Flash Summit, USA. https://www.flashmemorysummit.com/English/Collaterals/Proceedings/2011/20110809_F2C_Lee.pdf.

This presentation contained beneficial information on how SSD storage is organized, read and write times, and the technology behind how SSDs read and write information. In addition, the authors cover Flash Translation Layer (FTL), which is the layer between

the file system/OS and the storage device. FTL can be used to emulate traditional HDD devices. This article was the first article that introduced FTL to this project, and it allowed for further research through articles like Vieyra. The authors also cover the OpenSSD Project, which is a platform for developing open-source SSD firmware. While this isn't directly related to this project, it could be useful to see what possibilities are available for making SSDs more forensically-sound.

Morozov, R. (2018, Feb 12). Techno mode - The fastest way to access digital evidence on damaged SSDs. Forensic Focus. Retrieved February 10, 2021, from https://www.forensicfocus.com/articles/techno-mode-the-fastest-way-to-access-digital-evidence-on-damaged-ssds/.

Morozov, a researcher with ACE Lab (creator of the PC-3000 tool), presents invaluable information about storage types on SSDs. By analyzing how data is stored on SSDs, an investigator can better understand why a drive failed. While this article focuses specifically on damaged SSDs, Suhanov referenced this article for a solution to generally imaging SSDs while preventing data loss. Morozov introduces "techno mode", or factory mode, as a way to restore access to data on SSDs by rebuilding a broken translator. The author does not showcase how exactly to set the drive to techno mode, as they encourage readers to utilize the PC-3000 SSD utility, but provides a lot of good information for further research.

Neyaz, A., Shashidhar, N., &; Karabiyik, U. (2018, August). Forensic Analysis of Wear Leveling on Solid-State Media. In 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE) (pp. 1706-1710). IEEE.

This research was very interesting as the authors attempted to predict the effects of wear leveling and background processes on SSDs based on the device type, manufacturer, file system, and other characteristics. They explain, "The decay and persistence of the data are based on various factors…[and] once it is deleted from the media, it starts to get corrupted, modified or can be overwritten" (p. 1706). Most of the other sources used for this capstone research directly investigate the forensics implications of these internal processes, not how each device characteristic may affect the ability to recover data and files. This is not something that is explored in depth in this capstone research, but this research reinforces many topics outlined in the background of this research, as well as helping plan the testing framework for this project.

Suhanov, M. (2020, July 20). Trim and unallocated space. Retrieved February 10, 2021, from https://dfir.ru/2020/06/12/trim-and-unallocated-space/.

Suhanov provides a detailed yet concise description of the issues associated with SSDs. After first investigating the history behind SSDs and their evolution into why TRIM and other background processes were created, the author describes the basics of TRIM while focusing on the forensic implications. This is a fresh lens from many of the other sources for this report, as Suhanov specifically frames his research around the

impact on investigations, such as scenarios where TRIM does not deploy. Suhanov also points the reader to possible solutions like “techno mode”, as described in the article by Roman Morozov.

Uchiyama, J. J. (2014). Establishing professional guidelines for SSD forensics: A case study. *Auckland University of Technology.* Retrieved from https://core.ac.uk/reader/56364298.

This report goes into great detail about SSD inner processes and how they affect data retention and integrity. This provides not only a focused view of the research foundational to this project but also a beneficial macro view of the technology employed in SSDs, which was difficult to find in other sources. Although specific solutions to the SSD problem are not included, this technology overview is very useful when analyzing possible hardware solutions and how they impact the drive’s hardware. Similar to Geier, Uchiyama explains how SSDs differ from HDDs, and how those differences create hurdles in data recovery and forensics. “Data on SSDs is really hard to erase AND tremendously hard to recover,” they explain (39). Uchiyama’s research questions also closely align with the research performed in this capstone research.

Vieyra, J., Scanlon, M., & Le-Khac, N. (2019). Solid state drive forensics: Where do we stand? Forensics and Security Research Group, University College Dublin, Ireland. https://markscanlon.co/papers/SSDForensics.pdf.

The authors of this article explain how the inner workings of SSDs affect data integrity and the impacts they have on digital forensics and the legal process. This article

was rooted in why I chose this topic. In addition to the information provided by Lee and Kim, Vierya also touches on Flash Translation Layer (FTL), which is a protocol layer between the OS and the SSD. This article mainly touches on it in terms of the traditional, expensive, and time consuming process of SSD chip off for data acquisition, but it still is useful for this project because it provides the basis for further research of the solution.

Yohannes, F. (2011). Solid state drive (SSD) digital forensics construction. Politecnico Di Milano. https://www.politesi.polimi.it/bitstream/10589/37402/3/SSD%20Digital%20forensics%20Construction.pdf.

This report provides a low level overview of SSD forensics, related to the entire forensic process from acquisition to analysis. The author overviews how SSDs are difficult for data recovery and forensics, specifically in how to recover deleted files, but also fails to consider important aspects of SSD acquisition. For example, the author references using a write-blocker for acquisition, which is known not to stop SSDs inner processes from occurring. Hardware and software imaging tools are also covered, but not specific to SSDs. While this report gives a good overview of information necessary for this project, it appears to be essentially a large literature review of a lot of the other sources, as it synthesizes a lot of information.

Chapter 3: Methodology

This chapter will focus on the design of this project. Specifically, this section will show the steps necessary to prepare the drive before each experiment and the steps to image and analyze the drive after each experiment in order to find the best solution to disabling the drive's inner processes long enough to take a forensic image. The proposed methodology integrates existing forensic procedures with supported techniques from similar research projects performed in the last six years, as well as addressing any potential flaws of the process.

3.1 Overview

In order to preserve forensic integrity as much as possible, specific steps have been outlined for the processes of prepping, manipulating, and imaging of the drives. Three different SSDs will be tested to analyze how each proposed solution affects the retention of deleted evidence and the drive's hash value over time, as well as using an HDD as an experiment control. The results may vary from drive to drive, based on the drive's controller chip, memory cell capabilities, and existing drive degradation so multiple drives will be used to attempt to create a broad analysis of possible solutions. Due to time constraints, only three drives could be tested. This method combines quantitative with qualitative approaches in order to interpret how each proposed solution can preserve a drive's data integrity.

The goal of this research is to find a method of creating a forensic image while maintaining the drive's hash value and preventing as much loss of files previously deleted by a user as possible. "The step of verifying the integrity generally includes a comparison of the digital fingerprint between the initial image and the evidence

presented. This digital evidence mostly consists of a hash value of the image, meaning a computed checksum of the data." Geier explains, "All hash algorithms produce a nearly unique fingerprint, which will always be the same given the same input" (Geier, 2015, p. 24). That said, Sheward explains that even if the same data is present on the drive with no changes, internal processes can cause the two hash checksums to vary depending on the time it takes to image the drive. "This makes it impossible for the investigator to confirm the integrity of the evidence and with that, widely accepted forensics best practice is rendered useless" (Sheward, 2012). This point is acknowledged in case any of the images have resulting varying hash values during verification.

