

St. Cloud State University

theRepository at St. Cloud State

---

Culminating Projects in Information Assurance

Department of Information Systems

---

12-2019

## Forensic Aspects of Various Flash Memory Devices

Shivendran Divakar Tiruchanpalli  
tdshivendran94@gmail.com

Follow this and additional works at: [https://repository.stcloudstate.edu/msia\\_etds](https://repository.stcloudstate.edu/msia_etds)

---

### Recommended Citation

Tiruchanpalli, Shivendran Divakar, "Forensic Aspects of Various Flash Memory Devices" (2019).  
*Culminating Projects in Information Assurance*. 95.  
[https://repository.stcloudstate.edu/msia\\_etds/95](https://repository.stcloudstate.edu/msia_etds/95)

This Thesis is brought to you for free and open access by the Department of Information Systems at theRepository at St. Cloud State. It has been accepted for inclusion in Culminating Projects in Information Assurance by an authorized administrator of theRepository at St. Cloud State. For more information, please contact [rswexelbaum@stcloudstate.edu](mailto:rswexelbaum@stcloudstate.edu).

**Forensic Aspects of Various Flash Memory Devices**

by

Shivendran Divakar Tiruchanpalli

A Starred Paper

Submitted to the Graduate Faculty of

St. Cloud State University

in Partial Fulfillment of the Requirements

for the Degree of

Master of Science

in Information Assurance

December, 2019

Starred Paper Committee:  
Mark Schmidt, Chairperson  
Dennis Guster  
Balasubramanian Kasi

### **Abstract**

Flash memory devices provide high storage volume with low power consumption and faster read-write operations when compared to HDD. This makes FLASH memory devices to be considered as an efficient storage unit thus bringing huge demand for the usage of FLASH memory devices. One of the major problems faced by forensic investigators is extracting deleted data from flash memory devices, as some of the flash memory devices prevent extraction of deleted data using the standard forensic techniques. This paper focuses on exploring forensic opportunities for various flash-based memory devices. This is done by a thorough study of physics of flash memory, the development of flash transition layers, and the file systems that support these devices. It then conducts forensic experiments on various types of flash-based storage media and summarizes the results of each media. This paper also tries to explore various practices to be applied on flash storage media thus enabling them to retrieve deleted information with the use of standard forensic techniques.

## Table of Contents

	Page
List of Tables .....	6
List of Images .....	7
Chapter	
I. Introduction .....	11
Introduction.....	11
Problem statement.....	15
Objective of the Study .....	16
Study Questions .....	16
Limitations of Study .....	16
Summary .....	18
II. Background and Literature Review .....	19
Introduction.....	19
Background.....	19
History of Flash Memory.....	19
Physics of Flash Memory.....	20
NOR Flash vs NAND Flash.....	21
Working of NAND Flash.....	22
Flash Endurance and Limitations.....	25
NAND system architecture .....	26
Flash Transition Layer .....	27
Applications of Flash memory.....	29

	4
Literature Review.....	30
Image acquisition.....	30
Remnant Data.....	31
Cannot delete .....	33
Summary .....	34
III. Methodology.....	35
Introduction.....	35
Design of study .....	35
Data Collection Model.....	37
Tools and Techniques .....	41
Hardware and Software Requirements .....	42
Summary .....	43
IV. Data Presentation and Analysis .....	44
Introduction.....	44
Data presentation .....	44
Creation of a case file .....	44
Copying contents into flash devices .....	49
Creating Images part 1 – After copying contents into devices .....	52
Creating Images part 2– After deleting certain contents from devices.....	72
Data analysis .....	78
Analyzing USB_IMG 01 .....	78
Analyzing USB_IMG 02 .....	80

	5
Analyzing SD_CARD_IMG 01 .....	82
Analyzing SD_CARD_IMG 02.....	84
Analyzing SSD_IMG 01 .....	86
Analyzing SSD_IMG 02.....	87
Summary .....	89
V. Results, Conclusions And Recommendations .....	90
Introduction.....	90
Results.....	90
Conclusions.....	94
The TRIM Command.....	95
TRIM on external SSD .....	95
Self-corrosion of SSD.....	97
Future Work.....	98
References.....	99

**List of Tables**

Table	Page
1. Comparison of NOR vs NAND flash .....	14
2. Definition of Terms.....	17
3. Key differences in NAND and NOR flash (R, J, & R, 2015).....	22
4. Flash memory devices used for the experiment.....	37
5. Hardware and software requirements .....	42
6. Number of files associated with each keyword search .....	92
7. Number of hits associated with each keyword search .....	92

## List of Images

Figure	Page
1. Flash chip on a USB Drive (Woodford, 2017) .....	12
2. Digital forensics model (Satti & Jafari, 2015) .....	13
3. Flash memory transistor (Woodford, 2017).....	23
4. Flash transistor holding electrons (Woodford, 2017) .....	24
5. Comparison of NAND flash memory (Rouse, TLC flash (triple-level cell flash), n.d.) .....	25
6. Data storage flow in flash media (Deng & Zhou, 2011).....	27
7. Different types of MTD architecture (Huang, Chang, Kuo, Hsieh, & Lin, 2008).....	28
8. Electrical interface of NAND flash chip (Breeuwsma, Jongh, Klaver, Knijff, & Roeloffs, 2007) .....	30
9. Design flow of the experiment.....	36
10. Sandisk USB flash drive – 8 GB.....	38
11. Kingston SD card - 16 GB .....	39
12. Transcend SSD - 32 GB.....	40
13. Contents of Case Folder.....	45
14. Contents of Emails folder .....	46
15. Contents of Password protected folder .....	47
16. Contents of Passwords folder.....	48
17. Copying contents of Case file into USB drive.....	49
18. Copying contents of Case file into SD card.....	50
19. Copying contents of Case file into SSD .....	51



20. Selecting logical drive for all the devices .....	52
21. Select USB as source for image creation .....	53
22. Dialogue box for adding options to USB image.....	54
23. Dialogue box for selecting the type of image .....	55
24. Providing additional image information for USB.....	56
25. Providing destination and image file name for USB image .....	56
26. Dialogue box before starting image creation for USB.....	57
27. Image creation process for USB .....	58
28. USB image creation completion .....	59
29. Images of USB drive part 1 .....	60
30. Select SD card as source for image creation.....	61
31. Providing additional image information for SD card .....	62
32. Providing destination and image file name for SD card image .....	63
33. Dialogue box before starting image creation for SD card .....	64
34. Image creation process for SD card .....	65
35. SD card image creation completion .....	65
36. Images of SD card part 1 .....	66
37. Select SSD as source for image creation .....	67
38. Providing additional image information for SSD .....	68
39. Providing destination and image file name for SSD image .....	68
40. Dialogue box before starting image creation for SSD .....	69
41. Image creation process for SSD.....	70

42. SSD image creation completion.....	71
43. Images of SSD part 1 .....	72
44. Files that are deleted on each device.....	73
45. Contents of USB drive after deleting.....	74
46. Contents of SD card after deleting.....	74
47. Contents of SSD after deleting .....	75
48. Images of USB drive part 2 .....	76
49. Images of SD card part 2I .....	76
50. Images of SSD part 2 .....	77
51. FTK toolkit processing image file USB_IMG 01.....	79
52. Reading contents of the image file.....	79
53. Search for specific files in USB.....	80
54. FTK toolkit processing image file USB_IMG 02.....	80
55. Search results from USB_IMG 02.....	81
56. Deleted items on USB.....	82
57. FTK toolkit processing image file SD_CARD_IMG 01 .....	83
58. Search results for SD_CARD_IMG 01.....	83
59. FTK toolkit processing image file SD_CARD_IMG 02 .....	84
60. Search results from SD_CARD_IMG 02 .....	84
61. Deleted items on SD card .....	85
62. FTK toolkit processing image file SSD_IMG 01 .....	86
63. Search results from SSD_IMG 01 .....	87

