

# A Study on the Impact of TRIM and Garbage Collection on Forensic Data Recovery of SSDs at Varying Times and Disk Usage Levels

Vinay Mathew Varghese

A thesis submitted to the Faculty of Design and Creative Technologies  
Auckland University of Technology  
In partial fulfilment of the requirements for the degree of  
Master of Information Security and Digital Forensics

2022

School of Engineering, Computer and Mathematical Sciences

## Abstract

The application of flash memory, as the primary storage medium, in modern computers and consumer electronics is becoming broader. Solid State Drives (SSDs), based on flash memory technology, is rapidly replacing the Hard Disk Drives which are based on conventional magnetic storage technology. Along with the many advantages including improved performance and smaller size, SSDs pose new challenges to digital forensics investigators. An SSD is considered to be self-corrosive due to its internal self-management processes which causes permanent data loss without external interference, making forensic recovery of evidence challenging.

This study reviews relevant literature to understand the underlying technology of SSDs and the characteristics of the internal processes which are TRIM, garbage collection and wear leveling. Whilst many previous studies have been carried out to understand the changes that these processes impose on the data stored on SSDs, this study investigates the time variant changes in data recoverability of SSDs due to the impact of these processes. The research objectives also include understanding the change in data recoverability at varying disk usage levels. In order to fulfil the research objectives, this research interrogates whether the effects of time can be predicted on the amount of deleted data that can be recovered and what is the effect on the same as SSD disk usage increases.

Experiments were conducted to collect data for analysis. The tests were carried out on five different SSDs using three different operating system/file system combinations. The analysis of the results of the experiments shows that the effect of time on the data recoverability of SSDs can be predicted. Identifiable trends were observed in the change in data recoverability of the SSDs as the disk usage progressed from one level to another. Another

observation made from the findings is that larger files are more prone to permanent data loss compared to smaller files. This study also suggests possible reasons for the observed behaviour of the SSDs, in terms of data recoverability, in certain test scenarios. The knowledge acquired through this research, on the effect of time and disk usage on the amount of recoverable data, will assist forensic investigators to take adequate measures and to act in a timely manner, when the investigation involves evidence extraction from SSDs.

## Table of Contents

Abstract .....	i
List of Figures .....	vii
List of Tables.....	xi
Attestation of Authorship .....	xiii
Acknowledgements.....	xiv
Chapter 1 - Introduction .....	1
1.1 Background .....	1
1.2 Motivation.....	4
1.3 Research Approach .....	5
1.4 Thesis Structure.....	5
Chapter 2 - Literature Review .....	7
2.1 Introduction .....	7
2.2 Digital Forensics Investigation .....	8
2.2.1 Forensic Process.....	8
2.2.2 Fundamental Principles.....	11
2.3 Hard Disk Drives (HDD) .....	13
2.3.1 File System .....	16
2.3.2 Slack Space .....	17
2.3.3 Data Recovery .....	19
2.4 Solid State Drives (SSD) .....	20
2.4.1 Types of Semiconductor Memory.....	21
2.4.2 NAND Flash Technology .....	22
2.4.2.1 NAND Array.....	24
2.4.2.2 Read .....	26
2.4.2.3 Program.....	26
2.4.2.4 Erase.....	27
2.4.3 NAND Memory Reliability .....	28
2.4.4 Scaling of NAND Flash Memory .....	34
2.4.4.1 Multi-Level Cell (MLC) .....	34
2.4.4.2 Charge Trap Cells.....	36
2.4.4.3 3D NAND Flash Memory .....	36
2.4.5 NAND Flash Memory Controller .....	40
2.4.6 Host Interface.....	41

2.4.7 Flash Translation Layer.....	43
2.4.8 Wear Leveling.....	44
2.4.9 Garbage Collection .....	46
2.4.10 TRIM .....	47
2.5 Related Work.....	48
2.6 Conclusion .....	53
Chapter 3 – Research Design and Methodology.....	55
3.1 Introduction .....	55
3.2 Research Questions.....	55
3.3 Research Method .....	57
3.3.1 Test Environment.....	58
3.3.1.1 Solid State Drives .....	58
3.3.1.2 Target Computers .....	61
3.3.1.3 Forensic Workstation .....	62
3.3.1.4 USB 3.0 Hub.....	63
3.3.1.5 Hard Drive Enclosures .....	64
3.3.1.6 External Hard Disk Drives .....	65
3.3.1.7 Data Acquisition Tool .....	65
3.3.1.8 Data Recovery Tools.....	66
3.3.1.9 Operating Systems and File Systems .....	66
3.4 Test Plan .....	69
3.5 Payload Files.....	71
3.6 Conclusion .....	71
Chapter 4 – Findings.....	72
4.1 Introduction .....	72
4.2 Microsoft Windows 10 Home with NTFS .....	72
4.2.1 25% Drive Usage .....	73
4.2.2 50% Drive Usage .....	78
4.2.3 75% Drive Usage .....	82
4.3 Ubuntu 21.10 with EXT4 .....	85
4.3.1 25% Drive Usage .....	88
4.3.2 50% Drive Usage .....	92
4.3.3 75% Drive Usage .....	96
4.4 Apple macOS Catalina with APFS .....	99
4.4.1 25% Drive Usage .....	101

4.4.2 50% Drive Usage .....	104
4.4.3 75% Drive Usage .....	108
4.5 Conclusion .....	111
Chapter 5 - Discussion .....	113
5.1 Introduction .....	113
5.2 Research Questions.....	113
5.2.1 Question 1 .....	113
5.2.1.1 Data Recoverability by Operating System/File System.....	114
5.2.1.1.1 Microsoft Windows 10 Home with NTFS.....	114
5.2.1.1.2 Ubuntu 21.10 with EXT4 .....	116
5.2.1.1.3 Apple macOS Catalina with APFS.....	117
5.2.1.2 Summary .....	118
5.2.2 Question 2 .....	119
5.2.2.1 Data Recoverability by SSD .....	120
5.2.2.1.1 Samsung 870 EVO .....	120
5.2.2.1.2 Kingston A400 .....	122
5.2.2.1.3 Crucial BX500 .....	124
5.2.2.1.4 Lexar NS 100.....	126
5.2.2.1.5 Lexar NS 100 - 1.....	128
5.2.2.2 Data Recoverability by Operating System/File System.....	130
5.2.2.2.1 Microsoft Windows 10 Home with NTFS.....	130
5.2.2.2.2 Ubuntu 21.10 with EXT4 .....	131
5.2.2.2.3 Apple macOS Catalina with APFS.....	132
5.2.2.3 Summary .....	132
5.3 Hypothesis.....	135
5.4 Supplementary Discussion .....	136
5.5 Conclusion .....	137
Chapter 6 – Conclusion .....	138
6.1 Introduction .....	138
6.2 Summary of Research .....	138
6.3 Implications.....	140
6.4 Limitations.....	141
6.5 Recommendations for Future Research .....	142
6.6 Conclusion .....	143
References.....	145

Glossary .....	156
Appendix A .....	159
Recovered Files .....	159

## List of Figures

Figure 2.1. Four-Phase Digital Forensics Process .....	9
Figure 2.2. Disk Geometry of a Hard Disk Drive .....	14
Figure 2.3. Seagate Barracuda ST19171N SCSI Hard Disk Platter Deck .....	15
Figure 2.4. Volume Slack Space .....	18
Figure 2.5. File System Slack Space .....	18
Figure 2.6. File Slack Space .....	19
Figure 2.7. Block Diagram of an SSD .....	21
Figure 2.8. Schematic Representation of a Floating Gate Memory Cell .....	24
Figure 2.9. NAND String and NAND Array .....	25
Figure 2.10. Read Disturb Representation in a NAND Flash Array .....	30
Figure 2.11. Pass Disturb and Program Disturb Representation in a NAND Flash Array .....	31
Figure 2.12. Multi-Level Storage in NAND Flash Memory .....	35
Figure 2.13. Cross-Sectional View of a 3D Charge Trap NAND Flash Memory Cell .....	38
Figure 2.14. P-BiCS NAND Strings .....	40
Figure 4. 1. Creating Data Partition Using Disk Management Utility .....	73
Figure 4. 2. Digital Erasure of a Drive Partition .....	73
Figure 4. 3. Forensic Imaging Using dcfldd Tool .....	74
Figure 4. 4. SSDs Connected to the Forensic Workstation Using USB 3.0 Hub .....	74
Figure 4. 5. Changes in Remaining Files: Windows 10 Home With NTFS - 25% Drive Usage .....	75
Figure 4. 6. Comparative View of the Recovered Files: Windows 10 Home With NTFS - 25% Drive Usage .....	76
Figure 4. 7. Average Data Recoverability: Windows 10 Home With NTFS - 25% Drive Usage .....	78
Figure 4. 8. Changes in Remaining Files: Windows 10 Home With NTFS - 50% Drive Usage .....	79
Figure 4. 9. Comparative View of the Recovered Files: Windows 10 Home With NTFS - 50% Drive Usage .....	80
Figure 4. 10. Average Data Recoverability: Windows 10 Home With NTFS - 50% Drive Usage .....	81
Figure 4. 11. Changes in Remaining Files: Windows 10 Home With NTFS - 75% Drive Usage .....	83
Figure 4. 12. Comparative View of the Recovered Files: Windows 10 Home With NTFS - 75% Drive Usage .....	84



Figure 4. 13. Average Data Recoverability: Windows 10 Home With NTFS - 75% Drive Usage.....	85
Figure 4. 14. Shrinking the OS Partition .....	86
Figure 4. 15. Creating Partitions for the OS and Data.....	87
Figure 4. 16. Digitally Erasing the Data Partition .....	87
Figure 4. 17. Formatting the Data Partition to the Ext4 File System .....	88
Figure 4. 18. List of Resulting Partitions.....	88
Figure 4. 19. Execution of TRIM Using the fstrim Command.....	89
Figure 4. 20. Changes in Remaining Files: Ubuntu 21.10 With EXT4 - 25% Drive Usage	90
Figure 4. 21. Comparative View of the Recovered Files: Ubuntu 21.10 With EXT4 - 25% Drive Usage .....	91
Figure 4. 22. Average Data Recoverability: Ubuntu 21.10 With EXT4 – 25% Drive Usage .....	92
Figure 4. 23. Changes in Remaining Files: Ubuntu 21.10 With EXT4 – 50% Drive Usage .....	93
Figure 4. 24. Comparative View of the Recovered Files: Ubuntu 21.10 With EXT4 – 50% Drive Usage .....	94
Figure 4. 25. Average Data Recoverability: Ubuntu 21.10 With EXT4 - 50% Drive Usage .....	95
Figure 4. 26. Changes in Remaining Files: Ubuntu 21.10 With Ext4- 75% Drive Usage .	97
Figure 4. 27. Comparative View of the Recovered Files: Ubuntu 21.10 With Ext4- 75% Drive Usage .....	98
Figure 4. 28. Average Data Recoverability: Ubuntu 21.10 With EXT4 - 75% Drive Usage .....	99
Figure 4. 29. Creating the Data Partition in macOS Using the Disk Utility .....	100
Figure 4. 30. Enabling TRIM in macOS .....	100
Figure 4. 31. Verification of the Execution of TRIM.....	101
Figure 4. 32. Changes in Remaining Files: Apple macOS Catalina With APFS - 25% Drive Usage.....	102
Figure 4. 33. Comparative View of the Recovered Files: Apple macOS Catalina With APFS - 25% Drive Usage .....	103
Figure 4. 34. Average Data Recoverability: Apple macOS Catalina With APFS – 25% Drive Usage .....	104
Figure 4. 35. Changes in Remaining Files: Apple macOS Catalina With APFS – 50% Drive Usage.....	105
Figure 4. 36. Comparative View of the Recovered Files: Apple macOS Catalina With APFS – 50% Drive Usage.....	106

Figure 4. 37. Average Data Recoverability: Apple macOS Catalina With APFS – 50% Drive Usage .....	107
Figure 4. 38. Changes in Remaining Files: Apple macOS Catalina With APFS – 75% Drive Usage.....	109
Figure 4. 39. Comparative View of the Recovered Files: Apple macOS Catalina With APFS – 75% Drive Usage.....	110
Figure 4. 40. Average Data Recoverability: Apple macOS Catalina With APFS – 75% Drive Usage .....	111
Figure 5. 1. Representation of the Average Data Recoverability Among the Drives at the Key Timelines: Windows 10 Home With NTFS.....	115
Figure 5. 2. Representation of the Average Data Recoverability Among the Drives at the Key Timelines: Ubuntu 21.10 With Ext4 .....	116
Figure 5. 3. Representation of the Average Data Recoverability Among the Drives at the Key timelines: Apple macOS Catalina With APFS.....	118
Figure 5. 4. Comparative View of Data Recoverability at Different Disk Usage Levels - Samsung 870 EVO .....	121
Figure 5. 5. Average Data Recoverability at Different Disk usage Levels - Samsung 870 EVO.....	121
Figure 5. 6. Comparative View of Data Recoverability at Different Disk Usage Levels - Kingston A400 .....	123
Figure 5. 7. Average Data Recoverability at Different Disk Usage Levels - Kingston A400 .....	123
Figure 5. 8. Comparative View of Data Recoverability at Different Disk Usage Levels - Crucial BX500 .....	125
Figure 5. 9. Average Data Recoverability at Different Disk Usage Levels - Crucial BX500 .....	125
Figure 5. 10. Comparative View of Data Recoverability at Different Disk Usage Levels - Lexar NS 100.....	127
Figure 5. 11. Average Data Recoverability at Different Disk Usage Levels - Lexar NS 100 .....	127
Figure 5. 12. Comparative View of Data Recoverability at Different Disk Usage Levels - Lexar NS 100 - 1.....	129
Figure 5. 13. Average Data Recoverability at Different Disk Usage Levels - Lexar NS 100 - 1.....	129
Figure 5. 14. Average Data Recoverability at Different Disk Usage Levels - Microsoft Windows 10 Home With NTFS .....	130
Figure 5. 15. Average Data Recoverability at Different Disk Usage Levels – Ubuntu 21.10 With Ext4.....	131

Figure 5. 16. Average Data Recoverability at Different Disk Usage Levels – Apple macOS Catalina With APFS.....	132
Figure 5. 17. Comparison of the Average Data Recoverability Among the Three Operating System/File System Combinations at Different Disk Usage Levels Between the five SSDs.....	133
Figure 5. 18. Comparison of Average Data Recoverability Among the SSDs at Different Disk Usage Levels Between the Three OS/File System Combinations.....	135

## List of Tables

Table 3. 1. List of Main Components of the Test Environment .....	58
Table 3. 2. Samsung 870 EVO SSD Specification .....	59
Table 3. 3. Kingston A400 SSD Specification .....	59
Table 3. 4. Crucial BX500 SSD Specification .....	60
Table 3. 5. Lexar NS 100 SSD Specification .....	60
Table 3. 6. Lexar NS 100 - 1 SSD Specification .....	61
Table 3. 7. HP Pavilion Laptop Specification .....	61
Table 3. 8. MacBook Pro Specification.....	62
Table 3. 9. Dell Latitude SpecificationS.....	63
Table 3. 10. USB 3.0 Hub Specification .....	63
Table 3. 11. Orico Hard Drive Enclosure 2520U3 Specification .....	64
Table 3. 12. Orico Hard Drive Enclosure 2526C3 Specification .....	65
Table 3. 13. Seagate Hard Disk Drives Specifications .....	65
Table 3. 14. Test Files .....	71
Table 4. 1. Summary of Remaining Files: Windows 10 Home With NTFS - 25% Drive Usage.....	75
Table 4. 2. Summary of Recovered Files: Windows 10 Home With NTFS - 25% Drive Usage.....	76
Table 4. 3. Data Recoverability: Windows 10 Home With NTFS - 25% Drive Usage.....	78
Table 4. 4. Summary of Remaining Files: Windows 10 Home With NTFS - 50% Drive Usage.....	79
Table 4. 5. Summary of Recovered Files: Windows 10 Home With NTFS - 50% Drive Usage.....	80
Table 4. 6. Data Recoverability: Windows 10 Home With NTFS - 50% Drive Usage.....	81
Table 4. 7. Summary of Remaining Files: Windows 10 Home With NTFS - 75% Drive Usage.....	82
Table 4. 8. Summary of Recovered Files: Windows 10 Home With NTFS - 75% Drive Usage.....	83
Table 4. 9. Data Recoverability: Windows 10 Home With NTFS - 75% Drive Usage.....	85
Table 4. 10. Summary of Remaining Files: Ubuntu 21.10 With EXT4 - 25% Drive Usage.....	89
Table 4. 11. Summary of Recovered Files: Ubuntu 21.10 With EXT4 - 25% Drive Usage .....	90
Table 4. 12. Data Recoverability: Ubuntu 21.10 With EXT4 – 25% Drive Usage .....	92
Table 4. 13. Summary of Remaining Files: Ubuntu 21.10 With EXT4 – 50% Drive Usage .....	93

Table 4. 14. Summary of Recovered Files: Ubuntu 21.10 With EXT4 – 50% Drive Usage .....	94
Table 4. 15. Data Recoverability: Ubuntu 21.10 With EXT4 - 50% Drive Usage .....	95
Table 4. 16. Summary of Remaining Files: Ubuntu 21.10 With Ext4- 75% Drive Usage.	96
Table 4. 17. Summary of Recovered Files: Ubuntu 21.10 With EXT4- 75% Drive Usage	97
Table 4. 18. Data Recoverability: Ubuntu 21.10 With EXT4 - 75% Drive Usage .....	99
Table 4. 19. Summary of Remaining Files: Apple macOS Catalina With APFS - 25% Drive Usage.....	102
Table 4. 20. Summary of Recovered Files: Apple macOS Catalina With APFS - 25% Drive Usage.....	103
Table 4. 21. Data Recoverability: Apple macOS Catalina With APFS – 25% Drive Usage .....	104
Table 4. 22. Summary of Remaining Files: Apple macOS Catalina With APFS – 50% Drive Usage.....	105
Table 4. 23. Summary of Recovered Files: Apple macOS Catalina With APFS – 50% Drive Usage.....	106
Table 4. 24. Data Recoverability: Apple macOS Catalina With APFS – 50% Drive Usage .....	107
Table 4. 25. Summary of Remaining Files: Apple macOS Catalina With APFS – 75% Drive Usage.....	108
Table 4. 26. Summary of Recovered Files: Apple macOS Catalina With APFS – 75% Drive Usage .....	109
Table 4. 27. Data Recoverability: Apple macOS Catalina With APFS – 75% Drive Usage .....	111
Table 5. 1. Average Data Recoverability Among the Drives: Windows 10 Home With NTFS.....	115
Table 5. 2. Average Data Recoverability Among the Drives: Ubuntu 21.10 With Ext4.. .....	116
Table 5. 3. Average Data Recoverability Among the Drives: Apple macOS Catalina With APFS.....	117
Table 5. 4. Data Recoverability at Different Disk Usage Levels - Samsung 870 EVO ....	120
Table 5. 5. Data Recoverability at Different Disk Usage Levels - Kingston A400.....	122
Table 5. 6. Data Recoverability at Different Disk Usage Levels - Crucial BX500 .....	124
Table 5. 7. Data Recoverability at Different Disk Usage Levels - Lexar NS 100 .....	126
Table 5. 8. Data Recoverability at Different Disk Usage Levels - Lexar NS 100 - 1 .....	128

## Attestation of Authorship

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined in the acknowledgements), nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning.

Vinay Mathew Varghese

4<sup>th</sup> July 2022

## Acknowledgements

I would like to thank my research supervisor, Dr Alastair Nisbet, for providing valuable guidance, support and feedback throughout the entire process of completing this study. I would also like to extend my thanks to my family and parents for their support and encouragement. Special thanks to my 13-year-old elder son, Shawn Vinay, for helping me in the preparation of the test environment, especially, by loading the operating systems on the SSDs.

## Chapter 1– Introduction

### 1.1 Background

Flash memory is widely used in computers and mobile devices as the main storage technology (Jin & Lee, 2019). Solid State Drives (SSDs) built on flash memory technology are becoming more common as storage media replacing Hard Disk Drives (HDDs) in enterprise and client applications. SSDs meet the storage requirements of both enterprise and client applications by their higher performance and lower power consumption (Micheloni et al., 2018, p. 1-2). Rizvi and Chung (2010) observe considerable increase in the use of solid-state memory in consumer electronics including camcorders, mobile phones, laptops and personal computers. Superior performance, smaller size, increased access speed and shock resistance, as there are no mechanical moving parts like HDDs, are some of the main attributes which contribute to the usage preference of SSDs over HDDs for electronic devices (Rizvi & Chung, 2010). HDDs have mechanical moving parts which include magnetic platters and actuator arms. The performance of the HDDs is limited by the rotational speed of the magnetic platters and the seek time of the actuator arms. The absence of complex mechanical moving parts in SSDs accounts for the lower latency and reduced rate of failure compared to HDDs. SSDs also have higher bandwidth, better performance on random access of data and are more reliable than HDDs (Jin & Lee, 2019).

#### **Brief History of SSDs**

In 1978, StorageTek invented the first SSD which was RAM based (Jin & Lee, 2019). The SSD manufactured by StorageTek, STC 4305, was for the IBM mainframe plug compatible market and it was seven times faster than IBM's



2305 HDD (Kerekes, n.d). SSDs with flash memory were created by Western Digital in 1989 (Jin & Lee, 2019). The world's first 2.5" SSD with flash memory, having 20 MB storage capacity, was manufactured and shipped to IBM by SunDisk, later known as SanDisk, in 1991 (Kerekes, n.d). Earlier versions of SSDs were built with flash memory based on NOR gates due to their high performance (Jin & Lee, 2019). Flash memory based on NOR gates are characterised by faster read operation (Fazio, 2006). In 1995, M-System developed SSDs using higher density flash memory structure with NAND gates. Due to higher cost their use was limited to certain applications (Jin & Lee, 2019).

Samsung developed the first 1GB NAND flash memory in 1999 and started mass producing the same in 2002 (Samsung, n.d). World's first SSD systems with 1 TB of storage capacity were made available in 2003 by Texas Memory Systems and Imperial Technology (Kerekes, n.d). The cost of NAND flash memory dropped significantly by 2004 and it started revolutionising the storage market by replacing magnetic storage devices in mobile phones, laptops and even desktop computers (Jin & Lee, 2019).

### **Digital Forensics and SSDs**

Digital forensics is a branch of forensic science (Raji et al., 2018) which deals with the extraction of digital evidence for investigations (Baca et al., 2013). The information stored in digital devices could be used as evidence in courts (Carrier, 2002). Digital investigation includes the acquisition of digital evidence from computers and digital storage media by engaging accepted procedures and techniques (Aldaej et al., 2017). Collection, preservation, examination, analysis and presentation of digital data are the major phases of digital forensic investigation (Mir et al., 2016). Amato et al. (2020) state that the data collection process covers all the operations involved in the

extraction of data stored in digital devices maintaining the integrity of the data. The integrity of the data extracted from the digital devices using approved procedures by the forensic investigators must be able to be independently verifiable (Yusoff et al., 2011). It is difficult to identify a unique procedure for acquiring digital evidence due to the heterogeneity of the digital devices. The data acquisition processes need to be adapted to the different digital devices subjected to forensic investigation (Amato et al., 2020).

The underlying technology used in SSDs for storing data is entirely different from that of HDDs (Bell & Boddington, 2010), but the forensic examination remains the same for both the devices (Reddy, 2019, p. 398). To improve the performance and the longevity of the NAND cells, SSDs implement certain background processes. These are wear leveling, garbage collection and TRIM. The wear leveling process moves around the data stored in an SSD to make the memory cells wear out evenly whereas the garbage collection process prepares the previously used memory cells for new data to be stored. TRIM is an operating system command that is used to inform the SSDs about the memory locations holding redundant data which can be deleted (Vieyra et al., 2018). The implementation of these processes is vendor specific, which they tend to keep confidential, due to the absence of proper accepted industry standards (Reddy, 2019, p. 386). These built-in processes rapidly sanitise the data stored in the SSDs and also make it challenging for the digital forensic investigators to extract and preserve the data, maintaining the integrity, in a manner admissible to court (Aldaej et al., 2017). The internal processes in SSDs leave forensic investigators with stochastic forensics (Reddy, 2019, p. 398). These processes are discussed in detail in Sections 2.4.8, 2.4.9 and 2.4.10.

## 1.2 Motivation

Along with the rapid development of technology and the increasing use of SSDs in computers and other electronic devices to store data, more research is being done in the area of forensic acquisition of data from SSDs because of the anatomy of the SSDs and the challenges it introduces to the digital forensic investigation. As many of the previous research focused on the impact of garbage collection and TRIM on the data stored on SSDs, there are few studies conducted to understand the changes of the data over time. Understanding the effect of the background functions of the SSDs on the data over time may help forensic investigators to act appropriately and in a timely manner. Joshi and Hubbard (2016) conducted experiments to evaluate the effect of the TRIM functionality in forensic recovery of data over different operating systems and SSDs. Their study found that deleted files were able to be recovered within limited time when TRIM was enabled and disabled. More files were recovered when TRIM was disabled. Research conducted by Nisbet and Jacob (2019) using six different SSDs and three different operating systems reveals that there were data losses caused by the background processes of SSDs. The data changes caused by the processes were forensically identifiable and may be predictable by a forensic investigator. This research aims to complement the previous studies in this field, which were conducted to analyse the time variant changes to the deleted data on SSDs, by studying the change in data recoverability of SSDs in relation to elapsed time and disk usage.

Following are the research questions this study aims to answer.

1. Can the effects of time be predicted on the amount of deleted data that can be recovered?

2. What is the effect as SSD disk usage increases on the amount of deleted data that can be recovered?

### 1.3 Research Approach

In order to answer the research questions, experiments using multiple SSDs and operating system/file system combinations with appropriate test cases were designed, which were derived from similar studies and are presented in Chapter 3. The experiments were conducted to gather data to obtain insights into the data recoverability of SSDs at different timelines and disk usage levels.

### 1.4 Thesis Structure

The thesis is organised into six Chapters. This chapter introduced the research topic by providing a background of digital storage media and a brief history of the evolution of SSDs. A brief discussion on digital forensics along with the main challenges that SSDs pose in that aspect has also been presented. The motivation for carrying out this research, the state of current research in this field and the significance of this study were discussed. The chapter also presented the high-level findings of this research.

Chapter 2 provides an in-depth review of relevant literature covering the digital forensic investigation process, the anatomy of SSDs and the underlying technologies. The internal processes of SSDs, which are autonomous in nature, that impact the data recoverability are discussed in this chapter. The chapter also presents the review of similar studies that were conducted in this field.

Chapter 3 defines the research questions based on the findings from the literature review. The chapter develops a suitable research design, that was

derived from similar studies, describes the test environment and provides the list of all the hardware and software resources required for conducting the experiments.

Chapter 4 presents the results obtained by conducting the experiments and provides some analysis of the findings. The results and the analysis are organized by operating system/file system combinations selected for the experiments and the different disk usage levels defined for the data collection.

Chapter 5 provides further analysis of the findings. The research questions and the hypothesis are answered by analysing the findings from different perspective. The analysis of the findings is presented by SSDs and by operating system/file system combinations, using tables and graphs.

Chapter 6 concludes the thesis by providing a summary of the study. The chapter describes the implications of this research and identifies the limitations as well. The chapter also put forth some recommendations for future research that could complement the knowledge gained through this study.

## Chapter 2 - Literature Review

### 2.1 Introduction

SSD is an emerging technology used in heterogeneous digital devices for storing data. Manufacturers of computing equipment, striving to improve the performance, find SSD as a solution (Bednar & Katos, 2011). Data is stored in magnetic material in the traditional storage devices including HDD, however SSD stores data in flash memory chips. An SSD offers non-volatile data storage similar to an HDD, but the retention of the data depends on various factors (Neyaz et al., 2019).

The data stored in the digital devices has very high value in digital forensic investigations (Romero et al., 2019). As it is very important that the reliability and integrity of the evidence collected from the digital devices should be able to be verified, it is equally important to properly identify any modification or data loss that occurred during the recovery process (Shah et al., 2014).

Electronic devices leave traces of evidence of computing activities on the information storage media. Hard Disk Drives have well identified forensic properties which are favourable for the retrieval of evidence in a later point of time, using forensic methods. While SSDs which are built on the latest technologies offer many advantages over the HDDs, SSDs have some limitations; especially when it is viewed from a digital forensics' perspective. SSDs have a tendency to wipe the stored information as part of its internal data sanitisation processes, including wear leveling and garbage collection, which are associated with the performance improvement of the storage media. Operating systems have no control over these processes. (Shah et al., 2014).

## 2.2 Digital Forensics Investigation

An investigation is a structured examination with the goal to identify or verify facts in relation to a crime or incident. Forensic science involves the application of scientific methods in an investigation. When it is applied to the digital information, it is called digital forensics and the corresponding investigation is known as digital forensics investigation. A digital forensics investigation should follow professionally recognised and established digital forensics principles, standards and processes in order to be forensically sound (Årnes, 2017, p. 2-6).

### 2.2.1 Forensic Process

A systematic investigation of the digital evidence that is present in any device used to store or process digital information is defined by a forensic process. A digital investigation process is similar to a physical investigation process with the difference that the evidence is digital in case of the digital investigation which introduces new challenges on identifying and gathering the evidence relevant to the case. The process ensures that the identified evidence is managed properly as it is essential to prove the case in a court of law. A digital forensic investigation process is often considered to be universal as it can be applied to digital investigations of any type of crime or incidents (Årnes, 2017, p. 14).

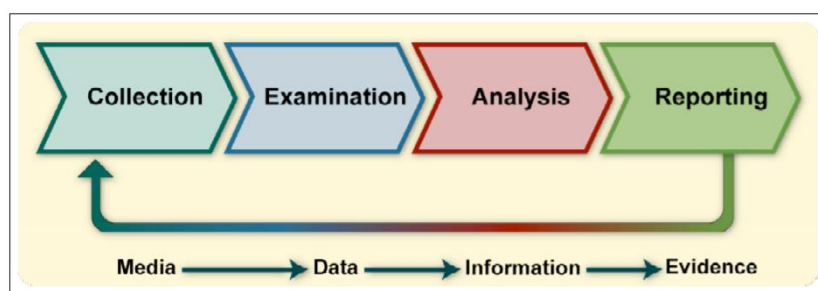
There are several models of digital forensic process. National Institute of Standards and Technology (2006) defines a simple model with four stages to describe the process which are Collection, Examination, Analysis and Reporting as shown in Figure 2.1 (Sammons, 2015, p. 28-29).

**Collection:** Identifying the potential sources of information and gathering the data from them is the first step in the forensic process. Following the identification of the possible sources of data, a plan for the data acquisition

needs to be developed. The plan should contain the list of the identified data sources, their priorities and the order to perform the data acquisition. Factors to be considered while prioritising the data sources include the potential value of the information source, volatility of the data and the effort required to extract the data from the source. The data acquisition process generally involves the collection of volatile data, creation of duplicate copies of the non-volatile data sources to collect the information and preservation of the original non-volatile data sources. The integrity of the acquired data needs to be verified which is usually done by computing the message digest of both the original and the copied data using appropriate tools, then compare the digests to ensure that they are the same. Every step in the data collection process needs to be logged in detail, including the details of any tools used, so that the entire process can be repeated by another person, if needed (National Institute of Standards and Technology [NIST], 2006).

**Figure 2.2.11**

*Four-Phase Digital Forensics Process*



*Note.* From “Guide to Integrating Forensic Techniques into Incident Response”,  
by National Institute of Standards and Technology, 2006, NIST Special  
Publication 800-86, p.25.



**Examination:** The collected data is subjected to examination in order to extract the relevant pieces of information. Often the number of files to be iterated through would be enormous which makes the task tedious. But there are various tools and techniques available for aid that can bring down the amount of data to be examined. For example, email logs having certain email address or documents containing references to certain persons' names can be identified by using text and pattern searches. Another useful technique is to engage a tool to identify the files of interest based on the type of their contents and exclude the others. The content types include text, graphics and music (National Institute of Standards and Technology [NIST], 2006).

**Analysis:** After the extraction of the relevant information through careful examination of the collected data, it is then studied and analysed, using appropriate methods and techniques which are well-documented (Sammons, 2015, p. 28), to draw conclusions. The analysis includes correlating information from different sources, using a methodical approach, in order to identify people, locations, objects and events that can help to reach a conclusion or to determine that no conclusion can be made yet. For example, an event raised by a network intrusion detection system (IDS) can lead to a host computer that can reveal the user account associated with the particular event through its audit logs and further insights on the actions performed by the user can be obtained by analysing the logs of the host's IDS. Technical tools are available to automate such process of gathering and correlating the information (National Institute of Standards and Technology [NIST], 2006). Throughout the process the evidence needs to be kept free from distortion and every file analysed needs to be recorded with details including the contents, location, date and ownership of the file (Varol & Sönmez, 2017).

**Reporting:** The reporting process, which is the final stage, includes the preparation and presentation of the information obtained through the analysis of the evidence (National Institute of Standards and Technology [NIST], 2006). The report may include the details of the actions performed, the tools and procedures engaged along with the reasons for selecting the same; any further actions to be performed which can include examination of additional information sources and enhancement of the current security measures; and recommendations to improve policies, procedures, tools and anything relevant to the forensic process (Sammons, 2015, p. 29).

### 2.2.2 Fundamental Principles

Evidence integrity and chain of custody are two fundamental principles of digital forensics investigation.

**Evidence Integrity:** Preservation of the evidence in its original form without any intentional or unintentional modifications is referred to as Evidence integrity (Årnes, 2017, p. 6). Maintaining the evidence integrity is of paramount importance as it has impact on the admissibility of the evidence in a court of law (Tobin et al., 2016).

Following the seizure of the digital media subjected to an investigation, a forensic image of the evidence is created which is carried out by trained professionals to ensure that the evidence is untampered (Reddy, 2019, p. 5). The Association of Chief Police Officers' (ACPO) Good Practice Guides for Digital Evidence defines the principles of digital evidence. The principle 2 states that any person accessing the original data must be competent to do so (Williams, 2011). The forensic image is a bit-by-bit copy of the physical storage device. The image, usually in the disk dump format (.dd) or in the Encase file format (.E01), contains all the files and folders including the deleted ones. Write blockers are used to prevent accidental damage or

deletion of the contents of the storage device during data acquisition (Reddy, 2019, pp. 5-9). Those are devices designed to prevent write access to the data storage devices but provide read-only access, in order to maintain the data integrity. There are hardware write blockers and software write blockers (Meffert et al., 2016).

In order to ensure the integrity of the forensic image, a hash value is taken using cryptographic hashing algorithms (Bell & Boddington, 2010). MD5 and Sha1 are examples of the hashing algorithms. The hashing algorithms compute the hash value based on the contents of the subject file and even a tiny difference in the contents results in a different hash value. Any such difference in the hash value of the forensic image, compared to the original one, indicates tampering of the evidence or a technical error during the imaging process and could make it inadmissible in a court of law (Reddy, 2019, p. 6). Although it is ideal to maintain the evidence integrity, it is often not achievable during investigations involving live computer systems and networks as data changes is almost certain. This makes documenting all the steps taken during the investigation much important (Årnes, 2017, p. 6).

**Chain of Custody:** Maintaining a proper chain of custody throughout the entire investigation process is very important (Reddy, 2019, p. 10). Chain of custody is a documentation which covers the acquisition, control, analysis and disposition of the evidence (Årnes, 2017, p. 6). It keeps track of the evidence throughout its life cycle starting from the first individual who took custody of the evidence to the last person who returned or destroyed it once the investigation is over. Any break in the chain of custody can make the evidence inadmissible to the court (Reddy, 2019, p. 10).

## 2.3 Hard Disk Drives (HDD)

HDD was invented by IBM in 1956. The main parts of an HDD are rotating circular platters and moveable read/write heads. Information is stored magnetically on the platters. Platters have a thin coating of magnetic material on both sides (Carrier, 2005) consisting of several magnetic segments which are called bit cells. Each bit cell typically contains around fifty to a hundred grains of magnetic material. A logical “0” or “1” is determined by the collective orientation of the magnetic grains in a bit cell. The hard disk drive can read the zeros and ones or reverse the magnetic polarization (write) of the bit cells by means of the read/write head (Fink, 2013). An HDD can have more than one platters, stacked on top of each other, rotating simultaneously. The heads for reading and writing data are attached to arms which move back and forth inside the disk with each arm having a head on the top and another on the bottom, but only either one of those can read or write at a time. Each head in the disk is assigned with an address (Carrier, 2005).

The materials used to make platters are glass or aluminium. It is then coated with multiple layers of different compounds using electroless plating followed by vapour deposition process. This prepares the disk for the magnetic material that stores the digital data, which is often an alloy of Cobalt, organised in concentric rings of around 250nm width and 25nm depth. These rings of magnetic material look like grains on microscopic scale (Evanson, 2020). Small magnetic domains, which are regions of the magnetic material having the same orientation, are created to store the information (Ismail-Beigi Research Group, n.d).

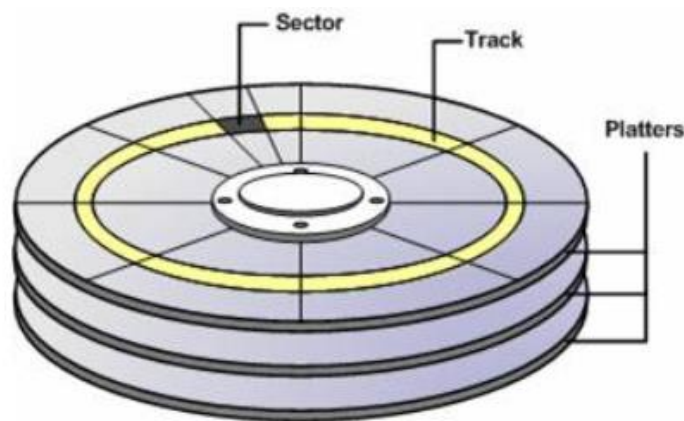
The concentric rings of magnetic material on the platters are called tracks. At a given position, the read/write head can read or write one track. Tracks

are assigned with numbers starting from zero (Di Marco, 2007). As the layout of all the platters are the same, tracks in different platters can have the same number or address. Tracks with the same address in multiple platters are collectively known as a cylinder which has its own address (Carrier, 2005). One surface of one of the platters is designated to store information on hardware track positioning and therefore not available to the operating system. Both surfaces of the other platters, if there are more than one platters, is available to the operating system for data storage. The hardware track positioning data, written to the disk by the manufacturer during assembly, is used by the system disk controller to position the read/write head in the correct position (NTFS.com, n.d).

Figure 2.2 shows the disk geometry of a Hard Disk Drive.

**Figure 2.2**

*Disk Geometry of a Hard Disk Drive*



*Note.* From What is Hard Disk? by Applexsoft, n.d.

(<https://www.applexsoft.com/glossary/hard-disk.html>).

Tracks are further divided into sectors which are the smallest addressable (Carrier, 2005) physical storage units on a disk. Adjacent tracks with the same number of sectors (Pitchumani et al., 2012) or having the same size

are called zones (Di Marco, 2007). Sectors are labelled using the hardware track positioning data. The starting address of each sector is identified by the sector identification data stored on the area before the contents of the sector (NTFS.com, n.d). Figure 2.3 shows an image of platters organised on a platter deck.

**Figure 2.3**

*Seagate Barracuda ST19171N SCSI Hard Disk Platter Deck*



*Note.* From File:Seagate-ST19171N-platters.JPG – Wikimedia Commons, by Rayshade, 2009, Wikimedia Commons (<https://commons.wikimedia.org/wiki/File:Seagate-ST19171N-platters.JPG>). CC BY-SA 3.0

A sector can be addressed by using the cylinder address (C) to identify the track, address or number of the head (H) to identify the platter including the side and finally the sector address (S) to identify the sector on the particular track. This type of sector addressing using the disk geometry is known as CHS addressing, which is almost obsolete now. Logical Block Addressing (LBA) which uses a single number, starting from zero, to address each sector has become the standard replacing the CHS addressing method

(Carrier, 2005). Host operating system addresses the sectors by using the logical block addresses. The disk maps those addresses to the corresponding physical locations (Pitchumani et al., 2012). One or more consecutive sectors together forms a cluster. A file can spread across multiple clusters which may not be contiguous (NTFS.com, n.d).

### 2.3.1 File System

An HDD must be formatted before data can be written to it for the first time. There can be one or more partitions which are logical divisions of storage locations in the hard disk. Different operating systems have different file systems which keep track of the physical location of the data on the disk (Bhat & Quadri, 2012). A file system organises information into files. It also controls the naming of files, storage, retrieval, modification and other file related operations including security depending on the type of file system. File Allocation Table (FAT), New Technology File System (NTFS) and High Performance File System (HPFS) are some examples of file systems (Lutkevich, n.d).

A file system keeps track of the storage locations of files by creating mapping records. As an example, when a new file is created NTFS creates a record corresponding to the file in the Master File Table (MFT) (Lutkevich, n.d). MFT is an important feature of the NTFS, organised as an array of records, which holds at least one entry for each file and directory. These entries are called file records, having a default size of 1024 bytes. The initial 42 bytes of the file record is used to store the MFT header and the remaining bytes are used to store the file attributes. Attributes are small data structures with specific purposes. \$STANDARD\_INFORMATION, \$FILE\_NAME and \$DATA are examples of attributes. Attributes are classified into resident and non-resident types. The content of a resident

attribute is stored directly in the MFT entry and that of a non-resident attribute is stored in the external clusters, in which case, a cluster run containing the list of clusters used is stored in the attribute MFT entry. A cluster represents a data unit in the NTFS and each cluster has an associated Logical Cluster Number (LCN) starting from 0 for the first cluster in the file system. A Virtual Cluster Number (VCN) is assigned to the clusters which belong to the same file and the mapping between the VCNs and the LCNs is provided by the cluster runs of the non-resident attributes (Huebner et al., 2006).

### 2.3.2 Slack Space

Slack spaces are those areas on the storage medium which cannot be used by the file system. The discrete nature of the storage space allocation is the cause of the formation of slack spaces. There are different types of slack spaces (Huebner et al., 2006).