Throughout the methodology proposed, the drive will be imaged then the resulting hash values and number of recoverable deleted files will be compared to analyze the best possible solution to imaging the drive while attempting to disable the inner processes that are triggered by the drive's controller chip.

3.2 Process Overview

Figure 3.1 outlines the process which will be performed on each drive for each proposed solution. These proposed solutions will be outlined in Chapter 4.

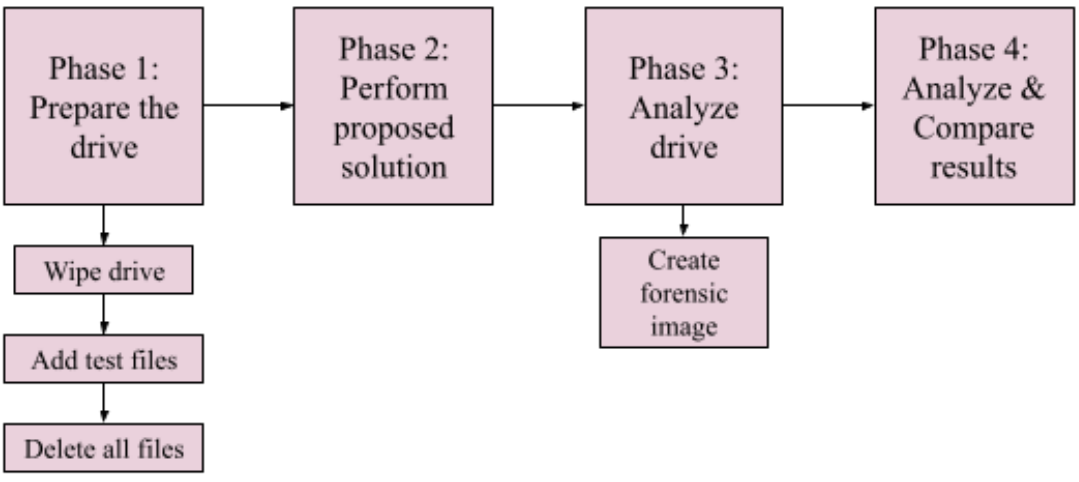

Figure 3.1: Methodology Flow Chart

3.2.1 Phase 1: Prepare the drive

Wipe the drive: To begin, it is important to establish that the drive doesn't have any existing data on it that could interfere with how the drive handles deletion of other evidence. Secure Erase software will be used to ensure the drive is completely erased. The drive will stay plugged in for as much time is necessary to ensure that the contents were completely erased. Additional tests may be used, such as investigating the drives contents to ensure that every existing file on the drive was completely erased. For this capstone project, EaseUs Partition Master was used to wipe data and create partitions for each drive before each test.

Add test files: Test files will then be added to the drive, which will be the same across every drive. For this experiment, the .jpg file shown in Figure 2.2 was copied to fill up a portion of the drive. This is similar to Geier's work, who utilized .JPGs files in their experiment (Geier, 2015, p. 29). Geier tested the recoverability of deleted files on SSDs as well as how the hash value changes over time, which provided the foundation

for this work. .JPG files are recognized by all major forensic software and it will be obvious throughout these tests if a .JPG file is corrupted or incomplete based on opening and viewing it.

Delete all files: All of the .JPG files will then be permanently deleted from the drive. This action will initiate the inner processes like TRIM and garbage collection, which will alter the drive's hash value and may permanently delete evidence.

Figure 3.2: Test .JPG file

3.2.2 Phase 2: Perform proposed solution

Once the OS is done deleting the files on the drive, immediately perform the proposed software solution. This step will vary based on the hypothesis for each round, and will be further outlined in the Experiments section. It is imperative to do this

quickly, because the inner processes like garbage collection may be working in the background anytime that the drive is powered on.

If hardware manipulation, unplugging the drive will be performed before this step is performed. If software manipulation, the drive will stay plugged in (without using a write blocker) in order to make changes to the drive. Then, the drive should be connected using a write blocker.

3.2.3 Phase 3: Analyze the drive

Once Phase 2 is completed, the drive is plugged into the computer, with or without a write blocker depending on the test to be performed. A write-blocker most likely will have no effect on the inner processes of the drive, but it will block new commands from being sent to the drive, such as TRIM, and is vital to a forensic investigator's process. Uchiyama explains that processes like wear leveling and garbage collection can not be stopped by a user, but TRIM can be stopped by a standard writeblocker because it is executed by the operating system (2014, p. 39). A write blocker will prevent TRIM from working further on the drive, if enabled.

Create first forensic image: Use forensic imaging software to create the first forensic image (.E01) of the drive. This should be done as soon as possible once the drive is plugged back into the computer to prevent the drive's inner processes from working too much. Once the forensic image is created obtain the verified hash value using the imaging software.

3.2.4 Analyze & Compare results

Use forensic analysis software to investigate the persistence of the files deleted in the experiment. Notably, look to see if each picture file is completely intact and viewable,

partially recoverable, or completely zeroed out. Compare the results from this image to the other images for each test.

Chapter 4: Experiments and Observations

This chapter describes the tests and experiments performed for this project. This section outlines more detailed instructions on execution of the experiments, hypotheses, and general results from each test. Integration of the methodology framework from Chapter 3 is utilized in order to create a detailed description of the proposed solutions so that other investigators can recreate the same results.

4.1 Overview

This chapter implements various proposed solutions to maintaining a drive's hash value and limiting the complete loss of deleted data. Additionally, it tests setups that would be common in most forensic investigations, such as using a write blocker with an SSD or disabling TRIM to illustrate how data integrity can be affected. Other solutions are also explored that seek to prevent the drive's inner processes from working, such as corrupting the SSD's partition table and putting the drive into factory mode. This section explores the different software solutions proposed throughout this research or by other investigators in order to maintain drive integrity with limited funds when factory access mode is inaccessible. Not all proposed solutions were fully tested, such as puting the drive into factory mode or corrupting the drive, but extensive research was performed to create a background for further research in the future.

4.2 Materials

Various hardware tools, drives, and software are necessary for completing this project. Table 4.2.1 describes each of the three SSD drives and the one HDD (control) that will be used for experimentation in this project.