	10
64. FTK toolkit processing image file SSD_IMG 02 .....	87
65. Search results from SSD_IMG 02 .....	88
66. Deleted items on SSD .....	88
67. Structure of the created case file .....	91
68. Structure of case file after deletion .....	91
69. Files results for keyword email in SSD .....	93
70. Files results for keyword email in SD card.....	94
71. Externally connected SSD using SATA-USB device.....	96
72. Performing trim operation on externally connected SSD .....	97

## **Chapter I: Introduction**

### **Introduction**

There has been a tremendous growth in the usage of portable devices which has led to rapid increase in consumer electronics. These portable devices make use of non-volatile storage medium that can save data electrically using semiconductor chips. The data on these chips can be electrically erased and can be programmed several times after it is written and deleted. The semiconductor chip (or transistor) can be integrated at a large scale on a very tiny chip. This allows for huge digital storage capacity on a tiny chip that is physically no bigger than the size of a human nail. These memory chips are known as flash memory and they bring a huge impact in the way the data is stored and retrieved. Compared to the traditional optical storage medium the flash memory devices operate at low power and offer high resistance to shock. Since these devices come in small physical sizes and huge storage space with the capability of rugged usage, it finds its applications in the military to the large-scale consumer usage.

The portable devices like phone, camera, PDA's, etc. has also been used in a criminal activity. Criminal activity has also equally grown with the improvements in the flash devices. Mostly these device uses the memory cards or any flash-based memory device which allow them to store data easily with improved portability and efficiency. For a forensic expert, extracting data from these devices is problematic nowadays. Current forensic methods and analysis do not allow for acquiring data that's present on these devices. This includes recovering the deleted data which might be useful in gathering evidences related to a criminal activity. Acquiring data from the flash devices is only possible by looking at the chip using a microscope and reading the chip at the lowest level like wear levelling and other physical properties of each silicon transistor.



*Figure 1: Flash chip on a USB Drive (Woodford, 2017)*

Flash memory exists in two different flavors, NOR flash and NAND flash. Manufacturing a NOR flash is expensive than manufacturing a NAND flash. NOR flash memory can read byte by byte data in a constant time which enables faster data access. NAND flash memory is comprised of blocks. In a NAND flash, data is stored in regions that is scaled down from a static predefined number of pages called blocks. A typical page size of a NAND flash is 512 bytes. Writing data into the NAND flash is achieved by a WRITE cycle that is injecting necessary data into a buffer one byte at a time (Sansurooah, 2009).

NAND flash devices offer large storage space and low read speed when compared to NOR flash devices. Thus, NOR flash is used primarily to hold and execute firmware. The parts of memory that are not used by firmware, cannot be used to store user information or other data storage. Therefore, most of the mobile storage units like USB, SD card etc. use NAND flash to store huge data in a compact storing medium (Breeuwsma, Jongh, Klaver, Knijff, & Roeloffs, 2007).

Digital forensics deals with the preservation, identification, extraction, documentation, and interpretation of computer data (Kruse & Heiser, 2001). Acquiring, authenticating and analyzing of data is the key functions involved in digital forensics. Data is acquired in a bit by bit copy of the hard drive and ensuring the copy of the acquired data with the help of checksums is called authentication. Analysis of the acquired data is the most important part in digital forensics as they provide the evidence related to the crimes (Bui, Enyeart, & Luong, 2003).

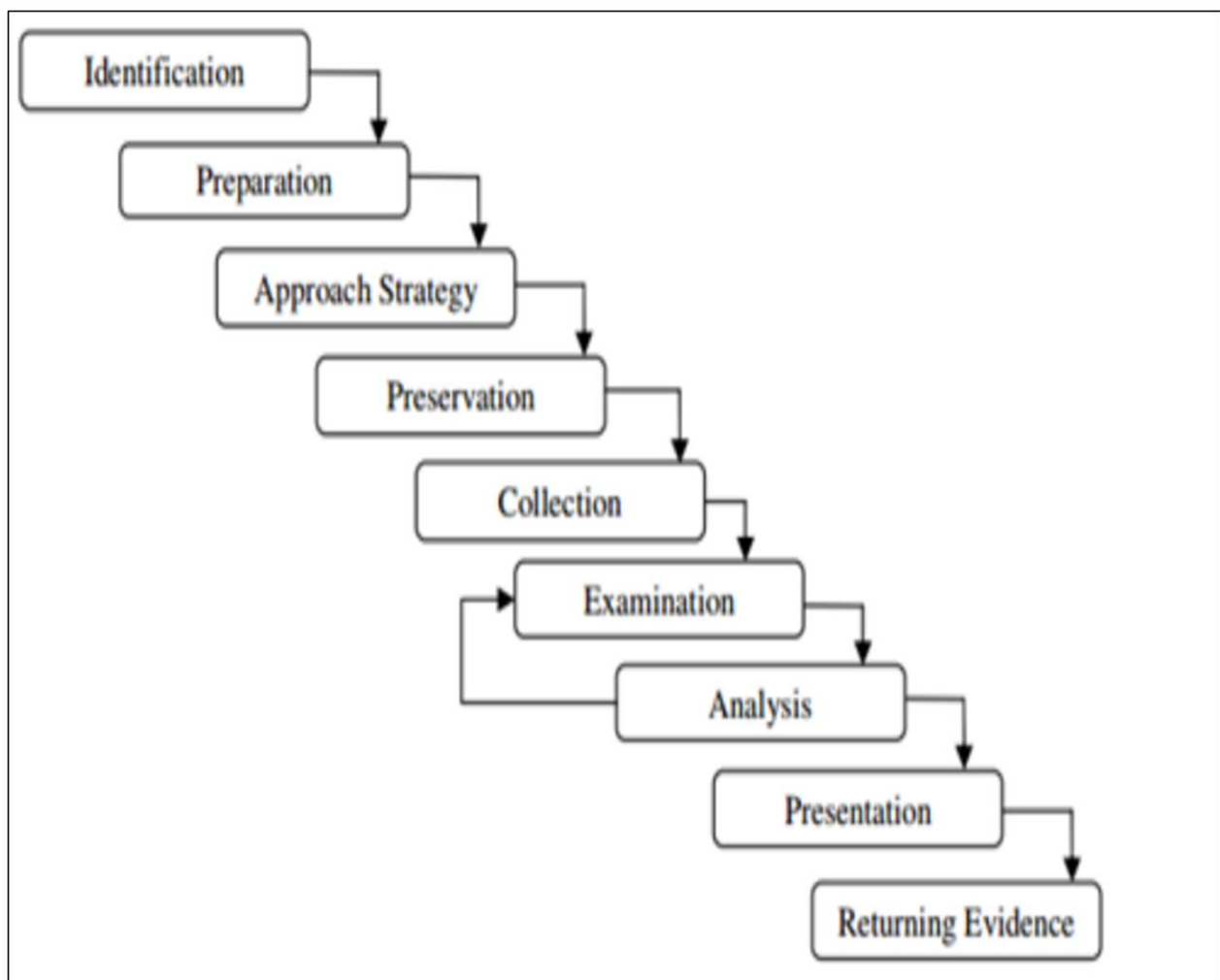


Figure 2: Digital forensics model (Satti & Jafari, 2015)