#### **Volume Slack**

The unused space between the end of a file system and the end of a partition is known as the volume slack (Huebner et al., 2006; Wani et al., 2020). When a partition is deleted, the operating system removes the reference to it, but any data stored in the partition still remains in it. As the reference to the partition is deleted, the space becomes inaccessible to the operating system for file storage (Berghel, 2007). Figure 2.4 shows an example of a volume slack space.



**Figure 2.4**

*Volume Slack Space*

Partition 1 NTFS	Partition 2 NTFS	Partition 3 Not Formatted
------------------	------------------	---------------------------

*Note.* Partition 3 is volume slack space.

**File System Slack**

The unused space after the end of the file system that is not part of any cluster is called the file system slack. The cause of the formation of the file system slack is the insufficiency of space to make a full cluster, mainly happens when the partition size is not a multiple of the cluster size (Huebner et al., 2006). Figure 2.5 shows an example of a file system slack space.

**Figure 2.5**

*File System Slack Space*

Sector 1	Sector 2	Sector 3	Sector 4	Sector 1	Sector 2
----------	----------	----------	----------	----------	----------

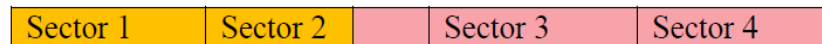
*Note.* Figure shows 4 sectors per cluster. The final 2 sectors are file system slack space as those cannot form a full cluster.

**File Slack**

The unused space between the end of a file and the end of the last allocated cluster is called file slack. RAM slack and drive slack are two types of file slack. RAM slack is the unused space between the end of a file and the end of the last partially used sector (Huebner et al., 2006). Operating systems prior to Windows 95 use this space to fill with data from the RAM giving it the name “RAM Slack” (Sindhu & Meshram, 2012). The drive slack space starts from the end of a file to the end of the last cluster (Huebner et al., 2006). Figure 2.6 shows an example of a RAM slack and a drive slack.

**Figure 2.6**

*File Slack Space*



*Note.* The final, unused part of Sector 2 is the RAM slack. Final part of Sector 2, Sector 3 and Sector 4 together form drive slack.

When a file is deleted, the file system deallocates the clusters used by the file (Pal & Memon, 2009). However, the actual data still remains at the physical location (Bhat & Quadri, 2012). The clusters which were allocated to the deleted file become available to store a new file, but the old data is not wiped at this point and can be recovered until it is overwritten with a new file. Even after the clusters are allocated to a new file, traces of the old file can still exist in the file slack space. This makes it possible to retrieve a part of the deleted file even long after its deletion, from an HDD (Casey, 2011, p. 455).

### 2.3.3 Data Recovery

Recovering data from the deleted or hidden (Reddy, 2019, p. 39) files from unstructured digital forensic images, in the absence of proper file system information, is known as “Carving” (Cohen, 2007). Carving is used in digital forensics to locate files in raw data stream including the unallocated clusters in the hard disk and recover the data (Casey, 2011, p. 445).

Most types of files have a specific structure designed by the developers of the corresponding software or standards bodies. This structure can be used to identify the file and recover the data fragments from the hard disk. Each type of file starts with few specific bytes at the header of the file. The data in the file can be stored in different locations and the file terminates with few specific bytes at the footer of the file. The distinctive common header

specific to the file type is known as the file signature or more casually, the magic number. The file signature can be used to locate and recover the deleted files, which is one of the traditional methods used for data recovery that is based on the file structures (Casey, 2011, p. 445; Pal & Memon, 2009). For example, a JPEG file starts with the byte sequence “0xFF 0xD8” and ends with “0xFF 0xD9” (Poisel & Tjoa, 2013). First generation of file structure-based carvers, which are software tools used for data recovery, extract all the data between a known file header and footer assuming no fragmentation and no missing information in between. This can result in junk data in the middle of the recovered file (Pal & Memon, 2009). File carvers have evolved, implementing advanced algorithms which take the fragmentation into account, with different approaches to identify the fragments and reassemble those in the original order to reconstruct the actual document (Pal & Memon, 2009).

## 2.4 Solid State Drives (SSD)

Flash memory and the microcontroller constitute the main components of an SSD; however, an SSD has other components as well. Those can include components to derive and stabilize the power supply, temperature sensors and cache memory. Often Double Data Rate (DDR) memory is introduced to cache the data. During the write operation it is used to store the data before moving to the flash memory which contributes to faster data updates without wearing out the flash (Micheloni et al., 2018, p. 2-3). Figure 2.7 shows the basic block diagram of an SSD.

The semiconductor properties and the internal architecture of SSDs add to its complexity and also have impact on their performance, reliability, power and security properties. SSDs are self-corrosive (Neyaz et al., 2019) and self-

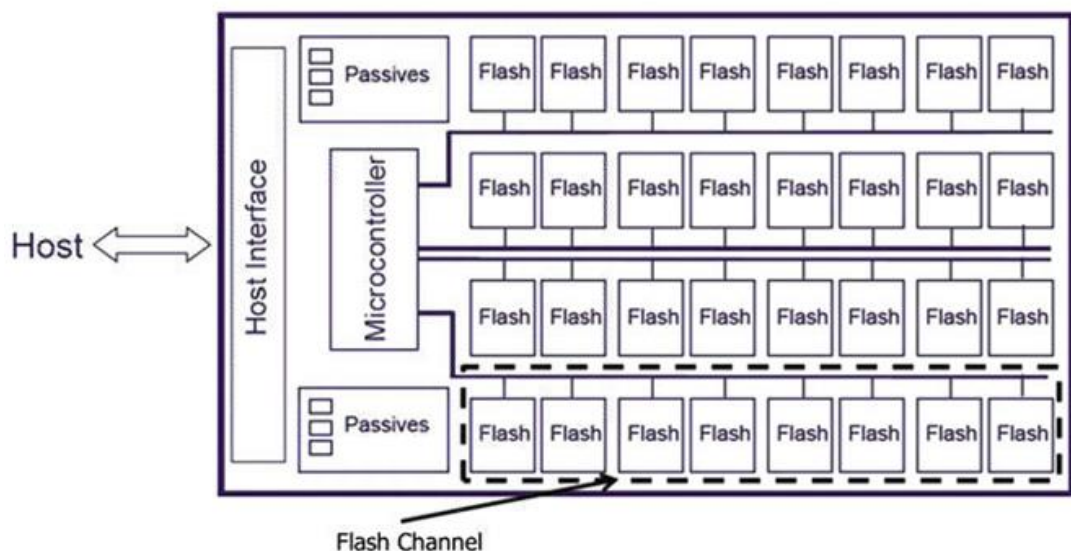
managing (Rajimwale et al., 2009). Following sections discuss more about the anatomy of SSDs.

### 2.4.1 Types of Semiconductor Memory

Random Access Memory (RAM) and Read Only Memory (ROM) are the two main categories of semiconductor-based memory. A RAM loses the data stored on it when the power is switched off whereas a ROM can hold it even in the absence of power. The content of a ROM is defined during its manufacturing and cannot be changed later. Another type of memory is the Non-Volatile Memory (NVM) which sits between the other two types. The content of NVMs can be altered electrically and NVMs do not need power to retain their contents as well. As an SSD is built on NAND flash memory (Xu et al., 2015) which do not require power to retain its data, it falls under the category of non-volatile memory (Huffman, 2015).

**Figure 2.7**

*Block Diagram of an SSD*



*Note.* From *Inside Solid State Drives (SSDs)* (2nd ed., p. 2), by R. Micheloni et al, 2018, Springer Singapore Pte. Limited. Copyright 2018 by Springer Nature Singapore Pte. Limited.

### 2.4.2 NAND Flash Technology

NAND flash memory stores electrons on a capacitor and retains the charge indefinitely (Cornwell, 2012). Flash memory cells are Complementary Metal Oxide Semiconductor (CMOS) transistors with variable threshold voltages. The data is stored in a flash memory cell as trapped charge on a metal layer, which is often referred to as the floating gate, implanted in the oxide layers between the control gate and the channel. The number of electrons stored on the floating gate determines the threshold voltage of the cell (Caulfield, 2013). Figure 2.8 shows the schematic representation of a floating gate memory cell.

The oxide layers, which provide electrical isolation for the floating gate, are tunnel oxide and inter-poly-oxide (IPO). The tunnel oxide controls the threshold voltage which represents the information stored in the memory cell. The oxide layers prevent the charge stored on the floating gate from leaking, which accounts to the non-volatility characteristic of the flash memory (Zuolo et al., 2017). While the floating gate is isolated, surrounded by oxide, the overlapping control gate is a contacted one to facilitate the gate terminal. The process of injecting electrons to the isolated floating gate is called programming and that of removing electrons from it is called erasing. The threshold voltage of the memory cell is altered by these operations. A higher gate voltage than the threshold voltage of the cell represents “1”, cell is in the ON state and a lower gate voltage than the cell’s threshold voltage represents “0”, cell is in the OFF state (Micheloni et al., 2018, p. 3). The NAND cells are programmed by applying high voltage pulses (Cornwell, 2012).

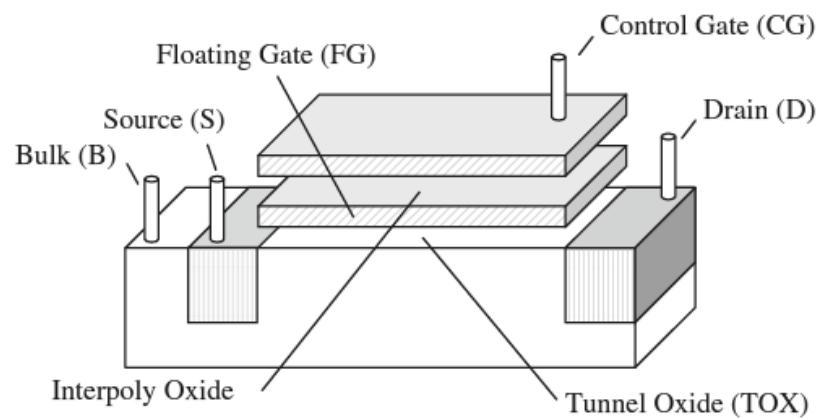
An SSD consists of many flash memory chips, which are operated via bus connection, organised by blocks with multiple pages in each block (Park et

al., 2019). Each page layout consists of user data and meta data sections. Meta data section holds information which include logical page address, logical block address, error correction codes, erase count number and bad blocks. It also includes flags reflecting the various states related to the page, indicating if the data stored is valid or not and if the page is free or occupied (McEwan & Mir, 2015).

Read, program (write) and erase are the three main flash operations (Jose & Pradeep, 2013). As flash memory is a type of Electrically Erasable Read Only Memory (EEPROM), in-place data update cannot be performed due to its inherited characteristics (Seung-Ho & Kyu-Ho, 2006). Before performing a write operation, the target physical block must be erased if it is not already empty. The erase operation sets all the bits in the entire block to “1” (Yan et al., 2014). Write operation clears the bits to “0”. Once a bit is set to “0”, only an erase operation can set it back to “1” (Jose & Pradeep, 2013). It is not possible to erase data from a single page but has to be done in the block level, as the smallest erasable unit is a block, which is a time consuming process (Aldaej et al., 2017). Programming and reading are done at the page level (Micheloni et al., 2018, p. 6). The erase operation takes more time than the program operation. Read operation is the fastest among the three main flash operations (Yan et al., 2014).

**Figure 2.8**

*Schematic Representation of a Floating Gate Memory Cell*



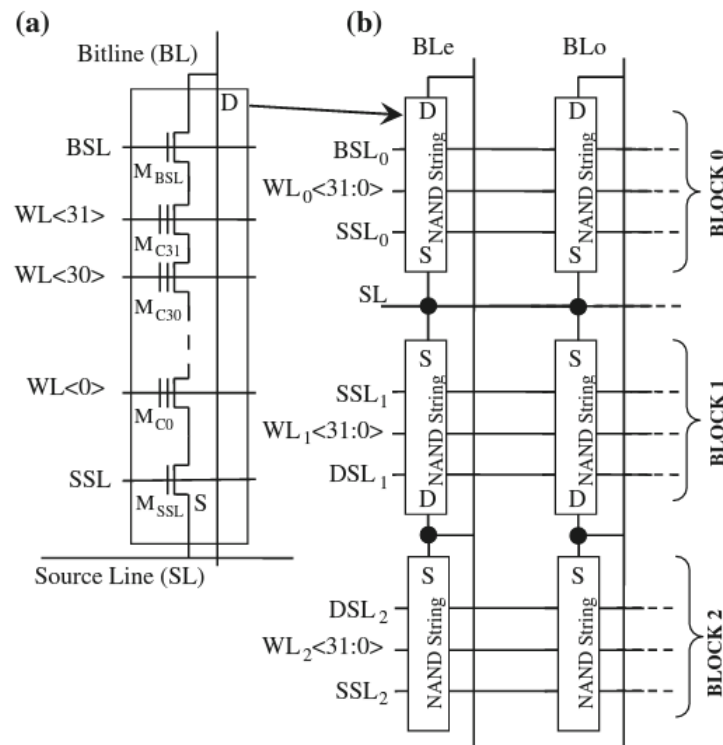
*Note.* From *Inside Solid State Drives (SSDs)* (2nd ed., p. 4), by R. Micheloni et al, 2018, Springer Singapore Pte. Limited. Copyright 2018 by Springer Nature Singapore Pte. Limited.

#### 2.4.2.1 NAND Array

Flash devices are made up of an array of floating-gate transistors which essentially acts as memory cells (Micheloni et al., 2018, p. 4). Depending on the type of architecture of the flash memory, whether it is single-level cell (SLC) or multilevel cell (MLC), the gate can store one or more bits of information (Zambelli et al., 2017). NAND strings arranged in a matrix array fashion are the basic elements of a NAND Flash memory. Figure 2.9 shows the schematic representation of a NAND string and a NAND array. There can be 32, 64 or 128 memory cells in a NAND string which are connected in series. (Micheloni et al., 2018, p. 4). The number of memory cells could even be 150. The matrix way of arrangement of the memory cells reduces the space on silicon (Micheloni & Crippa, 2016, p. 63).

**Figure 2.9**

*NAND String and NAND Array*



*Note.* (a) NAND String and (b) NAND Array. From *Inside Solid State Drives (SSDs)* (2nd ed., p. 5), by R. Micheloni et al, 2018, Springer Singapore Pte. Limited. Copyright 2018 by Springer Nature Singapore Pte. Limited.

A NAND string consists of two selection transistors which are connected at the edges of the string (Micheloni et al., 2018, p. 4). One of those, which connects to the source line (SL), is called the Source select line transistor ( $M_{SSL}$ ) (Joe et al., 2012) and the other which connects to the bit line (BL), is called the bit select line transistor ( $M_{BSL}$ ) (Bavandpour, 2020). The control gates of the memory cells are connected through the word lines (WLs). The adjacent NAND strings, in the direction of the word line, share the same word line, drain select line (DSL), bit select line (BSL) and source line. The same bit line contact is shared by adjacent NAND strings in the bit line direction. The NAND strings which belong to the same word line group form



a block. Memory cells sharing the same word line constitute a logical page (Michelsoni et al., 2018, p. 4).

#### 2.4.2.2 Read

Read operation addresses specific memory cells to fetch the stored information by measuring the voltage of the cell. The read operation is limited to one cell per bit line at a time. While reading a NAND memory cell, a read reference voltage ( $V_{\text{READ}}$ ) is applied at the control gate of the transistor of the cell being read. The cell is turned ON if the read reference voltage is higher than the cell's threshold voltage ( $V_{\text{TH}}$ ). All the other cells in the same string also have to be turned ON so that those cells can pass the information from the cell being read to the output (Cai, Luo, Ghose, et al., 2015). The pass transistors are applied with a pass-through voltage ( $V_{\text{PASS}}$ ) at their control gates which is usually 4V to 5V while the read reference voltage,  $V_{\text{READ}}$ , is usually 0V.  $V_{\text{PASS}}$  is applied independently from the threshold voltage,  $V_{\text{TH}}$ , of the pass cells and is also higher than  $V_{\text{TH}}$ .

#### 2.4.2.3 Program

The program or write operation modifies the threshold voltage of the memory cell which is being programmed. During a program operation, a voltage pulse with predefined amplitude and duration is applied at the gate of the cell by using Incremental Step Programming Pulse (ISPP) algorithm. This is followed by a verification operation (Richter, 2016, p. 80) so as to verify if the cell's  $V_{\text{TH}}$  has reached a higher value than a predefined voltage,  $V_{\text{VFY}}$ . The verification operation succeeds if the cell has reached the expected state and if so, no more program pulses are applied. If the cell has not reached the desired state, it is subjected to another ISPP cycle with slightly increased program voltage. During the program operation a high voltage is applied to the selected word line, as the program operation is

carried out along the word line, even though the programming is done memory cell-wise (Micheloni & Crippa, 2016, pp. 72-73). Multiple Cell Operation Principle is supported by NAND arrays which has an important role in the calculation of program performance of NAND flash memory (Richter, 2016, p. 56).

Fowler–Nordheim (FN) tunnelling is used to efficiently inject electrons to the floating gate (Jin & Lee, 2019), which also provides a high degree of parallelism while programming (Compagnoni et al., 2008). Other benefits of FN tunnelling include the low current requirement to charge the memory cells and the excellent linearity it provides between the program voltage and the cell's  $V_{TH}$  (Richter, 2016, pp. 56-57).

#### 2.4.2.4 Erase

Erase operation changes the state of the memory cell to “1” by removing electrons from the floating gate. To achieve this, a large negative voltage is needed which repels the electrons from the floating gate (Jin & Lee, 2019). Similar to the program operation, erase operation also uses FN tunnelling. The information stored in each block is deleted in few erase steps. The  $V_{TH}$  of the erased cells tends to be more negative due to the large electric field applied to the matrix while conducting the Electrical Erase operation. To compensate this a program-after-erase (PAE) phase is introduced to bring the  $V_{TH}$  of the erased cells close to 0V, but leaving enough margin for the read operation, which reduces floating gate coupling.

An erase verify (EV) operation is carried out after each erase step in order to check if any of the erased cells have  $V_{TH}$  higher than 0V. All the word lines are grounded during the EV phase. If the EV is not successful, indicating the presence of some cells which still remain in the programmed state, another erase pulse is applied. This cycle continues until it reaches the maximum

erase pulses, in which case the erase operation fails (Micheloni & Crippa, 2016, pp. 74-75). This cyclic method is similar to that engaged in the program operation and is known as incremental step-pulse erasing (Spinelli et al., 2017).

### 2.4.3 NAND Memory Reliability

The reliability of non-volatile memory is defined in terms of their capability to store information correctly, retain the stored information for a long period of time and retrieve it without errors. Errors occur in NAND flash memory when the  $V_{TH}$  of the memory cells retrieved during read operation differs from the actual programmed value. This can happen due to several reasons. The most common ones are program errors, disturb errors and data retention errors (Compagnoni & Spinelli, 2019).

#### **Program Errors**

Program errors are the result of incorrect rise in the cell's  $V_{TH}$  that happens during programming. The variations in the number of electrons transported to the storage layer from the channel due to anomalous or erratic tunnelling is the cause of this rise in  $V_{TH}$ . Anomalous tunnelling can occur due to the defects in the tunnel oxide or because of spontaneous changes in the rate of electron tunnelling caused by floating-gate depletion effect (Compagnoni & Spinelli, 2019). While FN tunnelling method has been proved to be sufficiently reliable, anomalous FN tunnelling currents can be generated at random times which significantly push the threshold voltage of the memory cells following programming. This phenomenon, which is termed "erratic bits", affects the performance of the memory cells and can eventually cause over programming issue due to the increase in the  $V_{TH}$  of the cells (Micheloni & Crippa, 2016, p. 32).

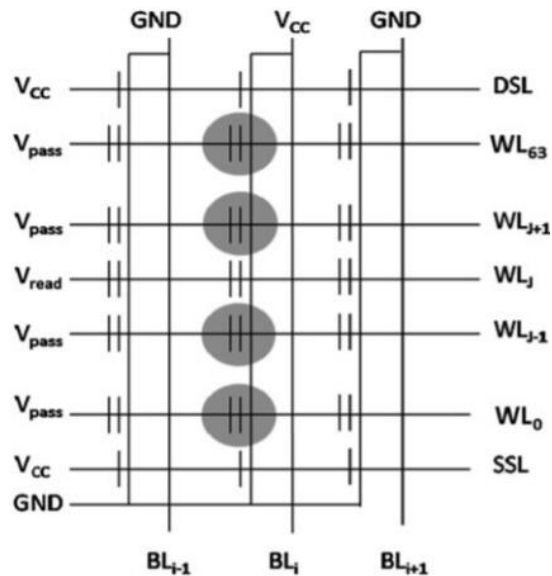
## **Disturbances**

Disturbance is a phenomenon which shifts the threshold voltage of the memory cells due to certain electrical effects caused by the array structure. This results in data errors when the shifted voltage exceeds the reference voltage. The intensity of the disturbance varies with the programming method engaged as some disturbances are intensified by certain programming pattern. Disturbances are not permanent physical damage and can be corrected by an erase operation (Zhang et al., 2021).

Read disturbance, which is the usual source of disturbance in NAND memory, occurs when the same cell is read multiple times without an erase operation in between. Figure 2.10 shows a read disturb representation in a NAND flash array. While performing a read operation on a cell, all the other cells in the same NAND string, the pass cells, also need to be in the ON state. The relatively high pass voltage that is repeatedly applied to the control gate of the pass cells can increase their stored charge. The resulting increase in the threshold voltage of the pass cells may cause read errors (Micheloni & Crippa, 2016, p. 34).

**Figure 2.10**

*Read Disturb Representation in a NAND Flash Array*

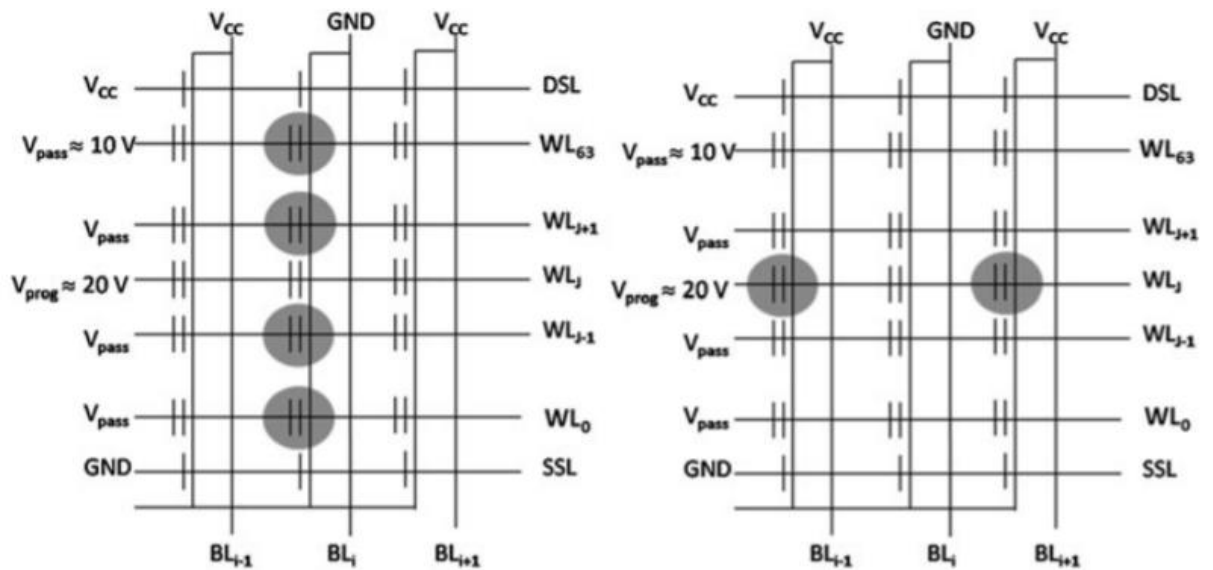


*Note.* The cells potentially affected by read disturb are marked in gray. From *3D Stacked NAND Flash Memories* (p. 34), by R. Micheloni & L. Crippa, 2016, Springer Nature. Copyright 2016 by Springer Science+Business Media Dordrecht.

Two main disturbances that occur during the programming operation are program disturbance and pass disturbance (Zhang et al., 2021). Program disturbance is defined as the phenomenon of increasing the threshold voltage of the cells which are not selected to program, while a selected cell is being programmed, due to weak programming (Chunmei et al., 2017). Program disturbance occurs when the channel potential is low and has effect on the cells belonging to the selected word line (Richter, 2016, pp. 57-58). On the other hand, pass disturbance affects the cells which belong to the same string, the cells belonging to the adjacent word lines, of the cell that is selected for programming, which is caused by the high pass voltage applied to the pass cells (Micheloni & Crippa, 2016, pp. 34-35; Richter, 2016, p. 58). Figure 2.11 shows a representation of pass disturb and program disturb in a NAND flash array.

**Figure 2.11**

*Pass Disturb and Program Disturb Representation in a NAND Flash Array*



*Note.* a) Pass Disturbance b) Program Disturbance. The cells potentially affected by the disturb are marked in gray. From *3D Stacked NAND Flash Memories* (p. 35), by R.

Micheloni & L. Crippa, 2016, Springer Nature. Copyright 2016 by Springer Science+Business Media Dordrecht.

Another disturbance that occurs during the program operation is the edge word line disturbance (Zhang et al., 2021). It affects the cells corresponding to the first and the last word lines which connect the cell strings to the selection transistors at the edges. The difference between the  $V_{TH}$  of the cells which belong to the word lines at the two edges of the cell strings and the average  $V_{TH}$  of all the other cells is the cause of the edge disturbance (Micheloni & Crippa, 2016, p. 35). A large number of electron-hole pairs can be generated by the edge word line unit because of the presence of a large Gate Induced Drain Leakage (GIDL). These electrons are accelerated toward the channel and injected to the floating gate of the memory cells which in turn increase the  $V_{TH}$  of the cells resulting in data errors (Zhang et al., 2021).

## Endurance

A flash memory is subjected to a huge number of program-erase (PE) cycles during its lifetime. Each PE cycle involves the application of strong electric field to the tunnel oxide. The reliability of the NAND flash memory depends on the ability of the tunnel oxide to function properly withstanding the stress. Program and erase operations transport charge into and from the storage layer through thin oxides using FN tunnelling. The oxide wears out slowly and continuously due to the traps created and interfacial damages during electron tunnelling, causing unwanted charge flow. This affects the program operation and the intensity of its effect increases with the number of PE cycles. For example, the tunnelling efficiency gets affected by the electron trapping which results in a decrease in the charge transported into and from the storage layer, under set voltage and time factors, as the PE cycle increases (Micheloni & Crippa, 2016, p. 30).

Endurance is a measure based on the number of program and erase cycles that the storage cells can withstand without causing fatal errors (Zambelli et al., 2017). Raw Bit Error (RBE) number represents the difference in the number of bits between the data read and the original programmed data, without error correction. RBE number is an important indication of the level of endurance change and the flash memory does not function normally when the RBE number exceeds certain value. The RBE number gets elevated and the endurance of the flash memory decreases as the number of PE cycles increases (Zhang et al., 2021). The rate of deterioration of different pages in a block can be different (Jimenez et al., 2014).

Most of the algorithms used to program NAND memory cells implement the technique of applying a sequence of program and erase pulses followed by a verification process to ensure the transfer of desired amount of charge

into and from the storage layer. This method is inevitable to control the amount of charge transferred in the desired direction of the storage layer (Micheloni & Crippa, 2016, p. 31).

### **Data Retention**

Non-volatile memory is characterised by its ability to retain the stored data without any alteration for long period of time (Micheloni & Crippa, 2016, p. 31). Although a NAND flash memory is non-volatile, charge leakage can happen over time leading to retention errors which remains as a major source of flash memory errors (Cai, Luo, Haratsch, et al., 2015). Charge leakage can occur by gradual electron loss even in the absence of any bias voltage, resulting in read errors. If the  $V_{TH}$  goes below the reference level, a programmed cell can be read as erased (Micheloni & Crippa, 2016, p. 31). The charge leakage can be attributed to various physical phenomena, but the prominent reasons are related to the defects of the tunnel oxide (Compagnoni & Spinelli, 2019). Operational temperature can also intensify the charge loss (Richter, 2016, p. 153).

The thickness of the bottom tunnel oxide is a physically limiting factor. This limitation, along with the type of material used and the physical stress caused by the increased number of program and erase cycles, affect the retention time of the flash memory. Retention time, which is a statistical value, is directly related to the number of erase operation performed on the flash memory. (Richter, 2016, p. 153). Flash correct and refresh (FCR) technique is one of the methods used to mitigate retention errors. FCR is basically conducting periodical read, correct and reprogram operations on the flash memory, until the total number of errors occurred over time reaches the error correction capability (Cai, Luo, Haratsch, et al., 2015).



## 2.4.4 Scaling of NAND Flash Memory

### 2.4.4.1 Multi-Level Cell (MLC)

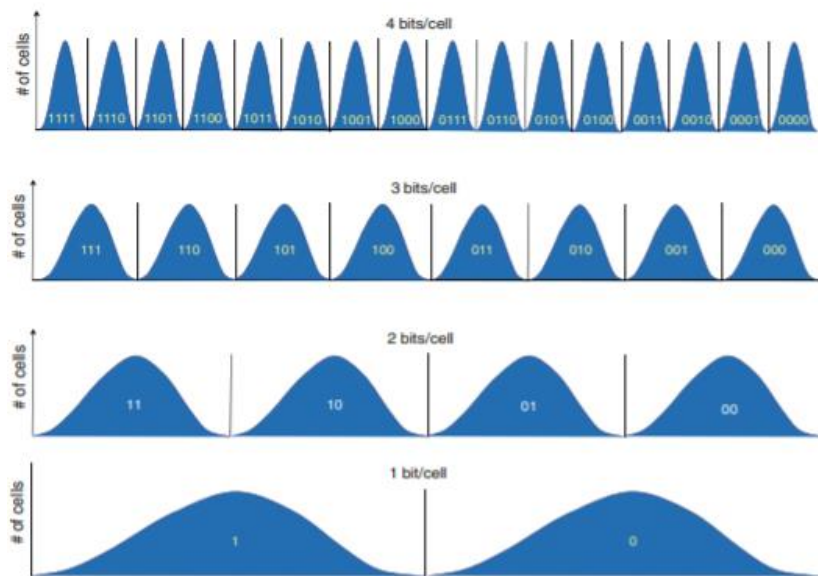
The memory density of flash memory can be increased by using multilevel cell (MLC) operation (Joe et al., 2012). While a single level cell (SLC) memory stores 1 bit per cell, MLC memory can store more than 1 bits per cell depending on the number of storage levels. Currently, flash memory that can store 1 bit, 2 bits, 3 bits and 4 bits per cell are available commercially (Caulfield, 2013) and those are known as SLC, MLC, 8LC - also known as TLC (Michelsoni & Crippa, 2016) and 16LC - also known as QLC (Michelsoni & Crippa, 2016) respectively (Michelsoni et al., 2018, p. 4). Figure 2.12 shows multi-level storage in NAND flash memory.

As the multilevel architecture increases the storage capacity of the flash memory, the increased memory density also introduces new challenges which include random telegraph noise, cell-to-cell interference, stress-induced leakage current (Compagnoni et al., 2008) and read disturbance (Joe et al., 2012). More  $V_{TH}$  levels are set in multilevel architecture depending on the number of bits stored per cell. For example, there are four  $V_{TH}$  levels in an MLC (2 bits per cell) while there are only two  $V_{TH}$  levels in an SLC. The separation between the  $V_{TH}$  levels decreases due to the setting of more levels of  $V_{TH}$  in the available  $V_{TH}$  window warranting intense cell state control. This can affect the  $V_{TH}$  stability causing reliability issues (Compagnoni et al., 2008). Increasing the number of bits stored per cell by moving from SLC to MLC architecture also reduces the durability, which is often referred to as endurance, of the storage cells by one or two orders (Zambelli et al., 2017). Read and program operations are slower in MLC devices compared to SLC devices. Due to this, SLC is preferred for applications requiring high performance and high endurance while MLC

finds its usage where high storage density is required (Koltsidas & Viglas, 2011).

**Figure 2.12**

*Multi-Level Storage in NAND Flash Memory*



*Note.* From *3D Stacked NAND Flash Memories* (p. 66), by R. Micheloni & L. Crippa, 2016, Springer Nature. Copyright 2016 by Springer Science+Business Media Dordrecht.

The memory chip size can be reduced by increasing the number of cells per NAND string. This is achieved by decreasing the cell size per bit. However, this causes a reduction of the cell current as well (Kim et al., 2009). The degradation of the on-cell string current, due to the increased number of pass cells, can affect the read operation resulting in erroneous verification of the state of the selected cell. Electron trapping in the tunnel oxide after program and erase cycles causes the degradation of the cell current even more severe. To remedy this and to ensure reliable read operation, a higher voltage than the threshold voltage of the pass cells needs to be applied to the pass cells. However, the high pass voltage may result in read disturbance (Joe et al., 2012).

#### 2.4.4.2 Charge Trap Cells

Capacitive networks of the flash memory cells need to be scaled while scaling the flash memory cells. When the adjacent floating gates become too close to each other, the coupling between the floating gates increases causing the adjacent cells to communicate through the capacitance between the cells and thereby the information stored in one cell impacts that of the other (Fazio, 2004). This phenomenon is called cell-to-cell interference, which increases heavily as the space between the cells decreases (Li & Quader, 2013). Decreasing the area of the capacitor by reducing the thicknesses of the floating and the control gates can minimise this effect. Alternatively, charge trapping (CT) technique can be used in which non-conductive dielectric charge trap layer having high trap density (Micheloni et al., 2018, p. 94) is employed to store the electrons replacing the conventional conductive floating gate (Fazio, 2004). The insulating charge trapping layer is mostly made up of silicon nitride storage or nanocrystal storage (Fazio, 2004) whereas conductive polycrystalline silicon material is used for floating gate (Wu et al., 2018).

Charge trapping NAND flash cells offer improved scalability with reduced coupling effects between adjacent cells (Wu et al., 2018). Considerable changes in the program, erase and read operations are needed for charge trapping cells (Fazio, 2004).

#### 2.4.4.3 3D NAND Flash Memory

The memory capacity per chip has been increased significantly and rapidly by engaging different approaches as the NAND flash technology evolved over the past decade (Lu, 2012). The gross bit storage density (GBSD), which is defined as the ratio between the storage capacity and the total chip area, has been constantly increased; initially by means of miniaturization of

memory cells followed by a shift from SLC to MLC by increasing the number of bits stored per cell to 2 and lately by engaging TLC technology further increasing the bits stored per cell to 3. However, these approaches used to increase the GBS in 2D (planar) NAND flash arrays also increased the process and system complexity. This was because of the requirement to reduce the technology feature size and also to address the issues resulting from the miniaturization, including the array reliability and performance issues. (Compagnoni & Spinelli, 2019).

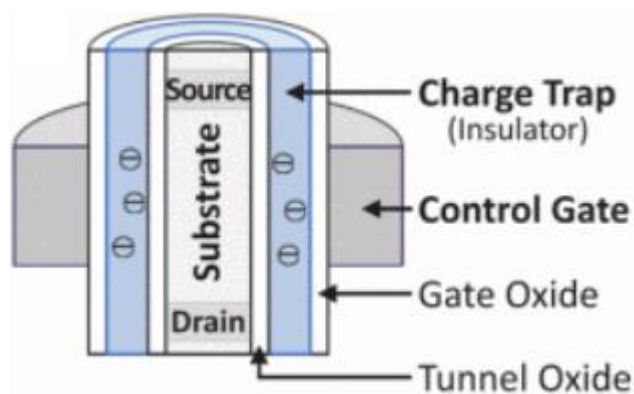
3D NAND flash arrays emerged as a promising technology that can provide high GBS, even with relatively larger memory cells which are stacked vertically, with better performance and reliability (Compagnoni & Spinelli, 2019). Instead of just reducing the minimum feature size, scaling in the Z-axis direction by stacking more memory cells vertically (Micheloni et al., 2018, p. 106) can drastically improve the storage density (Wu et al., 2018). Both charge trap and floating gate technologies can be used for building 3D arrays even though the former is more popular in the 3D architectures because of the simplicity in the fabrication process (Micheloni et al., 2018, p. 106). Another reason for preferring charge trap technology is its improved scalability. Although most of the NAND vendors opted to use the charge trap technology for making 3D NAND memory, Micron/Intel joint venture decided to stick with the floating gate technology as it is a technology that has been around for many generations with known failure modes, which have been addressed already (Micheloni & Crippa, 2016, p. 23).

The cross-section of a charge trapping cell in the 3D NAND flash memory is shown in Figure 2.13. The substrate, cylindrical in shape, is placed in the middle of the cell that is wrapped around by three layers. The inner most

layer is the tunnel oxide layer, middle one is the charge trap layer and the outer layer is the gate oxide. The entire cell is wrapped around by the control gate at the center, over the gate oxide. The cylindrical substrate has the transistor source at one end and the transistor drain at the other end. Current flows from the source to the drain when the cell is turned ON (Luo et al., 2018).

**Figure 2.13**

*Cross-Sectional View of a 3D Charge Trap NAND Flash Memory Cell*



*Note.* Adapted from “HeatWatch: Improving 3D NAND Flash Memory Device Reliability by Exploiting Self-Recovery and Temperature Awareness”, by Y. Luo, 2018, *2018 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, (<https://doi.org/10.1109/HPCA.2018.00050>). Copyright 2018 by IEEE. Adapted with permission.

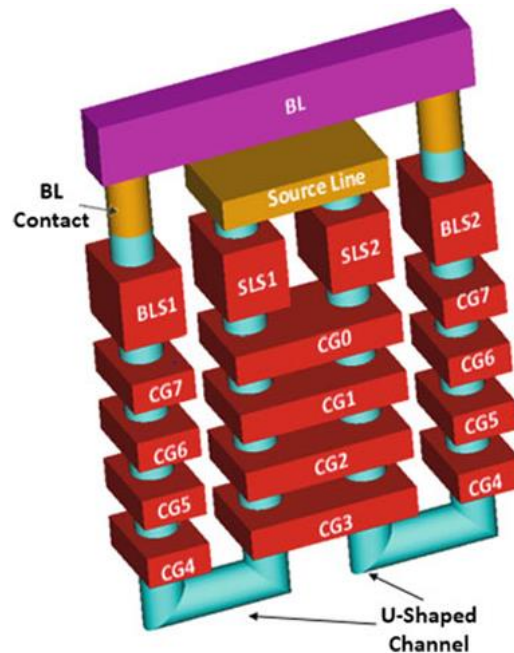
Building 3D NAND devices can be done by adopting two different approaches. The first approach is to use a thin polysilicon substrate to build the cell on it which is similar to the 2D planar arrays and pile more levels; while the second approach is to use a cylindrical channel, also known as vertical channel, to build a charge trapping cell. As the channel width is higher than the 2D arrays, the physical size of the cell is larger in both approaches. However, the introduction of multiple layers offers smaller equivalent area occupation. The second approach has the benefit of better

programming performance compared to the planar devices because of the shape of the charge trapping cell, which is termed “Gate-All-Around” (GAA) (Micheloni & Crippa, 2016, p. 41).

In the 3D devices, the data storage medium is made up of an array of word lines which are connected vertically along the channel side (Wu et al., 2018). Pipe Shaped Bit Cost Scalable (P-BiCS) is an important architecture used to build the vertical channel arrays. It is a modified version of the Bit Cost Scalable (BiCS) architecture improving the source line resistance. In the P-BiCS architecture, two vertical NAND strings are connected together at the bottom of the 3D structure to form a U-shaped single NAND string. Figure 2.14 below shows a P-BiCS array. The connections to the source line and the bit line are made possible through the source line and the bit line selection transistors at the two edges of the string, respectively. As both of the selection transistors are at the same height of the stack, those can be optimised and controlled at the same level contributing to better string functionality (Micheloni et al., 2018, pp. 106-110).

**Figure 2.14**

*P-BiCS NAND Strings*



*Note.* From *Inside Solid State Drives (SSDs)* (2nd ed., p. 111), by Micheloni et al, 2018, Springer Singapore Pte. Limited. Copyright 2018 by Springer Singapore Pte. Limited.

#### 2.4.5 NAND Flash Memory Controller

As discussed in the previous sections, a flash memory needs to be erased before writing to it and the endurance of the memory is limited to certain number of program-erase cycles. In order to manage such limitations, a flash memory controller has been introduced in between the host system and the flash memory. The controller, which is usually implemented as a system-on-a-chip (SoC), is engineered to manage the underlying memory effectively (Do et al., 2019). Designed as an application specific device, the controller consists of an eight-bits or 16-bits processor along with exclusively allocated hardware (Micheloni et al., 2018, p. 8) and own firmware to carryout performance critical tasks, making it a complex embedded system protecting the flash memory. A large static RAM is provisioned for the SSD firmware execution (Cornwell, 2012).

The entire SSD is managed by the controller by performing various tasks including error correction (Tjioe et al., 2012). The flash memory controller takes over the low level memory management from the host system and free-up the host system's resources (Khalifa et al., 2013). It provides suitable interface and protocols for the host system and the flash memory. Optimising data transfer rate, maintaining data integrity and improving the data retention capability of the flash memory are also the responsibilities of the flash memory controller (Micheloni et al., 2018, p. 8). An internal cache with volatile memory is utilized for read/write buffering to improve the data access speed (Tjioe et al., 2012).

#### 2.4.6 Host Interface

The host interface is a physical component placed between the host system and the SSD (Cornwell, 2012). SSDs usually use the traditional HDD interfaces including Parallel ATA, Serial ATA (SATA), Serial Attached SCSI (SAS) and Fibre Channel in order to easily replace HDDs and support most applications (Wong, 2013). These interfaces together with the corresponding device-level queues, Native Command Queuing (NCQ) and Tagged-Command Queueing (TCQ) are examples, enable SSDs to perform I/O scheduling independent of the host systems (Jung et al., 2020). SSDs with the latest storage interfaces USB and Peripheral Component Interconnect Express (PCIe), which are not used in HDDs, are also available in the market (Wong, 2013).