Table 4.2.1: Drives			
Device Type	**Manufacturer/Model**	**Storage Capacity**	**P/N**
SSD	KingDian S200	60GB	S200 60GB
SSD	Samsung 840 Pro MZ-7PD256	256GB	MZ7PD256HCGM - 1BW00
SSD	Transcend SSD370S	64GB	TS64GBSSD370S
HDD	Seagate Barracuda 7200.7 ST380011AS	80GB	9W2013-007

Additionally, the following auxiliary hardware tools will be used for creating forensic images (Table 4.2.2).

Table 4.2.2: Hardware used		
Device Type	**Manufacturer/Model**	**Notes**
Computer	2017 Macbook Pro, running macOS 10.15.7 2016 MSI laptop, running Windows 10 Home 64-bit	Any computer OS can be used
Write Blocker and connection cables	Coolgear write blocker for 2.5”/3.5” IDE or SATA HDD, Model 3FBCP	Common forensic tool to prevent writing to the evidence drive
SATA to USB cable	Skafil Tech USB3.0 to SATA III cable for 2.5 inch SSD	Used for connection between the SSDs and the computer when not using a write blocker

Table 4.3.3 details software which was used for imaging and analysis of the forensic images.

Table 4.2.3: Software used		
Software	**Version**	**Notes**

FTK Imager	3.1.1.8	Open source - used for creating forensic images (.E01) and verifying hash value
The Sleuthkit's Autopsy	4.17.0	Open source - used for analysis of the forensic images; used on Macbook Pro
EaseUs Partition Master	15.0 Free	Free software - used for wiping/secure erase and creating partitions on each drive prior to each test

4.3 Setup

This section outlines the general setup for the drive before each proposed experiment, expanded on from the Methodology section. The SSD or HDD (control) should be connected to the host computer using a SATA to USB connector, not an enabled write blocker in order to allow changes to be made to the drive.

Figure 4.1: SSD connected via disabled write blocker

4.3.1 Secure Erase

The drive needs to be completely wiped prior to each test to ensure no existing data on the drive interferes with the experiment. Secure Erase is a tool deployed in both HDDs and SSDs for completely wiping data from the drive. In SSDs, it is built into the SATA firmware. This process overwrites all of the bits on the drive, both previously written to and not, with all binary zeroes, binary ones, or random data, which can make it impossible to recover. Secure Erase can only be used for overwriting the entire drive, which is necessary for this project. Secure Erase is executed from the controller chip of the drive, which "wipes the entire contents of the drive at a hardware level" (Gubanovis & Afonin, 2012).

To begin, each drive was permanently wiped using the Secure Erase functionality. Manufacturer-specific software, such as Samsung Magician and Transcend SSD Scope, could be used for this step. That said, not every drive manufacturer produces such a tool, so the tool EaseUS Partition Master was used across all drives for these experiments.

4.3.2 Add test images, deletion

Once the drive was wiped, 9.35GB (600 files) of the test .JPG file was added to the drive through copy and paste from the host computer. The image is shown in Figure 3.2 in the Methodology section. This file's size is 16.7 MB, and 600 iterations of the file were used in a folder equating to 9.35 GB.

Then, all of these files were permanently deleted from the drive using the "SHIFT + DELETE" function. The project was originally planned to delete only half of the 600 .jpg files, but it was determined that the best illustration of TRIM would be showcased

with a higher number of deleted files. So, all of the 600 files were permanently deleted. Depending on the proposed solution, once the OS is done deleting the files (the progress bar pop-up shows completed), the drive should immediately be manipulated for the software test or be unplugged from the host computer for the hardware test

4.4 Perform proposed solution

This section outlines the proposed solutions for finding the best method for disabling or blocking the drive's inner processes in order to take a forensic image verified through a hash value. Table 4.4.1 shows the proposed tests to perform on the drive.

Table 4.4.1: Performed tests

Proposed Test	**Notes**
1. *Control*	Image HDD with write blocker
2. Write blocker, TRIM	Image SSD with write blocker, TRIM enabled
3. No write blocker, TRIM	*Default operating method*, Image SSD without write blocker, TRIM enabled
4. Write blocker, no TRIM	Image SSD with write blocker, TRIM disabled
5. No write blocker, no TRIM	Image SSD without write blocker, TRIM disabled
6. Disable Automount	Image SSD with write blocker, TRIM enabled, after disabling automount on host
7. Factory Mode or Corrupt the Drive	No test performed, only research

4.4.1 No Write Blocker

Solutions 3 and 5 don't utilize a write blocker. As previously stated, a write blocker is a staple in the forensics field for imaging HDDs as it prevents changes to the drive from the connected computer. According to researchers like Geier, Gubanovis, and

Afonin, write blockers work for HDDs, but not for SSDs as an SSD's inner processes occur without input from the connected computer (Gubanov & Afonin, 2012; Geier, 2015). That said, some researchers have indicated that write blockers help "stabilize" the drive for the forensic image to be created, meaning that it could result in better recovery rates (Lawson, 2015, p. 4).

While the inner processes of an SSD like Wear Leveling and Garbage Collection are deployed from the drive's controller chip, TRIM is deployed from the connected computer's OS. A write blocker won't inhibit the inner processes controlled by the controller, but it will stop TRIM from deploying, so a write blocker was used for experiments 1, 2, 4, and 6.

It is important to note that Gubanovis and Afonin describe that "write-blocking imaging hardware does not stop SSD self-corrosion. If the TRIM command has been issued, the SSD drive will continue erasing released data blocks at its own pace" (2014). That said, it is important to test how the write blocker affects the drive, as this is still a common use for many investigators when dealing with SSDs. This test aims to see if there is a higher rate of recovery for deleted files when utilizing a write blocker for imaging.

4.4.2 Disable TRIM

Solutions 4 and 5 involve disabling TRIM before the test to investigate how this affects deletion of evidence, specifically if Garbage Collection is deployed without TRIM. "A disabled TRIM functionality makes an SSD function similar to a hard disk drive by continuing storing deleted data; an enabled TRIM function triggers the internal routines to permanently delete almost all data" (Geier, 2015, p. 50). The hypothesis with

these solutions is that, if TRIM doesn't deploy to mark deleted files, then Garbage Collection will then not deploy. So, deleted files will possibly not be permanently deleted (Afonin, 2019).