Table 1: *Comparison of NOR vs NAND flash*

	NAND flash	NOR flash
Advantages	<ul style="list-style-type: none"> <li>• Fast Write</li> <li>• Fast Erase</li> </ul>	<ul style="list-style-type: none"> <li>• Random access</li> <li>• Byte by byte writing</li> </ul>
Disadvantages	<ul style="list-style-type: none"> <li>• Slow random access</li> <li>• Byte writing not possible</li> </ul>	<ul style="list-style-type: none"> <li>• Slow writes</li> <li>• Slow erase</li> </ul>
Applications	<ul style="list-style-type: none"> <li>• Disk applications</li> <li>• Large sequential data applications</li> </ul>	<ul style="list-style-type: none"> <li>• Replacement of EPROM</li> <li>• Direct execution from memory</li> </ul>

In an event of a crime, deleted data can become an important source of an evidence. One of the key roles for any forensic examiner is to look for any remnants of deleted data and investigate if the data is related to the crime. Most of the computer crimes involve deleting important files which a suspect uses for committing crimes. Optical storage medium easily allows for easy recovery of the deleted data when it undergoes a forensic investigation using traditional forensic techniques. (Bui, Enyeart, & Luong, 2003).

Deleting data on a flash memory device causes the data to be completely lost forever and cannot be recoverable. This poses a serious issue to forensic investigators to acquire remnant or deleted data from a flash memory device. With advances in technology and improved data storage techniques, criminals are finding a smarter way to commit crimes. Recent forensic statistics shows that has been a huge increase in the use of flash-based memory devices in the event of a crime (Gubanov & Afonin, Why SSD Drives Destroy Court Evidence, 2012).

Solid state drives, SD cards and USB thumb drive are three different types of storage devices that implements flash memory storage. USB thumb drives and SD cards are typically smaller in size and has low storage capacity when compared to solid state drives. Solid state drives or SSD are typically used as an internal storage drive for a computing system. Where as USB thumb drives and SD cards are used as a plug and play external storage device and are the smallest portable storage devices.

Deleting data on a traditional optical storage device, the data is not actually deleted but it is marked unimportant. New data is overwritten on the existing unimportant data in a traditional optical drive. In the case of solid - state drive, the cell has to be cleared in order to write new data on it. This implies that solid state drives are prone to permanent loss deleted data which is unrecoverable for the forensic investigators.

### **Problem statement**

Flash memory devices has introduced new ways of data storage when compared to traditional optical drives. And with the advancement of flash memory devices, storage space and efficiency has drastically improved thus opening a huge opportunity for flash memory devices to find its place in military and consumer usage. With the increase in the usage of flash memory devices, there is also equal increase in the number of computer crimes in which deleted data acts as a key evidence. Digital forensics plays a key role in helping finding evidences that is related to computer crimes (Gibson & Cohen, 2014). Traditional forensic techniques help in easy recovery of deleted data from a traditional optical drive than a solid-state drive. This is one of the major issues that is faced by the forensic investigators to find deleted data from the flash memory devices. This paper aims at investigating the key reasons that make forensics hard to recover



deleted files on flash memory devices. This paper also investigates how different types of flash-based storage device responds to forensic analysis.

### **Objective of the Study**

The main objective of this study is to investigate the key factors that make flash memory devices useless for finding deleted evidence during a forensic investigation. This study will also compare the results obtained from various flash memory devices upon forensic investigation. This research also aims in bringing recommendations that can bring consistent forensic results on each flash-based device.

### **Study Questions**

The study questions for this research revolves around the forensic investigations on various flash memory devices. What are the various types of flash memory devices? How each device responds to forensic investigation upon recovering deleted data? What are the key factors that are responsible for it? What can be done to obtain consistent forensic results from each device?

### **Limitations of Study**

This research study does not attempt to change the currently existing methods for forensic investigation to extract information from flash memory devices. It only explores the reasons behind why deleted data is unrecoverable in flash memory devices and how each flash memory device responds upon extraction of deleted data.

Table 2: *Definition of Terms*

<i>Digital forensics</i>	Digital forensics or computer forensic science is a branch of forensic science that encompasses with the process of uncovering and interpreting digital data. The main goal of this process is to preserve any evidence in its best original form while performing the investigation in a structural manner by collecting, identifying and validating the digital data that is used for the reconstruction of the past events (Techopedia, n.d.)
<i>Flash memory</i>	Flash memory is a type of non-volatile memory that can erase data in units called blocks. The block on a flash memory chip must be erased before any data is rewritten or programmed into the chip. The data retention of flash memory is extended over a period time whether the device equipped with flash memory is powered on or off (Rouse, The NAND flash, 2015).
<i>Solid state drive</i>	Solid state drive is a type of non-volatile storage device which stores persistent data using solid-state flash memory. These drives are not like traditional hard drives because they do not have any moving parts within them. This drive consists of an array of semiconductor memory that is organized as a disk drive with the use of integrated circuits. A solid-state drive can also be referred to as solid state disk (Rouse & Kranz, SSD, 2016).

## Summary

Flash memory devices are the most efficient and can be easily integrated on circuits for data storage. They occupy less space and offer huge storage capacities thus increasing the use of flash memory on portable devices. With the increasing computer crimes, deleted data plays a major role in finding evidences related to a crime. Digital forensics helps in finding deleted data to be used as an evidence for a criminal incident. However, with the case of flash memory devices, forensic investigators are having a tough time finding deleted data from them. Deleted data can be acquired by looking at each flash chip at a microscopic level and reading the wear leveling of the silicon chip. Sometimes it is almost impossible to recover deleted evidences from the flash memory devices.

Thus, this research paper aims at studying the key factors that makes flash memory devices useless for finding deleted evidence during a forensic investigation and provide related suggestions and provide results obtained from various flash memory devices upon forensic investigation. In next chapter, will discuss the problem in detail and the physics and operation of flash memory devices.

## **Chapter II: Background and Literature Review**

### **Introduction**

To explore the reasons behind limitations that's faced by the forensic investigators to find deleted data in flash devices, we must understand the characteristic features underlying the flash storage, the physics of the flash memory and logical characteristics related to storage of data in flash devices. This chapter discusses gives deep insight into the flash memory device and also dives into some of the previous works that is identical to this research problem.

### **Background**

#### **History of Flash Memory**

Flash storage started as an alternative to storing memory without the application of power to it. NOR flash was first introduced in 1981 by Fujio Masuoka when he patented a NOR flash chip that can hold memory electrically. The first working chip of a NOR flash was developed in 1984. Before the existence of NOR flash, the software that runs the computing resource has to be loaded from the magnetic storage into RAM before it is being executed. This is because RAM did not have the capability to hold data when the power is disconnected. NOR flash overcame this problem of holding the memory even when the power is disconnected. The NOR flash memory made its way into the applications like BIOS and firmware technologies due to its faster read speed and avoided software to be loaded into RAM.

Before the existence of the NAND flash, data like files and software used to be stored on disks that were huge in size and had less storage space. In the year 1990, the NAND flash devices went into the market trying to replace the traditional hard disks (Fulford, 2002). The

NAND flash memory overcame the limitations that was present in hard disk storage by introducing huge storage capacities on a compact chip.