PCIe is extensively used and finds its application in enterprise servers, personal computers, communication systems and industrial applications. It replaced the PCI and PCI-X bus standards. While traditional PCI employs shared parallel bus architecture, PCIe connects each device serially to the host system using separate links articulating an end-to-end topology that



provisions simultaneous two way, both upstream and downstream, data communication. As PCIe is based on serial technology, it reduces the complexity in board design compared to the bus architectures which are based on parallel technology, because of the elimination of wire count (Micheloni et al., 2018, p. 15).

Though the third generation SATA and SAS interfaces support 600 MB/s throughput, which is enough for even the fastest HDDs, it is way under to support the latest NAND flash architectures which deliver much higher Flash bandwidth. This resulted in the shifting of the SSD performance bottleneck from the storage media to the host interface. Faster host interface is required to overcome this performance issue and to take the full benefit of the flash technologies. PCIe comes as a solution for this performance bottleneck. Using PCIe, an HDD can be replaced with an SSD without any system hardware or driver software changes as the disk-drive form factor and the interface provide necessary support delivering higher performance (Micheloni et al., 2018, pp. 20-21).

While PCIe remains as the most preferred choice for high performance SSDs, offering multiple times higher throughput than the other traditional interfaces, Non-Volatile Memory Express (NVMe) further standardises the PCIe interface with CPU level storage queues (Malladi et al., 2017). High performance SSDs that connect to the host systems by means of PCIe can take advantage of NVMe which is the new standard interface (Jin et al., 2017). NVMe specification 1.0, released in March 2011, was collectively developed by more than eighty companies, known as Non-volatile Memory Host Controller Interface (NVMHCI) work group (Sanvido et al., 2008) or just NVMe work group. The goal was to make PCIe based SSDs easily adoptable and also to offer a scalable interface which can unleash the real

performance potential of SSDs. The specification defines optimised register interface, command set and feature set for SSDs using PCIe (Micheloni et al., 2018, pp. 24-25). NVMe can make use of the PCIe sockets to facilitate communication between the storage interface and the host system's CPU. The NVMe driver design is specific to SSDs built on flash memory. As NVMe uses the PCIe bus, rather than SATA bus, it tremendously increases the bandwidth for storage devices (kingston.com, 2021).

#### 2.4.7 Flash Translation Layer

The memory controller enables the SSD to interact with the host system in a manner that is similar to a standard HDD. Because of the differences in the physical characteristics of the flash memory and the disks, a NAND-flash translation layer (FTL) has been introduced in the SSD firmware as an abstraction layer for disk emulation (Chang & Du, 2010). As a result, ordinary disk based file systems became compatible with SSDs (Chang & Chang, 2013). Among the many other functionalities of the FTL, its main function is to map the logical blocks to the physical NAND pages and blocks (Cornwell, 2012). It acts as an indirection layer between the host system and the flash address locations (Malladi et al., 2017). FTL maintains a logical-to-physical address mapping table (Jin et al., 2017).

Each page of the NAND flash memory usually comes with an additional storage space known as spare area which is used to store few bytes of management information including the logical block addresses (LBA) (Chang & Chang, 2013). When the host issues read/write commands specifying the sector addresses and the size of the requests, similar to an HDD, the FTL translates those instructions into a sequence of flash memory intrinsic commands and physical addresses. FTL looks up the mapping table, which is originally constructed by scanning the spare area of the memory

pages, to do the translation. The logical to physical address mapping can be done at page level or block level. Although page level mapping allows more flexibility, it increases the size of the mapping table requiring more amount of SRAM to store it (Jesung et al., 2002).

Erase-before-write and in-order update are two main limitations of flash memory. As data cannot be written to the locations which already hold data, erase operation should precede the write operation. FTL addresses this erase-before-write limitation of the flash memory by preparing empty blocks, which have been erased, in advance and directs the incoming write requests to such empty blocks (Jung et al., 2014). The flash memory management is handled in a log structured fashion by the FTL; which means that the data will never be written back to its original physical location (Jin et al., 2017). Out-of-order data updates within a block are not permitted in modern SSDs. In order to handle this, FTL uses separate empty blocks and writes the data in-order. FTL then remaps the logical and physical addresses of the blocks (Jung et al., 2014).

#### 2.4.8 Wear Leveling

The longevity of the NAND flash-based storage devices is constrained by its physical characteristics. Wear levelling is a technique implemented in the firmware to improve the endurance of the flash storage devices (Chang & Chang, 2013).

The data change in different locations in the storage device differ in frequency. Some information is frequently updated while some are rarely updated. There could be even memory locations which are not at all updated in the entire lifetime of the storage device, after the initial write. The memory blocks which are subjected to frequent write/erase cycles are more stressed compared to those blocks which are less exposed to

write/erase cycles. This results in the uneven wear-out of the memory cells (Micheloni et al., 2018, pp. 10-11). The purpose of the wear leveling technique is to uniformly distribute the writes over the entire array of memory cells so as to prevent premature cell failures (Tjioe et al., 2012).

As the flash memory pages are characterised by “write-once property” which prevents another write operation to the same pages unless the existing data is erased, “out-place updates” are carried out so that the updated information is written to the other free pages. The pages holding the latest copy of the data are considered as live and the ones with the obsolete version are considered as dead. This warrants address translation to map the logical addresses of the data to the physical addresses (Chang & Chang, 2013). Wear levelling technique is based on this logical to physical address translation (Micheloni et al., 2018, p. 11).

On each occasion the host application raises an update request to a logical sector which already holds data, the memory controller maps that logical sector to a different physical sector (Micheloni et al., 2018, p. 11). Blocks with the least number of write/erase cycles are selected by the controller for writing the data (Reddy, 2019, p. 383). The controller keeps track of the mapping either in a specific table or by using pointers. The sector with the obsolete copy of the data is marked as invalid and is ready to be erased. This process ensures that all the physical blocks are uniformly used (Micheloni et al., 2018, p. 11). This is known as dynamic wear levelling. There is another type of wear levelling algorithm called static wear levelling. In static wear levelling, all the data blocks are distributed evenly irrespective of a data update request from the host application. To achieve this, the blocks with the least write/erase cycles from the static data pool

are selected by the controller and swapped with those with high write/erase cycles in the free data pool (Reddy, 2019, p. 383).

#### 2.4.9 Garbage Collection

Garbage collection is another technique incorporated in the FTL that is crucial to the performance and reliability of the SSD. FTL implements garbage collection and wear levelling as two separate modules. While wear levelling technique ensures uniform wear out of the memory cells by distributing the writes across all the memory sectors, garbage collection technique reclaims the dead pages and makes it available again for writing (Tjioe et al., 2012). Garbage collection functions independently without the need for any instruction from the operating system. It can be triggered when the SSD is powered on and start erasing the memory blocks (Shah et al., 2014).

When the amount of free sectors available for the write operation falls below a predefined threshold value or when the SSD is idle for some time (Cornwell, 2012), garbage collection kicks start its operation. It compacts the sectors and deletes multiple, invalid copies (Micheloni et al., 2018, p. 11). Garbage collection reclaims the obsolete pages by first copying the valid pages in the corresponding block to another block followed by the deletion of the block as a whole. However, Garbage collection imposes high toll on the performance of the SSD while the process is running (Tjioe et al., 2012). For minimising the impact of garbage collection on the performance, it can be run in the background (Micheloni et al., 2018, p. 12). Under immense stress, older SSDs had performance issues due to unavailability of blocks for garbage collection. To remedy this, modern SSDs over-provision the physical NAND flash in order to ensure sufficient free blocks and

prevent performance degradation from garbage collection (Cornwell, 2012).

#### 2.4.10 TRIM

TRIM is a data deletion method in the SSD (Reddy, 2019, p. 382). It is a command provided by modern operating systems to instruct the SSD's controller that certain blocks of data are no longer considered to be live and can be wiped internally. This helps the controller to do advance management of the garbage collection overhead (Shah et al., 2014). TRIM is an ATA command facilitating communication between the file system and the SSD controller (Reddy, 2019, p. 382) informing the SSD about the pages whose data have turned to be invalid, due to the erase operations carried out by the user or by the operating system itself (Kim & Shin, 2011), so that SSD can flag those pages as stale (Reddy, 2019, p. 382).

When a delete operation is carried out, the operating system marks the relevant sectors as free for fresh data and also sends a TRIM command to the SSD with the corresponding logical block addresses to be marked as invalid. This makes the SSD alerted that the data residing in those locations need not to be relocated while performing garbage collection. As a result, the number of writes to the flash memory gets minimised contributing to the longer endurance of the memory cells. The actual performance improvement resulting from the use of the TRIM command varies with SSDs as the implementation of the command is not uniform among the various SSD controllers (Kim & Shin, 2011).

TRIM, garbage collection and wear levelling collectively contribute to increase the life time of SSDs (Reddy, 2019, p. 382). At the same time, these internal processes of SSDs, affect the integrity of the data, cause faster data loss and also make data recovery, using techniques including carving, nearly

impossible (Reddy, 2019, pp. 385-386); unveiling new challenges in digital forensics as established practices are based on the way data is stored and located in the conventional HDD (Bednar & Katos, 2011).

## 2.5 Related Work

The anatomy of SSDs, that has been discussed in section 2.4 reveals that SSDs are prone to self-corrosion resulting in evidence destruction over time due to its internal management processes. Researchers have conducted several studies to understand the behaviour of these processes and their impact on the data loss in SSDs. While these studies provide valuable information on the behaviour of the underlying processes, which are controlled by the autonomous internal SSD controller, and their effect on permanent data loss in SSDs; the results of these studies also underscore the need to conduct further studies in this area to aid the forensic investigators to be better equipped as they deal with the modern primary storage devices.

Bell and Boddington (2010) conducted a study to understand the behaviour of an SSD in relation to the retention of deleted data as compared to an HDD and the extend of the corrosion of evidence in the SSD. Four different experiments were conducted, each analysing a different scenario. The test environment comprised of an SSD, an HDD and a forensic bridge. The researchers connected the SSD to the secondary SATA channel on the motherboard and created a single partition with an NTFS file system and filled the entire drive with multiple copies of a template file of 196KB. After filling the entire drive, the partition was formatted using the quick format option and then shutdown the computer. The computer was then restarted within a few seconds and the drive analysed for data which exhibited substantial loss of data, indicating the effect of garbage collection in a very

short span of time. The researchers left the SSD idle for another 15 minutes without any write instructions before removing it from the SATA channel on the mother board and then reattaching the drive to the computer using a USB forensic bridge in order to take a forensic image of the drive. The analysis of the forensic image showed that the drive was almost completely wiped which demonstrated the aggressiveness of the garbage collection process (Bell & Boddington, 2010). However, in the above test scenarios, all the data was recoverable in the case of the HDD.

Bell and Boddington (2010) conducted a further experiment to see if write blockers (forensic bridges) have any effect on the garbage collection process. After filling the entire drive with copies of the template file, followed by a quick format of the drive, the researchers removed the SSD from the SATA channel and reattached it to the computer using the USB write-blocker. The computer was then left idle for 20 minutes before analysing the drive for recoverable data. The analysis revealed that the write-blocker could not prevent the garbage collection process from running as there was data loss. However, only one-sixth of the drive was wiped in this scenario. The researchers are unsure about the reason for the difference in the amount of data wiped compared to the previous scenarios and suggest further research with varying experimental conditions and hardware combinations would provide more insight into this (Bell & Boddington, 2010).

King and Vidas (2011) carried out research on the amount of data loss in three different drive scenarios. These were high drive usage scenario depicting a heavily used drive with hundreds of files consuming almost all the available storage space, low drive usage scenario reflecting a new system with only a few files stored and the OS formatted drive scenario



where a user uses the default format options provided by the operating system to format the disk. The tests were conducted on 15 different SSDs from 10 different manufactures, out of which only six supported TRIM (King & Vidas, 2011). They also used one HDD control to compare and discuss the data retention in the above-mentioned scenarios. The experiments were performed using the operating systems Windows 7, Windows XP and Ubuntu 9.04; however, only Windows 7 had support for the ATA TRIM command. Using a large file of 650MB in size and another small file of 900KB for deletion and recovery, the researchers conducted 144 tests in total. The tests revealed significant data loss for the TRIM enabled drives when used with the TRIM supported operating system Windows 7, with 0% data recovery in most cases and especially for the large files. In contrast, there were nearly 100% data recoverability when the same SSDs were tested with the other two operating systems which lacked TRIM support. It was also observed that more data was able to be recovered in the high usage scenario compared to the low usage scenario. King and Vidas (2011) identify the potential for future research using different TRIM enabled operating systems as Windows 7 was the only operating system with TRIM support that was used in their experiments. They also emphasise the need for future studies taking the time factor in consideration in conjunction with the TRIM command.

Nisbet et al. (2013) explored the effect of TRIM on the data retention capability of an SSD with three different TRIM enabled file systems on 64-bit operating systems. They conducted experiments using NTFS on Windows 7 (SP1), Ext4 on Ubuntu 11.10 and HFS+ on Mac OS X 10.7. The tests were designed in such a way to simulate the combination of idle and active workload scenarios with low and high drive usage conditions. The drive was filled with dummy files to reduce the free space to between 5%

and 10% of the total size of the drive to depict high usage, whereas the drive had only a few files along with the system files and drivers for the low usage scenario. The payload consisted of two small and two large files. The researchers examined the drive for the amount of data it retained at the one-hour and five-hour marks after the deletion of the payload files, as one of the objectives of the research was to identify the effect of time on data retention in conjunction with TRIM. The results of the experiments showed the aggressiveness of TRIM and the researchers observed that the SSD quickly erased the corresponding memory cells following the issuance of a TRIM command. However, the researchers did not anticipate this to be the case always. The results of the examination of the drive at the one-hour mark showed considerable reduction in the size of the recoverable data for the NTFS and HFS+ file systems, bringing it down to below 0.5% of the deleted data; while Ext4 stood out from the others displaying minimal data loss which has been attributed to its batch discard method of handling the TRIM instructions. The researchers also observed that the differences in the data recoverability between the one-hour and the five-hour marks was negligible (Nisbet et al., 2013).

The study conducted by Morningstar (2018) reveals that the triggering of the garbage collection process has direct relation to the ratio of the available free space to the total size of the disk partition, when TRIM is not enabled. However, the results show that the garbage collection process starts within a few minutes of data deletion when TRIM is enabled. The study was conducted using three different SSDs for test scenarios with NTFS and ext4 file systems on Windows 10 Pro and Ubuntu 16.04 respectively. The test with the APFS file system was carried out using a MacBook Air 2013 with a built-in Apple SSD loaded with Mac OS High Sierra 10.13.4. The payload for the experiments consisted of twelve different files, varying in

type and size. The researcher observed that the unavailability of free space in the drive triggers the garbage collection process and also identifies that larger files have less chance of recoverability in such a case (Morningstar, 2018). However, the experiments were designed to test the behaviour in very low and very high drive usage levels only.

The experiments conducted by Nisbet and Jacob (2019) using six different SSDs and three different TRIM enabled file systems reveal that garbage collection, TRIM and wear leveling processes create data changes in the SSDs which are able to be identified by a forensic investigator. Payload sets of 5GB in size consisting of 6 files were used to conduct the tests. The researchers conducted a total of 18 iterations of payload additions and deletions with a 12-hour time gap between iterations for data collection. The test cases included TRIM enabled and disabled scenarios. The results of the tests show that data loss was more during TRIM periods among the iterations but the changes in data were relatively less compared to the changes identified following data additions (Nisbet & Jacob, 2019).

Recent research conducted by Hadi et al. (2021) using two SSDs from different manufactures analysed the effect of TRIM and garbage collection on deleted data over elapsed time. The tests were conducted using the 64-bit version of the Windows 7 Professional Edition, with TRIM enabled, for data deletion and disk format scenarios. Four different datasets, each differing in the range of file sizes and the number of files, with various types of files were used for the experiments. The forensic images of the drives were taken in 1-minute, 1-hour and 12-hour intervals of elapsed time, after the deletion of the datasets or formatting the drives, based on the test cases. The analysis of the results of the experiments, in the drive format test cases, showed that none of the files in the datasets were able to be

recovered for both the SSDs. In the case of the test cases for the delete option, the results exhibit differences between the drives. For one of the SSDs, all of the files in all the four datasets were recovered in the 1-minute and 1-hour marks with the exception of the loss of two of the files from the first dataset in the one-hour mark while all the four datasets lost only a few files in the 12-hour mark (Hadi et al., 2021). However, it is observed that none of the recovered files had matching hashes with the original hashes and all of those were corrupted unable to be opened with the associated software applications, which demonstrated the immediate effect of TRIM and garbage collection on the deleted data. In contrast to the results obtained with the first SSD, the majority of the files from all the four datasets were able to be recovered with the original hash values with the other SSD. Only a few files were corrupted and two of the files were wiped in the 12-hour mark. From their study, Hadi et al. (2021) conclude that time has a significant influence in the amount of recoverable data and state that it reduces with the duration of the elapsed time.

## 2.6 Conclusion

The rapid advancement of the NAND flash technology that resulted in the massive increase in the memory density, reduced chip size and reduction in the cost, along with the higher performance, has motivated the manufacturers of electronic devices to turn to SSDs with NAND flash memory for a data storage solution. The ability of SSDs to use the traditional host interfaces has made the transition from HDD to SSD even easier. Although SSDs using NAND flash memory are categorised as non-volatile, the data retention capability of SSDs depends on various physical factors. The internal processes designed to address the physical limitations of SSDs

also exhibit the volatility of the data stored in SSDs when approached from a digital forensics angle.

Information stored in the HDD does not deplete over time and is recoverable as long as the physical memory locations are not overwritten by new data. Literature, covering the underlying technology of NAND flash memory and the architecture of SSDs, reveals that the data stored in SSDs corrodes over time due to the self-management ability of SSDs, making forensic recovery of digital evidence challenging. Digital forensic practices and processes, which are mostly based on the traditional storage devices, have to be adapted to suit the modern storage devices.

## Chapter 3 – Research Design and Methodology

### 3.1 Introduction

The purpose of this chapter is to define the scope of this research and to develop an appropriate design for this study. The research questions, covered in section 3.2, have been developed based on the knowledge acquired through the literature review presented in Chapter 2 and the opportunities identified for further research by reviewing the current body of research in the related area.

The methodologies adopted by previous studies related to the data recoverability in SSDs and the various components of the corresponding test environments are discussed in Chapter 2 (section 2.5). The method developed for this research and the test environment, which consists of several hardware and software components, are detailed in section 3.3. A high-level plan for carrying out the experiments for the data collection and the payload files used for the tests are outlined in sections 3.4 and 3.5 respectively.

### 3.2 Research Questions

From the literature review presented in Chapter 2, it is evident that SSDs are prone to data loss contributed by the internal processes of the SSDs. While many studies have been conducted to understand the effect of TRIM, garbage collection and wear levelling on the deleted data, very few of those have focused on the time-variant changes on the deleted data caused by these processes. Nisbet et al. (2013) included test cases in their experiments to understand the effect of elapsed time on the recoverability of the deleted data in an SSD. However, the test cases were designed to assess the effect of a short duration of time which was up to 5 hours.

Another limitation was that the experiments were conducted on a single SSD. Hadi et al. (2021) also focused on the impact of the SSDs internal processes on data recoverability over elapsed time, but the tests were conducted only using Windows 7 operating system.

It has been demonstrated by researchers that drive usage levels and activity levels impact the execution of the SSDs internal processes. Although King and Vidas (2011) observe that there is a difference in data recoverability between low and high drive usage levels, the test conditions mainly focused at both extremes of the usage levels. Nisbet et al. (2013) also designed the test cases for their experiments in such a way to understand the behaviour of an SSD, in terms of data recoverability, at very low and very high drive usages. Morningstar (2018) states that insufficiency of free space in the drive causes the triggering of garbage collection and also emphasizes that it has a correlation to the percentage of the available free space in the drive.

The review of the related studies unveils the opportunity to carry out a new study, which complements the existing research in this field, to understand the impact of TRIM and garbage collection on the deleted data in SSDs at varying levels of drive usage in conjunction with different elapsed durations. Having better understanding about the data recoverability in relation with different drive usage levels and elapsed timelines may help forensic investigators to make better action plans when dealing with SSDs. This research aims to attain insights on the relation of disk usage and elapsed time to the recoverability of data from SSDs.

Following is the questions formulated for the purpose of this research.

**Question 1 (Q1).** Can the effects of time be predicted on the amount of deleted data that can be recovered?

**Question 2 (Q2).** What is the effect as SSD disk usage increases on the amount of deleted data that can be recovered?

The following hypothesis has been formulated to explore the relationship of data recoverability to drive usage and elapsed time after data deletion.

**Hypothesis 1 (H1).** The data recoverability of SSDs has an inverse relationship with elapsed time after data deletion and a positive correlation with drive usage.

### 3.3 Research Method

The high-level objective of this research was to understand the variation in the rate of data recovery from SSDs in relation to the drive usage and elapsed time after deletion of data. This research design is quantitative, mainly involving the quantitative variables; percentage of drive usage, elapsed time and the amount of data recovered. The test processes for the data collection was adapted from previous studies in this field which mainly involved forensic imaging of the test drives, data recovery using forensic tools and analysis of the recovered data.

To measure the rate of data recovery, experiments were conducted using different test cases wherein payloads consisting of a set of files, with the files varying in size and type, were loaded to the SSDs followed by the deletion of a set of files keeping the drive usage at predetermined levels. Forensic images of the SSDs were then taken at predefined time intervals, which were later subjected to data recovery processes using digital forensic tools. The rate of data recovery, which is expressed as a percentage of the deleted data, is calculated as below.

$$\begin{aligned} & \text{Rate of Data Recovery (\%)} \\ &= \frac{\text{Total number of recovered files}}{\text{Total number of deleted files}} \times 100 \end{aligned}$$



The experiments were conducted on multiple SSDs from different manufacturers. Data collection was done for three different operating systems on each SSD.

### 3.3.1 Test Environment

The test environment comprised various hardware and software components. The test environment setup in the previous studies, that were reviewed, formed the basis for the test environment preparation for this research. It mainly consists of the components listed in Table 3.1 (Uchiyama, 2014).

**Table 3. 1**

*List of Main Components of the Test Environment*

Component	Purpose
Solid State Drives	Subjected to tests for rate of data recovery
Target Computers	Used to install the target SSDs to perform the tests
Forensic Workstation	Computer, installed with forensic tools, that is used to conduct forensic investigation
Forensic Bridge	Prevents modification to the source while performing forensic tasks
Data Acquisition Tools	Used for the forensic imaging of the source drives
Data Recovery Tools	Used for the forensic recovery of lost or deleted data

The following sections detail the resources used to conduct the tests for the purpose of this study.

#### 3.3.1.1 Solid State Drives

The SSDs for the experiments were selected based on the availability in the New Zealand market and from popular manufacturers based on user votes (Ranker., 2019). Five SSDs from four different manufacturers were used for the tests. All the five SSDs were unused previously as those were bought

brand new. The SSDs were also digitally erased using Kali Linux before each experiment, as defined in the test plan mentioned in Section 3.4.

## I. Samsung 870 EVO

Table 3.2 shows the details of the Samsung 870 EVO SSD.

**Table 3. 2**

*Samsung 870 EVO SSD Specification*

Samsung	
Model Number	MZ-77E500BW
Serial Number	S6P6NM0RC03816
Size	500GB
Interface	Serial ATA
Form Factor	2.5"
NAND Type	MLC
Controller	MKX
Firmware Revision	SVT02B6Q
TRIM Supported	Yes

## II. Kingston A400

Table 3.3 shows the details of the Kingston A400 SSD.

**Table 3. 3**

*Kingston A400 SSD Specification*

Kingston	
Model Number	SA400S37480G
Serial Number	50026B7784581F21
Size	480GB
Interface	Serial ATA
Form Factor	2.5"
NAND Type	3D-NAND TLC
Controller	Phison PS3111-S11
Firmware Revision	S3B00101
TRIM Supported	Yes

### III. Crucial BX500

Table 3.4 shows the details of the Crucial BX500 SSD.

**Table 3. 4**

*Crucial BX500 SSD Specification*

Crucial	
Model Number	BX500
Serial Number	2203E5FE041F
Size	480GB
Interface	Serial ATA
Form Factor	2.5"
NAND Type	3D-NAND TLC
Controller	SM2258
Firmware Revision	M6CR054
TRIM Supported	Yes

### IV. Lexar NS 100

Table 3.5 shows the details of the first Lexar NS 100 SSD used for this study.

**Table 3. 5**

*Lexar NS 100 SSD Specification*

Lexar	
Model Number	NS 100
Serial Number	MA36542011899
Size	256GB
Interface	Serial ATA
Form Factor	2.5"
NAND Type	3D-NAND TLC
Controller	Marvell 88NV1120
Firmware Revision	V4.15.0
TRIM Supported	Yes

### V. Lexar NS 100 - 1

Table 3.6 shows the details of the second Lexar NS 100 SSD used for this study.

**Table 3. 6**

*Lexar NS 100 - 1 SSD Specification*

Lexar - 1	
Model Number	NS 100
Serial Number	MA36542011901
Size	256GB
Interface	Serial ATA
Form Factor	2.5"
NAND Type	3D-NAND TLC
Controller	Marvell 88NV1120
Firmware Revision	V4.15.0
TRIM Supported	Yes

### 3.3.1.2 Target Computers

An HP Pavilion laptop and a MacBook Pro were used, depending on the operating system and file system corresponding to the test cases, as the target computers. The tests were conducted by connecting the target SSD to the target computers using the internal SATA cable. The target SSD was designated as the primary drive and the laptop was booted from the drive for payload additions and deletions. Table 3.7 shows the details of the HP Pavilion laptop and Table 3.8 shows the details of the MacBook Pro used for this study.

**Table 3. 7**

*HP Pavilion Laptop Specification*

HP Pavilion G7	
Screen Size	17"
RAM	4GB DDR3
Graphics	Intel HD Graphics 3000
Processor	2.4GHz Dual-Core Intel Core i5

**Table 3. 8**

*MacBook Pro Specification*

MacBook Pro Mid 2012	
Screen Size	13"
RAM	8GB DDR3
Graphics	Intel HD Graphics 4000
Processor	2.5GHz Dual-Core Intel Core i5

### 3.3.1.3 Forensic Workstation

A Dell Latitude laptop hosting Windows 10 Professional version was used as the forensic workstation. Oracle VM VirtualBox 6.1, which is a cross-platform virtualization software enabling users to run multiple operating systems as virtual machines (Oracle Corporation, n.d), was installed in this laptop to run the Kali Linux operating system, version 2021.1. Kali Linux OS is an open-source Debian-based Linux distribution that is pre-loaded with hundreds of information security and digital forensics tools (Offensive Security, 2022). The choice to use the Kali Linux OS was made as it comes with all the required forensic tools needed for conducting the experiments for the purpose of this study. As Kali Linux does not auto-mount drives, it prevents the source data from modification while conducting the forensic tasks and therefore it negated the need for any additional forensic bridge. Using Kali Linux served as a viable option in terms of the project cost as well, as it is a freely available Linux distribution. Table 3.9 shows the details of the Dell Latitude laptop.

**Table 3. 9***Dell Latitude Specification*

Dell Latitude	
Screen Size	14"
RAM	16GB DDR3
Graphics	NVIDIA NVS 5200M
Processor	2.6GHz Quad-Core Intel Core i7

#### 3.3.1.4 USB 3.0 Hub

A 7-port USB 3.0 hub, manufactured by j5create, was used to connect the SSDs to the forensic workstation for imaging using the forensic tools available in the Kali Linux OS. The j5create USB 3.0 hub, featured as SuperSpeed with data transfer rate of 5GB per second, made it suitable for the experiments as big amount of data transfer was needed at several time intervals. Another useful feature was that the device was self-powered which helped to keep the SSDs powered without being connected to a computer between the imaging time intervals. Table 3.10 shows the details of the USB 3.0 hub.

**Table 3. 10***USB 3.0 Hub Specification*

USB 3.0 Hub	
Manufacturer	J5create
Model Number	JUH377
Number of ports	7
Power Mode	Self-powered/Bus-powered
Cable	USB 3.0 Micro B To USB Type-A
Supported OS	Windows/Mac OS/Linux/Chrome OS

### 3.3.1.5 Hard Drive Enclosures

Hard drive enclosures were used to connect the SSDs to the forensic workstation through the USB 3.0 Hub. Following are the details of the enclosures used.

#### I. Orico 2.5 Inch Hard Drive Enclosure 2520U3

Four hard drive enclosures of this model were used. A different model from the same manufacturer was selected for the fifth enclosure due to insufficient stock of this model in the shop at the time of purchase. Table 3.11 lists the details of this model of enclosures.

**Table 3. 11**

*Orico Hard Drive Enclosure 2520U3 Specification*

Hard Drive Enclosure - Orico 2520U3	
Manufacturer	Orico
Model Number	2520U3
Input	SATA3.0
Output	USB3.0 Micro-B
Transmission Rate	5GBps
Cable	USB Type-A/M To USB Micro-B/M
Supported OS	Windows/Mac OS/Linux

#### II. Orico 2.5 Inch Hard Drive Enclosure 2526C3

One hard drive enclosure of this model was used. Table 3.12 lists the details of this enclosure.

**Table 3. 12**

*Orico Hard Drive Enclosure 2526C3 Specification*

Hard Drive Enclosure - Orico 2526C3	
Manufacturer	Orico
Model Number	2526C3
Input	SATA3.0
Output	USB3.1 Gen1 Type-C
Transmission Rate	5GBps
Cable	USB Type-A/M To USB Type-C/M
Supported OS	Windows/Mac OS/Linux

### 3.3.1.6 External Hard Disk Drives

Two external hard disk drives were used to copy the forensic images of the target SSDs for later examination. Table 3.13 lists the details of the drives.

**Table 3. 13**

*Seagate Hard Disk Drives Specifications*

No#	Manufacturer	Model	Capacity
1	Seagate	Expansion Portable 1TB USB 3.0 Hard Drive SRD00F1	1TB
2	Seagate	Backup Plus Slim Portable Hard Drive STHN2000401	2TB

### 3.3.1.7 Data Acquisition Tool

The “dcfldd” tool that comes with the Kali Linux OS was used for imaging the target SSDs. The “dcfldd” tool is an updated version of the original raw imaging tool “GNU dd”, a Linux command line tool that can be used to



convert and copy files (Muntaha, 2019). The main features of the “dcfldd” tool include quick disk wipes, on-the-fly hashing of the input data as it is being transferred and status output to user on the progress of the data transfer (Filho, 2021).

#### 3.3.1.8 Data Recovery Tools

Two different forensic tools that are available in Kali Linux were used for the data recovery to minimise the possibility of missing out any recoverable file due to any difference in the underlying file carving techniques engaged by each tool. The tools used are listed below.

##### **I. Autopsy**

Autopsy is a graphical interface to several digital forensic tools including The Sleuth Kit (Carrier, n.d-a), which is a set of command line forensic tools for file and volume system analysis. The tools recover deleted and hidden files from raw data (.dd files), EnCase files, AFF (Advanced Forensic Format (Garfinkel et al., 2006, p. 13)) files and disk images without relying on the operating system (Carrier, n.d-b).

##### **II. Foremost**

Foremost is another forensic tool available in Kali Linux that can be used for the recovery of lost or deleted data. The data recovery technique is based on the file headers, footers and the internal data structure. Foremost can recover data from images produced by different tools including dd, Safeback and EnCase (Offensive Security, 2021).

#### 3.3.1.9 Operating Systems and File Systems

Western Governors University (2021) identifies the five most popular operating systems among the contemporary operating systems. Although there are numerous operating systems, including paid and free versions,

the predominant ones are Microsoft Windows, Apple macOS, Google Android OS, Apple iOS and Linux operating system. Among these, Google Android OS and Apple iOS are operating systems for mobile devices. While Microsoft Windows and Apple macOS are proprietary operating systems for personal computers, Linux is an open-source OS for the same purpose.

A global survey conducted by Statista Research Department (2022) on the market share held by the leading operating systems reveals that Microsoft Windows is the most popular operating system followed by Apple's macOS and iOS. It also identifies Linux OS versions as the widely used open-source operating system. For the purpose of this research Microsoft Windows 10 Home Edition, Ubuntu 21.10 (Impish Indri) which is a Debian-based Linux distribution (Wang, 2018, p. 9) and Apple macOS 10.15 (Catalina) were selected as it is reasonable to conduct the tests on the popular operating systems. The file systems used with each of these operating systems for this study are discussed below.

### **I. Microsoft Windows 10 Home with NTFS**

Microsoft released the NTFS (New Technology File System) in July 1993 replacing the previously used FAT file system (Neagu, 2018) which has several limitations including 4GB maximum size for partitions (Microsoft, 2021b). NTFS was released with the along with Windows NT 3.1 (Datarecovery.com, 2015) and has become the primary file system used in the newer versions of the Windows operating systems as it has many advanced features including disk quotas, encryption and rich metadata (Microsoft, 2021a). The Windows NT security model is fully supported by NTFS. It also has support for Portable Operating System Interface for uni-X (POSIX) (Microsoft, 2021b), which is a family of standards developed by IEEE and issued by ANSI and ISO (Indiana University, 2021).

## **II. Ubuntu 21.10 with EXT4**

The ext2, ext3 and ext4 file systems are generally known as second extended, third extended and fourth extended file systems respectively and are engineered for extensibility and backward compatibility. These are identified as Linux file systems and serve as the default file systems for many Linux distributions. Ext3 is an enhanced version of ext2 with journaling feature incorporated; while ext4 is an advanced version of ext3 with more scalability and reliability supporting large file system (Kerrisk, 2021). Extended file systems are open-source file systems (Nordvik, 2022b).

## **III. Apple macOS Catalina with APFS**

The Apple File System (APFS), developed by Apple, was deployed on iPhone and iPad in March 2017 and on macOS later in the same year. Since then, APFS has been the standard file system for Apple devices replacing the HFS+ file system (Nordvik, 2022a). The APFS differs itself from HFS+ by not being a journaling file system. Secure file system transactions are made possible using atomic operations and checkpoint system. It does not use conventional partition tables to define the storage partitions but introduced a new way of structuring the volumes. The file system consists of two layers wherein the external layer, which is termed the container, acts as a managing unit while the internal layer, the volumes, handles the management of files and directories (Göbel et al., 2019). The container is responsible for managing the entire file system and holds the metadata for the container, snapshots and volumes (Nordvik, 2022a). While there would be only one container per implementation of APFS, multiple volumes are

possible depending on the size of the container. Unlike traditional partitions, the volumes don't have reserved blocks or size but they have dynamic size which they share with other volumes in the container (Göbel et al., 2019).

### 3.4 Test Plan

A test plan was developed to conduct the experiments for the data collection in order to study the data recoverability of the SSDs; when the drives are filled with valid data up to 25%, 50% and 75% of the total capacity of the drives, in 8-hour intervals for 24 hours. The following steps explain the plan.

1. Create two partitions in the target SSDs – one for the operating system and another for the test data.
2. Install the operating system corresponding to the test case in the OS partition of the SSD under test.
3. Attach the SSD to the forensic workstation running Kali Linux using USB connection and conduct digital erasure of the data partition.
4. Format the data partition to the target file system.
5. Detach the SSD from the forensic workstation and connect it to the target computer using the internal SATA connection and boot from the drive.
6. Login to the target computer and ensure that TRIM is enabled.
7. Fill the data partition up to 50% of the size of the partition with the payload files and then delete (Shift+Del) 25% of the files to bring the size of the valid data in the partition to 25% of the size of the partition.

8. Initiate the operating system's TRIM command and then shutdown the drive and reattach it to the forensic workstation using USB connection.
9. Using the "dcfldd" tool that is available in the Kali Linux, take the forensic image of the data partition every 8 hours for the next 24 hours.
10. At the end of the 24 hours data collection, conduct digital erasure of the data partition and reformat the data partition to the target file system.
11. Detach the SSD from the forensic workstation and connect it to the target computer using the internal SATA connection and boot from the drive.
12. Fill the data partition up to 75% of the size of the partition with the payload files and then delete (Shift+Del) 25% of the files to bring the size of the valid data in the partition to 50% of the size of the partition.
13. Repeat steps 8 to 11
14. Fill the data partition up to 100% of the size of the partition with the test files and then delete (Shift+Del) 25% of the files to bring the size of the valid data in the partition to 75% of the size of the partition.
15. Repeat steps 8 to 9.
16. Subject the images taken in the previous steps to data recovery tasks using the digital forensic tools Autopsy and Foremost.
17. Analyse the recovered data to understand the patterns related to data recoverability for making conclusions.

### 3.5 Payload Files

Files of different types and sizes were selected for the test files in order to understand if files of particular types or size ranges are specifically targeted by the TRIM and garbage collection processes. Ten files were included in a set of payload files, with sizes varying from 24KB to 2.46GB, adding together to 5GB in total size. Table 3.14 shows the list of test files used for the experiments.

**Table 3. 14**

*Test Files*

No#	File	Type	Size	Sha1
1	01.zip	Zip	1GB	90ca840948bde21fbe9777127bae080bc3eee993
2	1.mp4	Video	35.1MB	7bcb29c7c9e18b372622e95917cb339bd0fa76e7
3	2.mp4	Video	237MB	43f9040aa64a0d19fc59e9cb6c3099f6ae3ec2bc
4	3.jpg	Image	24KB	a1c5d538af9989aaf9a68f41beee6157b2441338
5	4.doc	Document	68KB	64f38d6f0380e0cf3484ae03532fc7b36a5f8935
6	5.pdf	Document	136KB	03ac97dab3fbb1b72c272eeaeccdead22aa2404f
7	6.pdf	Document	548KB	157ac6c978a3c0a774ab8e15ae61455cbc114c45
8	7.DAT	Video	641MB	a853c1fff87d8e3631a020061702ce94eea9e10a
9	8.DAT	Video	658MB	2e7781d6bf0aedaefe357a625fd99bc5d9d4148c
10	9.mp4	Video	2.46GB	acb9731f1c100afeb43e61aba2f0ed5de571347e

### 3.6 Conclusion

This chapter presented the opportunity identified to conduct a new study, that compliments the previous studies, which aims to understand the impact of drive usage and elapsed time on the data recoverability of SSDs. The chapter defined the research questions and detailed the proposed methodology. The test environment including the key resources used were also covered in this chapter.

## Chapter 4– Findings

### 4.1 Introduction

The literature review, presented in Chapter 2, served as the base for the formation of the research questions and the research methodology was established in Chapter 3. This chapter presents the findings obtained by conducting a series of experiments, using five different SSDs, which were designed for data collection in order to investigate the research questions.

For conducting the experiments, two partitions were created in the SSDs, one for the operating system and the other for the test data. The data partition was created with 20GB in size. There were three iterations of the method, one iteration for each 25%, 50% and 75% drive usages, for each SSD and OS/file system combinations.

### 4.2 Microsoft Windows 10 Home with NTFS

Microsoft Windows 10 Home Edition was installed in each of the SSDs and the data partition was created using the Disk Management utility available in Windows. Figure 4.1 shows the creation of the data partition using the Disk Management utility. The data partition of each of the five SSDs was erased digitally using the `dcfldd` tool by connecting to the forensic workstation and then formatted it to the NTFS file system. Figure 4.2 shows the command executed to conduct digital erasure of the drive partition.

### Creating Data Partition Using Disk Management Utility



```
(root@kali)-[~]
# dcfldd if=/dev/zero of=/dev/sdc2 conv=sync,noerror
```

In order to conduct the test for 25% drive usage, each of the drives was connected to the target computer internally and two sets of the payload files were copied to the data partition, which comprised of 20 files in total that used 10GB space of the partition, followed by the deletion of one set of the payload files, consisting of 10 files and 5GB in size, to bring the drive usage level to 25%. Using the Optimize Drives option provided by the Windows 10 operating system, the TRIM command was then initiated



before shutting down the computer. This was followed by the forensic imaging of the drive using the `dcfldd` tool, as shown in Figure 4.3, and then in every 8-hour intervals for the next 24 hours, by connecting the five SSDs to the forensic workstation using the USB 3.0 hub. Figure 4.4 shows the forensic workstation performing the imaging task.

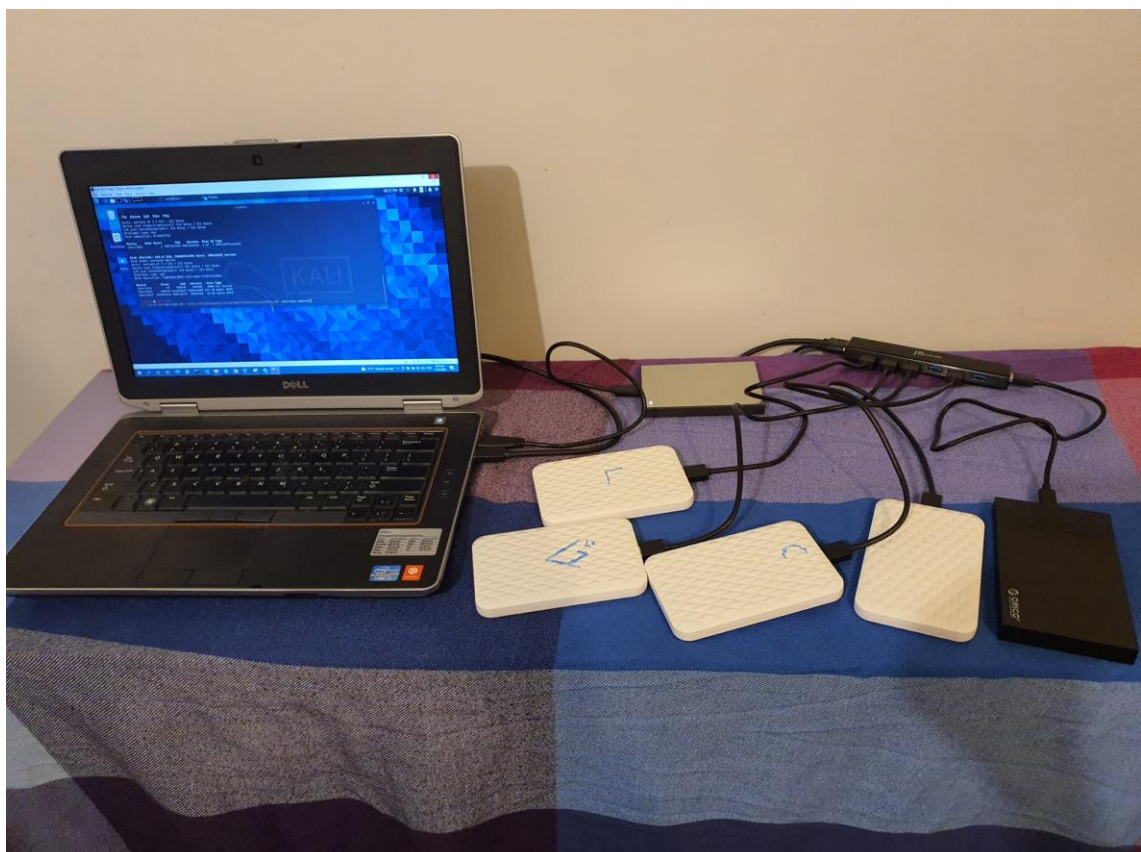
**Figure 4. 3**

*Forensic Imaging Using dcfldd Tool*

```
(root@kali)-[~]
# dcfldd if=/dev/sdc3 of="/media/root/backupplus/vinay/ddfiles/win/crucial/0hrs.dd" conv=sync,noerro
r
53504 blocks (1672Mb) written.
```

**Figure 4. 4**

*SSDs Connected to the Forensic Workstation Using USB 3.0 Hub*



The forensic images were later examined for the payload files. Table 4.1 shows the summary of the remaining files obtained from each of the SSDs,

including the files that were recovered using the forensic data recovery tools foremost and Autopsy.