Most SSDs and computers have TRIM enabled by default (Geier, 2015, p. 17). In a real scenario, an investigator may receive the SSD, and disable TRIM before imaging in order to prevent any further changes. While this may not make any difference to the drive as files may have already been processed for deletion by this time, it is hypothesized that computers with TRIM disabled to start with have complete recoverability of all files. That said, in certain circumstances, TRIM may be disabled by a user, and would prevent a different protocol from investigators if there is a difference in recoverability. Software like EaseUs or the following commands can be used to query and disable TRIM (Afonin, 2019; Geier, 2015, p. 17). Specifically, this guide was used for disabling and enabling TRIM on the host computer and involved the following commands for Windows:

Linux: `$ sudo hdparm -I /dev/sda | grep -i trim`

Windows:

Query what TRIM is set to. 0 means it is enabled, 1 means it is disabled:

```
fsutil behavior query DisableDeleteNotify
```

This command will set any NTFS drive to enabled. 1 can be used to disable:

```
fsutil behavior set DisableDeleteNotify NTFS 0
```

This command will set any reFS drive to enabled. 1 can be used to disable.

```
fsutil behavior set DisableDeleteNotify ReFS 0
```

Code 4: TRIM Commands

4.4.3 Disable Automount

Solution 6 involves disabling automount on the connected computer, which prevents TRIM from being performed, according to research by Josue Ferreria. He states, "disabling auto-mount will render the TRIM function and garbage collection ineffective, stabilising the volatile nature of SSDs, allowing for more than one identical bit-stream copy to be generated from the same device" (Ferreria, 2018). Lawson's experiments coincides with Ferreria's, explaining, "It was found that the presence of a write blocker helped stabilise the image, when connected quick enough and disabling the automount resulted in the recovery of more data." (2018, p. 4).

This guide was used for instruction on disabling and enabling automount. The commands shown in Figure 4.4 were used in an elevated command prompt. With automount enabled, each drive was wiped using EaseUs and the files were added then deleted from the drive. Then, the drive was disconnected from the computer.

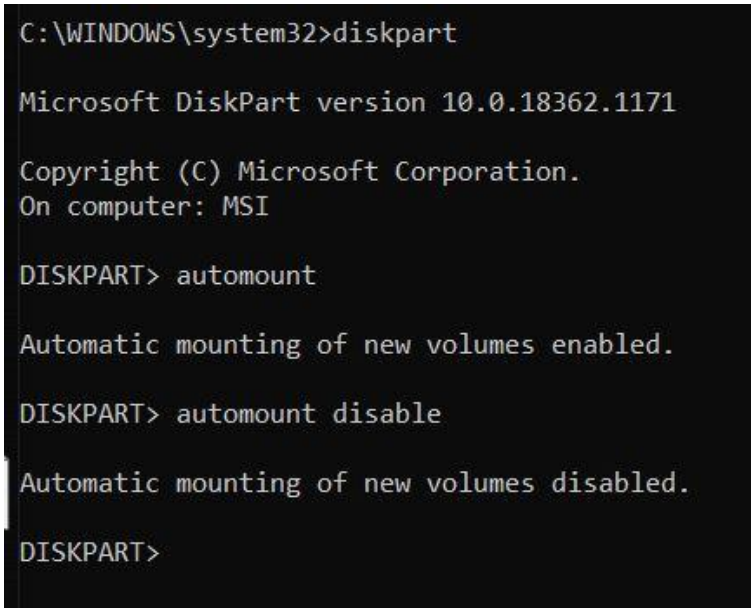

Figure 4.2: Disabling Automount

The following steps were taken for tests involving disabling automount:

1. With automount enabled, prepare the drive including wiping it, adding files, and deleting files. Unplug the drive.
2. Disable automount on the computer using the commands shown in Figure 4.4. Restart the computer.
3. With automount disabled, image the drive using a write blocker.

4.4.4 Factory Mode or Corrupt the Drive

Solutions like corrupting the partition table or putting the drive into factory mode will theoretically stop inner processes from happening, which will maintain the drive's hash and prevent further destruction of deleted files that are queued for garbage collection. Specifically factory mode is used by manufacturers to communicate over the ATA interface in order to upload firmware, test NAND memory, and diagnose problems; it is built into every SSD. Manufacturers also use this to directly access the NAND chips

and the system areas. Factory mode is accomplished by preventing the SSD controller from booting in standard mode and thereby blocking its access to the NAND chips. This means no background processes are started (Afonin, 2019). Similarly, some authors describe how corrupt areas of a drive can result in easier file recovery (Gubanov and Afonin, 2014).

Factory mode and corrupting the drive are the most difficult and time consuming potential solutions, but it is hypothesized that they would have the highest levels of recovery and maintain the drive's hash across multiple images. Due to the complexity of this, a solution was not tested throughout this capstone project, but intensive research was performed in order to contribute to the field of knowledge to encourage future work on the topic.

4.5 Imaging and Analysis

Once the test solution is completed, the drive will need to be imaged. For software solutions, the drive should be immediately unplugged from the computer, then plugged back in using a write blocker, if necessary for the experiment. A forensic image will be created, and in one test for each experiment, a second image will be created an hour after the first image is completed. This is to test how the drive's data changes over that period of time and whether the inner processes are working even with the presence of the proposed solution. It is hypothesized that, if full deletion of evidence was not completed prior to the first image being taken, then the inner processes would continue to work throughout this period. Additionally, other processes like wear leveling may alter the drive's hash value and be impossible to prove the original state of the drive for the first image. While the goal of most of the proposed solutions is to prevent complete

destruction of deleted files, it is also important to acknowledge the need to maintain the drive's hash value throughout an investigation. This wasn't completed on every drive due to time constraints.

4.5.1 Create Image

The image of the drive should be created as soon as possible after plugging the drive back into the computer, whether utilizing a write blocker or not (depending on the specific test). The forensic image is created using FTK Imager's capability of creating a forensic image from a connected physical drive. In this project, the .E01 file format was used because it is what is commonly used in forensic investigations.

4.5.2 Analysis 1

Upon creation of the forensic image, the hash value will appear in a confirmation, as shown in Figure 4.5.1. It is important to note the hash values produced, and if they match from the beginning to the end of imaging. Computed hash represents the hash of the physical drive, while Stored verification hash represents the hash of the .E01 file after creation.