NAND flash had many advantages over EPROM like the small size, low power consumption, and high storage density. Therefore, NAND flash was considered the best choice for non-volatile memory. With the rise in demand for mobile devices, there was an equal demand for the flash memory. The earliest commercial applications of flash memory date back to mid-1990s in which introduced CompactFlash, SmartMedia and multimedia cards developed by Sandisk. By 2000 the flash memory was commercially available as a plug and play media or a removable format portable device. Since 2001 various companies started producing USB flash drives which were an easy to use memory device. From the late 1990s to 2003 the NAND flash market accelerated by a 50% with the flash prices dropping by 30-40% (Burr, et al., 2008).

### **Physics of Flash Memory**

Flash memory is EEPROM (Electrically Erasable Programmable Read Only Memory) type of memory. This memory exists in two states, erased and not – erased. Flash has the potential to retain data even without the presence of power supply which makes it a non-volatile memory storage medium. Floating gate transistors are the key components that are used to build flash memory. This transistor is surrounded completely by an insulating material and is governed with the help of control gate. High energy electrons are injected through the isolating material and the electric isolating property of gate of the transistor traps the electron into the transistor. A trapped electron gives a negative charge to the transistor which is indicated with the logical 0 and the absence of the electron gives positive charge which is indicated as a logical 1. While performing write operations the transistor is programmed from a one to a zero (Regan, 2009).

## **NOR Flash vs NAND Flash**

There are two types of flash cells that are currently available. NAND flash and NOR flash. Each of them differs in the ways of connection of arrays and addressing for the purpose of read and write operations. In NOR flash the cells are connected in parallel and in NAND device, the cells are connected in series. The parallel connection in NOR flash allows for each cell to be individually read or programmed resembling a NOR gate type of connection. NOR flash allows for byte by byte read in constant time. The NAND flash has the cells connected in series which prevents individual cells to be read or programmed. Therefore, a total interconnected series of cells may or may not be programmed in NAND flash at a point of time. A bus is used to access each cell in a NAND flash memory whereas, in a NOR flash, a bus is used for addressing the memory cell for reading and write operations (Bez, Camerlenghi, Modelli, & Visconti, 2003).

NAND flash memory devices are more economical than the NOR flash devices. This is due to the lack of cell level accessibility present in the NAND flash device. This allows for increased density that helps in increasing the economic factors for NAND devices. NAND devices were considered to be the replacement for hard storage disks while NOR devices were considered to be the economic replacement for ROM. The main advantage of the NAND device is that it has faster erase time when compared to NOR flash. Due to the serial connection of cells in NAND flash, there is a multiplexed input/output bus that carries both address and data on the same. Typical buses will have 8 bit or 16-bit width which is small to carry address and data in the same cycle. Therefore, the access of data is done after the first three to five cycles of address. The same input/output bus is used to transfer data after the address is loaded (Breeuwsma, Jongh, Klaver, Knijff, & Roeloffs, 2007).

Table 3: *Key differences in NAND and NOR flash* (R, J, & R, 2015)

	<b>NAND</b>	<b>NOR</b>
<b>Arrangement of the memory cell</b>	Series arrangement of cells	Parallel arrangement of cells
<b>Capacity</b>	Mass data storage	Small code storage
<b>Non-volatile</b>	Yes,	Yes
<b>Interface</b>	I/O	Full memory
<b>Data access</b>	Random	Serial
<b>Access methods</b>	Sequential	Byte level
<b>Page mode access</b>	Yes	No
<b>Characteristics</b>	Fast read, Fast Write Fast erase	Fast read Slow Write Slow erase
<b>Price</b>	Low	High
<b>Life span</b>	$10^5 - 10^6$	$10^4 - 10^5$
<b>Write Cycles</b>	$10^6$	$10^6$

### Working of NAND Flash

Flash memories are made out of floating gate transistors in arrays. These transistors are like MOSFETs with two gates instead of one gate. The transistor consists of n-p-n sandwich with a control gate and a floating gate separated across a semiconductor oxide layer which is fully isolated and does not allow for the flow of current across both the gates.

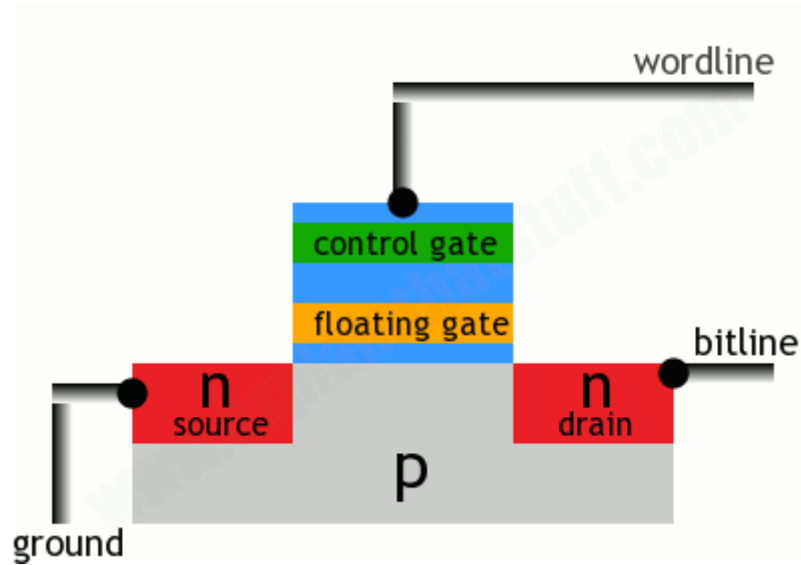


Figure 3: Flash memory transistor (Woodford, 2017)

Figure 3 shows the basic flash memory transistor in an off state that has three terminals namely word line also known as a drain, ground also known as source and bit line. Word line is connected to the control gate which allows for the holding of charges at the floating gate. In this state, there is no electrons present at the floating gate.

While performing the write operation, a positive voltage is applied at the word line and bit line. This makes the electrons to be pulled from the source to drain. Some of the high energy electrons try to pass through the oxide layer and is held at the floating gate.

Figure 2 shows the electrons present at the floating gate. The presence of electrons at the floating gate makes the transistor store a logical 1. Even when the positive voltages are removed at the bit line and worldline, electrons will stay indefinitely at the floating gate. In order to erase the data stored the electrons should be flushed out at the floating gate. To flush the electrons, a negative voltage is applied at the word line which repels the electrons out of the floating gate thus clearing the transistor data to store a zero again (Woodford, 2017).