**Table 4. 1**

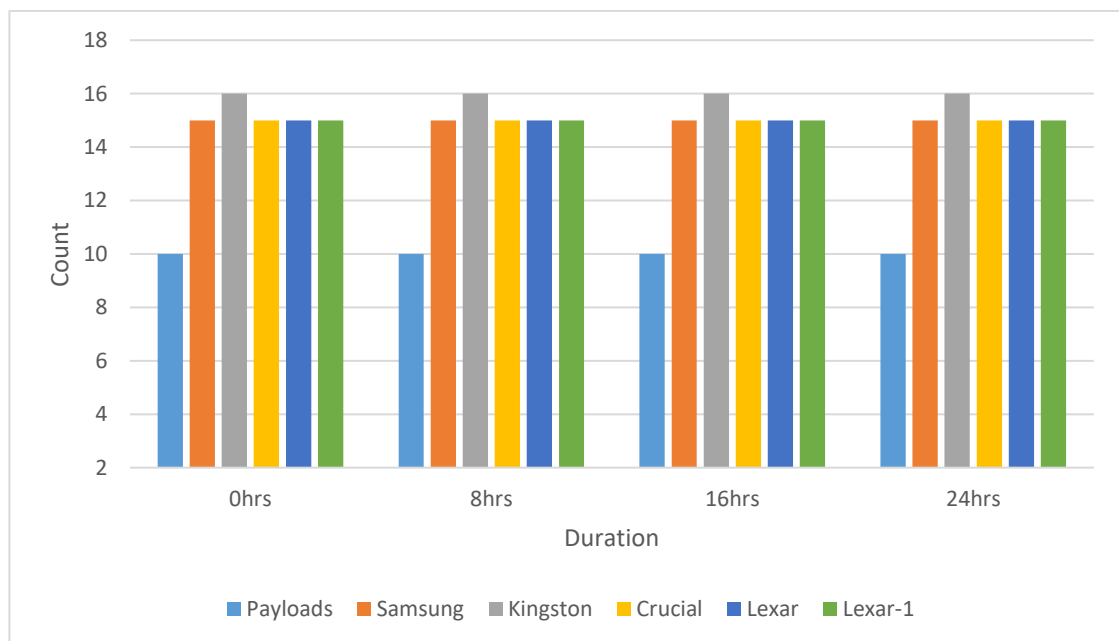
*Summary of Remaining Files: Windows 10 Home With NTFS - 25% Drive Usage*

Duration (hrs)	Payload Files	Samsung	Kingston	Crucial	Lexar	Lexar-1
0	10	15	16	15	15	15
8	10	15	16	15	15	15
16	10	15	16	15	15	15
24	10	15	16	15	15	15

Figure 4.5 shows a representation of the changes in the remaining files across the five different SSDs, in 8-hour intervals, as the time elapsed from 0 to 24 hours. The 0hrs corresponds to the forensic image of the drives that were taken soon after the initiation of the OS TRIM command.

**Figure 4. 5**

*Changes in Remaining Files: Windows 10 Home With NTFS - 25% Drive Usage*



Based on the summary of remaining files extracted from the SSDs, as listed in Table 4.1, Table 4.2 has been generated which provides a summary of the number of recovered files as compared to the total number of deleted files, for each of the drives.

**Table 4. 2**

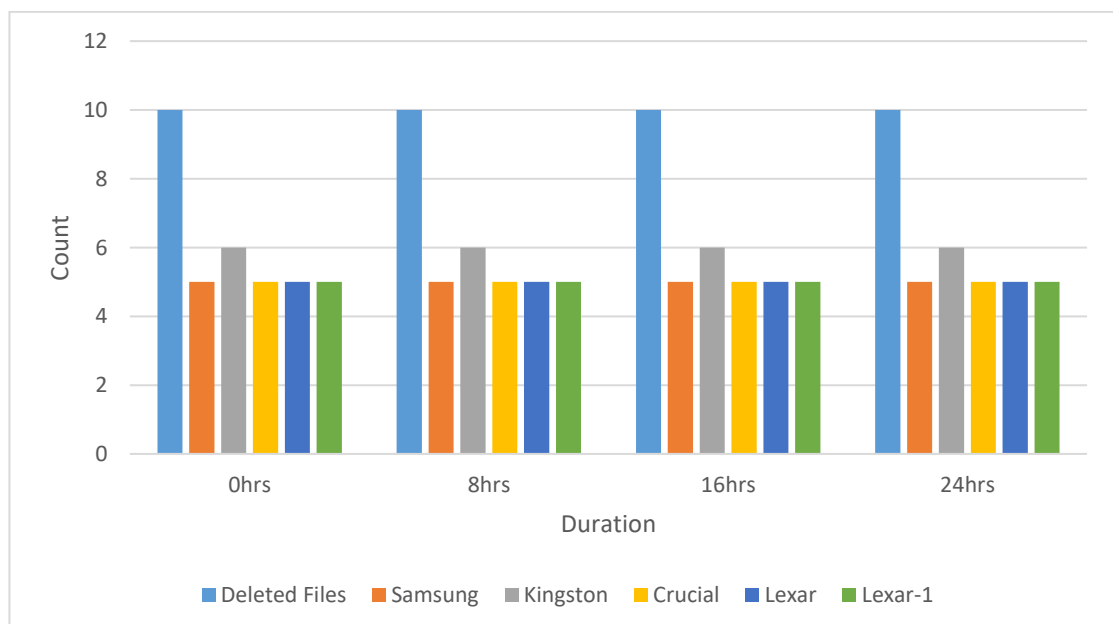
*Summary of Recovered Files: Windows 10 Home With NTFS - 25% Drive Usage*

Duration (hrs)	Deleted Files	Samsung	Kingston	Crucial	Lexar	Lexar-1
0	10	5	6	5	5	5
8	10	5	6	5	5	5
16	10	5	6	5	5	5
24	10	5	6	5	5	5

Figure 4.6 shows a comparative view of the number of recovered files from each of the SSDs, as the time elapsed from 0 to 24 hours, in 8-hour intervals.

**Figure 4. 6**

*Comparative View of the Recovered Files: Windows 10 Home With NTFS - 25% Drive Usage*



The details of the recovered files, from each of the drives, which shows the original file size, recovered file size, hash value of the original files and the hash value of the recovered files, corresponding to the 0Hrs image of the drives, can be found in Appendix A.

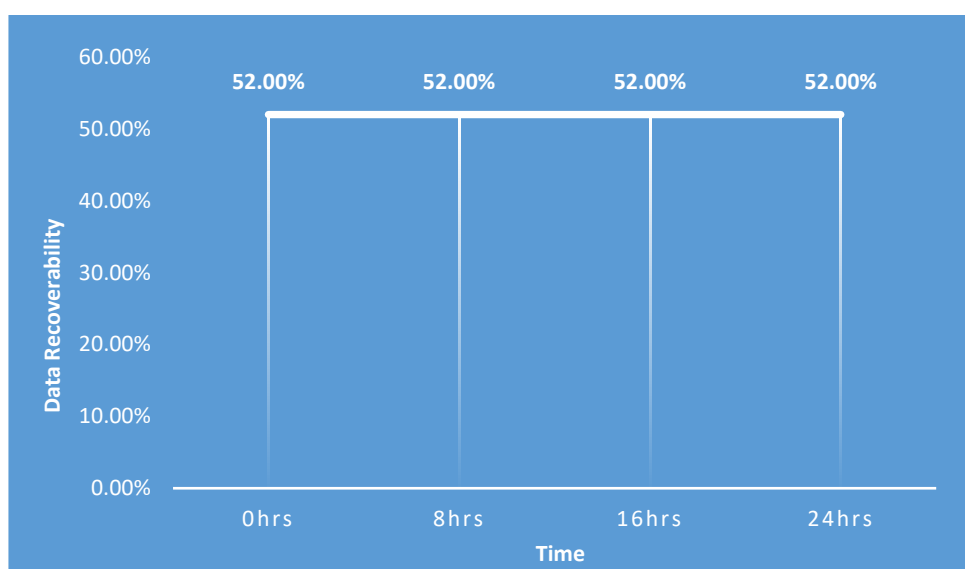
Auditing of the details of the recovered files showed that the Sha1 hash value of only one of the files, 3.jpg, matched with the Sha1 hash value of its original file. While all the other recovered files differed in the Sha1 hash value with that of the corresponding original files, the size of all of those were the same as of their corresponding original files apart from one of the files, 01.zip, which exhibited a significant difference from its original file's size. The results were consistent across all the SSDs.

The results remained the same throughout the different timelines in the 24 hours of data collection other than that one recovered file, 4.doc, displayed a difference in its sha1 hash value with the different timelines and across all the SSDs, except a few matches, even though its size remained the same.

The rate of data recoverability was calculated based on the formula presented in Section 3.3. Table 4.3 shows the percentage of data that was able to be recovered from each of the SSDs at the different timelines. It also shows the average data recoverability among the drives. Figure 4.7 shows the average data recoverability at the 25% drive usage level.

**Table 4. 3***Data Recoverability: Windows 10 Home With NTFS - 25% Drive Usage*

Time	0hrs	8hrs	16hrs	24hrs
Samsung	50.00%	50.00%	50.00%	50.00%
Kingston	60.00%	60.00%	60.00%	60.00%
Crucial	50.00%	50.00%	50.00%	50.00%
Lexar	50.00%	50.00%	50.00%	50.00%
Lexar-1	50.00%	50.00%	50.00%	50.00%
Average	52.00%	52.00%	52.00%	52.00%

**Figure 4. 7***Average Data Recoverability: Windows 10 Home With NTFS - 25% Drive Usage*

#### 4.2.2 50% Drive Usage

After completing the tests for 25% data usage, the data partition of each of the SSDs were digitally erased, formatted to NTFS file system and then detached from the forensic workstation. For conducting the tests for 50% drive usage, three sets of the payload files were copied to the data partition of each of the five SSDs subjected to the tests, consisting of 30 files in total that consumed 15GB space of the partition. Out of which, one set of the payload files was deleted to reduce the drive usage level to 50%. The computer was then shutdown following the initiation of the OS TRIM

command. The SSDs were then connected to the forensic workstation, via the USB 3.0 hub, to take the forensic images of the drives for the following 24 hours with 8-hour time gap in between. Table 4.4 shows the audit summary of the remaining files extracted from the forensic images using the forensic data recovery tools. The representation of the difference in the number of files obtained from the five different SSDs, at 50% drive usage level, is shown in Figure 4.8.

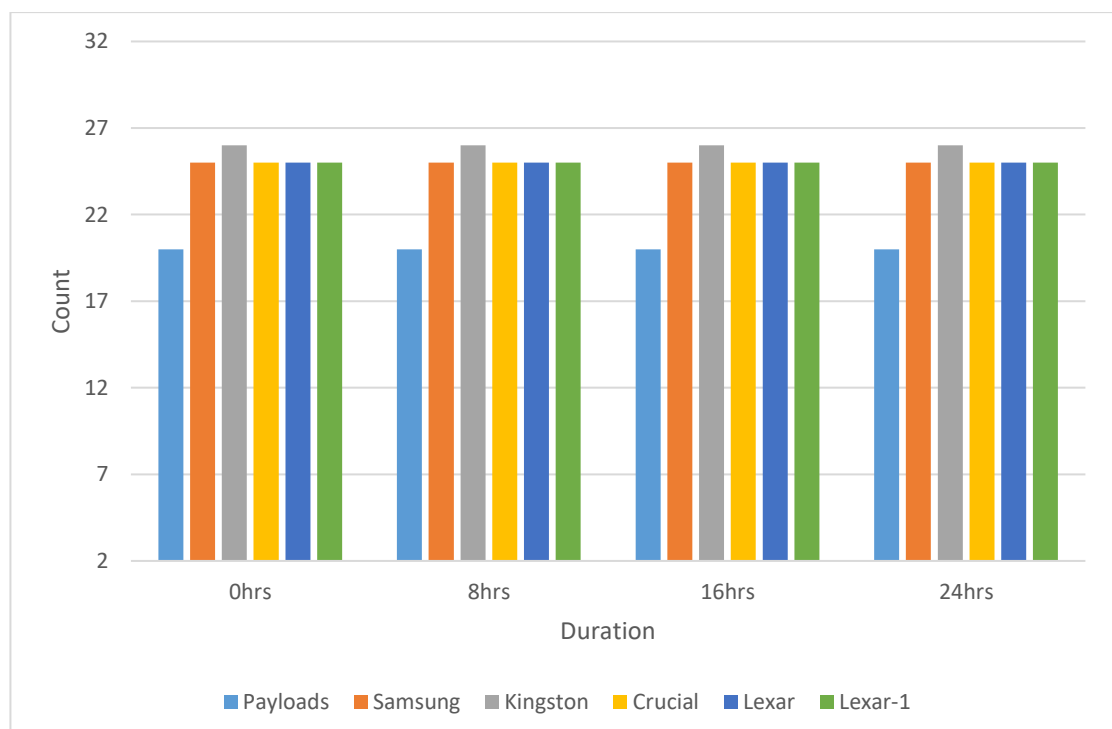
**Table 4. 4**

*Summary of Remaining Files: Windows 10 Home With NTFS - 50% Drive Usage*

Duration (hrs)	Payload Files	Samsung	Kingston	Crucial	Lexar	Lexar-1
0	20	25	26	25	25	25
8	20	25	26	25	25	25
16	20	25	26	25	25	25
24	20	25	26	25	25	25

**Figure 4. 8**

*Changes in Remaining Files: Windows 10 Home With NTFS - 50% Drive Usage*



Further analysis of the extracted files from the SSDs was conducted to obtain the number of recovered files as listed in Table 4.5. Out of the 10 deleted files, 6 files were recovered from the Kingston A400 SSD, while 5 files were recovered from the other four SSDs. Figure 4.9 shows a comparison of the recovered files from the SSDs, at 50% drive usage.

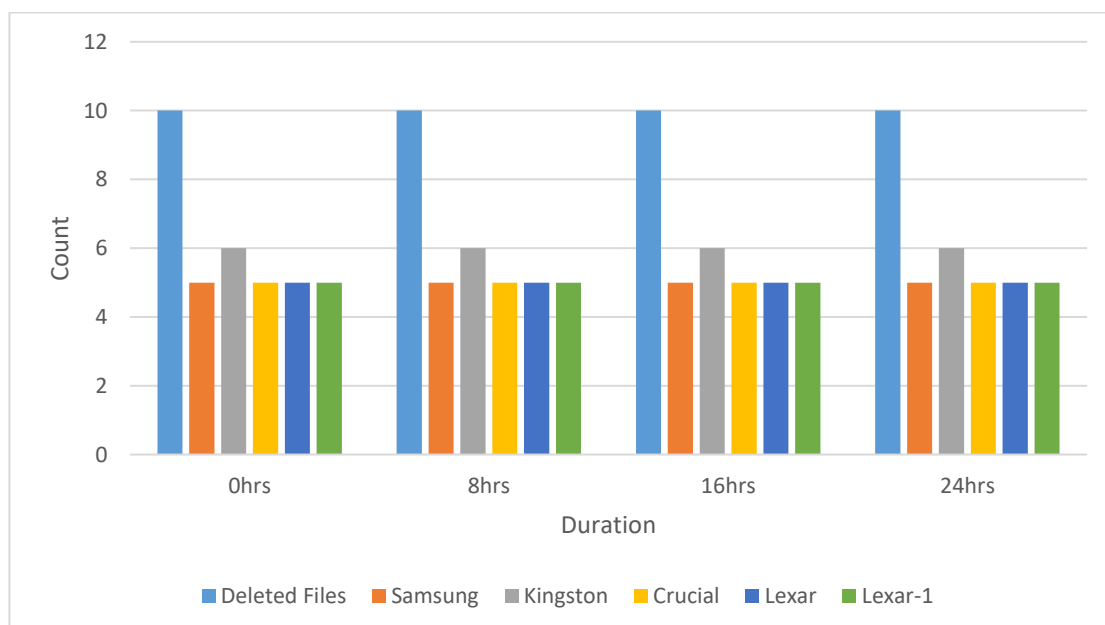
**Table 4. 5**

*Summary of Recovered Files: Windows 10 Home With NTFS - 50% Drive Usage*

Duration (hrs)	Deleted Files	Samsung	Kingston	Crucial	Lexar	Lexar-1
0	10	5	6	5	5	5
8	10	5	6	5	5	5
16	10	5	6	5	5	5
24	10	5	6	5	5	5

**Figure 4. 9**

*Comparative View of the Recovered Files: Windows 10 Home With NTFS - 50% Drive Usage*



It has been observed that the size of each of the recovered files matched with the size of the corresponding original file, except the file 01.zip. The

file 3.jpg was the only file which had the same hash value of its original file. Appendix A shows the details of the files recovered from the 0Hrs image of the drives. The hash value of all the recovered files remained constant throughout the key timelines, for each of the drives, defined for the data collection with one exception, 4.doc.

Table 4.6 shows the data recoverability at the various timelines and the average data recoverability at the 50% disk usage level. Figure 4.10 shows the average data recoverability at the 50% drive usage level.

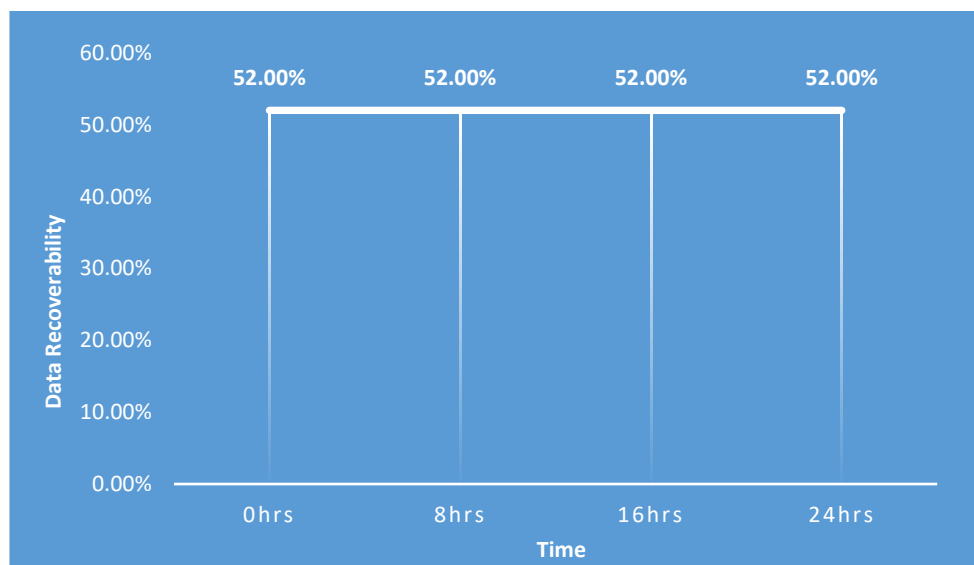
**Table 4. 6**

*Data Recoverability: Windows 10 Home With NTFS - 50% Drive Usage*

Time	0hrs	8hrs	16hrs	24hrs
Samsung	50.00%	50.00%	50.00%	50.00%
Kingston	60.00%	60.00%	60.00%	60.00%
Crucial	50.00%	50.00%	50.00%	50.00%
Lexar	50.00%	50.00%	50.00%	50.00%
Lexar-1	50.00%	50.00%	50.00%	50.00%
Average	52.00%	52.00%	52.00%	52.00%

**Figure 4. 10**

*Average Data Recoverability: Windows 10 Home With NTFS - 50% Drive Usage*





### 4.2.3 75% Drive Usage

To conduct the test for 75% disk usage, all the five SSDs were subjected to digital erasure and reformatted to NTFS file system. The experiment engaged the same process which was carried out for the 25% and 50% disk usage levels. Four sets of payload files, containing 40 files in total consuming 20GB space, were copied to the data drive of the SSDs and then deleted one of the sets, so as to depict 75% drive usage. This was followed by the triggering of the OS TRIM and then navigated through the further process of data collection for the following 24-hour time period. Table 4.7 shows the summary of the number of files extracted from the five SSDs at the different timelines. The representation of this information is shown in Figure 4.11.

**Table 4. 7**

*Summary of Remaining Files: Windows 10 Home With NTFS - 75% Drive Usage*

Duration (hrs)	Payload Files	Samsung	Kingston	Crucial	Lexar	Lexar-1
0	30	36	36	36	36	36
8	30	36	36	36	36	36
16	30	36	36	36	36	36
24	30	36	36	36	36	36

**Figure 4. 11**

*Changes in Remaining Files: Windows 10 Home With NTFS - 75% Drive Usage*

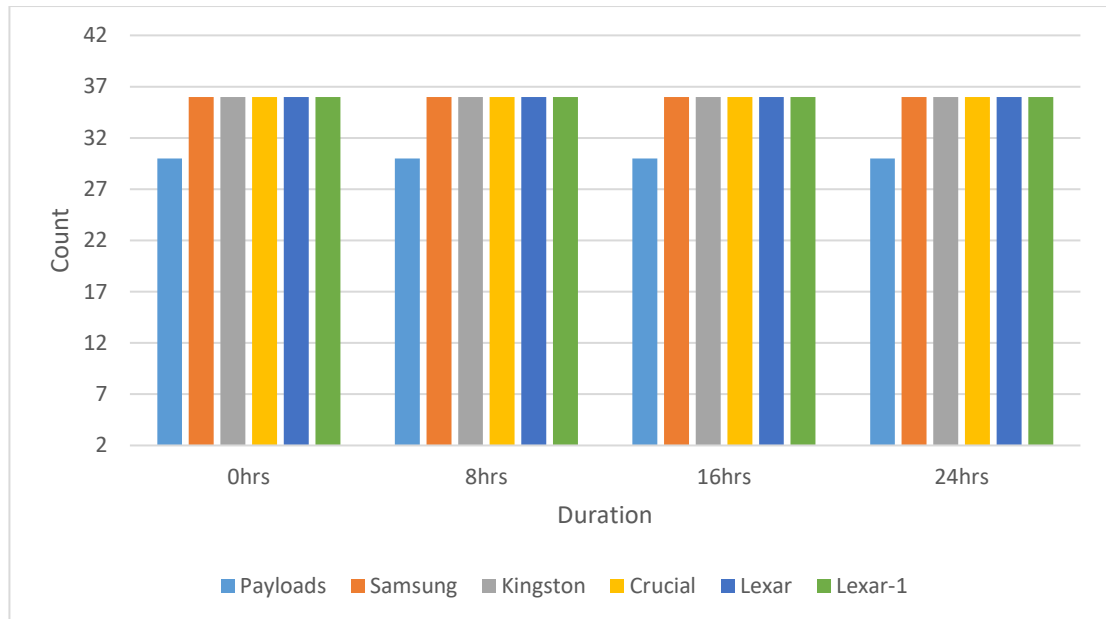


Table 4.8 has been derived from the summary of the remaining files, showing the number of deleted files which were able to be forensically recovered. Among the 10 deleted files, 6 files were able to be recovered from all the five SSDs that were subjected to the experiment. Figure 4.12 shows a representation of this information.

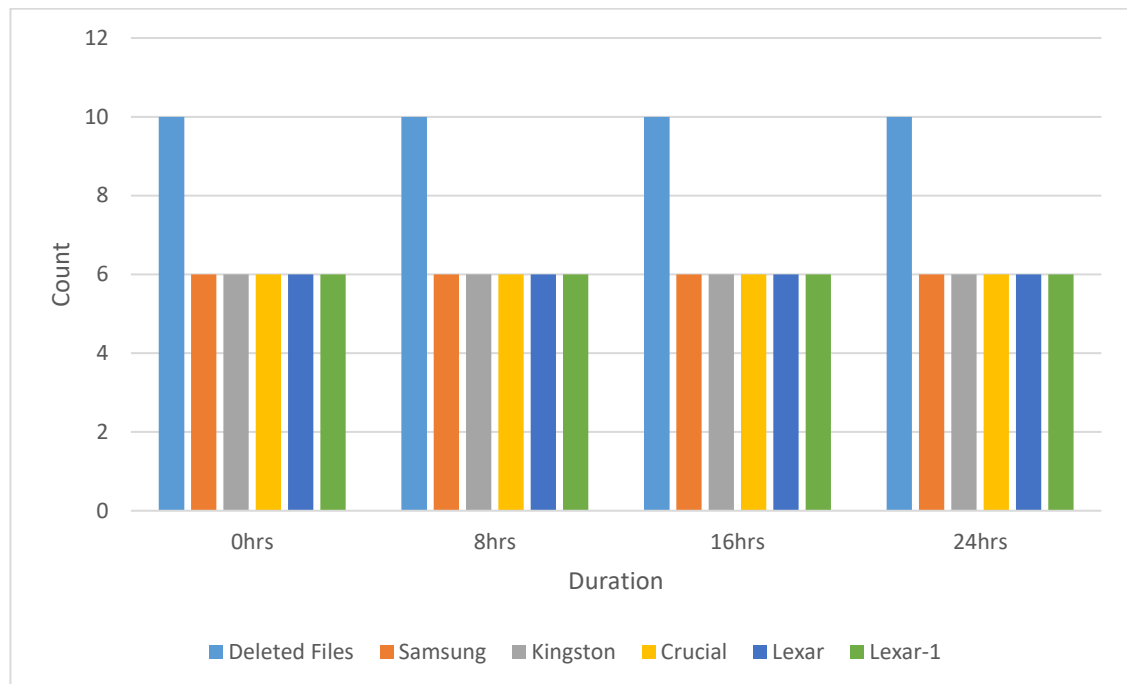
**Table 4. 8**

*Summary of Recovered Files: Windows 10 Home With NTFS - 75% Drive Usage*

Duration (hrs)	Deleted Files	Samsung	Kingston	Crucial	Lexar	Lexar-1
0	10	6	6	6	6	6
8	10	6	6	6	6	6
16	10	6	6	6	6	6
24	10	6	6	6	6	6

**Figure 4. 12**

*Comparative View of the Recovered Files: Windows 10 Home With NTFS - 75% Drive Usage*



The metadata of the recovered files, from the 0Hrs image of the SSDs, has been included in Appendix A. Review of the file metadata showed that the sha1 hash value of only one of the files, 3.jpg, matched with that of its original file. The size of five of the recovered files matched with the size of the corresponding original file while one file exhibited a difference, 01.zip. Although the hash values of five of the recovered files remained constant throughout the key timelines defined for the data collection, the file 4.doc displayed difference in hash value at different timelines, for each of the drives.

Table 4.9 shows the rate of data recoverability, derived from Table 4.8, across the SSDs at the different timelines including the average recoverability. Figure 4.13 represents the average data recoverability at the key timelines.

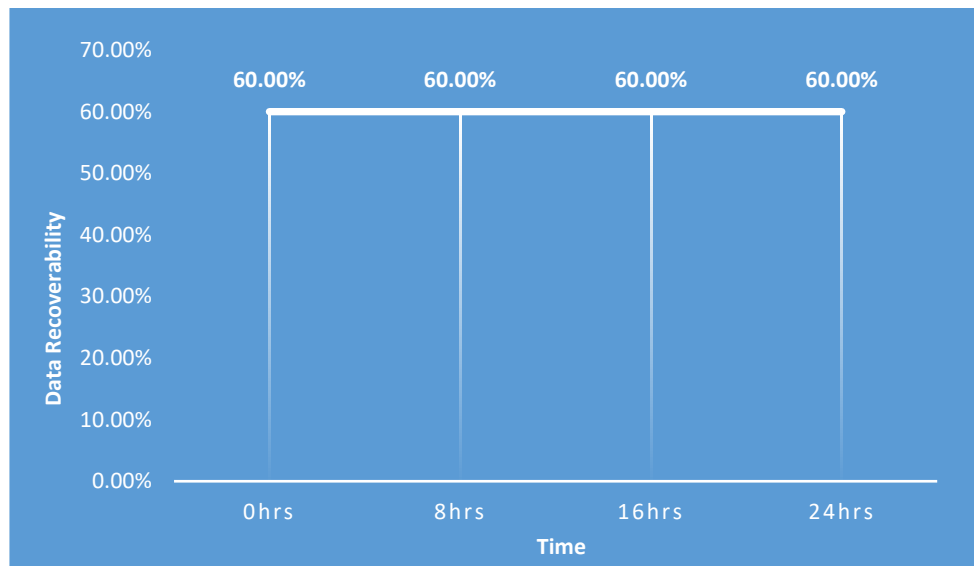
**Table 4. 9**

*Data Recoverability: Windows 10 Home With NTFS - 75% Drive Usage*

Time	0hrs	8hrs	16hrs	24hrs
Samsung	60.00%	60.00%	60.00%	60.00%
Kingston	60.00%	60.00%	60.00%	60.00%
Crucial	60.00%	60.00%	60.00%	60.00%
Lexar	60.00%	60.00%	60.00%	60.00%
Lexar-1	60.00%	60.00%	60.00%	60.00%
Average	60.00%	60.00%	60.00%	60.00%

**Figure 4. 13**

*Average Data Recoverability: Windows 10 Home With NTFS - 75% Drive Usage*



### 4.3 Ubuntu 21.10 with EXT4

The Windows 10 operating system, that was installed in the SSDs, was replaced by installing Ubuntu OS version 21.10. The Ubuntu OS installer removed the data partition, during the installation process, allocating all the available storage space to the OS partition. Hence, the data partition was recreated by performing the following process.

1. Shrunk the OS partition to 25Gb to free up the remaining space by using the 'e2fsck' and 'resize2fs' commands. The first command ensured that

the file system was clean and the second command resized the partition to 25GB. Figure 4.14 shows the execution of the commands against one of the SSDs.

2. Deleted the OS partition and created a new partition for the OS, starting at the same sector as of the original OS partition, allocating 25GB size.
3. Created another partition, which was meant to be used as the data partition for the experiments.

**Figure 4. 14**

*Shrinking the OS Partition*

```
(root@kali)-[~]
# e2fsck -f /dev/sdb3
e2fsck 1.46.1 (9-Feb-2021)
Pass 1: Checking inodes, blocks, and sizes
Pass 2: Checking directory structure
Pass 3: Checking directory connectivity
Pass 4: Checking reference counts
Pass 5: Checking group summary information
/dev/sdb3: 167695/15597568 files (0.1% non-contiguous), 3432736/62382848 blocks

(root@kali)-[~]
# resize2fs /dev/sdb3 25G
resize2fs 1.46.1 (9-Feb-2021)
Resizing the filesystem on /dev/sdb3 to 6553600 (4k) blocks.
The filesystem on /dev/sdb3 is now 6553600 (4k) blocks long.
```

Figure 4.15 shows the process of creating the OS and the data partitions on one of the SSDs. After the creation of the new data partition, it was subjected to digital erasure, as shown in Figure 4.16, and then formatted it to the ext4 file system using the following command.

`'mkfs -t ext4 /dev/sdb4'`

Figure 4.17 shows the formatting of the data partition to the ext4 file system. The resulting list of the partitions are shown in Figure 4.18.

**Figure 4. 15**

*Creating Partitions for the OS and Data*

```
root@kali: ~  
File Actions Edit View Help  
root@kali:~  
# fdisk /dev/sdb  
  
Welcome to fdisk (util-linux 2.36.1).  
Changes will remain in memory only, until you decide to write them.  
Be careful before using the write command.  
  
Command (m for help): p  
Disk /dev/sdb: 238.47 GiB, 256060514304 bytes, 500118192 sectors  
Disk model: External Device  
Units: sectors of 1 * 512 = 512 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disklabel type: gpt  
Disk identifier: B99EC4C2-E77E-4AF2-8EF7-14826809998B  
  
Device      Start      End      Sectors  Size Type  
/dev/sdb1    2048       4095     2048     1M BIOS boot  
/dev/sdb2    4096    1054719  1050624  513M EFI System  
/dev/sdb3   1054720  500117503  499062784  238G Linux filesystem  
  
Command (m for help): d  
Partition number (1-3, default 3): 3  
  
Partition 3 has been deleted.  
  
Command (m for help): n  
Partition number (3-128, default 3):  
First sector (1054720-500118158, default 1054720):  
Last sector, +/-sectors or +/-size{K,M,G,T,P} (1054720-500118158, default 500118158): +25G  
  
Created a new partition 3 of type 'Linux filesystem' and of size 25 GiB.  
Partition #3 contains a ext4 signature.  
  
Do you want to remove the signature? [Y]es/[N]o: n  
  
Command (m for help): n  
Partition number (4-128, default 4):  
First sector (53483520-500118158, default 53483520):  
Last sector, +/-sectors or +/-size{K,M,G,T,P} (53483520-500118158, default 500118158): +20G  
  
Created a new partition 4 of type 'Linux filesystem' and of size 20 GiB.
```

**Figure 4. 16**

*Digitally Erasing the Data Partition*

```
root@kali:~  
# dcfldd if=/dev/zero of=/dev/sdb4 conv=sync,noerror  
655360 blocks (20480Mb) written.  
655361+0 records in  
655360+0 records out
```

Figure 4. 17

### Formatting the Data Partition to the Ext4 File System

```
(root@kali)~# mkfs -t ext4 /dev/sdb4
mke2fs 1.46.1 (9-Feb-2021)
Creating filesystem with 5242880 4k blocks and 1310720 inodes
Filesystem UUID: 80fd56c3-ad56-4390-9494-9d0619848f74
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

(root@kali)~# lsblk -f
NAME        FSTYPE FSVER LABEL UUID                                 FSAVAIL FSUSE% MOUNTPOINT
sda
├─sda1      ext4    1.0             d1fa2eb5-318c-4a1f-879a-f230abf45cd3    59.9G   17% /
├─sda2
├─sda5      swap    1              76a2be20-235e-4abe-b906-35256de7e1e0                [SWAP]
sdb
├─sdb1
├─sdb2      vfat    FAT32           3758-AC0A
├─sdb3      ext4    1.0             2ac6aebc-2d74-4608-a1c4-22a89f13b496
├─sdb4      ext4    1.0             80fd56c3-ad56-4390-9494-9d0619848f74
sr0
```

Figure 4. 18

### List of Resulting Partitions

```
Disk /dev/sdb: 238.47 GiB, 256060514304 bytes, 500118192 sectors
Disk model: External Device
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: B99EC4C2-E77E-4AF2-8EF7-14826809998B

Device          Start      End    Sectors  Size Type
/dev/sdb1       2048      4095     2048    1M BIOS boot
/dev/sdb2       4096   1054719  1050624  513M EFI System
/dev/sdb3     1054720  53483519  52428800  25G Linux filesystem
/dev/sdb4     53483520  95426559  41943040  20G Linux filesystem
```

## 4.3.1 25% Drive Usage

Two sets of the payload files were copied to the data partition, by connecting the target drives internally to the target computer and then deleted one set of the files to maintain the drive usage level at 25%. The OS

TRIM command was triggered by executing the “fstrim” command as shown in Figure 4.19 and then the target computer was shut down. This was followed by the forensic imaging of the SSDs at the predetermined time intervals by connecting to the forensic workstation through the USB hub. Table 4.10 shows the summary of the files extracted from the forensic images of the drives with the help of the forensic data recovery tools and Figure 4.20 provides the comparative visualization of the number of files obtained from the SSDs at the different timelines.

**Figure 4. 19**

*Execution of TRIM Using the fstrim Command*

```

vinay@vinay-HP-Pavilion-g7-Notebook-PC: ~
vinay@vinay-HP-Pavilion-g7-Notebook-PC:~$ sudo mount /dev/sda4 /media/data
[sudo] password for vinay:
vinay@vinay-HP-Pavilion-g7-Notebook-PC:~$ sudo fstrim -v /media/data
/media/data: 19.5 GiB (20957421568 bytes) trimmed
vinay@vinay-HP-Pavilion-g7-Notebook-PC:~$

```

**Table 4. 10**

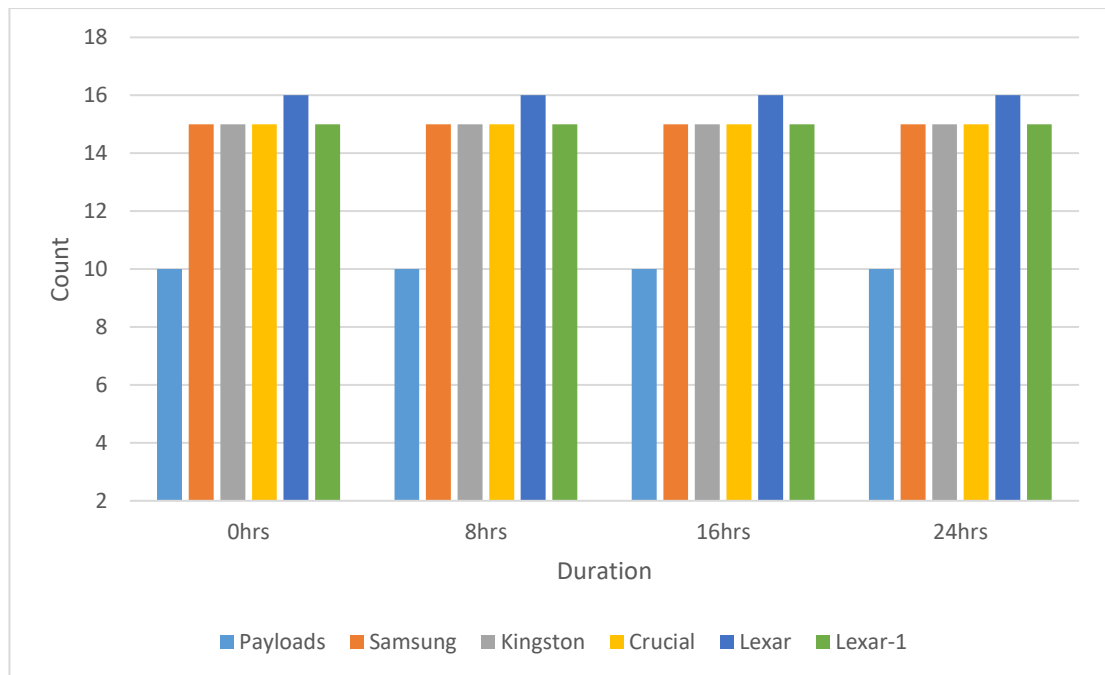
*Summary of Remaining Files: Ubuntu 21.10 With EXT4 - 25% Drive Usage*

Duration (hrs)	Payload Files	Samsung	Kingston	Crucial	Lexar	Lexar-1
0	10	15	15	15	16	15
8	10	15	15	15	16	15
16	10	15	15	15	16	15
24	10	15	15	15	16	15



**Figure 4. 20**

*Changes in Remaining Files: Ubuntu 21.10 With EXT4 - 25% Drive Usage*



Using the information from Table 4.10, the number of files recovered from each of the five SSDs has been compiled as per Table 4.11. Figure 4.21 represents this information.