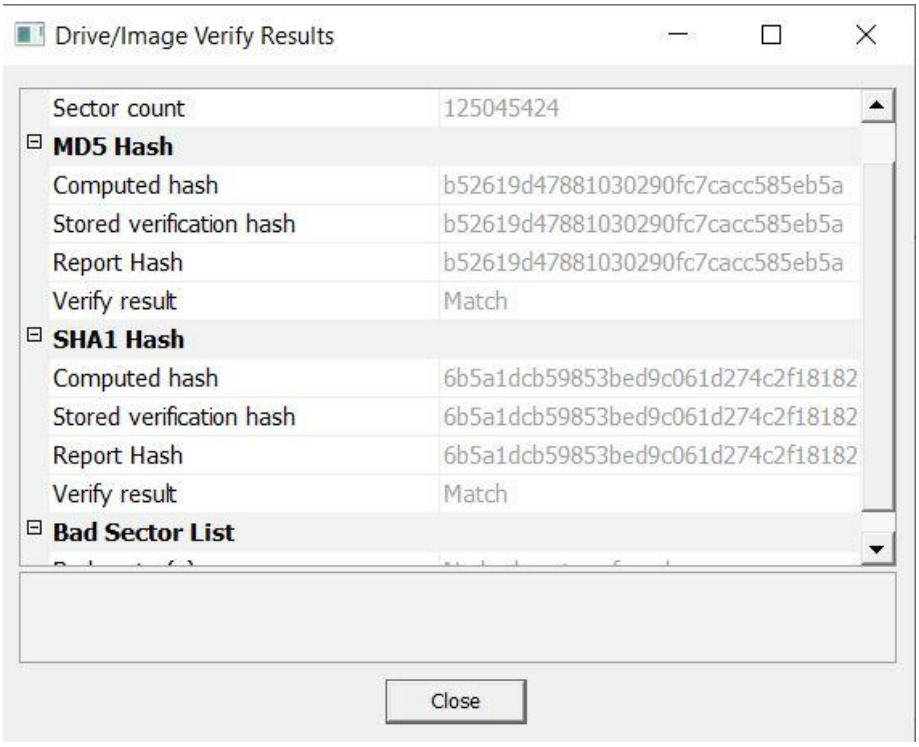

Figure 4.3: FTK hash results

The forensic image created during step 4.5.1 will then be loaded into the forensic analysis software Autopsy for analysis. Specifically, investigate the .E01 file for the number of files present, how many are detected as deleted and recoverable, and how many are completely corrupt or removed. For the purposes of analysis, each forensic image was noted with the following metrics: fully recoverable files (completely intact), partially recoverable files (picture unviewable but fragments of data found), and completely unrecoverable files (zeroed out or overwritten).

As described previously, one subtest from each test will be left plugged in for an hour after the first image, then create a second image. The two images will be compared in depth if the hash values differ to investigate changes to the drive.

4.6 Test Actions and Results

This section details the changes made to the drive for each proposed solution. Then, the results are listed for each test in a table format, specifically detailing the number of files which were recoverable and unrecoverable. Each drive's hash values were noted for each test, and FTK Imager verified matches for every drive throughout imaging.

Detailed analysis and comparison of results will be provided in Chapter 5.

*4.6.1 *Control**

The control experiment involved imaging a standard hard disk drive (HDD) with a write blocker. This was used as the control because it is a known and expected scenario that the drive will maintain its hash value after the files are deleted. This test was performed exactly the same as the other tests involving a write blocker, except that the drive was an HDD.

This test yielded the expected results, as shown in Table 4.6.1.

Table 4.6.1: HDD, Yes Write Blocker

Drive name	**Fully recoverable files**	**Partially recoverable files**	**Unrecoverable files**
Seagate Barracuda 7200.7 80 GB	600	0	0

4.6.2 Write blocker, TRIM

This was the first test using an SSD. For this test, a write blocker was used for imaging, and TRIM was enabled on the drive. This is the most realistic test for most investigators given that TRIM is enabled by default on drives, and a write blocker will be used by most investigators despite many experts saying write blockers have little effect on maintaining the state of an SSD.

Following the steps outlined in sections above, each drive was plugged in, EaseUs software was used to perform Secure Erase, and TRIM was confirmed to be enabled on the host computer. The files were added, then deleted before imaging each drive.

Table 4.6.1 displays the results from this test across multiple drives.

Table 4.6.2: SSD Trim Enabled, Yes Write Blocker			
Drive name & Test #	**Fully recoverable files**	**Partially recoverable files**	**Unrecoverable files**
Transcend #1	124	0	476
Transcend #2	600	0	0
KingDian #1	600	0	0
KingDian #2	0	0	600
Samsung #1	600	0	0
Samsung #2	600	0	0

These tests yielded high rates of recovery, with four out of 6 tests having complete recoverability. Transcend test #1 and King Dian test #2 had some number of completely unrecoverable files, meaning the files were completely overwritten with zeros. This can possibly be attributed to how quickly TRIM and garbage collection can work once a file is marked for deletion.

Analysis of each forensic image showed files most often marked as deleted and recoverable. In Transcend test #1 and Samsung test #1, not all files were marked as deleted by the file system, which may indicate the drive was unplugged too quickly after deleting the large number of files, so TRIM was not able to fully mark all files for deletion.

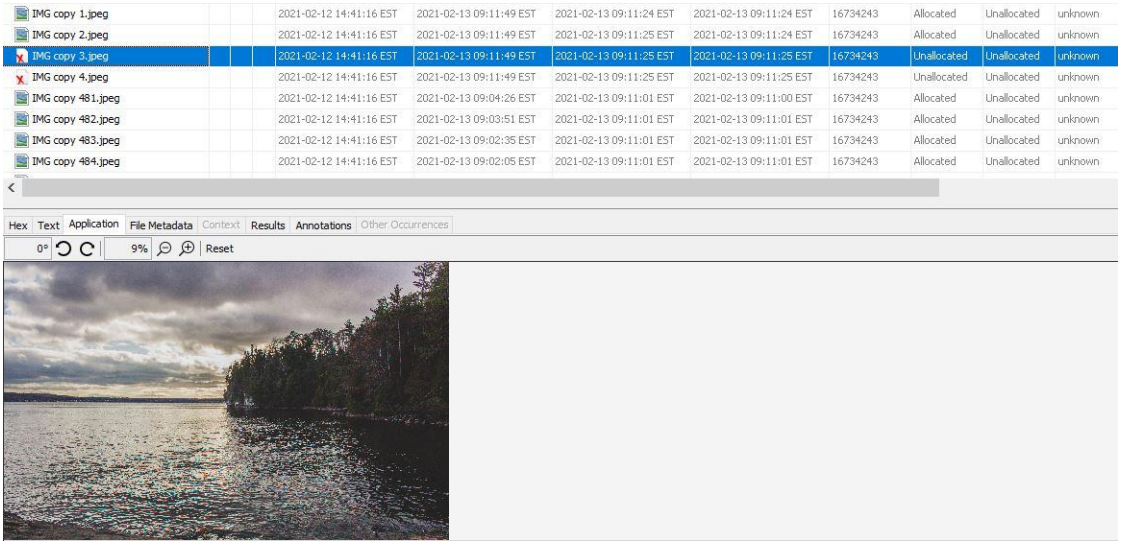

Figure 4.4: Autopsy view

Additionally, Samsung test #2 was left plugged in after imaging. It was imaged a second time and resulted in the same hash values and same recoverable number of files. This provides some confirmation that the write blocker has a higher effect on the drive than noted by many researchers.

4.6.3 No write blocker, TRIM

Experiment 3 is the same as experiment 2 except no write blocker was used. The drives were immediately moved from deleting the files to imaging without unplugging the drive.