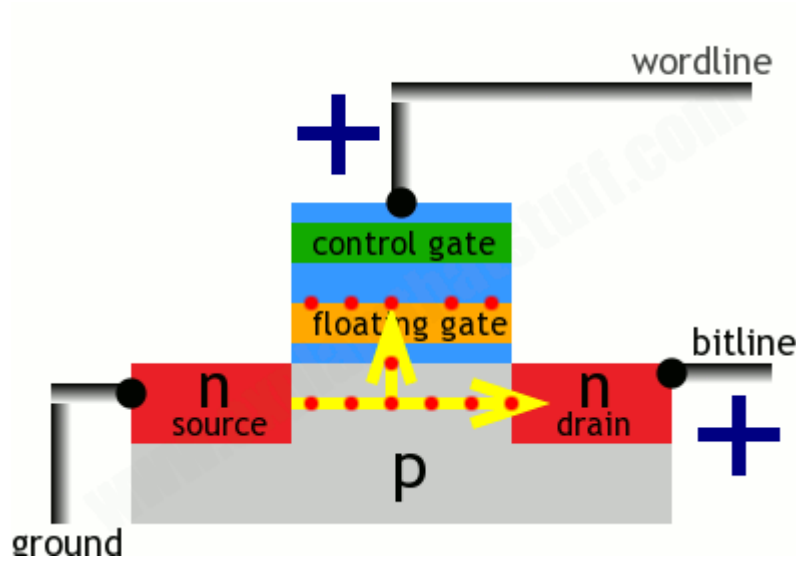


Figure 4: Flash transistor holding electrons (Woodford, 2017)

The above working model is so called a SLC (Single level cell) flash since it can store single bit 0 or 1 per cell. If the flash has the capability to store multiple bits in a single cell, then it is called MLC (Multi level cell). Flash devices having the capability to store 3 bits in a single cell then the flash is referred to as TLC (Triple level cell). Each type of flash cells has its own advantages and disadvantages. But TLCs are the only flash cells that are cheapest to manufacture among the other types of flash cells. This makes TLC to have its applications consumer storage devices. TLCs are mostly found in SSD which is otherwise called solid state drives, which is a hard drive with flash memory units instead of a memory disk (Audrey, 2015).

## NAND flash memory comparison

TYPE	DESCRIPTION	ENDURANCE (PROGRAM/ERASE CYCLES)	
		PLANAR/2D NAND	3D NAND
Single-level cell (SLC)	Stores one bit per cell and two levels of charge	50,000 to 100,000	Not manufactured
Multi-level cell (MLC)	MLC commonly stores two bits per cell and four levels of charge, although theoretically, MLC can store multiple bits per cell and multiple levels of charge	MLC: 3,000 Enterprise-grade MLC (eMLC): 10,000	30,000 to 35,000
Triple-level cell (TLC)	Stores three bits per cell and eight levels of charge	300 to 1,000	1,500 to 3,000
Quadruple-level cell (QLC)	Stores four bits per cell and 16 levels of charge	Not manufactured	150 to 1,000


©2017 TECHTARGET. ALL RIGHTS RESERVED. 

Figure 5: Comparison of NAND flash memory (Rouse, TLC flash (triple-level cell flash), n.d.)

### Flash Endurance and Limitations

The number of write erase cycles on flash is limited and ranges. The number of write erase cycles of a typical flash ranges from  $10^4$  to  $10^6$  times. This limitation of the flash is known as endurance (Regan, 2009). On a typical flash memory, write erase cycle cause flash memory to wear out, which decreases the lifetime of the flash. The wear mechanism happens because the tunnel oxide layer present at the floating gate degrades upon each write erase cycle (Poole, n.d.). Typically, SLC flash has greater write endurance when compared to MLC or TLC flash. When compared with NAND flash, the NOR flash has a higher endurance (Gal & Toledo, 2005).

To overcome this limitation and to increase the lifetime of flash devices, manufacturers have come up with a technique called wear levelling scheme which makes the wear even across all the flash cells. This technique will not improve the lifetime of a single flash cell rather it tries to write across all the cells thus achieving even wear across all the cells in a flash memory and improving the lifetime of the entire flash memory. Once a cell is no longer useful to be written data, the flash cell is permanently marked as a bad cell. According to (Breeuwsma, Jongh, Klaver, Knijff, & Roeloffs, 2007), 2% of the NAND flash memory devices that are shipped will already contain some bad cells in it.

### **NAND system architecture**

NAND flash chips are comprised of banks, pages and blocks. Erase operations on a NAND flash is performed at the block level which is comprised of fixed number of pages. Read and write operations on a NAND flash are performed at the page level. Whenever a data is written into page, the data is termed “live” until the page is erased and written with new data. Each page can write data only once. Over writing of data is not possible on pages. Erased data is considered as “dead”. Storage of live data makes the page valid and the pages are called “valid pages”. Dead data in a page marks the page as invalid. When the count of free pages falls below a minimum amount, the invalid pages undergoes a erase cycle to create more free pages (Huang, Chang, Kuo, Hsieh, & Lin, 2008).

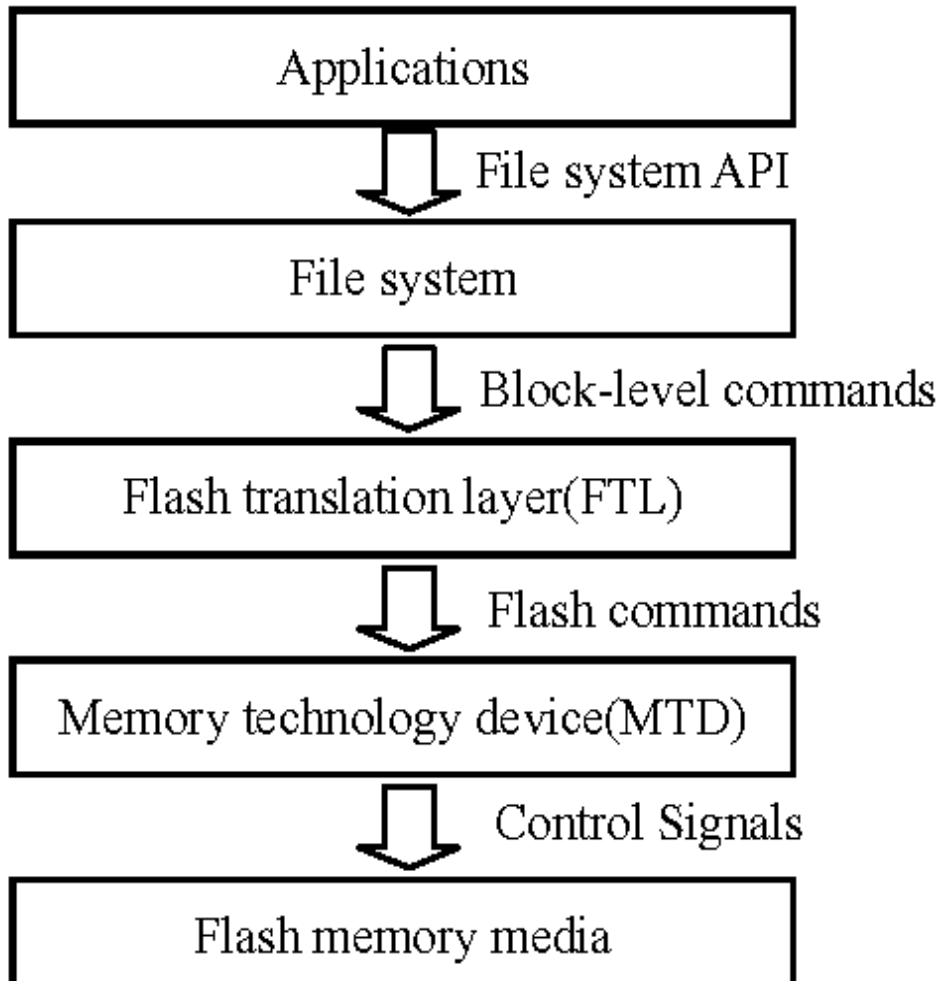


Figure 6: Data storage flow in flash media (Deng & Zhou, 2011)

### Flash Transition Layer

Flash transition layer or FTL is a driver that was introduced to act as an interface between the systems and the flash device. This introduces protocols that enable the interaction between NAND flash and other computing resources like operating systems, file systems, and embedded applications. FTL driver imitates the flash device as a block and provides functions like address translation and garbage collection to the operating system. MTD or memory technology device is a driver that is responsible for providing functions like read, write, and erase

over the flash storage. The combination of MTD and FTL gave rise to two different types of flash devices.