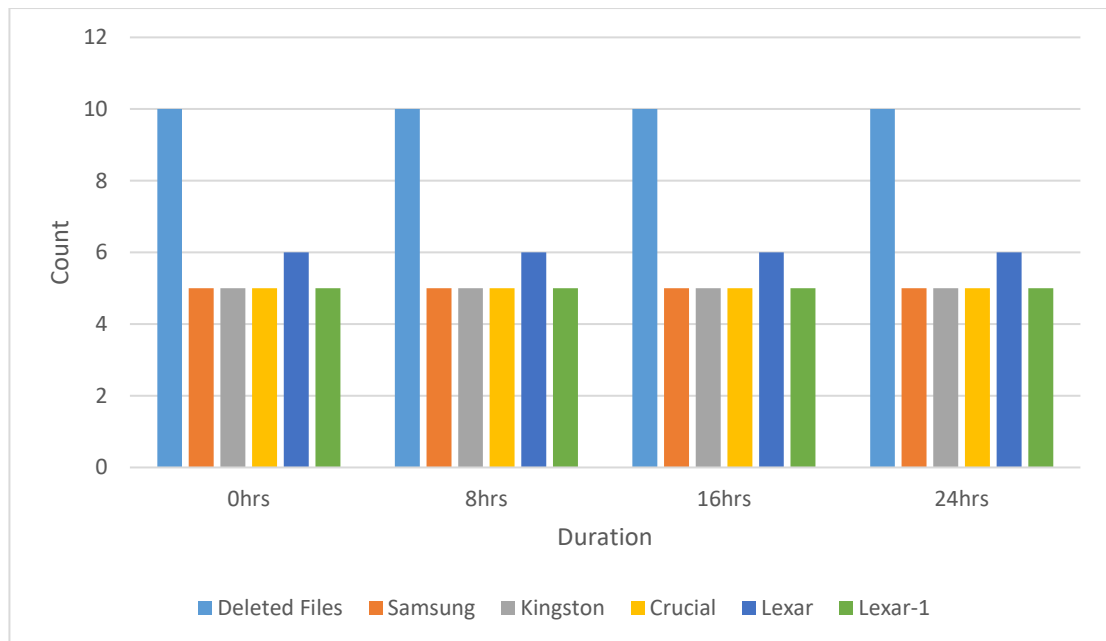
**Table 4. 11**

*Summary of Recovered Files: Ubuntu 21.10 With EXT4 - 25% Drive Usage*

Duration (hrs)	Deleted Files	Samsung	Kingston	Crucial	Lexar	Lexar-1
0	10	5	5	5	6	5
8	10	5	5	5	6	5
16	10	5	5	5	6	5
24	10	5	5	5	6	5

**Figure 4. 21**

*Comparative View of the Recovered Files: Ubuntu 21.10 With EXT4 - 25% Drive Usage*

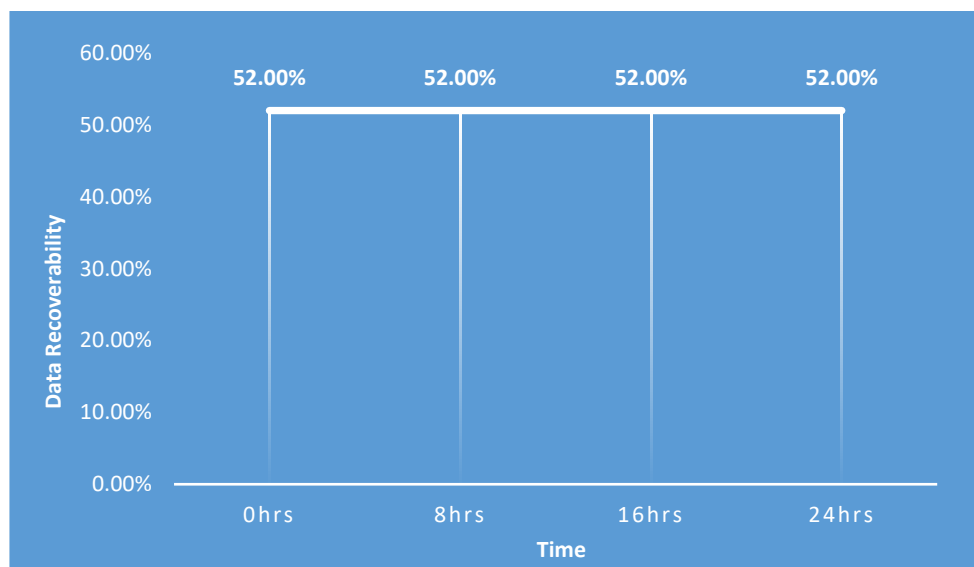


The details of the recovered files, corresponding to the image of the SSDs taken at the 0-hour timeline, can be found in Appendix A. Analysis of the information showed that only one file, 3.jpg, had the same hash for both the original and recovered versions. The hash value of all the recovered files except one file, 4.doc, remained consistent throughout the different forensic imaging timelines, for each of the drives. With only one exception, 01.zip, all the recovered files matched their corresponding original file in size.

Table 4.12 shows the data recoverability among the SSDs at the various timelines. It also shows the average data recoverability along the 24-hour time. Figure 4.22 represents the average data recoverability at the 25% disk usage level at the different timelines.

**Table 4. 12***Data Recoverability: Ubuntu 21.10 With EXT4 – 25% Drive Usage*

Time	0hrs	8hrs	16hrs	24hrs
Samsung	50.00%	50.00%	50.00%	50.00%
Kingston	50.00%	50.00%	50.00%	50.00%
Crucial	50.00%	50.00%	50.00%	50.00%
Lexar	60.00%	60.00%	60.00%	60.00%
Lexar-1	50.00%	50.00%	50.00%	50.00%
Average	52.00%	52.00%	52.00%	52.00%

**Figure 4. 22***Average Data Recoverability: Ubuntu 21.10 With EXT4 – 25% Drive Usage*

#### 4.3.2 50% Drive Usage

To conduct the test for 50% drive usage, the data partition of each of the drives was erased digitally and reformatted it to the ext4 file system. Following the same process that was engaged to conduct the experiment for the 25% drive usage, 3 sets of the payload files were copied to the data partition of the target drives followed by the deletion of one set of the payload files and then triggered the OS TRIM command. The SSDs were connected to the forensic workstation through the USB hub and subjected

to forensic imaging at the key timelines in the same way as in case of the 25% drive usage experiment.

Table 4.13 shows the number of files obtained from each of the SSDs from the forensic images of the SSDs which were taken in 8-hour intervals for the 24 hours, following the deletion of the files and execution of the OS TRIM command, using the digital forensic data recovery tools. Figure 4.23 shows the changes in the remaining files across the test drives at the various timelines.

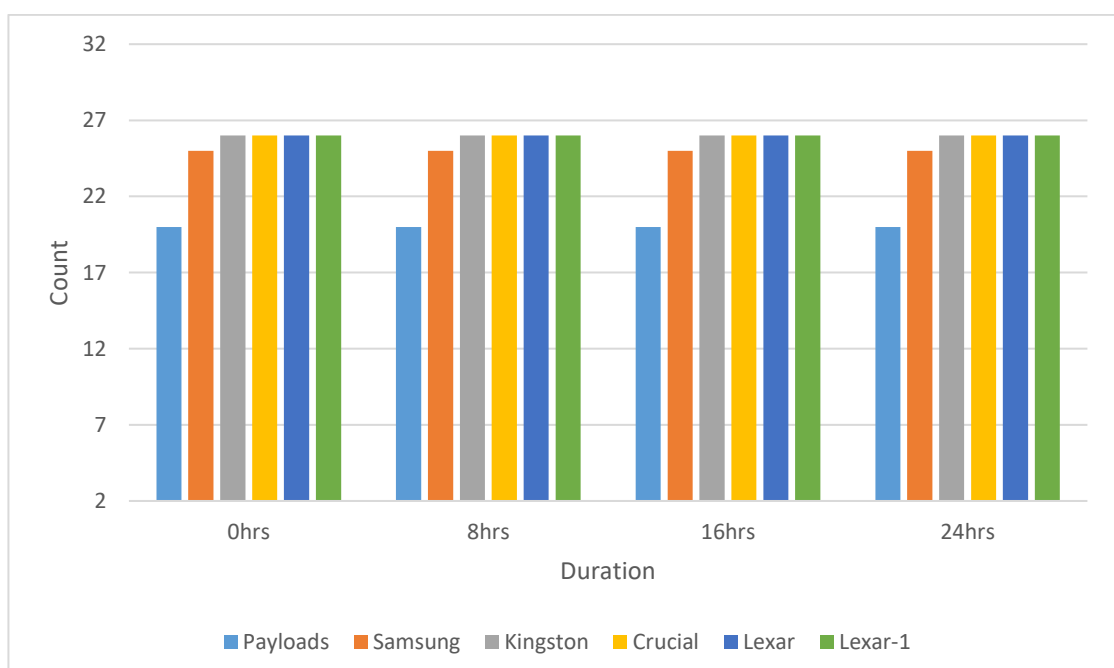
**Table 4. 13**

*Summary of Remaining Files: Ubuntu 21.10 With EXT4 – 50% Drive Usage*

Duration (hrs)	Payload Files	Samsung	Kingston	Crucial	Lexar	Lexar-1
0	20	25	26	26	26	26
8	20	25	26	26	26	26
16	20	25	26	26	26	26
24	20	25	26	26	26	26

**Figure 4. 23**

*Changes in Remaining Files: Ubuntu 21.10 With EXT4 – 50% Drive Usage*



Processing of the information listed in Table 4.13 provides insight to the number of recovered files, which were originally deleted, from the forensic images of the drives. Table 4.14 compiles the information on the number of files recovered from each of the SSDs and Figure 4.24 shows a comparative view of the data.

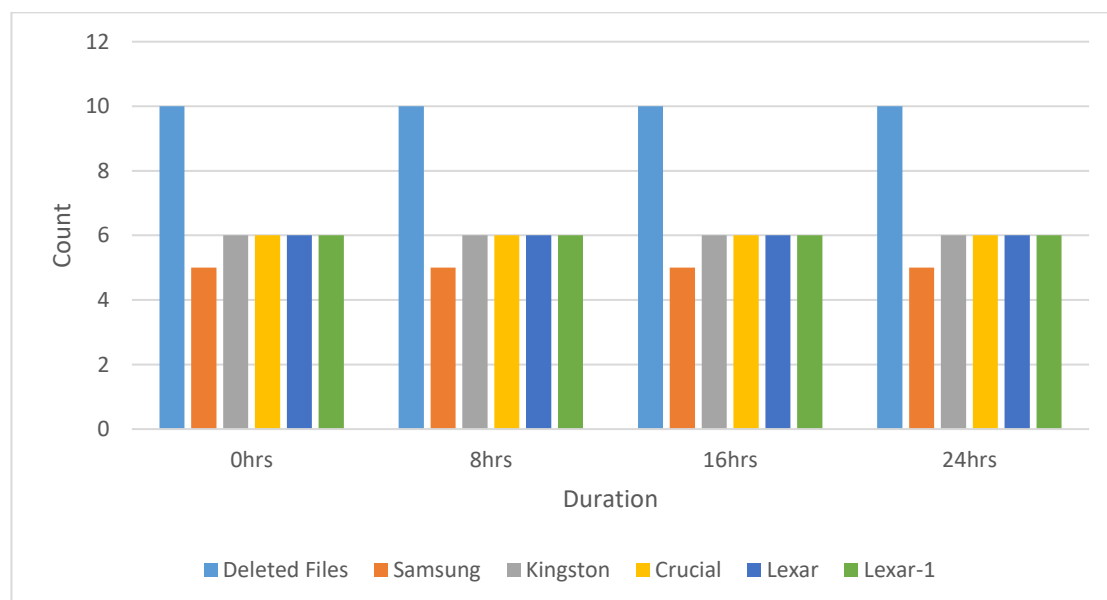
**Table 4. 14**

*Summary of Recovered Files: Ubuntu 21.10 With EXT4 – 50% Drive Usage*

Duration (hrs)	Deleted Files	Samsung	Kingston	Crucial	Lexar	Lexar-1
0	10	5	6	6	6	6
8	10	5	6	6	6	6
16	10	5	6	6	6	6
24	10	5	6	6	6	6

**Figure 4. 24**

*Comparative View of the Recovered Files: Ubuntu 21.10 With EXT4 – 50% Drive Usage*



By auditing the metadata of the recovered files, extracted from the 0Hrs image of the SSDs, which can be found in Appendix A, it has been observed that all of the recovered files, except the file 3.jpg, generated a different hash value from that of the corresponding original file even though the size

of each of the recovered files, except the file 01.zip, matched with the size of its original file. 4.doc was the only file that exhibited a difference in the hash value with the different timelines, for each of the drives.

Table 4.15 displays the data recoverability across the SSDs along the various timelines. It also shows the average data recoverability among the SSDs at the timelines. Figure 4.25 visualize the average data recoverability.

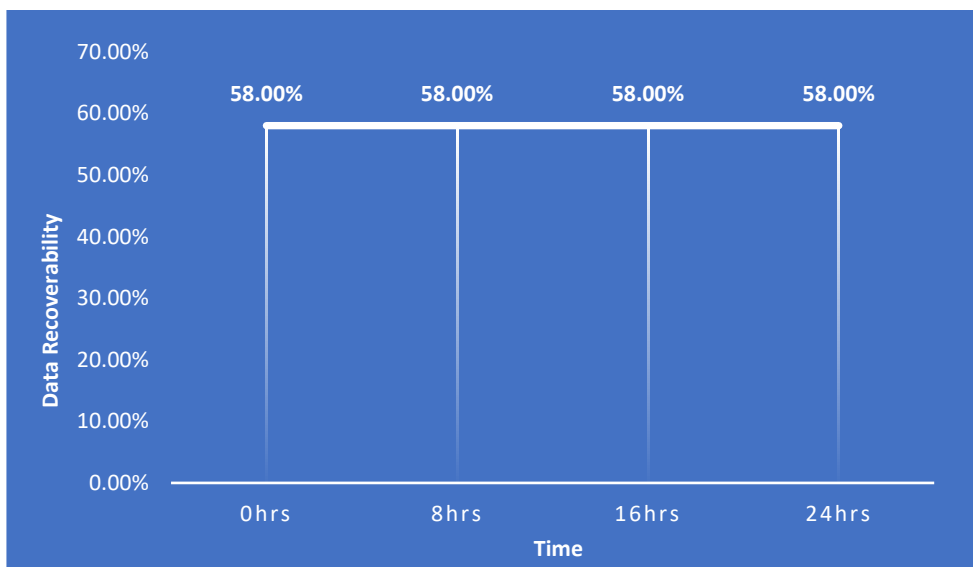
**Table 4. 15**

*Data Recoverability: Ubuntu 21.10 With EXT4 - 50% Drive Usage*

Time	0hrs	8hrs	16hrs	24hrs
Samsung	50.00%	50.00%	50.00%	50.00%
Kingston	60.00%	60.00%	60.00%	60.00%
Crucial	60.00%	60.00%	60.00%	60.00%
Lexar	60.00%	60.00%	60.00%	60.00%
Lexar-1	60.00%	60.00%	60.00%	60.00%
Average	58.00%	58.00%	58.00%	58.00%

**Figure 4. 25**

*Average Data Recoverability: Ubuntu 21.10 With EXT4 - 50% Drive Usage*



### 4.3.3 75% Drive Usage

The SSDs were prepared for the 75% data usage test by engaging the same process as in the case of the 25% and 50% data usage scenarios. After preparing the drives for the experiment, 4 sets of the payload files were loaded on to the data partition of the SSDs and then removed one set from it bringing the data usage level to 75%. Following this, the OS TRIM command was initiated and then shut down the target computer. The drives were then attached to the forensic workstation via the USB hub and subjected to forensic imaging for the next 24 hours at 8-hour interval. Table 4.16 shows the files extracted from those forensic images across the five drives. Figure 4.26 represents this information.

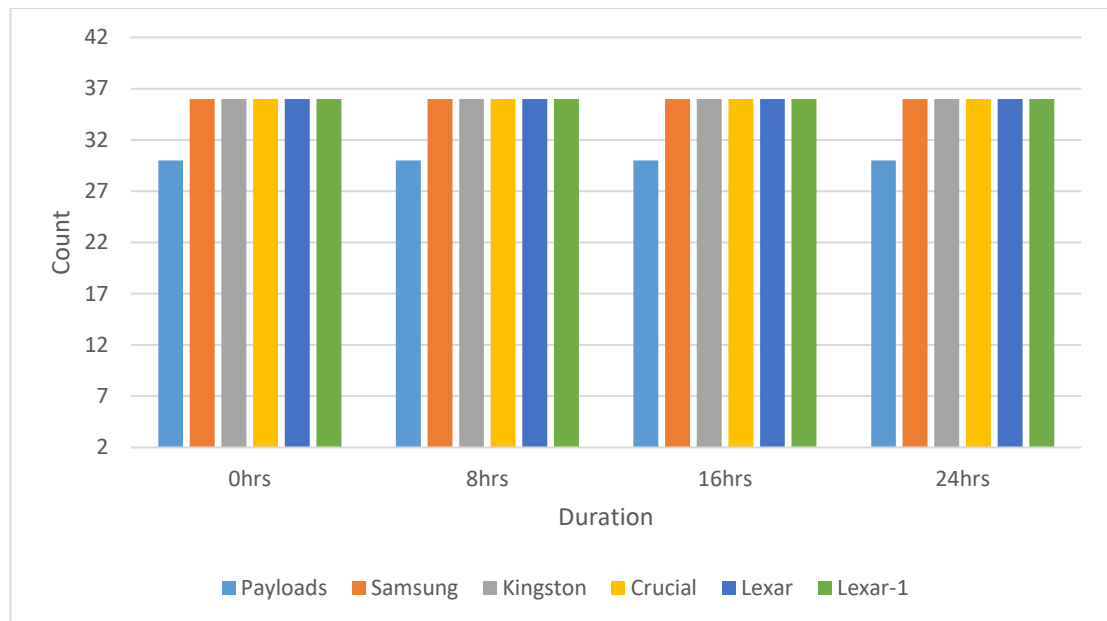
**Table 4. 16**

*Summary of Remaining Files: Ubuntu 21.10 With Ext4- 75% Drive Usage*

Duration (hrs)	Payload Files	Samsung	Kingston	Crucial	Lexar	Lexar-1
0	30	36	36	36	36	36
8	30	36	36	36	36	36
16	30	36	36	36	36	36
24	30	36	36	36	36	36

**Figure 4. 26**

*Changes in Remaining Files: Ubuntu 21.10 With Ext4- 75% Drive Usage*



The summary of the forensically recovered files from each of the five drives is shown in Table 4.17. Figure 4.27 shows a drive-wise comparison of the number of files recovered from the SSDs.

**Table 4. 17**

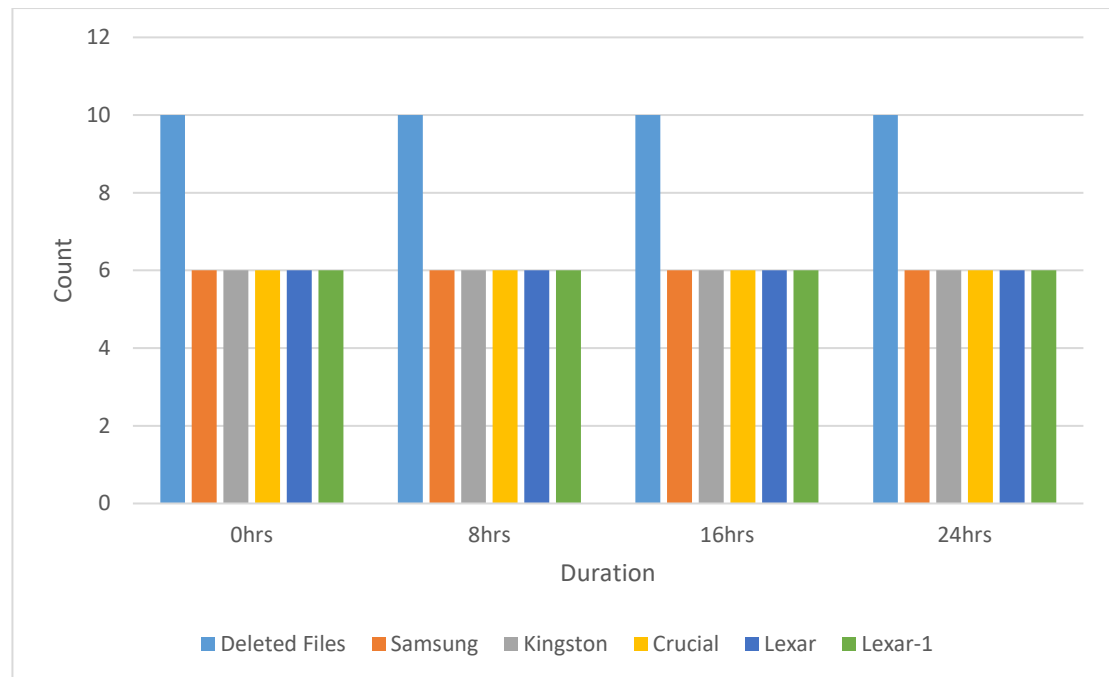
*Summary of Recovered Files: Ubuntu 21.10 With EXT4- 75% Drive Usage*

Duration (hrs)	Deleted Files	Samsung	Kingston	Crucial	Lexar	Lexar-1
0	10	6	6	6	6	6
8	10	6	6	6	6	6
16	10	6	6	6	6	6
24	10	6	6	6	6	6



**Figure 4. 27**

*Comparative View of the Recovered Files: Ubuntu 21.10 With Ext4- 75% Drive Usage*

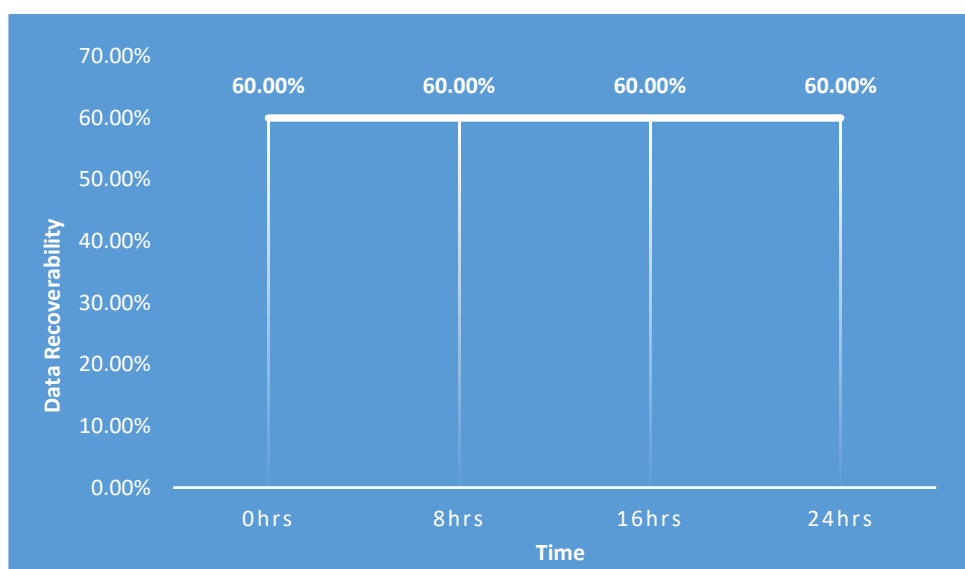


Further analysis of the recovered files revealed that the properties of the recovered files, in the 75% test scenario, were consistent with the properties of the files recovered in the other test cases. The only recovered file that had the same hash value of its original file was 3.jpg and the one file, among the recovered files, which had a different size from its original file was 01.zip. Except the file 4.doc, all the other recovered files maintained the same hash value along the different forensic imaging timelines, for each of the drives.

Table 4.18 displays the data recoverability among the SSDs and the average data recoverability across the different timelines. Figure 4.28 displays the average data recoverability across the key timelines.

**Table 4. 18***Data Recoverability: Ubuntu 21.10 With EXT4 - 75% Drive Usage*

Time	0hrs	8hrs	16hrs	24hrs
Samsung	60.00%	60.00%	60.00%	60.00%
Kingston	60.00%	60.00%	60.00%	60.00%
Crucial	60.00%	60.00%	60.00%	60.00%
Lexar	60.00%	60.00%	60.00%	60.00%
Lexar-1	60.00%	60.00%	60.00%	60.00%
Average	60.00%	60.00%	60.00%	60.00%

**Figure 4. 28***Average Data Recoverability: Ubuntu 21.10 With EXT4 - 75% Drive Usage*

#### 4.4 Apple macOS Catalina with APFS

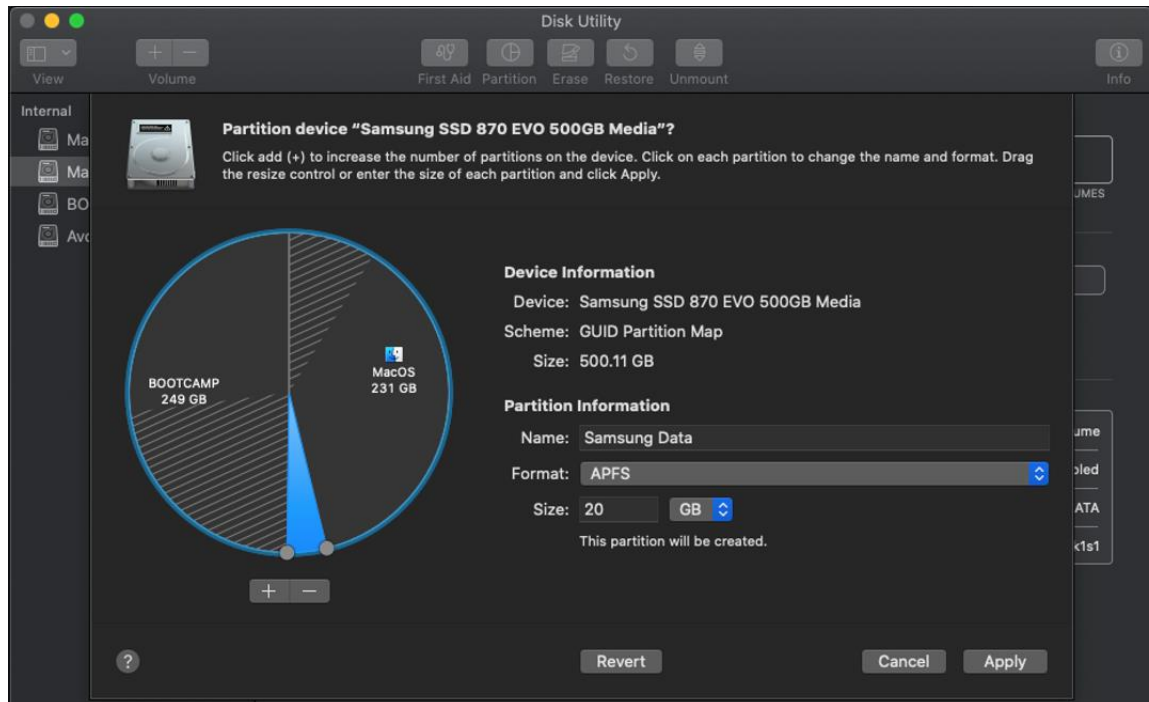
The Ubuntu 21.10 operating system that was installed in the SSDs was replaced by installing Apple macOS Catalina. The data partition to copy the payload files was created using the Disk Utility that comes with the macOS, with 20GB size, formatting to APFS file system. Figure 4.29 shows the creation of the data partition using the Disk Utility. The TRIM functionality was enabled by executing the following command.

`'sudo trimforce –enable'`

Figure 4.30 shows the process of enabling TRIM in macOS using the “trimforce” command.

**Figure 4. 29**

*Creating the Data Partition in macOS Using the Disk Utility*



**Figure 4. 30**

*Enabling TRIM in macOS*

```
vinayvarghese@Vinays-MBP ~ % sudo trimforce enable
Password:
IMPORTANT NOTICE: This tool force-enables TRIM for all relevant attached
devices, even though such devices may not have been validated for data
integrity while using TRIM. Use of this tool to enable TRIM may result in
unintended data loss or data corruption. It should not be used in a commercial
operating environment or with important data. Before using this tool, you
should back up all of your data and regularly back up data while TRIM is
enabled. This tool is provided on an "as is" basis. APPLE MAKES NO WARRANTIES,
EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION THE IMPLIED WARRANTIES OF
NON-INFRINGEMENT, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE,
REGARDING THIS TOOL OR ITS USE ALONE OR IN COMBINATION WITH YOUR DEVICES,
SYSTEMS, OR SERVICES. BY USING THIS TOOL TO ENABLE TRIM, YOU AGREE THAT, TO THE
EXTENT PERMITTED BY APPLICABLE LAW, USE OF THE TOOL IS AT YOUR SOLE RISK AND
THAT THE ENTIRE RISK AS TO SATISFACTORY QUALITY, PERFORMANCE, ACCURACY AND
EFFORT IS WITH YOU.
Are you sure you wish to proceed (y/N)? y
Your system will immediately reboot when this is complete.
Is this OK (y/N)?
```

### 4.4.1 25% Drive Usage

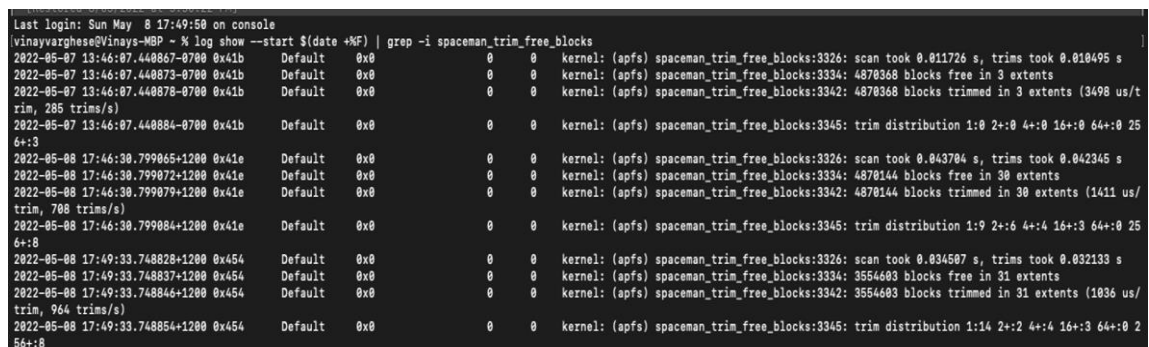
The experiment to collect data at the 25% drive usage scenario was conducted by loading two sets of the payload files to the data partition of each of the SSDs, by connecting internally to the MacBook Pro, which was then followed by the deletion of one set among those. The target computer, the MacBook Pro, was restarted after the deletion of the files, as Apple macOS initiates TRIM command when it detects the file system in a connected TRIM supported device or when the computer is restarted. There is no manual TRIM option available in macOS (Plugable Technologies, 2021). The initiation of the TRIM command was verified after restarting the MacBook Pro using the following command, which lists the TRIM performed on the current date.

```
`log show --start $(date +%F) | grep -i spaceman_trim_free_blocks`
```

Figure 4.31 shows the verification of the execution of TRIM by executing the command, on one of the drives.

**Figure 4. 31**

*Verification of the Execution of TRIM*



```
Last login: Sun May 8 17:49:50 on console
vinayvarghese@Vinays-MBP: ~ % log show --start $(date +%F) | grep -i spaceman_trim_free_blocks
2022-05-07 13:46:07.448867-0700 0x41b Default 0x0 0 0 kernel: (apfs) spaceman_trim_free_blocks:3326: scan took 0.011726 s, trims took 0.010495 s
2022-05-07 13:46:07.448873-0700 0x41b Default 0x0 0 0 kernel: (apfs) spaceman_trim_free_blocks:3334: 4870368 blocks free in 3 extents
2022-05-07 13:46:07.448878-0700 0x41b Default 0x0 0 0 kernel: (apfs) spaceman_trim_free_blocks:3342: 4870368 blocks trimmed in 3 extents (3498 us/t
rim, 285 trims/s)
2022-05-07 13:46:07.448884-0700 0x41b Default 0x0 0 0 kernel: (apfs) spaceman_trim_free_blocks:3345: trim distribution 1:0 2+:0 4+:0 16+:0 64+:0 25
6+:3
2022-05-08 17:46:30.799065+1200 0x41e Default 0x0 0 0 kernel: (apfs) spaceman_trim_free_blocks:3326: scan took 0.043704 s, trims took 0.042345 s
2022-05-08 17:46:30.799072+1200 0x41e Default 0x0 0 0 kernel: (apfs) spaceman_trim_free_blocks:3334: 4870144 blocks free in 30 extents
2022-05-08 17:46:30.799079+1200 0x41e Default 0x0 0 0 kernel: (apfs) spaceman_trim_free_blocks:3342: 4870144 blocks trimmed in 30 extents (1411 us/
trim, 708 trims/s)
2022-05-08 17:46:30.799084+1200 0x41e Default 0x0 0 0 kernel: (apfs) spaceman_trim_free_blocks:3345: trim distribution 1:9 2+:6 4+:4 16+:3 64+:0 25
6+:8
2022-05-08 17:49:33.748828+1200 0x454 Default 0x0 0 0 kernel: (apfs) spaceman_trim_free_blocks:3326: scan took 0.034507 s, trims took 0.032133 s
2022-05-08 17:49:33.748837+1200 0x454 Default 0x0 0 0 kernel: (apfs) spaceman_trim_free_blocks:3334: 3554603 blocks free in 31 extents
2022-05-08 17:49:33.748846+1200 0x454 Default 0x0 0 0 kernel: (apfs) spaceman_trim_free_blocks:3342: 3554603 blocks trimmed in 31 extents (1036 us/
trim, 964 trims/s)
2022-05-08 17:49:33.748854+1200 0x454 Default 0x0 0 0 kernel: (apfs) spaceman_trim_free_blocks:3345: trim distribution 1:14 2+:2 4+:4 16+:3 64+:0 2
56+:8
```

The target drives were then connected to the forensic workstation through the USB hub in order to carry out the forensic imaging process at 8-hour intervals for the following 24 hours. The resulting forensic images were

subjected to forensic data recovery process. Table 4.19 shows the summary of the remaining files obtained from the images of each of the drives at the various predetermined time marks. A visualization of the data is shown in Figure 4.32

**Table 4. 19**

*Summary of Remaining Files: Apple macOS Catalina With APFS - 25% Drive Usage*

Duration (hrs)	Payload Files	Samsung	Kingston	Crucial	Lexar	Lexar-1
0	10	15	16	16	15	15
8	10	15	16	16	15	15
16	10	15	16	16	15	15
24	10	15	16	16	15	15

**Figure 4. 32**

*Changes in Remaining Files: Apple macOS Catalina With APFS - 25% Drive Usage*

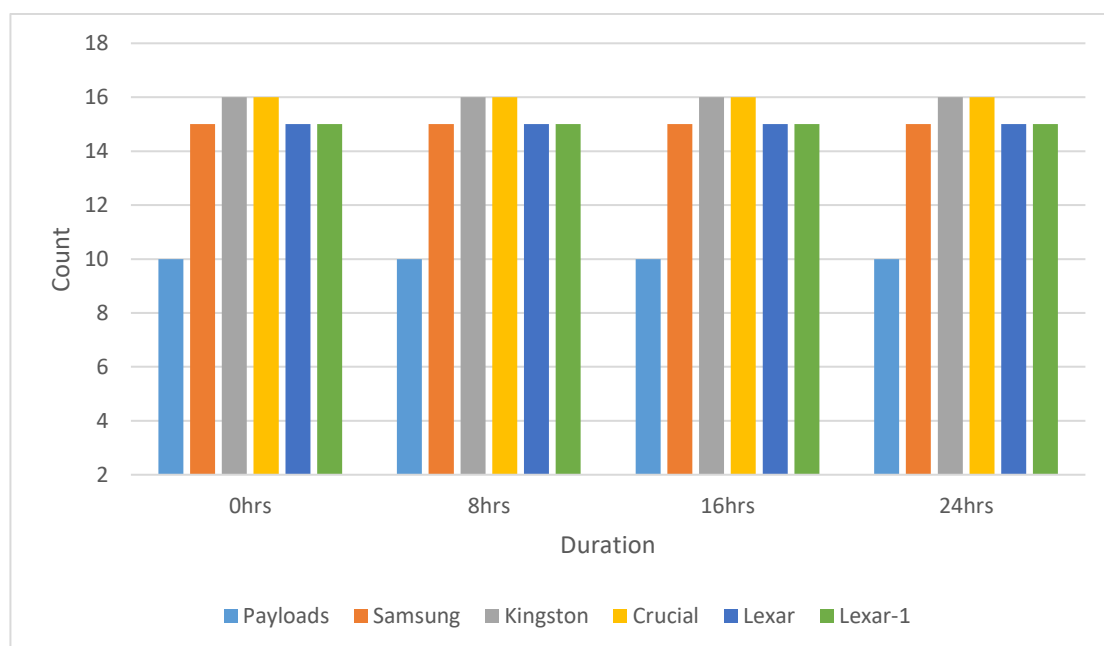


Table 4.20 lists the number of recovered files, from the subject SSDs that underwent the experiment, which has been derived from Table 4.19. A visual comparison of the number of recovered files among the drives is shown in Figure 4.33.

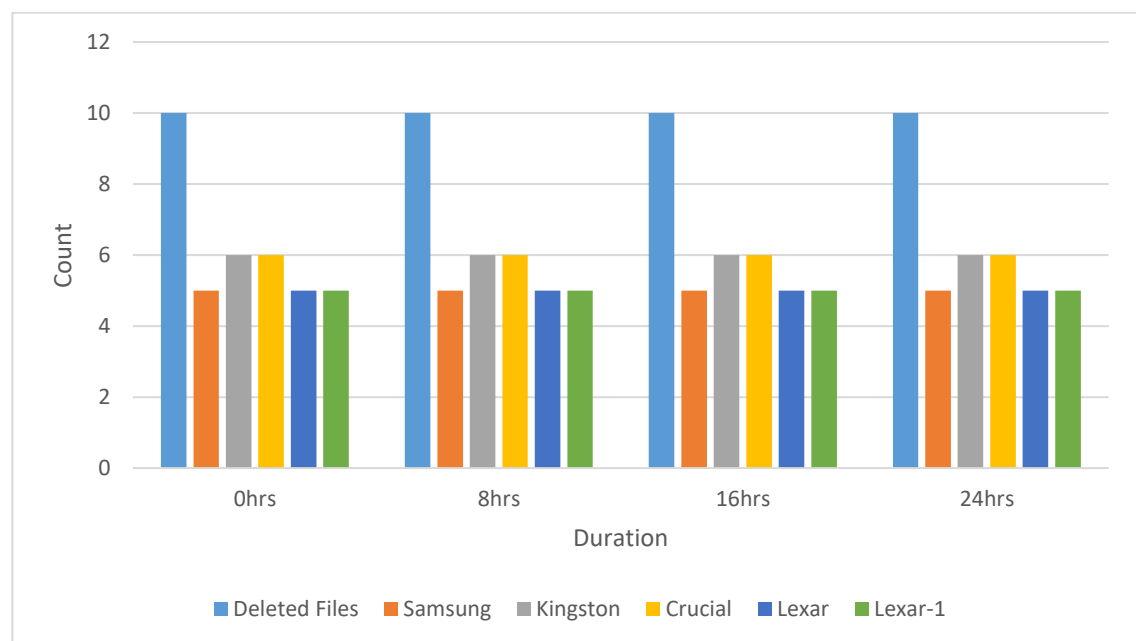
**Table 4. 20**

*Summary of Recovered Files: Apple macOS Catalina With APFS - 25% Drive Usage*

Duration (hrs)	Deleted Files	Samsung	Kingston	Crucial	Lexar	Lexar-1
0	10	5	6	6	5	5
8	10	5	6	6	5	5
16	10	5	6	6	5	5
24	10	5	6	6	5	5

**Figure 4. 33**

*Comparative View of the Recovered Files: Apple macOS Catalina With APFS - 25% Drive Usage*



The details of the recovered files can be found in Appendix A. Auditing the metadata of the recovered files showed that only one file had a different size compared to the size of its original file while all the files, but one, had a different hash value compared to the hash value of the corresponding original file. Except one file, all the other files that were recovered had consistency with the hash value across the different timelines.

Table 4.21 shows the rate of data recoverability among the SSDs at the different timelines, including the average data recoverability and Figure

4.34 displays the average data recoverability at 25% drive usage along the various timelines.

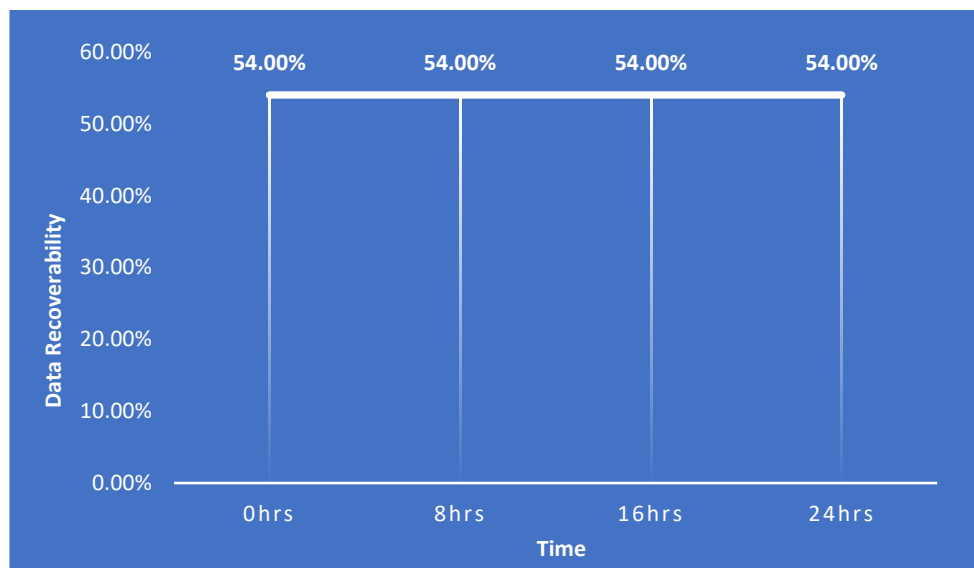
**Table 4. 21**

*Data Recoverability: Apple macOS Catalina With APFS – 25% Drive Usage*

Time	0hrs	8hrs	16hrs	24hrs
Samsung	50.00%	50.00%	50.00%	50.00%
Kingston	60.00%	60.00%	60.00%	60.00%
Crucial	60.00%	60.00%	60.00%	60.00%
Lexar	50.00%	50.00%	50.00%	50.00%
Lexar-1	50.00%	50.00%	50.00%	50.00%
Average	54.00%	54.00%	54.00%	54.00%

**Figure 4. 34**

*Average Data Recoverability: Apple macOS Catalina With APFS – 25% Drive Usage*



#### 4.4.2 50% Drive Usage

The SSDs were prepared for the 50% drive usage experiment by conducting digital erasure of the data partition and then reformatting it to the APFS file system. In order to depict 50% drive usage, three sets of the payload files were copied to the data partition of the five SSDs that were undergoing the test and deleted one of the sets of the payload files. The target computer

was then restarted to trigger the OS TRIM function followed by shutdown. The drives were reattached to the forensic workstation via the USB hub, after detaching from the target computer, for performing the forensic imaging at the pre-set time intervals. The summary of the number of extracted files at the key timelines across the SSDs is shown in Table 4.22. Figure 4.35 illustrates the changes in the remaining files in the drives at the different timelines.