Table 4.6.2 shows the results from this test across multiple devices. This test had similar results to experiment #2, but slightly worse rates of irrecoverability. Transcend test #1 and KingDian test #1 both were completely unrecoverable. Unlike experiment #2, all files in all of these tests were marked as deleted by the file system when viewed for analysis in Autopsy.

Table 4.6.3: SSD Trim Enabled, No Write Blocker			
Drive name & Test #	**Fully recoverable files**	**Partially recoverable files**	**Unrecoverable files**
Transcend #1	0	0	600
Transcend #2	600	0	0
KingDian #1	0	0	600
KingDian #2	600	0	0
Samsung #1	600	0	0
Samsung #2	600	0	0

KingDian Test #2 was left plugged in and imaged again an hour after the first image. This resulted in all the files still being recoverable, but the hash had changed. This indicates that, while the write blocker may have prevented permanent deletion of evidence by preventing TRIM from working, the inner processes were still working and altering the drive, so the drive's hash was altered.

4.6.4 Write blocker, no TRIM

Test 4 involved using a write blocker, but disabling TRIM. It was hypothesized that this test would have the highest rate of recoverability for Tests 2 through 5 between both preventing TRIM from being further executed and using a write blocker. Only two tests were performed for this experiment because, even though files were deleted using the "SHIFT + DELETE" function to permanently delete files, all files were completely recoverable when analyzed in Autopsy, as expected.

Table 4.6.4: SSD TRIM disabled, Yes Write Blocker			
Drive name & Test #	**Fully**	**Partially**	**Unrecoverable**

	recoverable files	**recoverable files**	**files**
Transcend #1	600	0	0
KingDian #1	600	0	0

4.6.5 No write blocker, no TRIM

Test 5 involved not using a write blocker and disabling TRIM. It was used to contrast with tests 3 and 4 to see the effect that TRIM and using a write blocker have on a drive. Similar to the results from experiment 5, all files were completely recoverable as expected.

Table 4.6.5: SSD TRIM disabled, No Write Blocker			
Drive name & Test #	**Fully recoverable files**	**Partially recoverable files**	**Unrecoverable files**
Transcend #1	600	0	0
KingDian #1	600	0	0

4.6.6 Disable Automount

Disabling automount involved the methodology outlined in section 4.4.3. This experiment had the highest rates of recovery, with only Samsung test #1 resulting in unrecoverable files. This is attributed to the possibility that TRIM and garbage collection executed across all files before the drive was unplugged for imaging. On the other side, it is important to note that Transcend test #1, Samsung test #2, and King Dian test #2 resulted in some files not being marked as deleted when viewed using Autopsy.

Table 4.6.6: SSD TRIM Enabled, Yes Write Blocker, Automount Disabled

Drive name & Test #	Fully recoverable files	Partially recoverable files	Unrecoverable files
Transcend #1	600	0	0
Transcend #2	600	0	0
KingDian #1	600	0	0
KingDian #2	600	0	0
Samsung #1	0	0	600
Samsung #2	600	0	0

4.6.7 Factory Mode or Corrupt the Drive

A solution was not found for putting an SSD into factory mode without the use of software like the PC3000 or sending the drive directly to the manufacturer for recovery. This section compiles research by many other researchers. Most forensic analysts working with SSDs agree that factory mode is the best way to stop the drive's inner processes and maintain the state of the drive. Afonin explains in their 2019 article that

> Factory access mode can and should be used for forensic imaging of solid-state media. This is the only mode forensic experts can use to extract information from SSD drives without the risk of losing evidence due to background wiping, and this is the only mode allowing access to non-addressable, overprovisioned blocks…. This is the closest one can get to chip-off without actually pulling the chips.

Many researchers mention how chip-off is the only sure way to get all data written to a drive's NAND chips at the state it was received, but chip-off is extremely time consuming and requires a high level of expertise. Additionally, chip-off difficulty is compounded or even made impossible in the presence of encryption or RAIDs. Factory

mode is a better solution because it will allow the data directly from NAND chips to be read, and encrypted data will be decrypted by the controller chip (Morozov, 2018). It will also enable investigators to bypass and detect passwords set for a drive, which can be extremely beneficial if a suspect uses the same password on multiple accounts (Morozov, 2018). Factory mode would enable investigators without the depth of knowledge needed for manual decryption and rebuilding the system architecture to recover data for investigations.

That said, there is no established way to switch the drive into factory mode, which all drives on the market are enabled with; it is almost never documented or understood except by manufacturers (Afonin, 2019; Morozov, 2018). Additionally, because SSD controllers are not regulated, some manufacturers have their own versions of factory mode, so it can vary widely from one drive to the next (Morozov, 2018). Most experts recognize the only method to putting the drive into factory mode as sending it directly to the manufacturer or using expensive software such as ACE laboratories PC-3000, which is unattainable to most investigators, especially in local law enforcement.

An alternative to stopping a drive's inner processes is to corrupt parts of the drive to prevent inner processes from executing over the drive and force the drive into a safe mode, or even factory mode in some drives (Morozov, 2018). No extensive research on this topic was found, but it is acknowledged in many research papers on SSDs in forensics. Many researchers mention that anything preventing the drive from booting in standard mode, such as corrupting the boot sector or partition table or damaging the translator, will prevent the drive's controller chip from being able to read the NAND chips, and thus, not be able to deploy the inner processes against the data (Afonin, 2019,

Gubanovis and Afonin, 2014). “Surprisingly, SSD drives with corrupted system areas (damaged partition tables, skewed file systems, etc.) are easier to recover than healthy ones. The TRIM command is not issued over corrupted areas because files are not properly deleted” (Gubanovis and Afonin, 2014).

Questions of forensicality were brought up throughout this research, but it was considered that the solution of corrupting the drive in order to read files is an attempt to protect evidence. While the drive as a whole may be altered, specific file evidence needs to remain the same. Essentially, the goal should be to put the drive safely into factory mode, but methods of corrupting the drive in order to prevent the controller chip from being able to read the data should also be considered in situations where deleted evidence is likely. Neither of these possible solutions were tested due to time constraints and expertise level, but it is highly recommended to explore further in the future how damaging the drive can enable factory mode or prevent inner processes.

Chapter 5: Results and Evaluations

This chapter reviews the observed results from the experiments. Additionally, this section looks at how these experiments model specific real-life scenarios within the digital forensics field. Through comparing these results, a framework was created with recommendations on best practices for forensic investigators when imaging SSDs.