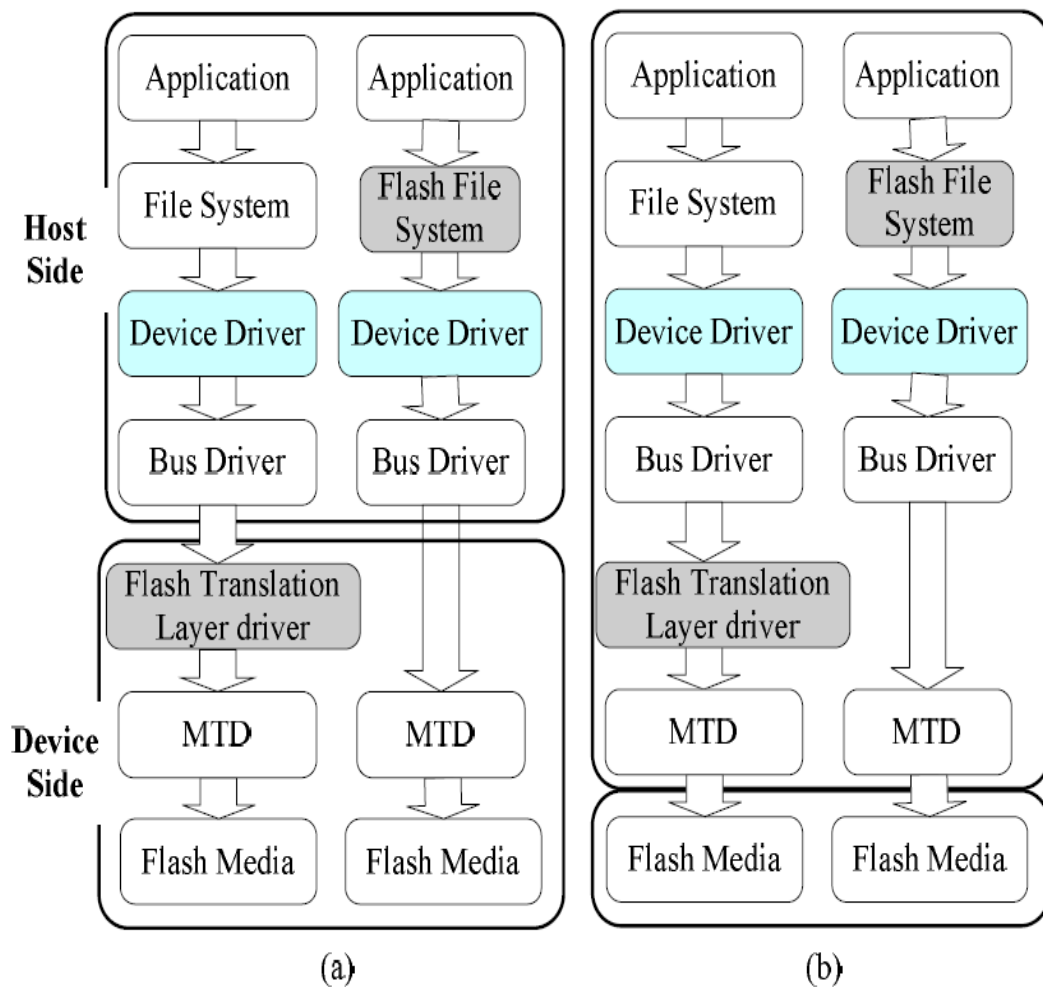


Figure 7: Different types of MTD architecture (Huang, Chang, Kuo, Hsieh, & Lin, 2008)

Flash devices like USB integrates both MTD and FTL as a single package as shown in Figure 7 (a). Figure 7(b) refers to the second type of architecture in which the MTD is not included with the flash memory device.

## **Applications of Flash memory**

USB drives: USB drives were introduced in 2002 to offer high capacity storage with fast transfer rates in a small package with the advantages of flexibility and mobility. They are also built with hardware encryption and built in password protection tools to make them even more secure. When compare to floppy drives or disk drives, USB drives offer high storage capacity and a fast data transfer rate with the help of a USB interface (Kay, 2010).

Memory cards: Introduced in 1994, these devices come with the size of postal stamp size, with higher capacity storage and fast transfer speeds. These devices are available as miniSD and microSD cards and they find their applications in providing storage for mobile devices, cameras, PDAs etc. (Kay, 2010)

Solid State drives: Solid state drives are the newest form fo flash devices that are used for the replacement of the hard drive storage on a computer. These drives have no moving parts and are quieter and smaller when compared to the traditional hard drives that are used on computers. They offer a huge storage capacity and comes at a lower price (Kay, 2010).

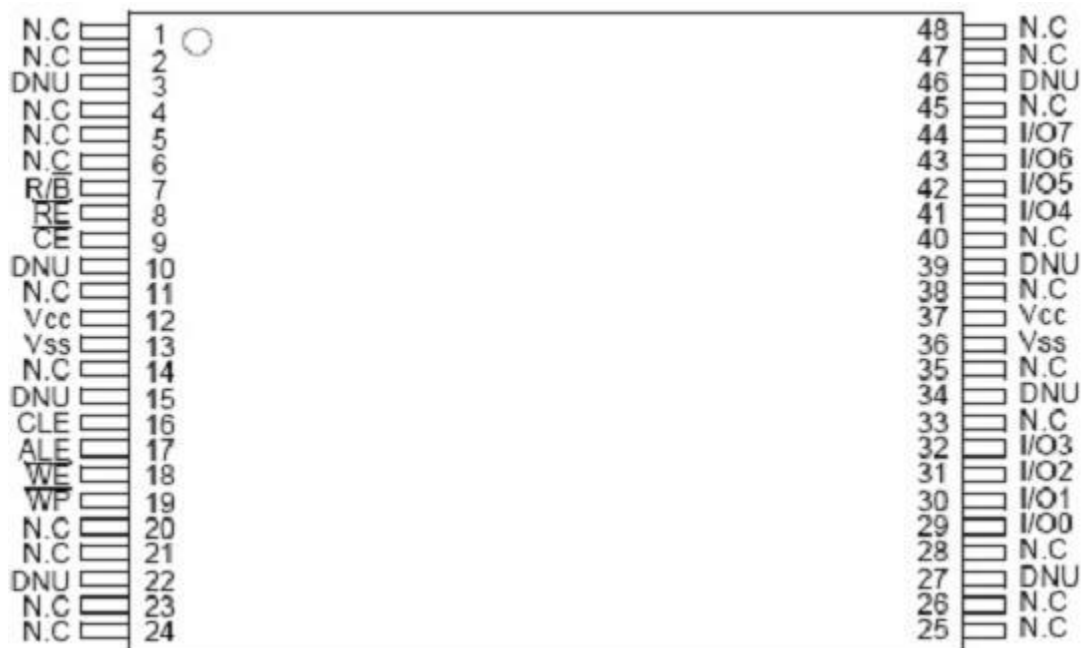


Figure 8 : Electrical interface of NAND flash chip (*Breeuwsma, Jongh, Klaver, Knijff, & Roeloffs, 2007*)

### Literature Review

Several methods to identify or recover data from flash devices are categorized as works to acquire data from logical and physical images. Most of these works discuss the attempts made to acquire data by recognizing the peculiarities of the flash memory. These peculiarities include wear leveling and the way levelling impacts data retention in the flash devices.