**Table 4. 22**

*Summary of Remaining Files: Apple macOS Catalina With APFS – 50% Drive Usage*

Duration (hrs)	Payload Files	Samsung	Kingston	Crucial	Lexar	Lexar-1
0	20	25	25	26	25	26
8	20	25	25	26	25	26
16	20	25	25	26	25	26
24	20	25	25	26	25	26

**Figure 4. 35**

*Changes in Remaining Files: Apple macOS Catalina With APFS – 50% Drive Usage*

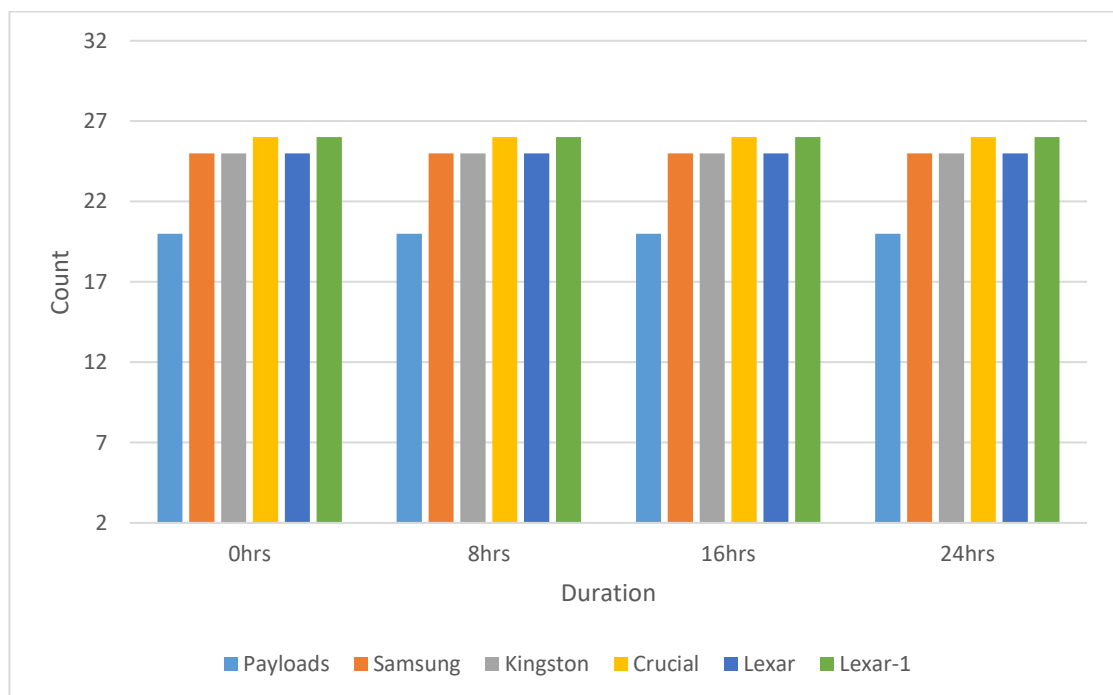




Table 4.23 contains information on the number of recovered files from the drives at the various timelines. Figure 4.36 provides a comparative view of the number of files recovered across the five SSDs.

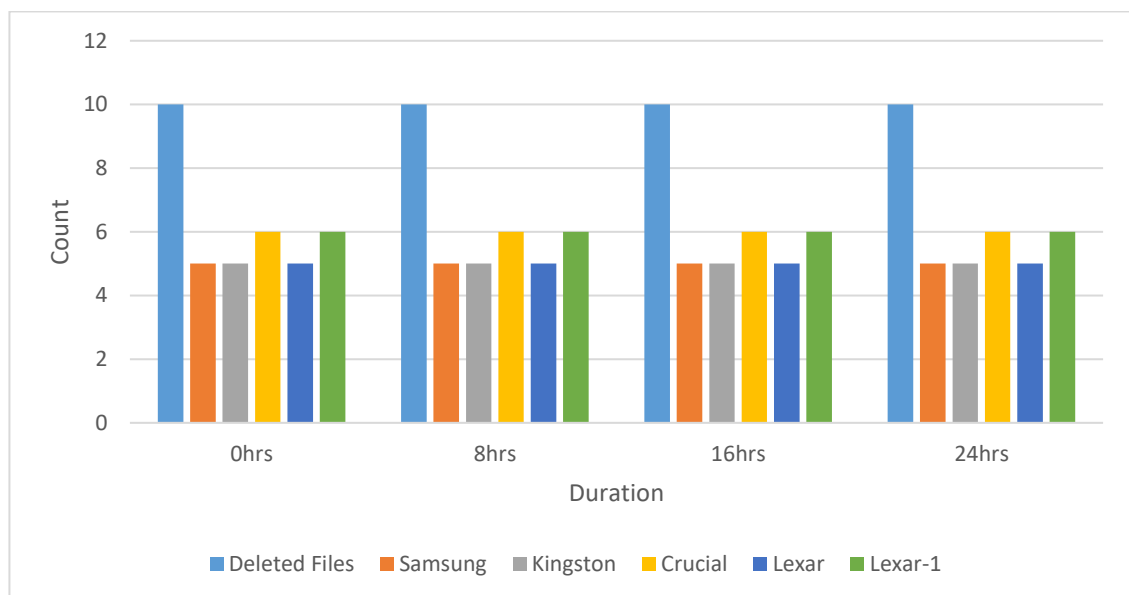
**Table 4. 23**

*Summary of Recovered Files: Apple macOS Catalina With APFS – 50% Drive Usage*

Duration (hrs)	Deleted Files	Samsung	Kingston	Crucial	Lexar	Lexar-1
0	10	5	5	6	5	6
8	10	5	5	6	5	6
16	10	5	5	6	5	6
24	10	5	5	6	5	6

**Figure 4. 36**

*Comparative View of the Recovered Files: Apple macOS Catalina With APFS – 50% Drive Usage*



Analysis of the details of the recovered files, which is included in Appendix A, showed that the size of the recovered files matched with the size of the corresponding original file with a single exception. While the hash value of one of the recovered files matched with the hash value of its original file,

all the other files exhibited a difference. The file 4.doc displayed inconsistency in hash value at the different timelines.

The rate of data recoverability across the five SSDs along the different timelines, which was derived from Table 4.23, is shown in Table 4.24. It also displays the average data recoverability among the drives. Figure 4.37 shows the average data recoverability among the SSDs at the different timelines.

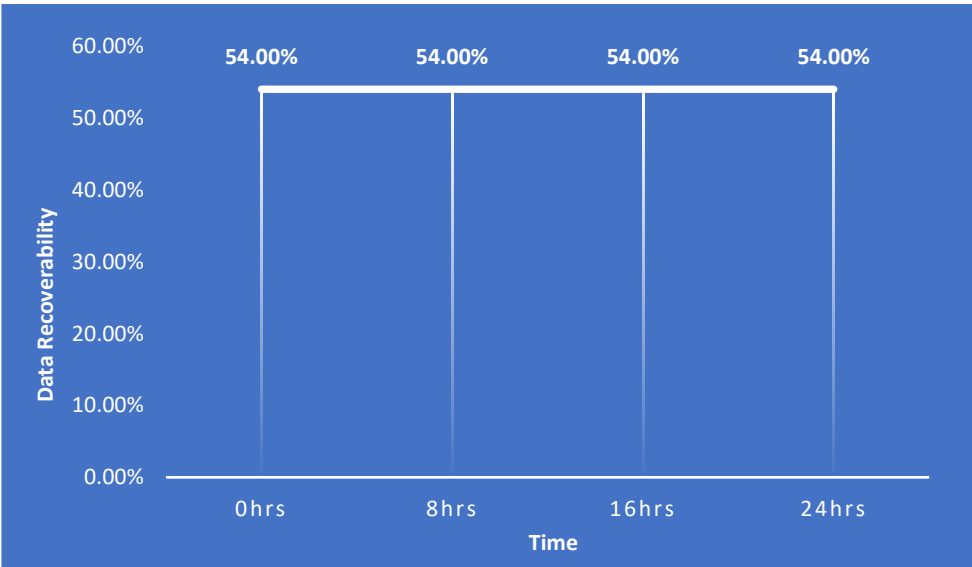
**Table 4. 24**

*Data Recoverability: Apple macOS Catalina With APFS – 50% Drive Usage*

Time	0hrs	8hrs	16hrs	24hrs
Samsung	50.00%	50.00%	50.00%	50.00%
Kingston	50.00%	50.00%	50.00%	50.00%
Crucial	60.00%	60.00%	60.00%	60.00%
Lexar	50.00%	50.00%	50.00%	50.00%
Lexar-1	60.00%	60.00%	60.00%	60.00%
Average	54.00%	54.00%	54.00%	54.00%

**Figure 4. 37**

*Average Data Recoverability: Apple macOS Catalina With APFS – 50% Drive Usage*



#### 4.4.3 75% Drive Usage

The process followed to conduct the test for the 75% drive usage level was similar to that of the 50% drive usage. Four sets of the payload files were copied to the data partition of the SSDs and then one set of files was deleted. This was followed by the initiation of TRIM and the target computer was shutdown. The forensic images of the SSDs were then taken at various time intervals by connecting the drives to the forensic workstation. Table 4.25 shows the remaining files obtained from the SSDs at the key timelines. Figure 4.38 provides a representation of the change in the remaining files across the SSDs along the different timelines.

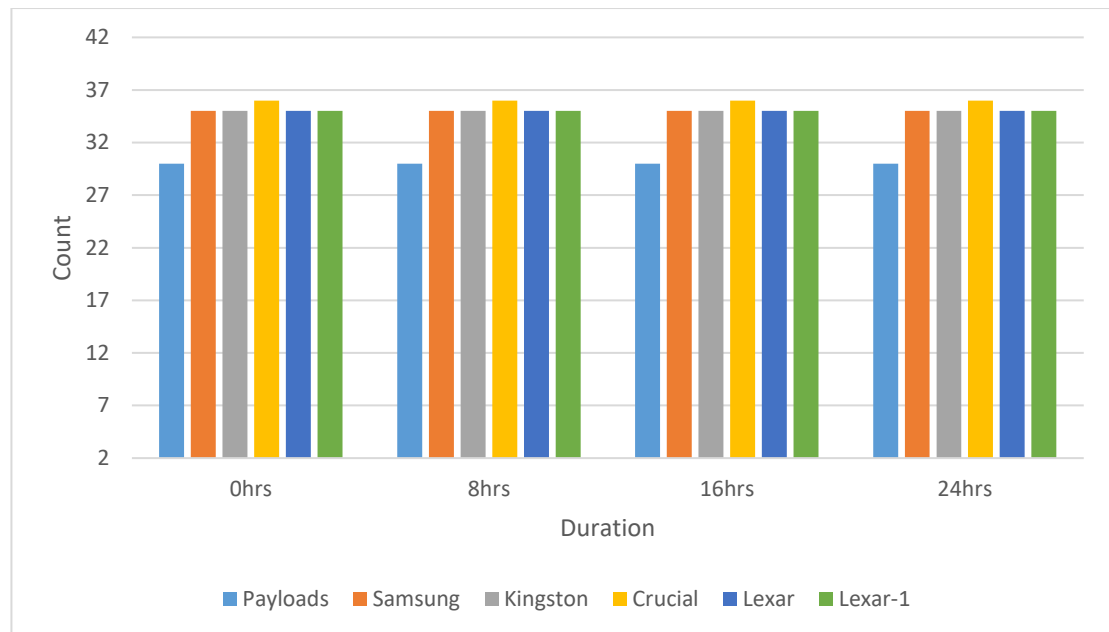
**Table 4. 25**

*Summary of Remaining Files: Apple macOS Catalina With APFS – 75% Drive Usage*

Duration (hrs)	Payload Files	Samsung	Kingston	Crucial	Lexar	Lexar-1
0	30	35	35	36	35	35
8	30	35	35	36	35	35
16	30	35	35	36	35	35
24	30	35	35	36	35	35

**Figure 4. 38**

*Changes in Remaining Files: Apple macOS Catalina With APFS – 75% Drive Usage*



The summary of the recovered files from the SSDs is shown in Table 4.26, which is derived from Table 4.25, and a comparative view of the same is shown in Figure 4.39.

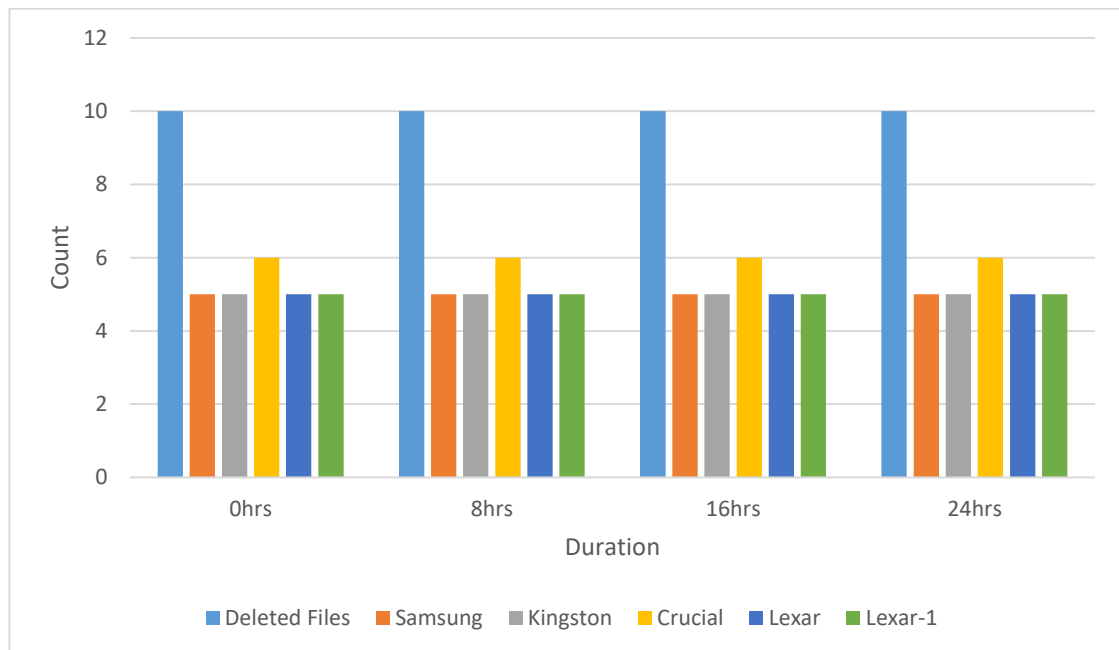
**Table 4. 26**

*Summary of Recovered Files: Apple macOS Catalina With APFS – 75% Drive Usage*

Duration (hrs)	Deleted Files	Samsung	Kingston	Crucial	Lexar	Lexar-1
0	10	5	5	6	5	5
8	10	5	5	6	5	5
16	10	5	5	6	5	5
24	10	5	5	6	5	5

**Figure 4. 39**

*Comparative View of the Recovered Files: Apple macOS Catalina With APFS – 75% Drive Usage*

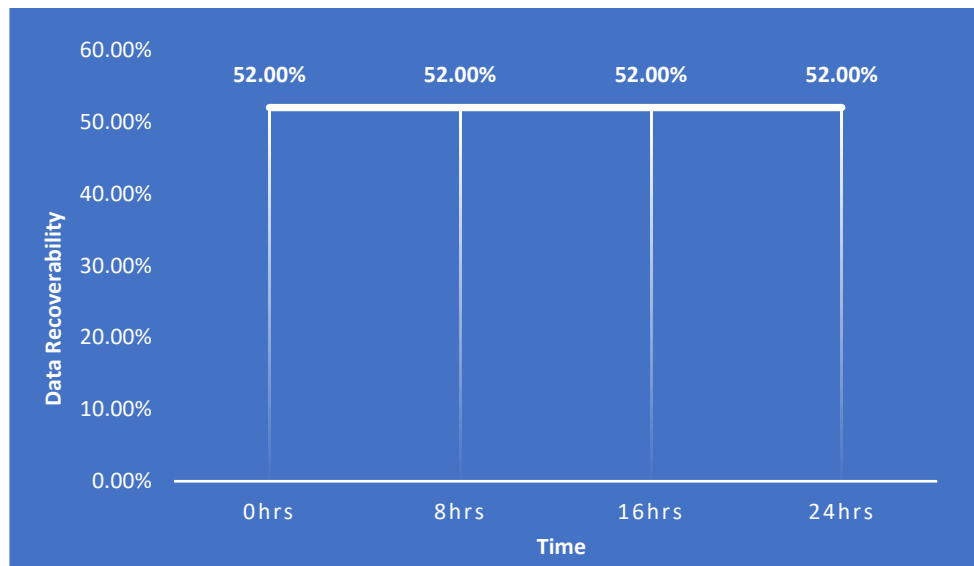


The findings of the analysis of the recovered file showed similarity with the results obtained from the experiments conducted for other test cases. One of the recovered files had a different size from its original file while the rest of the recovered files had the same size of the corresponding original file. The hash value of one of the recovered files matched the hash value of its original file. All the other files had a different hash value compared to the hash value of their original file. Except one file, all the other recovered files maintained the same hash value along the various timelines.

Table 4.27 shows the rate of data recoverability across the SSDs at the different timelines, including the average data recoverability. Figure 4.40 shows the average data recoverability along the key timelines at the 75% disk usage level.

**Table 4. 27***Data Recoverability: Apple macOS Catalina With APFS – 75% Drive Usage*

SSD	0hrs	8hrs	16hrs	24hrs
Samsung 870 EVO	50.00%	50.00%	50.00%	50.00%
Kingston A400	50.00%	50.00%	50.00%	50.00%
Crucial BX500	60.00%	60.00%	60.00%	60.00%
Lexar NS 100	50.00%	50.00%	50.00%	50.00%
Lexar NS 100 -1	50.00%	50.00%	50.00%	50.00%
<b>Average</b>	52.00%	52.00%	52.00%	52.00%

**Figure 4. 40***Average Data Recoverability: Apple macOS Catalina With APFS – 75% Drive Usage*

## 4.5 Conclusion

Chapter 4 has compiled the results obtained from the experiments conducted on five different SSDs using three different operating system and file system combinations, by following the test plan defined in Section 3.4. The results segregated for each SSD and operating system were presented using tables and graphs to visualize the data in different perspectives. The results reported and analysed, provide insights into the change in data recoverability at different drive usage levels and the effect of time on it. The

findings will be further discussed in Chapter 5 to answer the research questions formulated for this study.

## Chapter 5 – Discussion

### 5.1 Introduction

Chapter 3 posed the research questions and established a research design for this study, based on the similar studies that were reviewed in Chapter 2. Chapter 5 provides further analysis of the results presented in Chapter 4, that were obtained by conducting the experiments following the plan presented in Chapter 3. This Chapter also discusses their relationship to the findings from the literature review regarding the data recoverability of SSDs. Section 5.2 and Section 5.3 answer the research questions and the hypothesis, respectively, with detailed analysis of the findings and discuss if the findings support or contradict the findings from the literature review.

### 5.2 Research Questions

The following sections answer the two research questions that were formulated and presented in Chapter 3.

#### 5.2.1 Question 1

**Q1.** Can the effects of time be predicted on the amount of deleted data that can be recovered?

Answer:

The findings of this research show that it is possible to predict the effects of time on the amount of recoverable data.

Discussion:

The literature review presented in Chapter 2 revealed that the ATA TRIM command enables the file system to instruct the SSD's controller (Reddy, 2019, p. 382) about the pages whose data have become stale, as a result of the delete operations carried by a user or by the operating system, so that



garbage collection can avoid relocating those data but can erase those locations (Kim & Shin, 2011). The results obtained from the experiments underscore this as the SSDs exhibited significant data loss after the execution of the operating system's TRIM command.

Relevant previous research that were reviewed in Chapter 2 demonstrated the aggressiveness of TRIM, causing substantial data loss, soon after the issuance of the command by the operating system (Nisbet et al., 2013). It was also identified that, following an OS TRIM command, the garbage collection process gets initiated within a matter of few minutes (Morningstar, 2018). The results of this study support this, as significant data loss was exhibited at the initial point of time of the forensic data recovery, for all the drives tested.

The evidence from the findings shows that the amount of data that can be recovered, the rate of data recoverability that was calculated based on the formula mentioned in Section 3.3, exhibited a sharp decline at the beginning and maintained the same rate of recovery throughout the key timelines, for each of the drives and operating system/file system combinations. The behaviour remained the same at the different drive usage levels as well.

#### 5.2.1.1 Data Recoverability by Operating System/File System

The following sections discuss the average data recoverability among the SSDs, along the key timelines at different disk usage levels, with each of the operating system/file system combinations.

##### 5.2.1.1.1 Microsoft Windows 10 Home with NTFS

Table 5.1 shows the average data recoverability among the five SSDs, along the key timelines at the different disk usage levels, based on the results

obtained when tested using Microsoft Windows 10 Home with NTFS. Figure 5.1 shows a representation of this information.

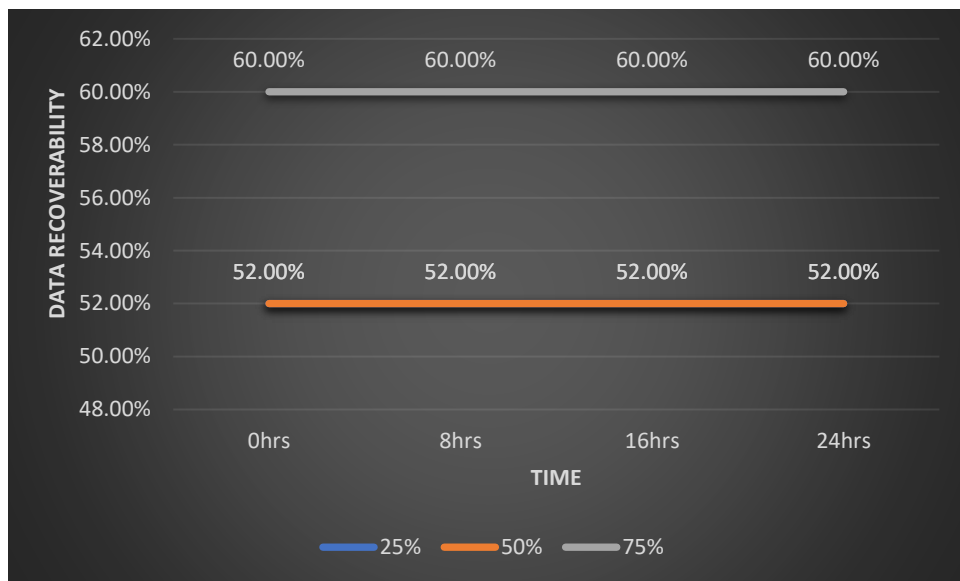
**Table 5. 1**

*Average Data Recoverability Among the Drives: Windows 10 Home With NTFS*

Disk Usage	Duration			
	0hrs	8hrs	16hrs	24hrs
25%	52.00%	52.00%	52.00%	52.00%
50%	52.00%	52.00%	52.00%	52.00%
75%	60.00%	60.00%	60.00%	60.00%

**Figure 5. 1**

*Representation of the Average Data Recoverability Among the Drives at the Key Timelines: Windows 10 Home With NTFS*



For the test cases with Windows 10 Home with NTFS, the data recoverability at 25%, 50% and 75% disk usage levels remained to be 52%, 52% and 60% respectively, throughout the 24-hour duration. The horizontal straight lines displayed in Figure 5.1 provide a clear picture of the static

nature of the data recoverability from 0 to 24-hour duration, at all the disk usage levels.

#### 5.2.1.1.2 Ubuntu 21.10 with EXT4

Table 5.2 shows the average rate of data recovery among the five SSDs, along the key timelines at the different disk usage levels, which was derived from the results obtained when tested using Ubuntu 21.10 with EXT4. Figure 5.2 shows a representation of this information.

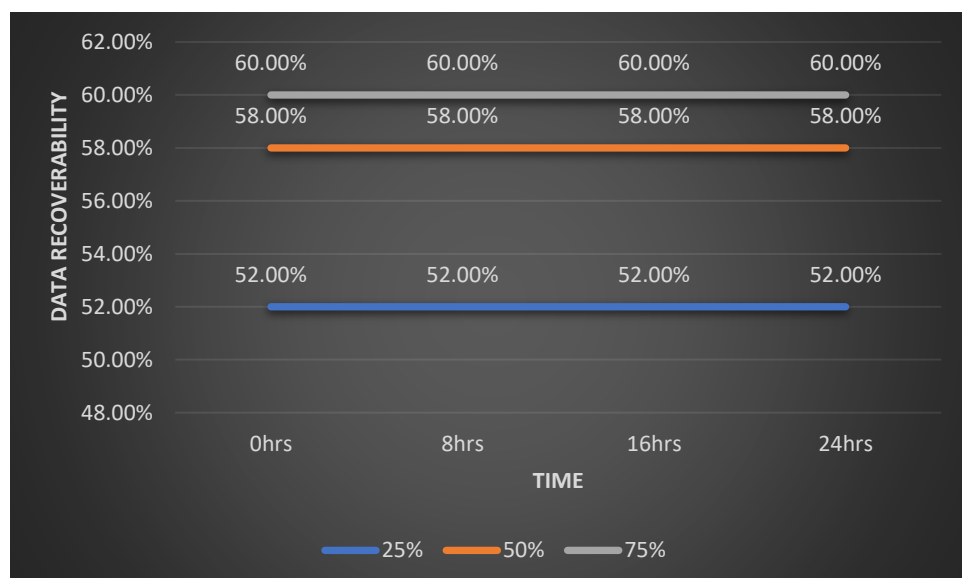
**Table 5. 2**

*Average Data Recoverability Among the Drives: Ubuntu 21.10 With Ext4*

Disk Usage	Duration			
	0hrs	8hrs	16hrs	24hrs
25%	52.00%	52.00%	52.00%	52.00%
50%	58.00%	58.00%	58.00%	58.00%
75%	60.00%	60.00%	60.00%	60.00%

**Figure 5. 2**

*Representation of the Average Data Recoverability Among the Drives at the Key Timelines: Ubuntu 21.10 With Ext4*



The data recoverability at 25%, 50% and 75% disk usage levels were 52%, 58% and 60% respectively, throughout the 24-hour duration, when experimented with Ubuntu 21.10 with EXT4. The data recoverability exhibited a static nature during the first 24 hours, at each of the predefined disk usage levels, which is clear from the horizontal straight lines displayed in Figure 5.2.

#### 5.2.1.1.3 Apple macOS Catalina with APFS

The average data recoverability for Apple macOS Catalina with APFS, calculated from the results of the experiments that were conducted on the five SSDs, along the key timelines at the different disk usage levels is shown in Table 5.3. Figure 5.3 provides a representation of this information.

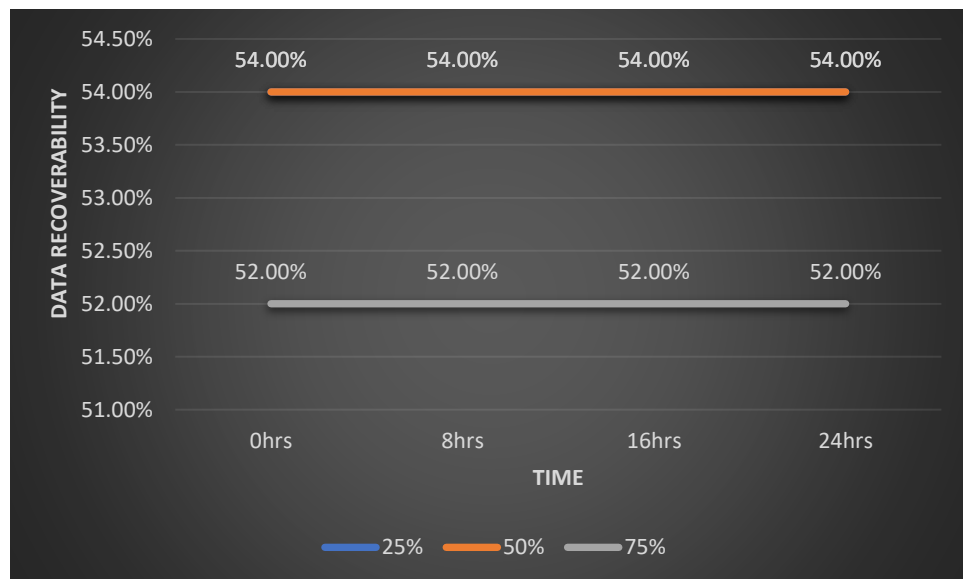
**Table 5. 3**

*Average Data Recoverability Among the Drives: Apple macOS Catalina With APFS*

Disk Usage	Duration			
	0hrs	8hrs	16hrs	24hrs
25%	54.00%	54.00%	54.00%	54.00%
50%	54.00%	54.00%	54.00%	54.00%
75%	52.00%	52.00%	52.00%	52.00%

**Figure 5. 3**

*Representation of the Average Data Recoverability Among the Drives at the Key timelines: Apple macOS Catalina With APFS*



The results from the tests conducted using Apple macOS Catalina with APFS shows that the data recoverability at 25%, 50% and 75% disk usage levels were maintained at 54%, 54% and 52% respectively, throughout the 24-hour duration. The data recoverability remained unchanged across the key timelines, at all the disk usage levels, which is represented by the horizontal straight lines displayed in Figure 5.3.

#### 5.2.1.2 Summary

The literature review identified that the reduction in the amount of recoverable data was significant at the one-hour mark after the deletion of data, and maintained the same state throughout the test time frame, which was up to 5 hours (Nisbet et al., 2013). The findings of this study complement those results as the data recoverability remained unchanged for the first 24-hours after the initial decline, providing indication that the effects of time on the data recoverability can be predictable, although the literature review also showcased an instance of change in data

recoverability between 1-hour and 12-hour marks after the deletion of data (Hadi et al., 2021). Further research with extended timelines could reveal more insights into the effects of time on the data recoverability of SSDs.

### 5.2.2 Question 2

**Q2.** What is the effect as SSD disk usage increases on the amount of deleted data that can be recovered?

Answer:

The amount of deleted data that can be recovered from SSDs tends to increase as the disk usage increases for Microsoft Windows 10 Home with NTFS and Ubuntu 21.10 ext4 file system, while it tends to decrease for Apple macOS Catalina with APFS.

Discussion:

The literature review found that garbage collection process gets initiated when the amount of available free space becomes low or when the SSD is inactive (Cornwell, 2012). It deletes the memory blocks containing obsolete data and makes the blocks available for writing (Micheloni et al., 2018, p. 11), by first relocating the pages within the blocks which contain valid data to other blocks (Tjioe et al., 2012). The TRIM process marks the pages whose data have become invalid, which helps the garbage collection process to avoid relocating the data contained in those pages (Shah et al., 2014). The findings of this study backed these findings from the literature review as the garbage collection process caused permanent data loss for many of the files which were deleted, following the execution of the OS TRIM.

### 5.2.2.1 Data Recoverability by SSD

The following sections discuss the data recoverability for each of the five SSDs that were subjected to the experiments, at different drive usage levels.

#### 5.2.2.1.1 Samsung 870 EVO

Based on the results obtained from the experiments, the data recoverability of the Samsung 870 EVO SSD, at the different disk usage levels, was calculated for each of the operating system/file system combinations, as shown in Table 5.4. It also displays the average data recoverability among the OS/file systems.

Figure 5.4 shows a comparative view of the data recoverability between the three OS/file system combinations. The data recoverability remains at 50% for all the three Operating System /file system combinations up to 50% disk usage. After the 50% drive usage level, the data recoverability increases to 60% for both Microsoft Windows 10 Home with NTFS and Ubuntu 21.10 with EXT4, exhibiting the same trend, while the data recoverability remains steadily at 50% for Apple macOS Catalina with APFS.

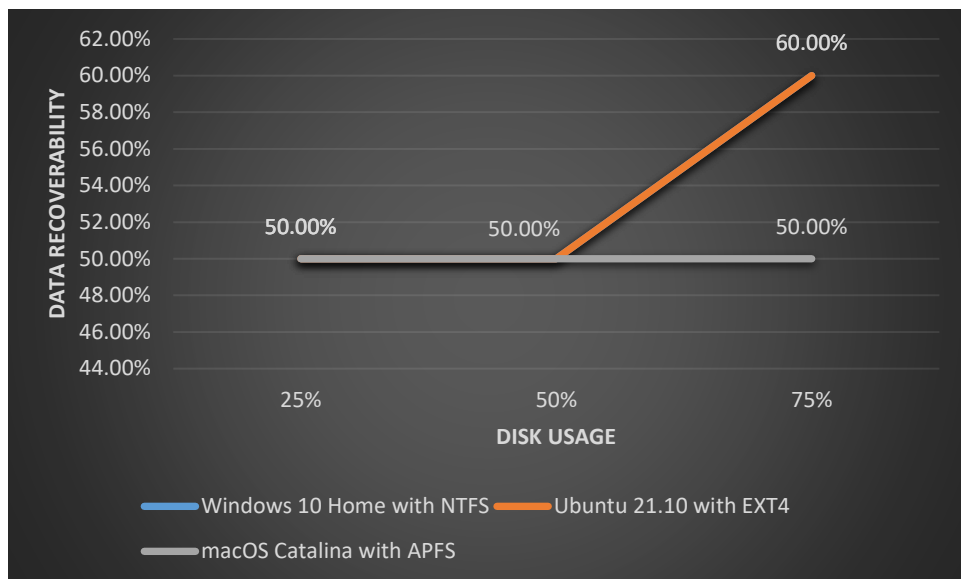
**Table 5. 4**

*Data Recoverability at Different Disk Usage Levels - Samsung 870 EVO*

OS/File System	Disk Usage		
	25%	50%	75%
Windows 10 Home with NTFS	50.00%	50.00%	60.00%
Ubuntu 21.10 with EXT4	50.00%	50.00%	60.00%
macOS Catalina with APFS	50.00%	50.00%	50.00%
Average	50.00%	50.00%	56.67%

**Figure 5. 4**

*Comparative View of Data Recoverability at Different Disk Usage Levels - Samsung 870 EVO*



**Figure 5. 5**

*Average Data Recoverability at Different Disk usage Levels - Samsung 870 EVO*

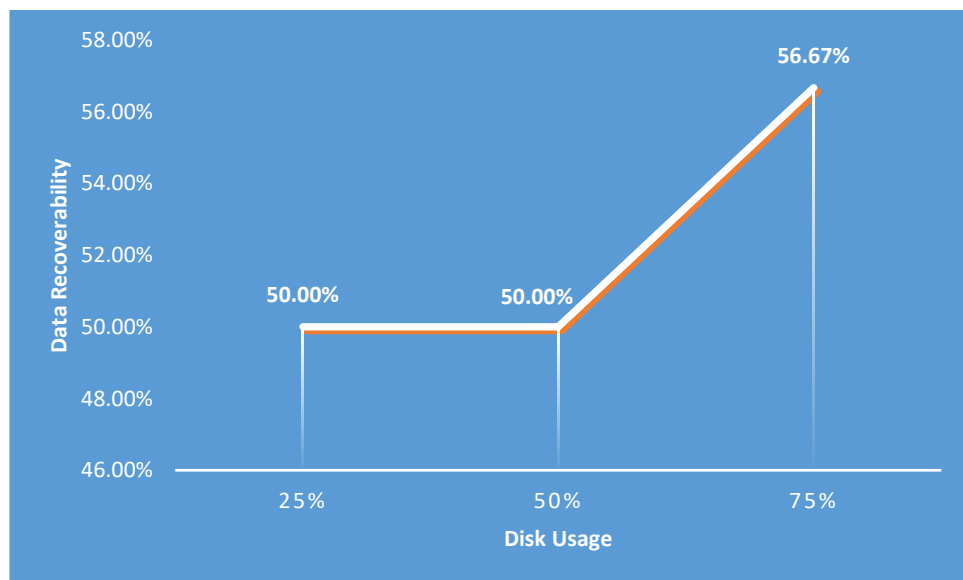


Figure 5.5 shows the average data recoverability of the Samsung 870 EVO SSD, among the three OS/file system combinations, at different drive usage levels. It shows that the data recoverability increased as the disk usage increased beyond 50%.



#### 5.2.2.1.2 Kingston A400

The rate of data recovery for the Kingston A400 SSD was calculated, using the formula presented in Section 3.3, at the different drive usage levels for each of the operating system/file system combinations used for the experiments and tabulated in Table 5.5, including the average data recoverability among the three operating system/filesystem combinations that were used for the experiments.

Figure 5.6 shows a comparative view of the data recoverability between the three operating system/file system combinations. While the rate of data recoverability remained static at 60% for Windows 10 Home with NTFS throughout the different disk usage levels, the other two operating system/file system combinations contrasted each other by showing the opposite trends. The data recoverability for Ubuntu 21.10 with EXT4 was 50% at 25% disk usage and increased to 60% when the disk usage increased to 50%, however, for Apple macOS Catalina with APFS, it advanced in the opposite direction with 60% data recoverability at 25% disk usage and then decreased to 50% when the disk usage increased to 50%. All the three operating system/file system combinations maintained their data recoverability at the same rate beyond 50% disk usage.

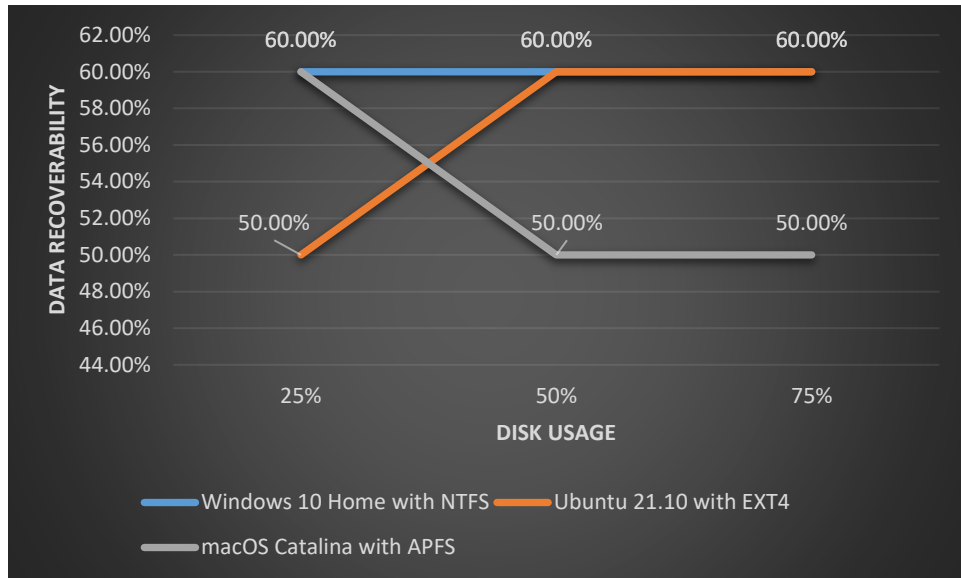
**Table 5. 5**

*Data Recoverability at Different Disk Usage Levels - Kingston A400*

OS/File System	Disk Usage		
	25%	50%	75%
Windows 10 Home with NTFS	60.00%	60.00%	60.00%
Ubuntu 21.10 with EXT4	50.00%	60.00%	60.00%
macOS Catalina with APFS	60.00%	50.00%	50.00%
Average	56.67%	56.67%	56.67%

**Figure 5. 6**

*Comparative View of Data Recoverability at Different Disk Usage Levels  
Kingston A400*



**Figure 5. 7**

*Average Data Recoverability at Different Disk Usage Levels - Kingston A400*

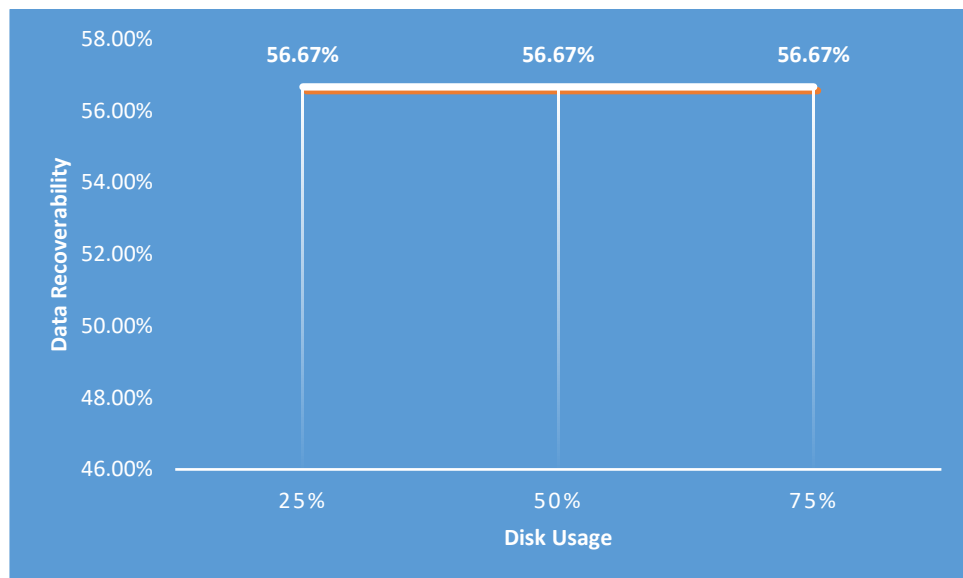


Figure 5.7 shows a representation of the average rate of data recovery, between the three operating system/file system combinations, for the

Kingston A400 SSD. The graph shows that the data recoverability remained static throughout the different disk usage levels.

#### 5.2.2.1.3 Crucial BX500

The data recoverability of the Crucial BX500 SSD, that was used for the experiments, at the different disk usage levels in conjunction with the three operating system/file system combinations is shown in Table 5.6.

Figure 5.8 shows a comparative view of the data recoverability of the Crucial BX500 SSD, between the three different operating system/file system combinations, at the different disk usage levels. With Windows 10 Home with NTFS, the data recoverability was constant at 50% until 50% disk usage and then it increased to 60% as the disk usage increased to 75%, while with Ubuntu 21.10 with EXT4, the data recoverability of the Crucial BX500 SSD exhibited an early increase to 60% from 50% as the disk usage increased from 25% to 50%. The data recoverability displayed no variation for Apple macOS Catalina with APFS along the varying disk usage levels. At the highest level of disk usage, the data recoverability for all the operating system/file system combinations was also at the highest level.