5.1 Evaluation of Results

Of the 6 tests performed (outlined in Table 4.4.1), disabling automount combined with the usage of a write blocker resulted in the best recoverability rates for deleted files in the most real-world scenario. This was also hypothesized to result in the best recoverability based on the research performed by Lawson and Ferreira. Ferreira explained that disabling automount will temporarily disable TRIM and garbage collection, "stabilising the volatile nature of SSDs, allowing for more than one identical bit-stream copy to be generated from the same device" (Ferreira, 2018).

That said, disabling TRIM, with or without a write blocker, resulted in fully recoverable files in all tests. This is not a very realistic standard as almost all SSDs and computers have TRIM enabled by default, which will begin working right after a file is deleted. A knowledgeable criminal would not disable TRIM because this would all but guarantee recoverability of any files they delete on their system.

The biggest surprise throughout these experiments was the effect write blockers do have on SSDs, despite many researchers such as Geier suggesting that write blockers have no effect on the inner workings of SSDs (2015, p. 28). While there was still decently high recoverability of files from tests not involving a write blocker, there was a

noticeable decline in results when compared to tests using a write blocker. It was concluded that not using a write blocker could cause even higher rates of loss of deleted data and using a write blocker could result in higher rates of recovery. Write blockers are staples in forensic labs, so it is recommended to use one either way.

The hash values provided by FTK Imager to verify the state of the drive against the forensic image taken remained the same in all experiments. In the two tests involving taking a second image an hour later in the same environment as the first, the test involving a write blocker maintained the hash while the test not involving a write blocker resulted in a different hash from the first test. Researchers seem to conflict on the effectiveness of a write blocker for imaging an SSD. Vierya et al found that the hash would always stay the same when using a write blocker, which is conclusive with the results found throughout this project. These experiments seem to indicate that the inner processes may not be working while a forensic image is being taken because of the read processes, but may change after when the drive's processing power is freed up.

An interesting observation was that all but one test had either all files fully recoverable or fully unrecoverable. Only one test had a partial amount between recoverable and unrecoverable (yes TRIM, yes write blocker, Transcend #1). This was thought to be because all of the tests were caught either before garbage collection or after garbage collection, but only one test was able to isolate in between this short window of time in which garbage collection is actively erasing files.

5.2 Recommendations

The following will be published for use by local law enforcement and others as a straight-forward guide on how to approach SSD forensic acquisition.

Forensic imaging of Solid State Drives (SSDs) can be difficult when compared to Hard Disk Drives (HDDs). Investigators may try to image SSDs in the same manner as HDDs, but the two drive types are very different. SSDs have inner processes, controlled by an on-board controller chip, which permanently delete and alter evidence. These inner processes occur anytime the drive is connected to power, and cannot be disabled by a user (besides TRIM). Extra care and specific steps should be taken when creating a forensic image of an SSD, in order to preserve possibly deleted evidence on the drive.

1. *Use software like the PC3000 when possible.*
   Data recovery software like the PC3000 SSD Utility is able to access and read the NAND memory chips directly, without enabling the inner optimization processes like garbage collection and wear leveling.
2. *Use a write blocker.*
   Despite many experts indicating that traditional write blockers do not have an effect on SSDs like they do on HDDs, they are still vital to blocking new changes to the drive and many researchers explain that write blockers help "stabilize" SSDs. Specifically, if TRIM has been deployed by the host computer but not fully executed in marked files for deletion, then the write blocker will prevent the host computer from communicating these changes to the SSD. Of 6 tests across 3 different types of SSDs, deleted files were completely recoverable on 4 tests, and about 1/4th of deleted files were recoverable on 5th test. This is compared to 2 of 3 tests without a write blocker not being recoverable at all. While write blockers will not always block the inner processes which destroy evidence and alter the drive, the use of a write blocker provides extra security to an investigation and may prevent additional deletion of files.
3. *Disable TRIM, if possible, before plugging in the SSD.*
   TRIM is the command from the computer which tells the SSD to permanently erase files. Deleted files were completely recoverable on 4 tests performed when TRIM was disabled immediately after deleting files and right before imaging. This is compared to 8 of 12 tests with files found completely unrecoverable in tests where TRIM was left enabled.
   TRIM is enabled by default in almost all SSDs on the market today, but disabling TRIM on the host computer may have some effect on the efficiency and ability of the drive to fully delete files which have been marked by the OS for deletion.
4. *Disable automount on the forensic workstation before plugging in the SSD.*
   Automount in Windows prevents the OS from directly communicating with the drive. Disabling automount seems to be one of the easiest solutions for preventing TRIM in addition to manually disabling it. In 6 tests across 3 SSDs where automount

was disabled prior to imaging using a write blocker, the deleted files from 5 SSDs were completely recoverable. Many forensic researchers have concluded that automount is a viable solution without the presence of expensive SSD recovery software.

Also note the following throughout SSD investigations:

- A write blocker prevents TRIM from working on the drive, but it does not prevent the drive's inner processes from working.
- While the inner processes are run from the drive's controller chip, this chip is vital to operation of the drive and reading the data from the NAND chips. This is why the controller chip can't just be removed from the drive.
- Data that is completely unrecoverable does not indicate "intentional permanent erasure or corruption" by a user, as it would with an HDD. While this could be the case, it is also indicative of the operation and processing of deleted data by SSDs (Bell and Boddington, 2010, p. 12).
- If the post-image verified forensic hash value is different from the original hash value, and a write blocker was used, it can be assumed that the drive's inner optimization processes like wear leveling and garbage collection were operating and altering data on the drive. The investigator should understand that data may have been moved around or deleted files were permanently erased while the drive was connected to power.
- Formatting the drive, even quick formatting, will deploy garbage collection and permanently erase data from the drive.

Chapter 6: Conclusion and Future Work

This chapter aims to state what has been accomplished in this thesis, summarize the observed results, and to outline the future work that can be performed on this thesis.

6.1 Conclusion

The experiments described throughout this report reveal the differences between HDDs and SSDs, and propose potential solutions to imaging SSDs in order to prevent destruction of deleted evidence. By analyzing previous research performed by others in the digital forensics field, it was identified that a solid solution for imaging SSDs does not exist yet. The best option stands to be using sending the drive directly to the manufacturer, so they can employ factory access mode in order to create a forensic image, or using expensive data recovery software, which most local law enforcement does not have access to.

It was found that imaging an SSDs with a write blocker, thus enabling read-only access, and the host computer operating with a disabled automount feature produced the highest rates of recoverability of deleted files. All in all, without the availability of expensive data recovery software or the opportunity to have help directly from the manufacturer, investigators imaging SSDs should use HDD imaging tactics, but also employ strategies to prevent communication between the host computer and the drive, like automount. This research illustrates how SSDs should be treated differently than HDDS, but it also raises the question of why some researchers describe SSDs as not being susceptible to the effects of write blockers, while others agree with the results of this research that write blockers do have an effect and at least slow down data

degradation. This research provides further evidence to answer some of these questions and provide a guide to investigators imaging SSDs. While the tests performed were limited due to time constraints, the trends presented are clear and consistent with the array of knowledge compiled throughout the background section.