### Image acquisition

In the paper “An overall assessment of the mobile internal acquisition tool”, the author uses MIAT (Mobile Internal acquisition tool) as a tool to extract the data stored on Symbian and Windows based smart phones using the internal memory slot. This paper describes the logical acquisition of data by using the operating system of the mobile device. Here the author tries to achieve a method of data acquisition with minimum changes to data that is independent of the many cable interfaces that

is used by the smart phone manufactures. This tool also effectively allows for parallel acquisition using open source tools. This tool copies the root files and directories at a point of time and creates a hash value for each file copied. This tool acquires the logical system files, database entries but this tool lacks the property to acquire deleted data from the memory. The drawbacks of this tool is it does not recover deleted data, and complete data integrity is not guaranteed (Me & Distefano, 2008).

In the paper “Analysis of USB flash devices in a Virtual environment,” the authors discuss the various advantages and the repercussions of using a virtual machine for the analysis of contents on a USB flash device that is obtained for a forensic investigation. The paper does not discuss the properties of the flash file system, erase functionalities and the wear levelling of the flash devices. This paper describes the logical image acquisition of the flash drive using the FTK imager software through a dd function and then the paper proposes a situation in which a forensic investigator would mount the image file and search for the evidences without considering the integrity of the dd file. This paper describes the methodology to acquire data that is like acquiring an image from the disk drive (Bem & Huebner, 2007).

### **Remnant Data**

In the paper “Integrated approach to recovering deleted files from NAND devices.” A methodology is proposed by authors to recover deleted data from the NAND devices using metadata of the recovered file. This paper does not focus on obtaining the physical image, but does a FAT rebuilding process which builds a version table containing all the available versions of the sectors. This paper discusses the process of recovering files by analyzing the File allocation table. This helps in developing the construction of corrupted files by using the different versions of the same sectors and filling the missing sectors using null place holders thus enabling the corrupted files to load. The authors also make the use of Volume Boot Record which helps in the rebuilding process of the FAT



versions. Fragmentation of the flash memory is also discussed by the authors in this paper. Files on flash will become fragmented, and the File allocation table is left unfragmented and there is logical level recovery that is performed. This methodology proposes rebuilding of the files but it is not flash specific (Luck & Stokes, 2008).

Remnant data is the data that can be recovered from a storage media when new information is written over old data. Extracting remnant data from the cells that have been erased was introduced by Sergei Skorobogatov in the paper, “Data remanence in flash memory devices”. The remnant data is often associated in disk type storage or magnetic storage. This data is different from residual data which is the data that is left unintentionally in the computer system. In this paper, the author provides example target devices like smart cards or microcontrollers. Here the author does not target the NOR flash that is used for the booting the hardware. If the chip is password protected, the operating system of the chip destroys the data that is present on the chip before the new data is written on it. So, this will destroy the passwords and that the new code does not gain access to the passwords that is present on the chip. The author also talks about one of the features of the flash chip, which reduces the lifetime of the flash memory. This happens due to the electrons in the cell that is gradually accumulated in the writing process and cannot be released while erasing (Skorobogatov, 2005).

In the paper “A study of Information Privacy and Data Sanitization Problems”, the author discusses the privacy breach when the data present in the storage media is not sanitized upon the disposal of the device. This paper is based on the Department of Defense standard for sanitizing the flash EPROM. In this paper, the author also provides a brief overview of the sanitization tools and techniques along with the standards. Most of the paper tries to focus on hard drives, the author also describes an overview of flash drives and flash devices. A tool that is readily

available for sanitization is discussed. The author makes suggestions to erase the entire chip and write the entire chip with random characters which is a DoD standard sanitization (Roubos, et al., 2007).

Data recovery from USB flash devices is discussed in the paper “Recovering data from USB flash memory sticks that are damaged or electronically erased”. The authors of the paper discuss a series of experiments that first tries to physically destroy or damage the flash stick and then try to recover the data that is previously stored on the device. In these two methods are described to recover the data. One method is to connect the device to a computer and the other method is connecting the flash memory chip with a microcontroller (J., D., & R., 2008).

This experiment is performed by first saving text file and audio files in a number of flash devices. Then the flash devices are subjected to application of high voltage at the lies of the USB stick using a car battery, inducing corrosion in the flash memory by soaking the device in water, creating a short circuit to the flash devices, destroying the flash drive using petrol, stomping the device, striking with a hammer, shooting the device with a pistol, and cooking the device inside a microwave oven. The authors were able to successfully recover the files form several devices that is subjected to high voltage, stomped, and soaked in water. Data recovery was not possible from other devices. This paper successfully demonstrated the experiment of recovering data but did not measure the amount of damage that was necessary to make the device unusable (J., D., & R., 2008).

### **Cannot delete**

The currently available flash memory devices limit the number of write operations that is performed on the device. After certain write operations, the flash storage sectors wear off which and become permanently unusable. Wear levelling techniques are employed to overcome this issue. This technique allows to write into a different block of data instead of reusing or

modifying the used data blocks. This mechanism helps in scattering the data all over the memory chip. This improves the life span of the flash memory chips. Present flash device manufactures design the chips to hold 25% more data than the actual capacity of the flash chip. This additional capacity is not addressable nor can be accessed by the operating system or any other hardware devices. The contents in the additional storage cannot be wiped out by traditional means. This does not ensure the cleaning of data securely.

To resolve this issue, the implementation of ATA ANSI specification enabled a secure destruction of data being held in flash chips. This ensures the entire contents of the chip is wiped out at the hardware level using a secure erase SE command. Software tools with secure wipe option will try to over write the existing data with random data which is cryptographically secure. These types of tools are restricted to access the full storage capacity of the solid state device. (Brant, 2018)

### **Summary**

Flash devices are manufactured in two different flavors. NAND flash and NOR flash. Each type of the flash device has its own limitations and applications. NOR flash memory had its applications into firmware and other operating system related software due to its fast-read speed when compared to the NAND flash. Commercially NAND flash storage is the most popular option for storing huge amounts of data on a very tiny chip. And with the advancement of the technology, these devices had a huge growth in usage due to its improved efficiency and ruggedness. The architecture of NAND flash allows for data to be lost for ever and make them unrecoverable. This created a huge issue to forensic investigators to extract deleted evidences from NAND flash devices.