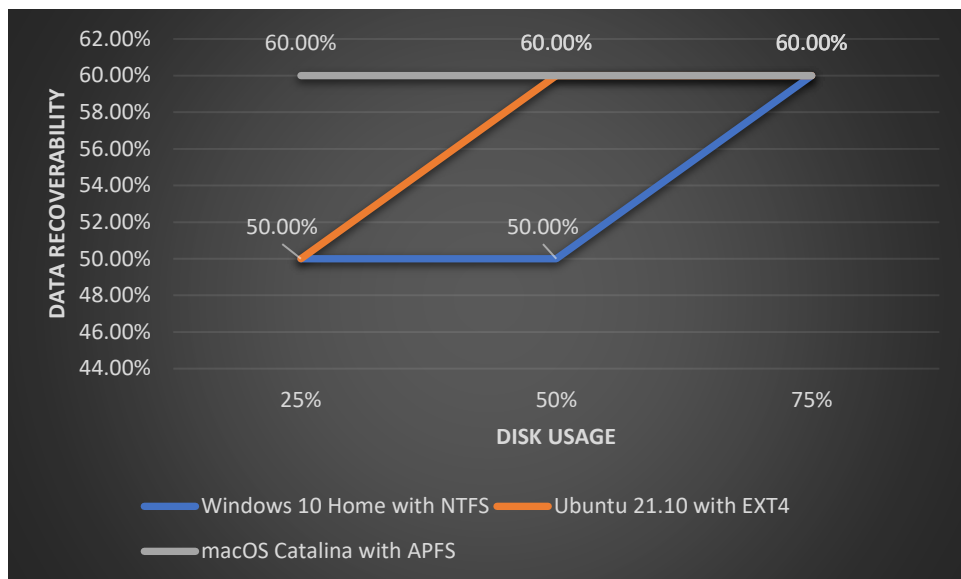
**Table 5. 6**

*Data Recoverability at Different Disk Usage Levels - Crucial BX500*

OS/File System	Disk Usage		
	25%	50%	75%
Windows 10 Home with NTFS	50.00%	50.00%	60.00%
Ubuntu 21.10 with EXT4	50.00%	60.00%	60.00%
macOS Catalina with APFS	60.00%	60.00%	60.00%
Average	53.33%	56.67%	60.00%

**Figure 5. 8**

*Comparative View of Data Recoverability at Different Disk Usage Levels - Crucial BX500*



**Figure 5. 9**

*Average Data Recoverability at Different Disk Usage Levels - Crucial BX500*

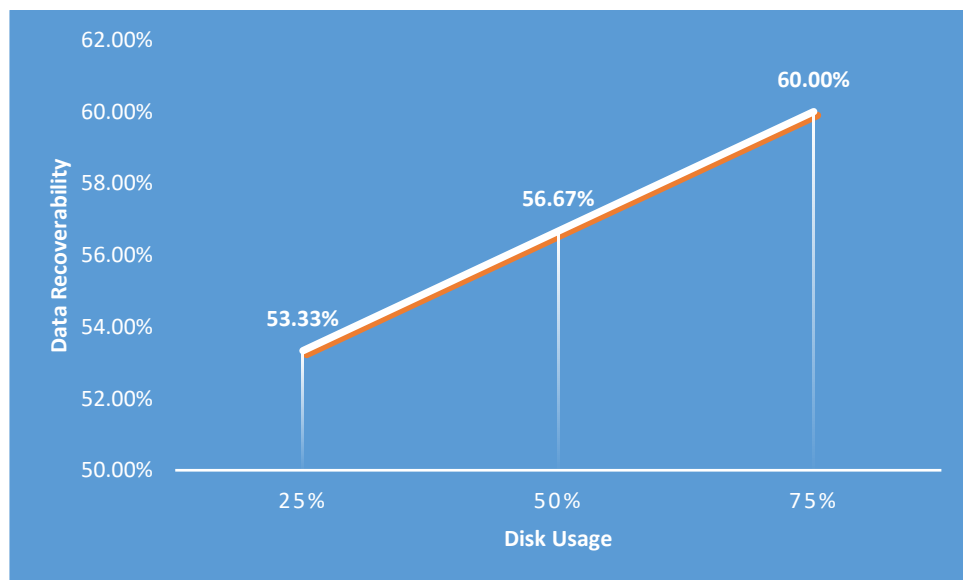


Figure 5.9 shows the average data recoverability, for the Crucial BX500 SSD, among the three different operating system/file system combinations at various disk usage levels. The chart exhibits a steady linear increase in data recoverability as the disk usage increases.

#### 5.2.2.1.4 Lexar NS 100

Table 5.7 shows the data recoverability of one of the Lexar NS 100 SSDs used for the experiments, at the different disk usage levels with three different operating system/file system combinations. It also shows the average data recoverability among the three combinations.

Figure 5.10 shows a comparison of the rate of data recovery with the three different operating system/file system combinations, used for the experiments, along with varying levels of disk usage. It shows that the data recoverability remained unchanged for Ubuntu 21.10 with EXT4 and Apple macOS Catalina with APFS, as the disk usage increased from 25% to 75%, while it increased by 20% as the disk usage increased from 50% to 75% for Windows 10 Home with NTFS. Until 50% disk usage the data recoverability remained static for all the three operating system/file system combinations.

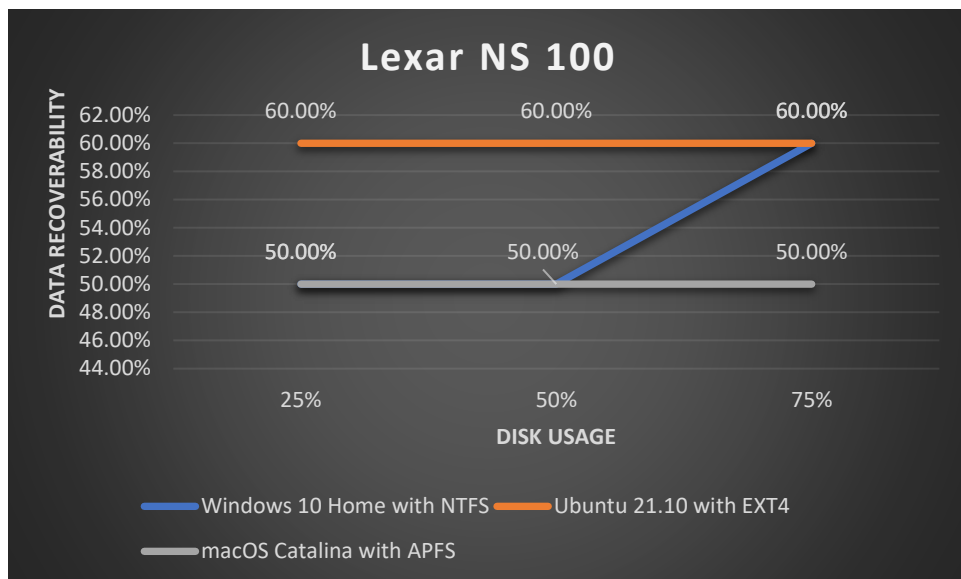
**Table 5. 7**

*Data Recoverability at Different Disk Usage Levels - Lexar NS 100*

OS/File System	Disk Usage		
	25%	50%	75%
Windows 10 Home with NTFS	50.00%	50.00%	60.00%
Ubuntu 21.10 with EXT4	60.00%	60.00%	60.00%
macOS Catalina with APFS	50.00%	50.00%	50.00%
Average	53.33%	53.33%	56.67%

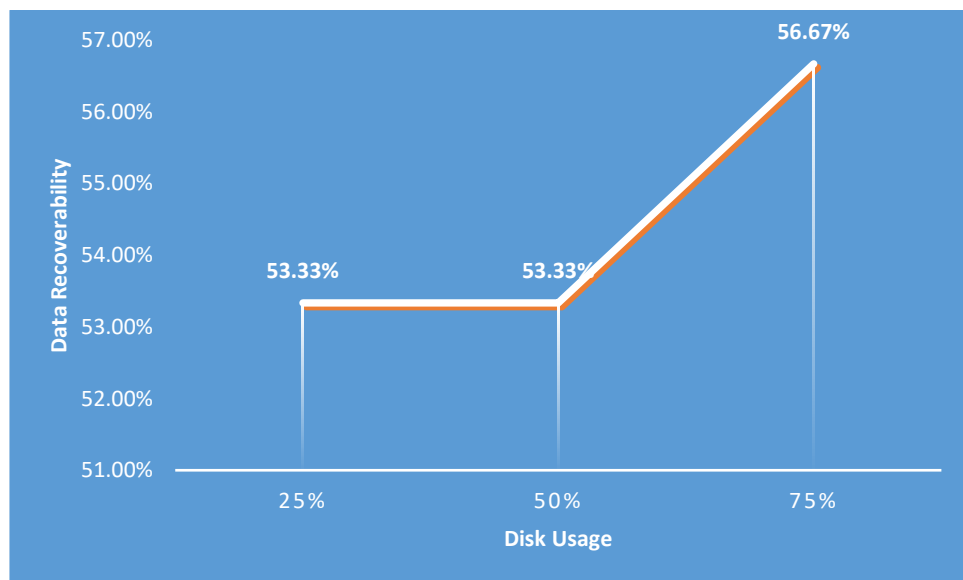
**Figure 5. 10**

*Comparative View of Data Recoverability at Different Disk Usage Levels - Lexar NS 100*



**Figure 5. 11**

*Average Data Recoverability at Different Disk Usage Levels - Lexar NS 100*



The average data recoverability, of the first Lexar NS 100 SSD that was subjected to the tests, among the three different operating system/file system combinations is shown in Figure 5.11. It shows that the rate of data

recovery increased by over 6% as the disk usage increased from 50% to 75%, while it remained static until 50% disk usage.

#### 5.2.2.1.5 Lexar NS 100 - 1

The data recoverability of the second Lexar NS 100 SSD, at the different disk usage levels, when tested with three different operating system/file system combinations, including the average data recoverability, is compiled in Table 5.8.

Figure 5.12 present a comparative view of the data recoverability among the three operating system/file system combinations. The chart shows 20% increase in data recoverability when the disk usage increased from 25% to 50%, in case of Ubuntu 21.10 with EXT4 and macOS Catalina with APFS, while it was constant for Windows 10 Home with NTFS. The behaviour was different for all the three combinations when the disk usage progressed beyond 50%. The data recoverability remained static for Ubuntu 21.10 with EXT4, while it increased for Windows 10 Home with NTFS and decreased for Apple macOS Catalina with APFS.

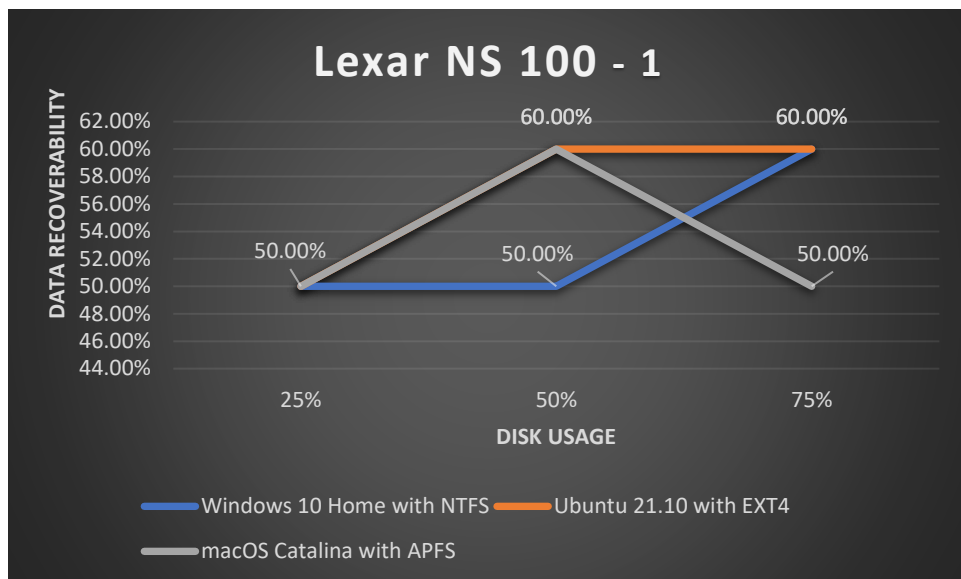
**Table 5. 8**

*Data Recoverability at Different Disk Usage Levels - Lexar NS 100 - 1*

OS/File System	Disk Usage		
	25%	50%	75%
Windows 10 Home with NTFS	50.00%	50.00%	60.00%
Ubuntu 21.10 with EXT4	50.00%	60.00%	60.00%
macOS Catalina with APFS	50.00%	60.00%	50.00%
Average	50.00%	56.67%	56.67%

**Figure 5. 12**

*Comparative View of Data Recoverability at Different Disk Usage Levels - Lexar NS 100 - 1*



**Figure 5. 13**

*Average Data Recoverability at Different Disk Usage Levels - Lexar NS 100 - 1*

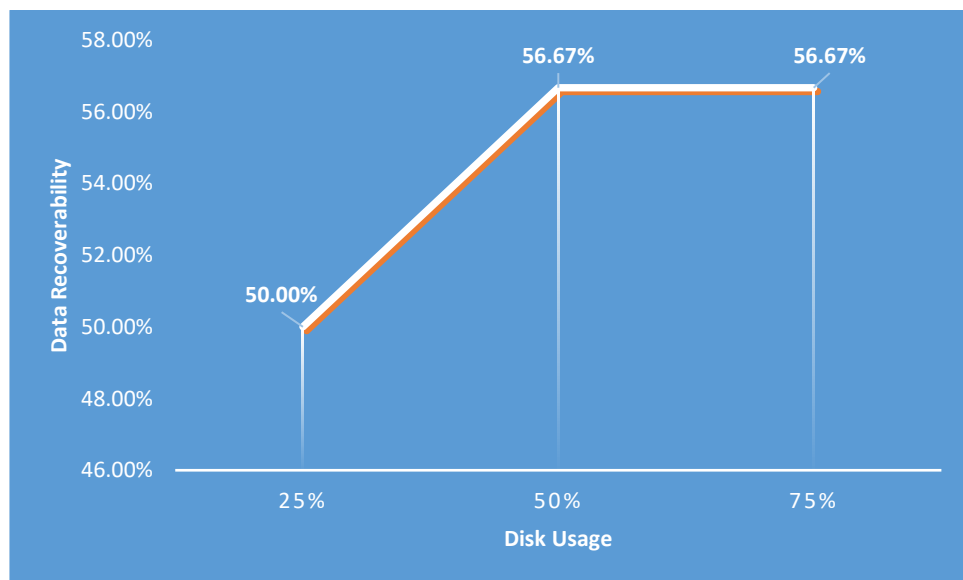


Figure 5.13 provides a visualization of the average data recoverability, of the second Lexar NS 100 subjected to the experiments, between the three operating system/file system combinations, as the disk usage increased from 25% to 75%. It shows an increase in data recoverability as the disk



usage progressed from 25% to 50% and it remained steady beyond 50% disk usage.

#### 5.2.2.2 Data Recoverability by Operating System/File System

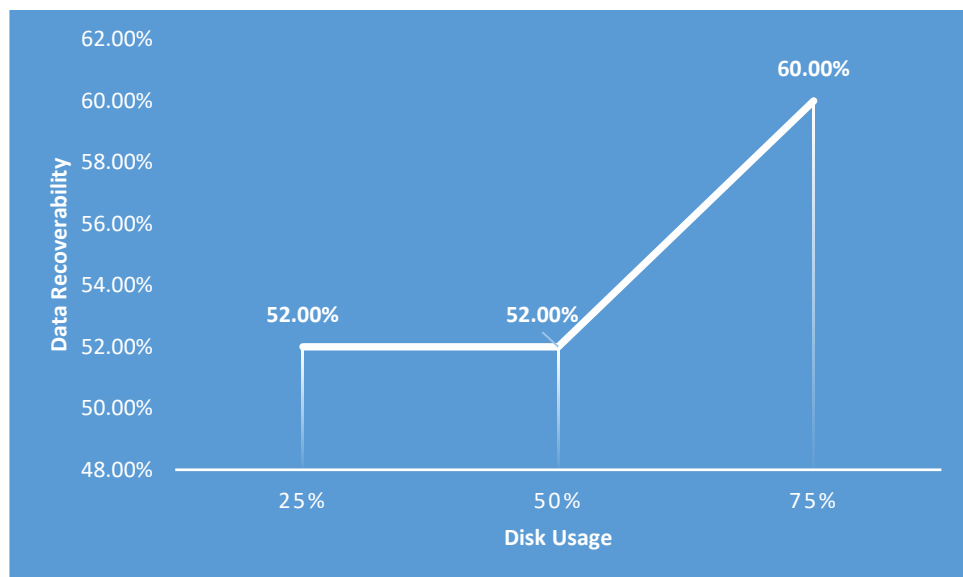
The following sections discuss the average data recoverability for each of the three operating system/file system combinations that were used for the experiments, at different drive usage levels.

##### 5.2.2.2.1 Microsoft Windows 10 Home with NTFS

Figure 5.14 visualize the average data recoverability with Microsoft Windows 10 Home with NTFS, among the five SSDs that were subjected to the experiments, along the different disk usage levels.

**Figure 5. 14**

*Average Data Recoverability at Different Disk Usage Levels - Microsoft Windows 10 Home With NTFS*



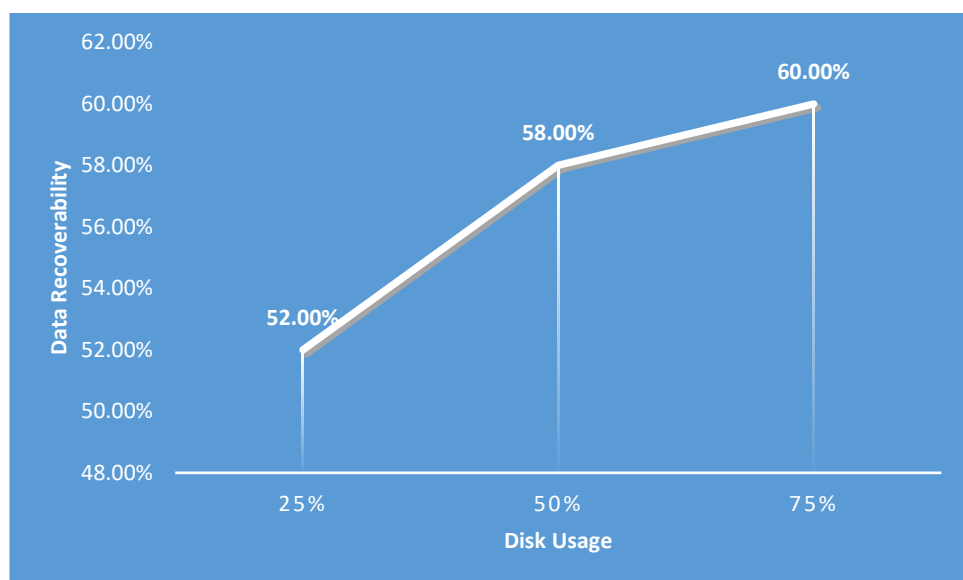
For Microsoft Windows 10 Home with NTFS, the average rate of data recovery climbed from 52% to 60% as the disk usage increased from 50% to 75%, showcasing an increase of more than 15%.

#### 5.2.2.2.2 Ubuntu 21.10 with EXT4

The average data recoverability at different disk usage levels, among the five SSDs that were tested, using Ubuntu 21.10 with EXT4 is shown in Figure 5.15.

**Figure 5. 15**

*Average Data Recoverability at Different Disk Usage Levels – Ubuntu 21.10 With Ext4*



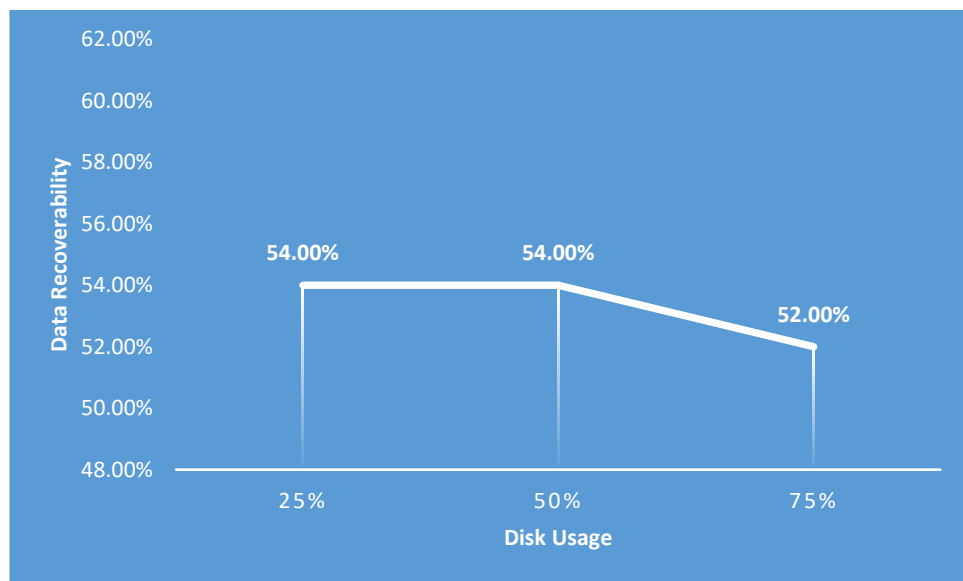
The average rate of data recovery, for Ubuntu 21.10 with EXT4, advanced positively as disk usage progressed from 25% to 75%. The data recoverability increased by 11.5%, climbing to 58% from 52%, as the disk usage increased from 25% to 50% and it further increased by 3.5%, climbing to 60% from 58%, when the disk usage increased from 50% to 75%.

### 5.2.2.2.3 Apple macOS Catalina with APFS

Figure 5.16 represents the change in the average rate of data recovery, among the five SSDs, for Apple macOS Catalina with APFS, at the different disk usage levels.

**Figure 5. 16**

*Average Data Recoverability at Different Disk Usage Levels – Apple macOS Catalina With APFS*



The average data recoverability of the SSDs, using Apple macOS Catalina with APFS, remained steady until 50% disk usage and then it declined from 54% to 52% as the disk usage increased to 75% from 50%. Among the three operating system/file system combinations used for conducting the experiments, only Apple macOS Catalina with APFS showed a decrease in data recoverability when the disk usage increased.

### 5.2.2.3 Summary

The analysis of the findings shows that four of the five SSDs that were tested demonstrated an increase in the average data recoverability at some stage of the disk usage increase, among the three operating system/file system

combinations, which accounts to 80% of the SSDs. This supports the findings from the literature review that identified increased data recoverability at high disk usage scenario (King & Vidas, 2011). Only one of the SSDs had static average data recoverability as the disk usage increased. The percentage of SSDs which exhibited a decline in the average data recoverability, among the three operating system/file system combinations, when the disk usage increased, was 0%. 40% of the SSDs showed an increase in the average data recoverability when the disk usage increased from 25% to 50%, while 60% of the SSDs exhibited an increase in the average data recoverability when the disk usage increased from 50% to 75%. Figure 5.17 shows a comparison of the average rate of data recovery, among the three operating system/file system combinations, at varying disk usage levels, between the five SSDs that were subjected to the experiments.

**Figure 5. 17**

*Comparison of the Average Data Recoverability Among the Three Operating System/File System Combinations at Different Disk Usage Levels Between the five SSDs*

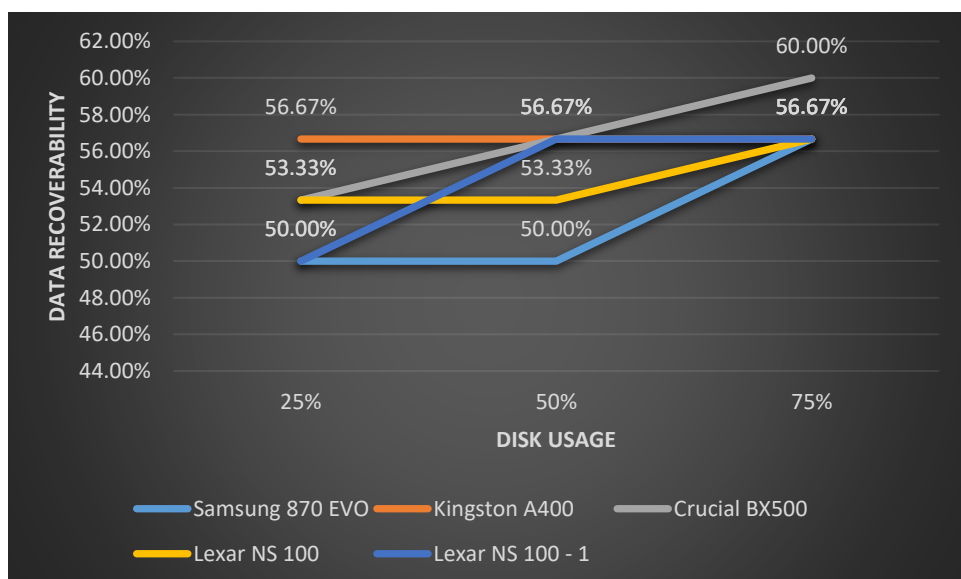
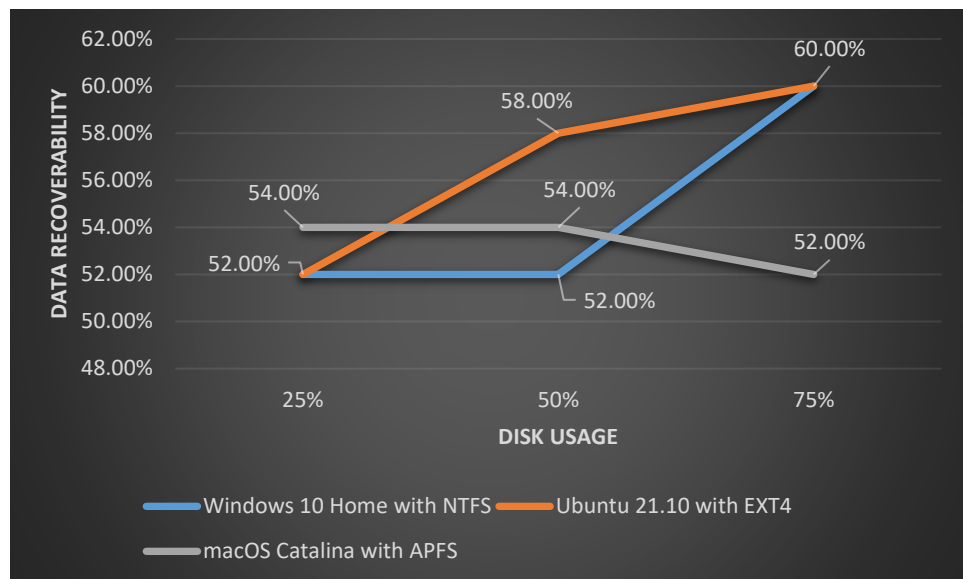


Figure 5.18 represents a comparative view of the change in the average data recoverability, among the five SSDs, between the three operating system/file system combinations at different disk usage levels. The average data recoverability among the SSDs, when experimented using Microsoft Windows 10 Home with NTFS and Ubuntu 21.10 with EXT4, exhibited a change in the positive direction as the disk usage advanced from 25% to 75% and both operating system/file system combinations had the same rate of data recovery, between them, at the lower and upper ends of the disk usage levels which represented the minimum and maximum data recoverability respectively. An increase in the average data recoverability was observed for both Windows 10 Home with NTFS and Ubuntu 21.10 with EXT4 when the disk usage increased from 50% to 75% while it changed only for Ubuntu 21.10 with EXT4 when the disk usage increased from 25% to 50%. Apple macOS Catalina with APFS showed the same trend as of Windows 10 Home with NTFS up to 50% disk usage. However, it stood out by showing a decline in the average data recoverability among the SSDs when the disk usage increased from 50% to 75%. It has been observed that instances of a decline in the data recoverability, with an increase in the disk usage, only occurred in experiments involving Apple macOS Catalina with APFS.

**Figure 5. 18**

*Comparison of Average Data Recoverability Among the SSDs at Different Disk Usage Levels Between the Three OS/File System Combinations*



### 5.3 Hypothesis

**H1.** The data recoverability of SSDs has an inverse relationship with elapsed time after data deletion and a positive correlation with drive usage.

Answer:

The results show that the data recoverability is independent of time for the first 24 hours, after the initial decline, in the absence of drive activity. With the drive usage, the results confirm a positive correlation between the drive usage and data recoverability for Windows 10 Home with NTFS and Ubuntu 21.10 with EXT4; however, for Apple macOS Catalina with APFS, the results exhibit an inverse relationship.

Discussion:

The findings of this research show that the data recoverability remains the same for the initial 24 hours, following the deletion of data. However, the behaviour does not reflect a scenario with drive activity. The static nature

of the data recoverability after the initial decline, when the SSD is idle, may be because once the TRIM is executed and the garbage collection had its first run, a drive activity might be required to trigger another run of the garbage collection and remove the residual files. Further research, with extended timelines and drive activities, is required to gain more insights into the effects of time on the data recoverability of SSDs.

The findings of this study confirm that the data recoverability of SSDs increased with an increase in disk usage for Windows 10 Home with NTFS and Ubuntu 21.10 with EXT4. The garbage collection process relocates the data from the pages containing valid data, that belong to the blocks to be deleted (Tjioe et al., 2012). In a high drive usage scenario, the garbage collection may struggle to find blocks with free pages to write those data, which may explain the increased data recoverability with increased drive usage. However, the results show that the data recoverability decreased with increased drive usage in case of Apple macOS Catalina with APFS, which warrants further research.

## 5.4 Supplementary Discussion

The literature review identified that larger files were susceptible to minimum data recovery (King & Vidas, 2011), which has been backed by the findings of this research. The analysis of the recovered files shows a strong inclination towards smaller files in terms of data recoverability. The literature review revealed that the smallest erasable unit in a flash memory is a block and individual pages cannot be erased (Aldaej et al., 2017). As discussed in the previous section, before erasing a block, the pages containing valid data need to be copied to free pages in other blocks, by the garbage collection process (Tjioe et al., 2012). Smaller files occupy fewer memory pages which could result in more pages containing valid data in the

corresponding memory blocks. In such a scenario, the garbage collection process may find it expensive and not worth to copy the pages holding valid data to other blocks but may leave the comparatively small number of pages corresponding to the deleted file intact, which could serve as a possible explanation for the greater data recovery of the smaller files.

## 5.5 Conclusion

Chapter 5 provided a discussion of the findings, from the experiments conducted to understand the various aspects of data recoverability of SSDs, presented in Chapter 4 and answered the research questions posed by this study. The analysis of the findings showed that the data recoverability of the SSDs remained unchanged during the initial 24 hours as the SSDs continued in an idle state. The results also demonstrated a positive correlation between the drive usage and data recoverability of the SSDs for Windows 10 Home with NTFS and Ubuntu 21.10 with EXT4, whilst an inverse relationship was observed for Apple macOS Catalina with APFS. This chapter also discussed the possible reasons for the particular behaviour of SSDs, in terms of data recoverability, in the given scenarios.



## Chapter 6 – Conclusion

### 6.1 Introduction

Chapter 1 introduced the research topic of data recoverability of SSDs, the motivation behind carrying out the research and defined the thesis structure. The literature review presented in Chapter 2 explored the technical details of SSDs and the challenges introduced by this modern storage medium, when it comes to digital forensics.

Chapter 3 formulated the research questions based on the findings from the literature review and developed a research design that was derived from similar studies reviewed in Chapter 2. The findings of the research, obtained by conducting the experiments defined in Chapter 3, were presented in Chapter 4. Further analysis and discussion of the findings were provided in Chapter 5, which also answered the research questions and the hypothesis.

This chapter concludes the thesis by providing a summary of the research and the implications of this study for the field of digital forensics. The identified limitations of this research and some recommendations for future research also have been included in this chapter.

### 6.2 Summary of Research

This research intended to investigate the changes in data recoverability of SSDs, over the elapsed time after the execution of the operating system TRIM command and as the disk usage varies. To obtain the data, experiments were conducted using five different SSDs, that are commonly available in New Zealand, from different manufactures with different storage capacities. The experiments for data collection were done using three different operating system/file system combinations.

The data recoverability of the SSDs that were tested showcased stability, after an initial fall, throughout the following 24 hours, as the SSDs remained in an idle state. The demonstrated behaviour was identical in all the test cases used for the experiments, irrespective of the operating system/file system combinations. This observation aligns with the findings of the literature review, even though exceptions were present. This shows that the effects of time on the amount of recoverable data is predictable which answered the research question Q1.

The recovered files were mainly smaller files, which would use fewer memory pages, resulting in memory blocks with more pages with valid data in the memory blocks corresponding to the smaller files. A possible explanation of the static nature of the data recoverability, following the initial decline, was identified that after the first run of the garbage collection, leaving aside the files which it finds expensive to delete based on the number of pages in the corresponding blocks that it must relocate valid data from, it may require a drive activity causing a change in the memory state of the drive, for another run.

The average data recoverability showed an increase at some stage, while otherwise remained static, when the disk usage increased from one level to another for Windows 10 Home with NTFS and Ubuntu 21.10 with EXT4, which supported the findings of the literature review. Whilst none of the SSDs exhibited a decline in data recoverability with a disk usage increase, when experimented with Windows 10 Home with NTFS and Ubuntu 21.10 with EXT4, there were instances of a decrease in data recoverability when tested with Apple macOS Catalina with APFS. These findings of the research showcase the effect of increased drive usage on the amount of deleted data that can be recovered, which answered the research question Q2. This

study also identified a possible reason for the increased data recoverability at high disk usage, as a potential result of a stressed garbage collection. Before deleting the memory blocks corresponding to the deleted files, the garbage collection process relocates the data in the memory pages having valid data, that are within those blocks. The garbage collection process may struggle with unavailability of free memory pages, at high disk usage, to perform the relocation task.

The research tested the hypothesis H1 that the data recoverability of SSDs has an inverse relationship with elapsed time after data deletion and a positive correlation with drive usage. The results of the experiments showed no variation in data recoverability during the initial 24 hours. In terms of drive usage, a positive correlation was observed between the drive usage and data recoverability for Windows 10 Home with NTFS and Ubuntu 21.10 with EXT4, which supports the hypothesis, while for Apple macOS Catalina with APFS it was the opposite.

### 6.3 Implications

The behaviour of SSDs, in terms of data recoverability, that was observed in this study contributes to the body of knowledge in the digital forensics field, as it aids the digital forensic investigators to plan better when the investigation involves extraction of evidence from SSDs, which are widely in use as the storage media in computers and mobile devices (Jin & Lee, 2019). Understanding of the effect of time on the data recoverability of SSDs helps the digital forensic investigators to respond in a timely manner. The knowledge of the amount of data that could be recovered at various disk usage levels, taking the time factor into account, may help the digital forensic investigators to implement appropriate strategies to prevent further data loss. The findings of this study also helps to create an

awareness of the possibility to recover the files with specific characteristics, in terms of size and type, in conjunction with time and disk usage levels, which would enable the digital forensic investigators to take informed decision to prioritize the files to be targeted, while performing the data recovery task.

#### 6.4 Limitations

One of the goals of this study was to understand the effect of time on the deleted data in SSDs. The research design was derived from the review of similar studies, most of which conducted data collection for up to 24 hours, with some extending to few more days. This study also aimed to understand the effect of disk usage on the amount of recoverable data, which required the entire data collection process to be conducted for each of the predetermined disk usage levels. This imposed a limitation of time constraint on this research and the maximum duration of data collection at a disk usage level was limited to 24 hours. Consequently, the conclusions made on the effect of time on the recoverable data from SSDs during the 24-hour period might not be an accurate reflection of the behaviour when the time elapse beyond 24 hours.

The operating systems and the corresponding file systems were selected based on their popularity. Although three of the most popular operating systems (Statista Research Department, 2022; Western Governors University, 2021) were selected for the experiments, significant operating systems including Google Chrome OS that is exclusively used in Chromebooks, the educational laptops (Google.com, n.d) that has gained much popularity recently and overtook Apple's Macs in 2020 (BBC News, 2021), which ship with an SSD or an eMMC (Vättö, 2014) and server

operating systems were excluded. The exclusions were mainly because of time and cost constraints.

The SSDs selected for the experiments were commonly available in New Zealand and all those SSDs had TRIM support. The time and cost constraints limited the number of SSDs to five, which in turn limited the sample size. A deviation from the trend in data recoverability, that was observed in this study, may be possible if experimented with a wider set of heterogeneous SSDs.

The experiments for this research were designed to understand the change in data recoverability, with time and disk usage parameters, in the absence of disk activity. This has limited the scope of the findings of this study to the idle state of SSDs. The introduction of disk activity in the experiment design may produce varying results. Also, the number of files in a set of payload files was limited to ten as an effort to minimize the complexity of the analysis of the results. Files of different types and sizes were included in the set of test files to reasonably mimic a real-world scenario. However, care should be taken while interpreting the results as experiments using a greater number of files of different types and sizes may not necessarily generate identical results.

## 6.5 Recommendations for Future Research

The purpose of this study was to gain insights into the effect of time on the amount of data that could be recovered from SSDs and the change in data recoverability at varying disk usage levels. The analysis of the findings of this research unveiled many potential areas for further research to enhance the knowledge gained from this study.

The experiments for this research were designed to collect the data to understand the change in data recoverability of SSDs, for a maximum duration of 24 hours following the execution of operating system's ATA TRIM command. This time limit has been identified as a limitation of this study which provides opportunity for future research with extended time frames to understand the effect of greater duration of time on the amount of recoverable data from SSDs.

The recent prominence gained by Google Chrome OS, overtaking Apple's macOS in popularity, that has been recognised by this study, highlights the need for future research to advance the knowledge obtained from this research, by including relevant contemporary operating systems and file systems that were not included in this study. The contradicting results observed, when experimented using Apple macOS Catalina with APFS, that showed a decline in data recoverability with an increase in drive usage, underscores the importance of the need for further research in this aspect. Also, further enhancements can be made by conducting future research with greater number of SSDs, with a variety of specifications, from different manufactures.

This research was designed to collect data when the SSDs remained in idle state. The findings of this research can be further elaborated by including disk activity in the experiment design along with the time and disk usage parameters. Another opportunity to enhance the overall knowledge gained from this research exists by widening the set of payload files with greater number of heterogeneous files.

## 6.6 Conclusion

The popularity of SSDs as a storage medium in consumer electronics, including personal computers and laptops, is on the increase. This study

explored the underlying technical architecture of SSDs and the challenges that these devices pose in terms of data recovery due to its internal self-management processes. This research collected empirical data, by conducting several experiments with SSDs, to understand the change in data recoverability at different timelines and disk usage levels.

This research revealed significant information regarding the change in data recoverability of SSDs in relation to time and disk usage, which provide value for digital forensic investigators as the knowledge help them to make better plans for data extraction from SSDs. This research also identified some opportunity for future research to enhance the knowledge that has been obtained through this study.