Further research on this topic is recommended, as outlined in the next section, but direct work between digital forensic researchers and SSD chip manufacturers will be vital to find a lasting and universal solution for blocking the inner processes of SSDs while imaging.

6.2 Recommendations and Future Work

A concrete solution still has not been found for the issues involving preventing SSDs from permanently deleting evidence. The research performed in this capstone project illustrates the necessity of further research to find a solution to imaging Solid State Drives.

Further research and tests could involve similar tests to the ones performed in this capstone project, but involve more actions taken on the drive to create more of a backlog of inner processes. This may imitate more of what a user would actually do, and also could provide more indication into how TRIM and garbage collection are completed. Additionally, the recoverability of files should also be investigated for drives that sit, powered-on for an extended period of time after permanently deleting files (i.e. imaging the drive after 5 hours). Investigating these effects would imitate more of a real-world scenario, as investigators almost never would receive a drive immediately after a suspect deletes a large number of files.

Uchiyama also explained how, “if one manufacturer could implement a dip switch or a jumper to enable read-only mode to preserve the state, most the issues discussed in this research will evaporate.” (2014, p. 107). Manufacturers should work directly with forensic investigators in order to find a solution to this problem. That said, Bell and Boddington explain that it is highly unlikely that international SSD controller manufacturers will be willing to work on a solution such as this. This might inhibit the performance of garbage collection, and the trend is that garbage collection will continue to operate more and more quickly to benefit consumers (2010, p. 14).

References

Afonin, Oleg (2019). Life after Trim: Using Factory Access Mode for Imaging SSD Drives. Elcomsoft Blog. Retrieved from https://blog.elcomsoft.com/2019/01/life-after-trim-using-factory-access-mode-for-imaging-ssd-drives/.

Bell, G. and Boddington, R. (2010, December). Solid State Drive Forensics: Where Do We Stand?. Journal of Digital Forensics, Secture and Law, Vol. 5(3). Retrieved April 26, 2020 from https://www.researchgate.net/publication/228662565_Solid_State_Drives_The_Beginning_of_the_End_for_Current_Practice_in_Digital_Forensic_Recovery.

Fa1c0n. (2020, April 11). How to perform SSD Forensics: Part-I. Retrieved October 23, 2020, from https://medium.com/@songchai.d01/how-to-perform-ssd-forensics-part-i-1158dd3975e.

Ferreira, Josue. (2018, March 13). Forensic acquisition of solid state drives with open source tools. Forensic Focus. Retrieved April 4, 2021 from https://www.forensicfocus.com/articles/forensic-acquisition-of-solid-state-drives-with-open-source-tools/.

Geier, F. (2015). The differences between SSD and HDD technology regarding forensic investigations (Doctoral dissertation, Linnaeus University). Retrieved from http://www.gti.bh/Library/assets/fulltext01-gshhsy652.pdf.

Gubanovis, Y., & Afonin, O. (2014, September 23). Recovering evidence from SSD drives in 2014: Understanding TRIM, garbage collection and exclusions. Retrieved October 23, 2020, from https://www.forensicfocus.com/articles/recovering-evidence-from-ssd-drives-in-2014-understanding-trim-garbage-collection-and-exclusions/.

Gubanovis, Y., & Afonin, O. (2012, October). Why SSD drives destroy court evidence, and what can be done about it. Retrieved November 2, 2020, from https://belkasoft.com/why-ssd-destroy-court-evidence.

Hafez, S. (2019, December). Flash storage: What Do SLC MLC TLC and QLC stand for? Retrieved April 26, 2021, from https://datastorageasean.com/blogs/flash-storage-what-do-slc-mlc-tlc-and-qlc-stand.

Harari, E., Norman, R. D., & Mehrotra, S. (1989). *Flash eeprom system.* U.S. Patent No. US5297148A. United States. Retrieved from https://patents.google.com/patent/US5297148.

Kronos (2011). The modern marvel of SSD. Superuser community blog. Retrieved 10 December, 2020 from https://blog.superuser.com/2011/02/10/the-modern-marvel-of-the-ssd/.

Lawson, M. (2018). Investigating the effects of garbage collection on potentially volatile data during the process of forensic extraction of SSDs. Retrieved from https://pats.cs.cf.ac.uk/@archive_file?p=962&n=final&SIG=574b37d36ac0ae5eb127d541c8628f551848025e6655da45126277ae8ece0bd4.

Lee, S. W., & Kim, J. S. (2011). Understanding SSDs with the OpenSSD platform. Flash Summit, USA. https://www.flashmemorysummit.com/English/Collaterals/Proceedings/2011/20110809_F2C_Lee.pdf.

Morozov, R. (2018, Feb 12). Techno mode - The fastest way to access digital evidence on damaged SSDs. Forensic Focus. Retrieved February 10, 2021, from https://www.forensicfocus.com/articles/techno-mode-the-fastest-way-to-access-digital-evidence-on-damaged-ssds/.

Neyaz, A., Shashidhar, N., &; Karabiyik, U. (2018, August). Forensic Analysis of Wear Leveling on Solid-State Media. In 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE) (pp. 1706-1710). IEEE.

Suhanov, M. (2020, July 20). Trim and unallocated space. Retrieved February 10, 2021, from https://dfir.ru/2020/06/12/trim-and-unallocated-space/.

Thatcher, Jonathan (2009, August 18). "NAND flash solid state storage performance and capability – an in-depth look" (PDF). SNIA. Retrieved December 10, 2020, from NAND Flash Solid State Storage Performance and Capability -- an In-depth Look.

Thornton, Scott (2018). SSDs vs. HDDs part 2: Sand or rust?. Microcontroller Tips. Retrieved 10 December, 2020 from https://www.microcontrollertips.com/ssds-vs-hdds-part-2-sand-or-rust/.

Uchiyama, Jay Junichiro. (2014). Establishing Professional Guidelines for SSD Forensics: A Case Study. Auckland University of Technology. Retrieved from https://core.ac.uk/reader/56364298.

Vieyra, John et al (2019). Solid State Drive Forensics: Where Do We Stand? Forensics and Security Research Group, University College Dublin, Ireland. Retrieved from https://markscanlon.co/papers/SSDForensics.pdf.

Yohannes, F. (2011). Solid state drive (SSD) digital forensics construction. Politecnico Di Milano. https://www.politesi.polimi.it/bitstream/10589/37402/3/SSD%20Digital%20forensics%20Construction.pd