## References

- Aldaej, A., Ahamad, M. G., & Uddin, M. Y. (2017, March 26-27). Solid state drive data recovery in open source environment. *Proceedings of the 2017 2nd International Conference on Anti-Cyber Crimes (ICACC), Abha, Saudi Arabia*, 228-231. <https://doi.org/10.1109/Anti-Cybercrime.2017.7905296>
- Amato, F., Castiglione, A., Cozzolino, G., & Narducci, F. (2020). A semantic-based methodology for digital forensics analysis. *Journal of Parallel and Distributed Computing*, 138, 172-177. <https://doi.org/10.1016/j.jpdc.2019.12.017>
- Årnes, A. (2017). *Digital Forensics*. John Wiley & Sons, Incorporated.
- Baca, M., Cosic, J., & Cosic, Z. (2013, June 24-27). Forensic analysis of social networks (case study). *Proceedings of the ITI 2013 35th International Conference on Information Technology Interfaces, Cavtat, Croatia*, 219-223. <https://doi.org/10.2498/iti.2013.0526>
- Bavandpour, M. (2020). *Toward Efficient Mixed-signal Neural Processors Using Non-volatile Memory Devices* (Publication Number 27828885) [Ph.D., University of California, Santa Barbara]. ProQuest Dissertations & Theses Global. Ann Arbor.
- BBC News. (2021). *Chromebooks overtake Macs in market share for first time - BBC News*. Retrieved June 23, 2022, from <https://www.bbc.com/news/technology-56116573>
- Bednar, P., & Katos, V. (2011, October 7). SSD: New Challenges for Digital Forensics. *Proceedings of the 8th Conference of the Italian Chapter of the Association for Information Systems, Rome, Italy*. <https://portal.research.lu.se/files/5456453/4318024.pdf>
- Bell, G. B., & Boddington, R. (2010). Solid state drives: the beginning of the end for current practice in digital forensic recovery? *Journal of Digital Forensics, Security and Law*, 5(3), 1. <https://doi.org/10.15394/jdfsl.2010.1078>
- Berghel, H. (2007). Hiding data, forensics, and anti-forensics. *Communications of the ACM*, 50(4), 15-20. <https://doi.org/10.1145/1232743.1232761>
- Bhat, W. A., & Quadri, S. M. K. (2012). After-deletion data recovery: myths and solutions. *Computer Fraud & Security*, 2012(4), 17-20. [https://doi.org/10.1016/S1361-3723\(12\)70032-5](https://doi.org/10.1016/S1361-3723(12)70032-5)
- Cai, Y., Luo, Y., Ghose, S., & Mutlu, O. (2015, June 22-25). Read Disturb Errors in MLC NAND Flash Memory: Characterization, Mitigation, and Recovery. *Proceedings of the 2015 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, Rio de Janeiro, Brazil*, 438-449. <https://doi.org/10.1109/DSN.2015.49>



- Cai, Y., Luo, Y., Haratsch, E. F., Mai, K., & Mutlu, O. (2015, February 7-11). Data retention in MLC NAND flash memory: Characterization, optimization, and recovery. *Proceedings of the 2015 IEEE 21st International Symposium on High Performance Computer Architecture (HPCA)*, Burlingame, CA, USA, 551-563.  
<https://doi.org/10.1109/HPCA.2015.7056062>
- Carrier, B. (2002). *Open source digital forensics tools: The legal argument*.
- Carrier, B. (2005). *File system forensic analysis*. Addison-Wesley Professional.
- Carrier, B. (n.d-a). *Autopsy*. Retrieved May 7, 2022, from <https://www.sleuthkit.org/autopsy>
- Carrier, B. (n.d-b). *The Sleuth Kit: File and Volume System Analysis*. Retrieved May 7, 2022, from <https://www.sleuthkit.org/sleuthkit/desc.php>
- Casey, E. (2011). *Digital evidence and computer crime : forensic science, computers and the internet* (Third edition. ed.). Academic Press.
- Caulfield, L. M. (2013). *Symbiotic Solid State Drives: Management of Modern NAND Flash Memory* (Publication Number 3567520) [Ph.D., University of California, San Diego]. ProQuest Dissertations & Theses Global. Ann Arbor.
- Chang, L.-P., & Du, C.-D. (2010). Design and implementation of an efficient wear-leveling algorithm for solid-state-disk microcontrollers. *ACM Trans. Des. Autom. Electron. Syst.*, 15(1), Article 6. <https://doi.org/10.1145/1640457.1640463>
- Chang, Y.-H., & Chang, L.-P. (2013). Efficient Wear Leveling in NAND Flash Memory. In R. Micheloni, A. Marelli, & K. Eshghi (Eds.), *Inside Solid State Drives (SSDs)* (pp. 233-257). Springer Netherlands. [https://doi.org/10.1007/978-94-007-5146-0\\_9](https://doi.org/10.1007/978-94-007-5146-0_9)
- Chunmei, Z., Yong, Z., Wei-Ting Kary, C., & Junyao, T. (2017, March 12-13). Fail mechanism of program disturbance for erase cells VT positive shift in NAND flash technology. *Proceedings of the 2017 China Semiconductor Technology International Conference (CSTIC), Shanghai, China*, 1-3. <https://doi.org/10.1109/CSTIC.2017.7919835>
- Cohen, M. I. (2007). Advanced carving techniques. *Digital Investigation*, 4(3), 119-128.  
<https://doi.org/10.1016/j.diin.2007.10.001>
- Compagnoni, C. M., & Spinelli, A. S. (2019). Reliability of NAND Flash Arrays: A Review of What the 2-D-to-3-D Transition Meant. *IEEE Transactions on Electron Devices*, 66(11), 4504-4516. <https://doi.org/10.1109/TED.2019.2917785>
- Compagnoni, C. M., Spinelli, A. S., Gusmeroli, R., Beltrami, S., Ghetti, A., & Visconti, A. (2008). Ultimate Accuracy for the nand Flash Program Algorithm Due to the Electron Injection

- Statistics. *IEEE Transactions on Electron Devices*, 55(10), 2695-2702.  
<https://doi.org/10.1109/TED.2008.2003230>
- Cornwell, M. (2012). Anatomy of a solid-state drive. *Communications of the ACM*, 55(12), 59-63. <https://doi.org/10.1145/2380656.2380672>
- Datarecovery.com. (2015). *What is the Windows 10 File System?* Retrieved May 6, 2022, from <https://datarecovery.com/rd/windows-10-file-system>
- Di Marco, A. (2007, September 27-28). Exploiting Commodity Hard-Disk Geometry to Efficiently Preserve Data Consistency. *Proceedings of the Formal Methods and Stochastic Models for Performance Evaluation, Berlin, Heidelberg*, 260-274.  
[https://doi.org/10.1007/978-3-540-75211-0\\_19](https://doi.org/10.1007/978-3-540-75211-0_19)
- Do, J., Lomet, D., & Picoli, I. L. (2019, July 1). Improving CPU I/O Performance via SSD Controller FTL Support for Batched Writes. *Proceedings of the 15th International Workshop on Data Management on New Hardware, Amsterdam, Netherlands*, Article 2.  
<https://doi.org/10.1145/3329785.3329925>
- Evanson, N. (2020). *Anatomy of a Storage Drive: Hard Disk Drives*. Retrieved September 10, 2021, from <https://www.techspot.com/article/1984-anatomy-hard-drive/>
- Fazio, A. (2004). Flash Memory Scaling. *MRS Bulletin*, 29(11), 814-817.  
<https://doi.org/10.1557/mrs2004.233>
- Fazio, A. (2006, May 8-12). Solid State Storage, Limits of Flash Memory. *Proceedings of the 2006 IEEE International Magnetism Conference (INTERMAG), San Diego, CA, USA*, 101-101. <https://doi.org/10.1109/INTMAG.2006.375601>
- Filho, J. E. M. (2021). *resurrecting-open-source-projects/dcfldd: Enhanced version of dd for forensics and security*. Retrieved May 16, 2022, from <https://github.com/resurrecting-open-source-projects/dcfldd>
- Fink, J. (2013). The Susceptibility of Magnetic Hard Disk Drives to External dc Magnetic Fields. *IEEE Potentials*, 32(6), 32-38. <https://doi.org/10.1109/MPOT.2012.2227523>
- Garfinkel, S., Malan, D., Dubec, K., Stevens, C., & Pham, C. (2006). Advanced Forensic Format: an Open Extensible Format for Disk Imaging. In M. S. Olivier & S. Sheno (Eds.), *Advances in Digital Forensics II* (Vol. 222, pp. 13-27). Springer US.  
[https://doi.org/10.1007/0-387-36891-4\\_2](https://doi.org/10.1007/0-387-36891-4_2)
- Göbel, T., Türr, J., & Baier, H. (2019, August 26-29). Revisiting Data Hiding Techniques for Apple File System. *Proceedings of the 14th International Conference on Availability, Reliability and Security, Canterbury, CA, United Kingdom*.  
<https://doi.org/10.1145/3339252.3340524>

- Google.com. (n.d). *Google Chromebooks*. Retrieved June 23, 2022, from <https://www.google.com/chromebook/discover/>
- Hadi, H. J., Musthaq, N., & Khan, I. U. (2021, November 23-25). SSD Forensic: Evidence Generation and Forensic Research on Solid State Drives Using Trim Analysis. *Proceedings of the 2021 International Conference on Cyber Warfare and Security (ICCWS), Islamabad, Pakistan*, 51-56. <https://doi.org/10.1109/ICCWS53234.2021.9702989>
- Huebner, E., Bem, D., & Wee, C. K. (2006). Data hiding in the NTFS file system. *Digital Investigation*, 3(4), 211-226. <https://doi.org/10.1016/j.diin.2006.10.005>
- Huffman, C. (2015). Chapter 3 - Storage. In C. Huffman (Ed.), *Windows Performance Analysis Field Guide* (pp. 57-91). Syngress. <https://doi.org/10.1016/B978-0-12-416701-8.00003-X>
- Indiana University. (2021). *About POSIX*. Retrieved May 6, 2022, from <https://kb.iu.edu/d/agjv>
- Ismail-Beigi Research Group. (n.d). *Hard Drives Methods And Materials | Ismail-Beigi Research Group*. Retrieved September 10, 2021, from <https://volga.eng.yale.edu/teaching-resources/hard-drives/methods-and-materials>
- Jesung, K., Jong Min, K., Noh, S. H., Sang Lyul, M., & Yookun, C. (2002). A space-efficient flash translation layer for CompactFlash systems. *IEEE Transactions on Consumer Electronics*, 48(2), 366-375. <https://doi.org/10.1109/TCE.2002.1010143>
- Jimenez, X., Novo, D., & lenne, P. (2014, February 17-20). Wear Unleveling: Improving NAND Flash Lifetime by Balancing Page Endurance. *Proceedings of the 12th USENIX Conference on File and Storage Technologies (FAST '14), Santa Clara, CA, USA*. <https://www.usenix.org/conference/fast14/technical-sessions/presentation/jimenez>
- Jin, Y., & Lee, B. (2019). Chapter One - A comprehensive survey of issues in solid state drives. In A. R. Hurson (Ed.), *Advances in Computers* (Vol. 114, pp. 1-69). Elsevier. <https://doi.org/10.1016/bs.adcom.2019.02.001>
- Jin, Y., Tseng, H., Papakonstantinou, Y., & Swanson, S. (2017, February 4-8). KAML: A Flexible, High-Performance Key-Value SSD. *Proceedings of the 2017 IEEE International Symposium on High Performance Computer Architecture (HPCA), Austin, TX, USA*, 373-384. <https://doi.org/10.1109/HPCA.2017.15>
- Joe, S., Jeong, M., Kang, M., Han, K., Park, S., & Lee, J. (2012). New Read Schemes Using Boosted Channel Potential of Adjacent Bit-Line Strings in nand Flash Memory. *IEEE Electron Device Letters*, 33(8), 1198-1200. <https://doi.org/10.1109/LED.2012.2202209>
- Jose, S. T., & Pradeep, C. (2013, March 22-23). Design of a multichannel NAND Flash memory controller for efficient utilization of bandwidth in SSDs. *Proceedings of the 2013 International Mutli-Conference on Automation, Computing, Communication, Control*

and Compressed Sensing (iMac4s), Kottayam, India, 235-239.

<https://doi.org/10.1109/iMac4s.2013.6526414>

- Joshi, B. R., & Hubbard, R. (2016, May 26-28). Forensics analysis of solid state drive (SSD). *Proceedings of the 2016 Universal Technology Management Conference (UTMC), Minnesota, United States of America*, 1-12.  
[https://www.researchgate.net/profile/Natalie-Walker-15/publication/303522421\\_Proceedings\\_of\\_2016\\_Universal\\_Technology\\_Management\\_Conference\\_UTMC\\_Minnesota\\_United\\_States\\_of\\_America\\_2016/links/57469b9208ae9ace84243f3c/Proceedings-of-2016-Universal-Technology-Management-Conference-UTMC-Minnesota-United-States-of-America-2016.pdf#page=3](https://www.researchgate.net/profile/Natalie-Walker-15/publication/303522421_Proceedings_of_2016_Universal_Technology_Management_Conference_UTMC_Minnesota_United_States_of_America_2016/links/57469b9208ae9ace84243f3c/Proceedings-of-2016-Universal-Technology-Management-Conference-UTMC-Minnesota-United-States-of-America-2016.pdf#page=3)
- Jung, M., Choi, W., Kwon, M., Srikantaiah, S., Yoo, J., & Kandemir, M. T. (2020). Design of a Host Interface Logic for GC-Free SSDs. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 39(8), 1674-1687.  
<https://doi.org/10.1109/TCAD.2019.2919035>
- Jung, M., Choi, W., Srikantaiah, S., Yoo, J., & Kandemir, M. T. (2014). HIOS: a host interface I/O scheduler for solid state disks. *SIGARCH Comput. Archit. News*, 42(3), 289–300.  
<https://doi.org/10.1145/2678373.2665715>
- Kerekes, Z. (n.d). *SSD Market History: Charting the Rise of the SSD Market*. Retrieved September 06, 2021, from  
<https://www.storagesearch.com/chartingtheriseofssds.html>
- Kerrisk, M. (2021). *ext4(5) — Linux manual page*. Retrieved May 6, 2022, from  
<https://man7.org/linux/man-pages/man5/ext4.5.html>
- Khalifa, K., Fawzy, H., El-Ashry, S., & Salah, K. (2013, December 16-18). Memory controller architectures: A comparative study. *Proceedings of the 2013 8th IEEE Design and Test Symposium, Marrakesh, Morocco*, 1-6. <https://doi.org/10.1109/IDT.2013.6727083>
- Kim, G., & Shin, D. (2011, November 29-Dec 1). Performance analysis of SSD write using TRIM in NTFS and EXT4. *Proceedings of the 2011 6th International Conference on Computer Sciences and Convergence Information Technology (ICCIT), Seogwipo, Korea (South)*, 422-423.
- Kim, T.-K., Chang, S., & Choi, J.-H. (2009). Floating gate technology for high performance 8-level 3-bit NAND flash memory. *Solid-State Electronics*, 53(7), 792-797.  
<https://doi.org/10.1016/j.sse.2009.03.019>
- King, C., & Vidas, T. (2011). Empirical analysis of solid state disk data retention when used with contemporary operating systems. *Digital Investigation*, 8, S111-S117.  
<https://doi.org/10.1016/j.diin.2011.05.013>

- kingston.com. (2021, February). *NVMe vs SATA: What is the difference?* Retrieved October 30, 2021 from <https://www.kingston.com/unitedstates/us/solutions/pc-performance/nvme-vs-sata>
- Koltsidas, I., & Viglas, S. D. (2011, June 12-16). Data management over flash memory. *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data, Athens, Greece*, 1209–1212. <https://doi.org/10.1145/1989323.1989455>
- Li, Y., & Quader, K. N. (2013). NAND Flash Memory: Challenges and Opportunities. *Computer*, 46(8), 23-29. <https://doi.org/10.1109/MC.2013.190>
- Lu, C. Y. (2012). Future prospects of NAND flash memory technology--the evolution from floating gate to charge trapping to 3D stacking. *J Nanosci Nanotechnol*, 12(10), 7604-7618. <https://doi.org/10.1166/jnn.2012.6650>
- Luo, Y., Ghose, S., Cai, Y., Haratsch, E. F., & Mutlu, O. (2018, February 24-28). HeatWatch: Improving 3D NAND Flash Memory Device Reliability by Exploiting Self-Recovery and Temperature Awareness. *Proceedings of the 2018 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, Vienna, Austria, 504-517. <https://doi.org/10.1109/HPCA.2018.00050>
- Lutkevich, B. (n.d). *What Is NTFS And How Does It Work?* Retrieved September 11, 2021, from <https://searchwindowserver.techtarget.com/definition/NTFS>
- Malladi, K. T., Chang, M., Niu, D., & Zheng, H. (2017, August 7-9). FlashStorageSim: Performance Modeling for SSD Architectures. *Proceedings of the 2017 International Conference on Networking, Architecture, and Storage (NAS)*, Shenzhen, China, 1-2. <https://doi.org/10.1109/NAS.2017.8026860>
- McEwan, A. A., & Mir, I. (2015, August 26-28). An Embedded FTL for SSD RAID. *Proceedings of the 2015 Euromicro Conference on Digital System Design, Madeira, Portugal*, 575-582. <https://doi.org/10.1109/DSD.2015.39>
- Meffert, C. S., Baggili, I., & Breitingner, F. (2016). Deleting collected digital evidence by exploiting a widely adopted hardware write blocker. *Digital Investigation*, 18, S87-S96. <https://doi.org/10.1016/j.diin.2016.04.004>
- Michelsoni, R., & Crippa, L. (2016). 3D Stacked NAND Flash Memories. In R. Michelsoni (Ed.), *3D Flash Memories* (pp. 63-83). Springer Netherlands. [https://doi.org/10.1007/978-94-017-7512-0\\_3](https://doi.org/10.1007/978-94-017-7512-0_3)
- Michelsoni, R., Marelli, A., & Eshghi, K. (2018). *Inside Solid State Drives (SSDs)* (2nd ed.). Springer Singapore Pte. Limited.
- Microsoft. (2021a). *NTFS overview*. Retrieved May 6, 2022, from <https://docs.microsoft.com/en-us/windows-server/storage/file-server/ntfs-overview>

- Microsoft. (2021b). *Overview of FAT, HPFS, and NTFS File Systems*. Retrieved May 6, 2022, from <https://docs.microsoft.com/en-us/troubleshoot/windows-client/backup-and-storage/fat-hpfs-and-ntfs-file-systems>
- Mir, S. S., Shoaib, U., & Sarfraz, M. S. (2016). Analysis of digital forensic investigation models. *International Journal of Computer Science and Information Security*, 14(11), 292.
- Morningstar, L. (2018). *Trim Command and Garbage Collection in Solid State Drives as an Antiforensic Technology* (Publication Number 13423658) [Master's thesis, Utica College]. ProQuest Dissertations & Theses Global.
- Muntaha, S. (2019). *Linux dd Command*. Retrieved May 16, 2022, from [https://linuxhint.com/linux\\_dd\\_command](https://linuxhint.com/linux_dd_command)
- National Institute of Standards and Technology. (2006). Guide to Integrating Forensic Techniques into Incident Response. *NIST Special Publication 800-86*. <https://doi.org/10.6028/NIST.SP.800-86>
- Neagu, C. (2018). *Simple questions: What is NTFS and why is it useful?* Retrieved May 6, 2022, from <https://www.digitalcitizen.life/what-is-ntfs-why-useful>
- Neyaz, A., Zhou, B., & Karpoor, N. (2019, December 9-12). Comparative Study of Wear-leveling in Solid-State Drive with NTFS File System. *Proceedings of the 2019 IEEE International Conference on Big Data (Big Data)*, Los Angeles, CA, USA, 4294-4298. <https://doi.org/10.1109/BigData47090.2019.9006067>
- Nisbet, A., & Jacob, R. (2019, August 5-8). TRIM, Wear Levelling and Garbage Collection on Solid State Drives: A Prediction Model for Forensic Investigators. *Proceedings of the 2019 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, Rotorua, New Zealand, 419-426. <https://doi.org/10.1109/TrustCom/BigDataSE.2019.00063>
- Nisbet, A., Lawrence, S., & Ruff, M. (2013, December 2-4). A forensic analysis and comparison of solid state drive data retention with trim enabled file systems. *Proceedings of the 11th Australian Digital Forensics Conference*, Perth, Western Australia. <https://doi.org/10.4225/75/57b3d766fb873>
- Nordvik, R. (2022a). APFS. In C. Hummert & D. Pawlaszczyk (Eds.), *Mobile Forensics – The File Format Handbook: Common File Formats and File Systems Used in Mobile Devices* (pp. 3-39). Springer International Publishing. [https://doi.org/10.1007/978-3-030-98467-0\\_1](https://doi.org/10.1007/978-3-030-98467-0_1)
- Nordvik, R. (2022b). Ext4. In C. Hummert & D. Pawlaszczyk (Eds.), *Mobile Forensics – The File Format Handbook: Common File Formats and File Systems Used in Mobile Devices* (pp. 41-68). Springer International Publishing. [https://doi.org/10.1007/978-3-030-98467-0\\_2](https://doi.org/10.1007/978-3-030-98467-0_2)

- NTFS.com. (n.d). *Hard Disk Drive Basics*. Retrieved September 10, 2021, from <http://ntfs.com/hard-disk-basics.htm>
- Offensive Security. (2021). *foremost | Kali Linux Tools*. Retrieved May 7, 2022, from <https://www.kali.org/tools/foremost/>
- Offensive Security. (2022). *What is Kali Linux?* Retrieved May 05, 2022, from <https://www.kali.org/docs/introduction/what-is-kali-linux>
- Oracle Corporation. (n.d). *Oracle VM VirtualBox*. Retrieved May 16, 2022, from <https://www.oracle.com/virtualization/technologies/vm/virtualbox.html>
- Pal, A., & Memon, N. (2009). The evolution of file carving. *IEEE Signal Processing Magazine*, 26(2), 59-71. <https://doi.org/10.1109/MSP.2008.931081>
- Park, J. K., Seo, Y., & Kim, J. (2019). A Flash-Based SSD Cache Management Scheme for High Performance Home Cloud Storage. *IEEE Transactions on Consumer Electronics*, 65(3), 418-425. <https://doi.org/10.1109/TCE.2019.2897686>
- Pitchumani, R., Hospodor, A., Amer, A., Kang, Y., Miller, E. L., & Long, D. D. E. (2012, August 7-9). Emulating a Shingled Write Disk. *Proceedings of the 2012 IEEE 20th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, Washington, DC, USA*, 339-346. <https://doi.org/10.1109/MASCOTS.2012.46>
- Plugable Technologies. (2021, August 19). *Trim an SSD in macOS*. Retrieved May 10, 2022, from <https://kb.plugable.com/data-storage/trim-an-ssd-in-macos>
- Poisel, R., & Tjoa, S. (2013, September 2-6). A Comprehensive Literature Review of File Carving. *Proceedings of the 2013 International Conference on Availability, Reliability and Security, Regensburg, Germany*, 475-484. <https://doi.org/10.1109/ARES.2013.62>
- Raji, M., Wimmer, H., & Haddad, R. J. (2018, April 19-22). Analyzing Data from an Android Smartphone while Comparing between Two Forensic Tools. *Proceedings of the SoutheastCon 2018, St. Petersburg, FL, USA*, 1-6. <https://doi.org/10.1109/SECON.2018.8478851>
- Rajimwale, A., Prabhakaran, V., & Davis, J. D. (2009, July 14-19). Block Management in Solid-State Devices. *Proceedings of the USENIX Annual Technical Conference, San Diego, USA*.
- Ranker. (2019, October 28). *The Best SSD Manufacturers*. Retrieved May 16, 2022, from <https://www.ranker.com/list/the-best-ssd-manufacturers/computer-hardware>



- Reddy, N. (2019). Solid State Device (SSD) Forensics. In N. Reddy (Ed.), *Practical Cyber Forensics: An Incident-Based Approach to Forensic Investigations* (pp. 379-400). Apress. [https://doi.org/10.1007/978-1-4842-4460-9\\_12](https://doi.org/10.1007/978-1-4842-4460-9_12)
- Richter, D. (2016). *Flash Memories*. Springer.
- Rizvi, S. S., & Chung, T. (2010, April 16-18). Flash SSD vs HDD: High performance oriented modern embedded and multimedia storage systems. *Proceedings of the 2010 2nd International Conference on Computer Engineering and Technology, Chengdu, China, 7*, V7-297-V297-299. <https://doi.org/10.1109/ICCET.2010.5485421>
- Romero, M., Macas, E., Rosero, D., Quisnancela, H., & Grijalva, J. (2019, November 20-22). OVJESMO Methodology: Data Collected for Forensic Analysis in Hard Disk Drives Applying the ISO / IEC Standard. *Proceedings of the 2019 International Conference on Information Systems and Computer Science (INCISCOS), Quito, Ecuador*, 136-143. <https://doi.org/10.1109/INCISCOS49368.2019.00030>
- Sammons, J. (2015). *Digital Forensics : Threatscape and Best Practices*. Elsevier Science & Technology Books.
- Samsung. (n.d). *Samsung's journey begins*. Retrieved September 08, 2021, from <https://www.samsung.com/semiconductor/minisite/ssd/worlds-no1-flash-memory/episode1/>
- Sanvido, M. A. A., Chu, F. R., Kulkarni, A., & Selinger, R. (2008). nand Flash Memory and Its Role in Storage Architectures. *Proceedings of the IEEE*, 96(11), 1864-1874. <https://doi.org/10.1109/JPROC.2008.2004319>
- Seung-Ho, L., & Kyu-Ho, P. (2006). An efficient NAND flash file system for flash memory storage. *IEEE Transactions on Computers*, 55(7), 906-912. <https://doi.org/10.1109/TC.2006.96>
- Shah, Z., Mahmood, A. N., & Slay, J. (2014). Forensic potentials of solid state drives. *Proceedings of the International Conference on Security and Privacy in Communication Networks, Beijing, China*, 113-126. [https://link.springer.com/chapter/10.1007/978-3-319-23802-9\\_11](https://link.springer.com/chapter/10.1007/978-3-319-23802-9_11)
- Sindhu, K. K., & Meshram, B. B. (2012). Digital Forensics and Cyber Crime Datamining. *Journal of Information Security, Vol.03*, 6, Article 21340. <https://doi.org/10.4236/jis.2012.33024>
- Spinelli, A. S., Compagnoni, C. M., & Lacaita, A. L. (2017). Reliability of NAND Flash Memories: Planar Cells and Emerging Issues in 3D Devices. *Computers*, 6(2), 16. <https://www.mdpi.com/2073-431X/6/2/16>
- Statista Research Department. (2022). *Market share held by the leading computer (desktop/tablet/console) operating systems worldwide from January 2012 to December*



2021. Retrieved May 6, 2022, from <https://www.statista.com/statistics/268237/global-market-share-held-by-operating-systems-since-2009>
- Tjioe, J., Blanco, A., Xie, T., & Ouyang, Y. (2012, June 28-30). Making Garbage Collection Wear Conscious for Flash SSD. *Proceedings of the 2012 IEEE Seventh International Conference on Networking, Architecture, and Storage, Xiamen, China*, 114-123. <https://doi.org/10.1109/NAS.2012.20>
- Tobin, P., Le-Khac, N., & Kechadi, M. (2016, August 24-26). A lightweight software write-blocker for virtual machine forensics. *Proceedings of the 2016 Sixth International Conference on Innovative Computing Technology (INTECH), Dublin, Ireland*, 730-735. <https://doi.org/10.1109/INTECH.2016.7845141>
- Uchiyama, J. J. (2014). *Establishing professional guidelines for SSD forensics: a case study* [Master's thesis, Auckland University of Technology]. Tuwhera. <https://openrepository.aut.ac.nz/handle/10292/7226>
- Varol, A., & Sönmez, Y. Ü. (2017, October 5-8). Review of evidence analysis and reporting phases in digital forensics process. *Proceedings of the 2017 International Conference on Computer Science and Engineering (UBMK), Antalya, Turkey*, 923-928. <https://doi.org/10.1109/UBMK.2017.8093563>
- Vättö, K. (2014). *Upgrading the SSD in Chromebook & MyDigitalSSD Super Boot Drive M.2 2242 SSD Review*. Retrieved June 23, 2022, from <https://www.anandtech.com/show/8543/upgrading-the-ssd-in-a-chromebook>
- Vieyra, J., Scanlon, M., & Le-Khac, N.-A. (2018). Solid state drive forensics: Where do we stand? *Proceedings of the International Conference on Digital Forensics and Cyber Crime*, 149-164. [https://doi.org/10.1007/978-3-030-05487-8\\_8](https://doi.org/10.1007/978-3-030-05487-8_8)
- Wang, K. C. (2018). *Systems Programming in Unix/Linux*. Springer International Publishing.
- Wani, M. A., AlZahrani, A., & Bhat, W. A. (2020). File system anti-forensics – types, techniques and tools. *Computer Fraud & Security*, 2020(3), 14-19. [https://doi.org/10.1016/S1361-3723\(20\)30030-0](https://doi.org/10.1016/S1361-3723(20)30030-0)
- Western Governors University. (2021). *5 Most Popular Operating Systems*. Retrieved May 6, 2022, from <https://www.wgu.edu/blog/5-most-popular-operating-systems1910.html>
- Williams, J. (2011). ACPO Good Practice Guide for Digital Evidence. *Metropolitan Police Service, Association of chief police officers, GB*, 1556-6013.
- Wong, G. (2013). SSD Market Overview. In R. Micheloni, A. Marelli, & K. Eshghi (Eds.), *Inside Solid State Drives (SSDs)* (pp. 1-17). Springer Netherlands. [https://doi.org/10.1007/978-94-007-5146-0\\_1](https://doi.org/10.1007/978-94-007-5146-0_1)

- Wu, F., Zhu, Y., Xiong, Q., Lu, Z., Zhou, Y., Kong, W., & Xie, C. (2018, October 7-10). Characterizing 3D Charge Trap NAND Flash: Observations, Analyses and Applications. *Proceedings of the 2018 IEEE 36th International Conference on Computer Design (ICCD), Orlando, FL, USA*, 381-388. <https://doi.org/10.1109/ICCD.2018.00064>
- Xu, C., Wang, W., Zhou, D., & Xie, T. (2015, September 8-11). An SSD-HDD Integrated Storage Architecture for Write-Once-Read-Once Applications on Clusters. *Proceedings of the 2015 IEEE International Conference on Cluster Computing, Chicago, IL, USA*, 74-77. <https://doi.org/10.1109/CLUSTER.2015.20>
- Yan, W., Wang, X., & Yu, X. (2014, April 26-28). Design and implementation of an efficient flash-based SSD architecture. *Proceedings of the 2014 4th IEEE International Conference on Information Science and Technology, Shenzhen, China*, 79-83. <https://doi.org/10.1109/ICIST.2014.6920336>
- Yusoff, Y., Ismail, R., & Hassan, Z. (2011). Common phases of computer forensics investigation models. *International Journal of Computer Science & Information Technology*, 3(3), 17-31. <https://doi.org/10.5121/ijcsit.2011.3302>
- Zambelli, C., Navarro, G., Sousa, V., Prejbeanu, I. L., & Perniola, L. (2017). Phase Change and Magnetic Memories for Solid-State Drive Applications. *Proceedings of the IEEE*, 105(9), 1790-1811. <https://doi.org/10.1109/JPROC.2017.2710217>
- Zhang, H., Wang, J., Chen, Z., Pan, Y., Lu, Z., & Liu, Z. (2021). An SVM-Based NAND Flash Endurance Prediction Method. *Micromachines*, 12(7), 746. <https://doi.org/10.3390/mi12070746>
- Zuolo, L., Zambelli, C., Micheloni, R., & Olivo, P. (2017). Solid-State Drives: Memory Driven Design Methodologies for Optimal Performance. *Proceedings of the IEEE*, 105(9), 1589-1608. <https://doi.org/10.1109/JPROC.2017.2733621>

## Glossary

2D	Two-dimensional
3D	Three-dimensional
ACPO	Association of Chief Police Officers
AFF	Advanced Forensic Format
ANSI	American National Standards Institute
APFS	Apple File System
ATA	Advanced Technology Attachment
BiCS	Bit Cost Scalable
BL	Bit Line
BSL	Bit Select Line
CMOS	Complementary Metal Oxide Semiconductor
CPU	Central Processing Unit
CT	Charge Trapping
DDR	Double Data Rate
DSL	Drain Select Line
EEPROM	Electrically Erasable Read Only Memory
EV	Erase Verify
FAT	File Allocation Table
FCR	Flash Correct and Refresh
FN	Fowler–Nordheim
FTL	Flash Translation Layer
GAA	Gate-all-around
GB	Gigabyte
GBSD	Gross Bit Storage Density
GIDL	Gate Induced Drain Leakage
HDD	Hard Disk Drive

HPFS	High Performance File System
IBM	International Business Machines Corporation
IDS	Intrusion Detection System
IEEE	Institute of Electrical and Electronics Engineers
IPO	Inter-poly-oxide
ISO	International Organization for Standardization
ISPP	Incremental Step Programming Pulse
JPEG	Joint Photographic Experts Group
KB	Kilobyte
LBA	Logical Block Addressing
LCN	Logical Cluster Number
MFT	Master File Table
MLC	Multi-level Cell
NCQ	Native Command Queuing
NIST	National Institute of Standards and Technology
NPCC	The National Police Chiefs' Council
NTFS	New Technology File System
NVM	Non-volatile Memory
NVMe	Non-volatile Memory Express
NVMHCI	Non-volatile Memory Host Controller Interface
OS	Operating System
PAE	Program-after-erase
P-BiCS	Pipe Shaped Bit Cost Scalable
PCI	Peripheral Component Interconnect
PCIe	Peripheral Component Interconnect Express
PE	Program-erase
POSIX	Portable Operating System Interface
QLC	Quad-level Cell

RAM	Random Access Memory
RBE	Raw Bit Error
ROM	Read Only Memory
SAS	Serial-Attached SCSI
SATA	Serial Advanced Technology Attachment
SCSI	Small Computer System Interface
SL	Source Line
SLC	Single-level Cell
SoC	System-on-a-chip
SP	Service Pack
SRAM	Static Random Access Memory
SSD	Solid State Drive
TB	Terabyte
TCQ	Tagged-Command Queueing
TLC	Triple-Layer Cell
USB	Universal Serial Bus
VCN	Virtual Cluster Number
WL	Word Line

## Appendix A

### Recovered Files

**Figure A.1**

Windows 10 Home With NTFS - 25% Drive Usage – Samsung 870 EVO SSD

No #	File	Size (Original )	Size (Recovered )	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> b9196b35a3d1439f7901e36bbe829ca643b12f92
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 2**

*Windows 10 Home With NTFS - 25% Drive Usage – Kingston A400 SSD*

No#	File	Size (Original)	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 3cee601d43b6233fa35ce36ff5b85639e9260374
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 3**

*Windows 10 Home with NTFS - 25% Drive Usage – Crucial BX500 SSD*

No #	File	Size (Original )	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> f696ba4a393bf269171a52a283d0eb9a47973ee9
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaeccdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5



**Figure A. 4**

Windows 10 Home With NTFS - 25% Drive Usage – Lexar NS 100 SSD

No #	File	Size (Original )	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 8fff4479773c5d2400f7e511700e979ad78822d5
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 5**

Windows 10 Home With NTFS - 25% Drive Usage – Lexar NS 100 SSD - 1

No #	File	Size (Original )	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> f696ba4a393bf269171a52a283d0eb9a47973ee9
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 6**

Windows 10 Home With NTFS - 50% Drive Usage – Samsung 870 EVO SSD

No #	File	Size (Original )	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 5152b069d7042a0b9366fa19e535dc6a86f623e9
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> 78de1d650b336b4655592963fcd16ea7a04e471b

**Figure A. 7**

Windows 10 Home With NTFS - 50% Drive Usage – Kingston A400 SSD

No #	File	Size (Original )	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 3406cdd14dbe70803e4870907423ecabb0f33fa7
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaeccdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 8**

*Windows 10 Home With NTFS - 50% Drive Usage – Crucial BX500 SSD*

No #	File	Size (Original )	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 70c05442c9f2a38e8e78a805bc5664c82261abc6
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 9**

Windows 10 Home With NTFS - 50% Drive Usage – Lexar NS 100 SSD

No #	File	Size (Original )	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 7d39f8a36ce8e01ab98426dff4550106b7448142
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> d2d999902114f927e5c198007cc2093e514127d4

**Figure A. 10**

Windows 10 Home With NTFS - 50% Drive Usage – Lexar NS 100 SSD - 1

No #	File	Size (Original )	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> af3315c176da6e5d34db1d4944ea7faf4141da8c
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 11**

Windows 10 Home With NTFS - 75% Drive Usage – Samsung 870 EVO SSD

No #	File	Size (Original )	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 57f5a9626e00ffafb222c74d4cb073030f0f6da7
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaeccdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5



**Figure A. 12**

Windows 10 Home With NTFS - 75% Drive Usage – Kingston A400 SSD

No #	File	Size (Original )	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 494f8f46fd341e8b4c4f04f1e8c605850b54c85e
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaeccdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 13**

Windows 10 Home With NTFS - 75% Drive Usage – Crucial BX500 SSD

No #	File	Size (Original )	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> dc92ce97fe4e514473fa35e15c382d8ae666d1d5
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaeccdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> 087af3f6446c7de9cf1ae84bb88b14873c507e38

**Figure A. 14**

Windows 10 Home With NTFS - 75% Drive Usage – Lexar NS 100 SSD

No #	File	Size (Original )	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 99778bdb6297d8fb2c3f01c913d7e73609db412e
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaeccdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 15**

Windows 10 Home With NTFS - 75% Drive Usage – Lexar NS 100 SSD - 1

No #	File	Size (Original )	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 6c4582f78e8f1efbe9ffb5e914be3b76fe93a6e6
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaeccdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 16**

*Ubuntu 21.10 With EXT4 - 25% Drive Usage – Samsung 870 EVO SSD*

No #	File	Size (Original )	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 105b5e27f40bf586400f2078713a671a0b9c2a78
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 17**

*Ubuntu 21.10 With EXT4 - 25% Drive Usage – Kingston A400 SSD*

No #	File	Size (Original )	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 9ded19a5944b56edb1477bfb55047a2b41d1964c
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 18**

*Ubuntu 21.10 With EXT4 - 25% Drive Usage – Crucial BX500 SSD*

No #	File	Size (Original )	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 153653bdba001b114bc55d36b703c2ccdbbb75f5
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 19**

*Ubuntu 21.10 With EXT4 - 25% Drive Usage – Lexar NS 100 SSD*

No #	File	Size (Original )	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> fd406f4f39d2ba1387976392f9a0c69400cb3cbf
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 50f80ca8f53814216d913b37a44e1ee96cdc8850
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5



**Figure A. 20**

*Ubuntu 21.10 With EXT4 - 25% Drive Usage – Lexar NS 100 SSD - 1*

No #	File	Size (Original )	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 73914b7a57e2d908b49f6a0f05c564ef2e3b2420
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 21**

*Ubuntu 21.10 With EXT4 - 50% Drive Usage – Samsung 870 EVO SSD*

No #	File	Size (Original )	Size (Recovered )	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 4ed44e2f12a8303e7c72cc3bc5a17973a821d09a
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 22**

*Ubuntu 21.10 With EXT4 - 50% Drive Usage – Kingston A400 SSD*

No#	File	Size (Original)	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 40fba4162de67b23a5cef8855fa324dd3a1c9ca2
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 23**

*Ubuntu 21.10 With EXT4 - 50% Drive Usage – Crucial BX500 SSD*

No #	File	Size (Original )	Size (Recovered )	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 9814f2095a4f9f92e28b7af5e4fa860ee4963be6
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 24**

*Ubuntu 21.10 With EXT4 - 50% Drive Usage – Lexar NS 100 SSD*

No#	File	Size (Original)	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 8df010ea46f39637e579aa5ffb6b169ccf93d495
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 25**

*Ubuntu 21.10 With EXT4 - 50% Drive Usage – Lexar NS 100 SSD - 1*

No#	File	Size (Original)	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> fb522035f5b033efc1c0dd09bf7cebbf566c08db
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaeccdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 26**

Ubuntu 21.10 With EXT4 - 75% Drive Usage – Samsung 870 EVO SSD

No #	File	Size (Original )	Size (Recovered )	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> c0fb54af00b9d76a5b01b23fec510b3c0847989b
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 27**

*Ubuntu 21.10 With EXT4 - 75% Drive Usage – Kingston A400 SSD*

No #	File	Size (Original )	Size (Recovered )	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> b6a692dd25262e569ef7989d8d1bf2bb4633983e
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5



**Figure A. 28**

*Ubuntu 21.10 With EXT4 - 75% Drive Usage – Crucial BX500 SSD*

No #	File	Size (Original )	Size (Recovered )	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> da6256d723ec0e7291e80f325d4f51b574d882de
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 29**

*Ubuntu 21.10 With EXT4 - 75% Drive Usage – Lexar NS 100 SSD*

No#	File	Size (Original)	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> b6a692dd25262e569ef7989d8d1bf2bb4633983e
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 30**

*Ubuntu 21.10 With EXT4 - 75% Drive Usage – Lexar NS 100 SSD - 1*

No #	File	Size (Original )	Size (Recovered )	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 112bad3c8ecbfee1f31e455f6f4a198e511a6145
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 31**

*Apple macOS Catalina With APFS - 25% Drive Usage – Samsung 870 EVO SSD*

No #	File	Size (Original )	Size (Recovered )	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> f696ba4a393bf269171a52a283d0eb9a47973ee9
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 32**

*Apple macOS Catalina With APFS - 25% Drive Usage – Kingston A400 SSD*

No #	File	Size (Original )	Size (Recovered )	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> a847468df2a89acfebf4f2bcec431b5a8585c81e
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 33**

*Apple macOS Catalina With APFS - 25% Drive Usage – Crucial BX500 SSD*

No#	File	Size (Original)	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 3d366c4ac6213b732216b5ba4965421ef6a00cac
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 34**

*Apple macOS Catalina With APFS - 25% Drive Usage – Lexar NS 100 SSD*

No#	File	Size (Original)	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> e8cbe2e572b1c74d5df186453b5321c097ad1f8d
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 35**

*Apple macOS Catalina With APFS - 25% Drive Usage – Lexar NS 100 SSD - 1*

No #	File	Size (Original )	Size (Recovered )	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> d9e7417ee261784b4cf044cca68e9b3f34b96f2f
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5



**Figure A. 36**

*Apple macOS Catalina With APFS - 50% Drive Usage – Samsung 870 EVO SSD*

No #	File	Size (Original )	Size (Recovered )	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> f696ba4a393bf269171a52a283d0eb9a47973ee9
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaeccdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 37**

*Apple macOS Catalina With APFS - 50% Drive Usage – Kingston A400 SSD*

No#	File	Size (Original )	Size (Recovered )	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> aa898c9ab9d19e48084314564c853994933d780c
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 38**

*Apple macOS Catalina With APFS - 50% Drive Usage – Crucial BX500 SSD*

No#	File	Size (Original)	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 3d366c4ac6213b732216b5ba4965421ef6a00cac
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 39**

*Apple macOS Catalina With APFS - 50% Drive Usage – Lexar NS 100 SSD*

No#	File	Size (Original)	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 9e677d9154533b3f21e190fa21b66b380565d701
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 40**

*Apple macOS Catalina With APFS - 50% Drive Usage – Lexar NS 100 SSD - 1*

No #	File	Size (Original )	Size (Recovered )	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> a847468df2a89acfebf4f2bcec431b5a8585c81e
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 41**

*Apple macOS Catalina With APFS - 75% Drive Usage – Samsung 870 EVO SSD*

No#	File	Size (Original)	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> f696ba4a393bf269171a52a283d0eb9a47973ee9
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 42**

*Apple macOS Catalina With APFS - 75% Drive Usage – Kingston A400 SSD*

No#	File	Size (Original)	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> bef55a7c1647448295b3d8a6fb619599bb4f3c84
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 43**

*Apple macOS Catalina With APFS - 75% Drive Usage – Crucial BX500 SSD*

No#	File	Size (Original)	Size (Recovered)	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	1.mp4	35.1MB	35.1MB	<b>Original File:</b> 7bcb29c7c9e18b372622e95917cb339bd0fa76e7
				<b>Recovered File:</b> a735fd181ff7af7d29c7a5f1ef1530075ef903c4
3	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
4	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 3d366c4ac6213b732216b5ba4965421ef6a00cac
5	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
6	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5



**Figure A. 44**

*Apple macOS Catalina With APFS - 75% Drive Usage – Lexar NS 100 SSD*

No #	File	Size (Original )	Size (Recovered )	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> 38bd5f3b9953cd7aa4c2976acaf67944d98c06e3
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaeccdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5

**Figure A. 45**

*Apple macOS Catalina With APFS - 75% Drive Usage – Lexar NS 100 SSD - 1*

No #	File	Size (Original )	Size (Recovered )	Sha1
1	01.zip	1GB	3.06KB	<b>Original File:</b> 90ca840948bde21fbe9777127bae080bc3eee993
				<b>Recovered File:</b> 6dab1d3a3aae0a38db8e5ebe7431241675945524
2	3.jpg	24KB	24KB	<b>Original File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
				<b>Recovered File:</b> a1c5d538af9989aaf9a68f41beee6157b2441338
3	4.doc	68KB	68KB	<b>Original File:</b> 64f38d6f0380e0cf3484ae03532fc7b36a5f8935
				<b>Recovered File:</b> f421e38fa9945cca95111ded6c4872b8e1f61966
4	5.pdf	136KB	136KB	<b>Original File:</b> 03ac97dab3fbb1b72c272eeaecdead22aa2404f
				<b>Recovered File:</b> d9ace7689883af527bed90022243116b883cf7aa
5	6.pdf	548KB	548KB	<b>Original File:</b> 157ac6c978a3c0a774ab8e15ae61455cbc114c45
				<b>Recovered File:</b> fcff17a47aec8d8716d3a57a2397355a714e7dc5