## **Chapter III: Methodology**

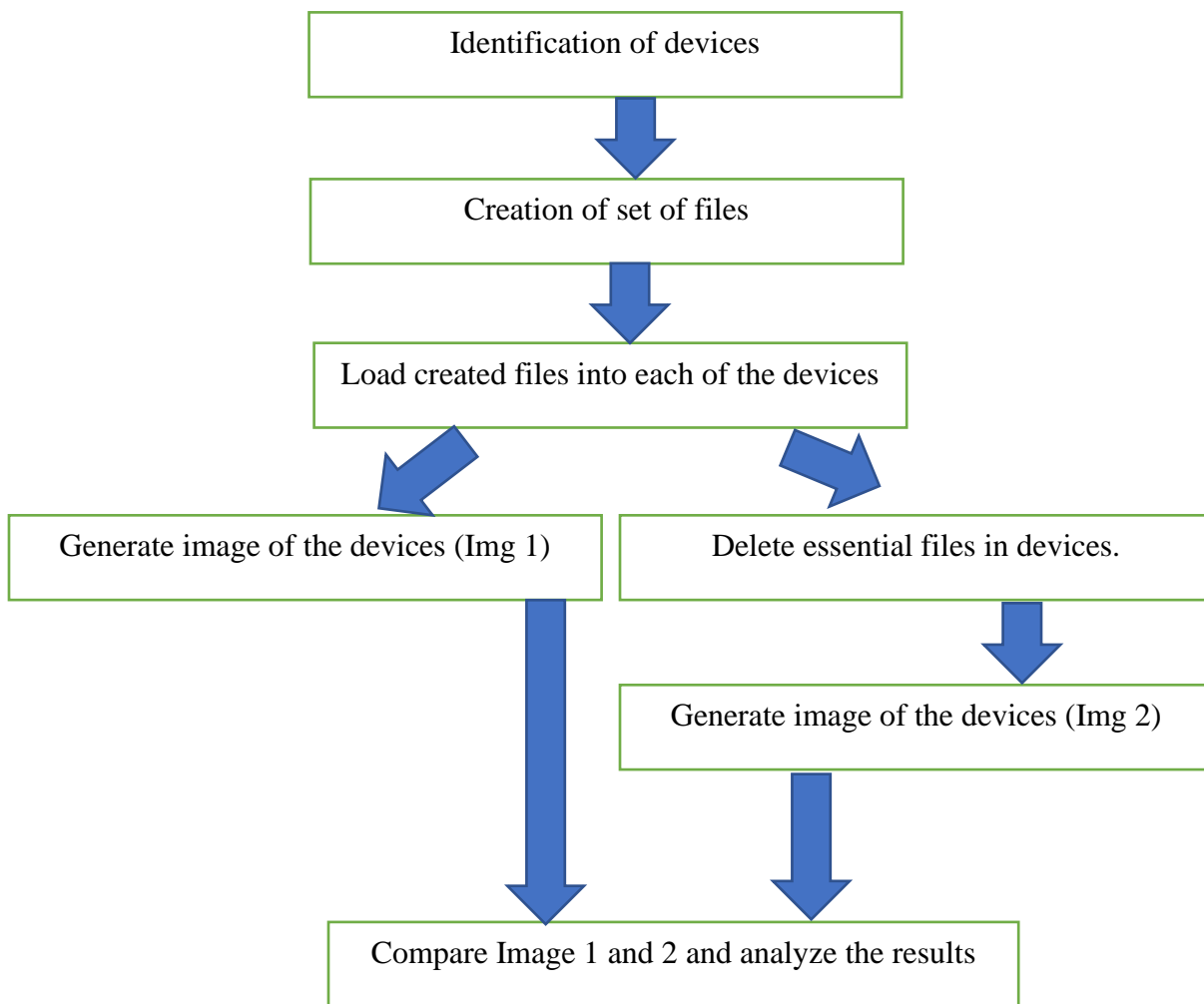
### **Introduction**

Finding deleted evidence in flash memory devices has become a serious challenge to the forensic investigators. The goal of this paper is to explore the reasons behind the challenges faced by forensics to extract deleted evidence in the flash memory devices. To achieve this goal, an experiment is performed on different types of flash memory devices. This involves performing forensic analysis on each of the different types of devices. This chapter discusses about the methods and the steps that are taken to perform this experiment. In addition, this chapter also discusses about the hardware and software requirements, tools that are essential for the experiment and the data collection model that will be best suited for achieving insightful results. Before the methodology is discussed, let's dive into how this study is designed and implemented.

### **Design of study**

The main objective of this study is to perform forensic analysis on different types of flash devices that are commercially available and to compare and study the results obtained from each of these devices. Initially some of the flash memory devices that are commercially available are identified that will be best suited for this experiment. One device from each type of flash memory will be used and supporting file systems will be investigated. A set of files and folders will be created for the sake of this experiment and will be used only to perform the study. The next step will be loading the created files into each of the flash memory device and an image of the device is created using a forensic imager tool and each device will be subjected to forensic analysis. Once the image is created and extracted, some of the essential files will be deleted on

each of the device. Each of the devices will be imaged and subjected to forensic analysis after performing the deletion of the data. Comparing each of these analysis before and after deletion will bring insightful information and understand the key factors that brings the challenges faced by the forensic investigators.



*Figure 9: Design flow of the experiment*

### Data Collection Model

The study focuses on analyzing the forensic results obtained from different types of flash memory devices. The three types of flash devices that are commercially available are USB thumb drive, SD memory card and a solid-state hard drive. Each of these device specifications like storage capacity and supported file system are presented in table 4.

Table 4: *Flash memory devices used for the experiment*

Type	USB flash Drive	SD card	Solid State Drive
Model / Make	SanDisk Cruzer Blade	Kingston Canvas	Transcend
Capacity	8 GB	16 GB	32 GB
Read Speed	15 MB/s	80 MB/s	560 MB/s
Write Speed	10 MB/s	10 MB/s	460 MB/s
Connector	USB Type-A	Push connector	SATA 3
File Systems Supported	NTFS, FAT, FAT32, exFAT.	FAT 16, FAT 32	NTFS, FAT32, exFAT



*Figure 10* : Sandisk USB flash drive – 8 GB



*Figure 11:* Kingston SD card - 16 GB





*Figure 12:* Transcend SSD - 32 GB

Before performing this experiment, the devices are inspected to check if there is any data is present in them. Any unwanted or existing data will be backed up and wiped out from the devices. A set of dummy files and folders will be created and copied on each of the device. Each of the device will have the exact same copy of the dummy files that are created. After copying the data in the devices, an image of each of the device is extracted. The image files will be saved as IMG\_101.iso, IMG\_102.iso and IMG\_103.iso obtained from USB, SD card and solid-state drive respectively. Some of the created files will be deleted on each of the device. Same copy of data is ensured on each of the devices. Once again, an image is extracted from each of the device

and saved as IMG\_201.iso, IMG\_202.iso and IMG\_203.iso. This process is repeated after formatting each of the drives. The image extracted from each device after formatting will be saved as IMG\_301.iso, IMG\_302.iso and IMG\_303.iso. All the images will be analyzed using a forensic analyzer tool and compared to the initial image of each of the device. This will be useful in gathering and analyzing the forensic response from each of the devices and that can draw conclusions to why forensics cannot reveal any deleted data from flash memory devices

### **Tools and Techniques**

Digital forensic analysis plays a major role in gathering and analyzing evidences in an event of a computer crime. This study tries to expose the reasons behind the limitation faced by the forensic analysis to extract evidences from flash memory devices. To perform this study, an experiment is conducted based on forensic analysis of different types of flash devices. FTK toolkit is used as a forensic investigation software for this experiment. FTK toolkit is a forensic software that is created by access data. This software performs image creation and looks for detailed information in the drive, it first obtains the image of the drive and then analyzes them for the required files that are useful for finding the evidences. This imaging program is a standalone application that is called as FTK imager which is a simple tool for creating the image of the storage media. FTK imager can extract image from a logical drive, physical drive and can also perform folder level analysis. This software creates the image that can be used by the forensic tool kit software to perform byte by byte analysis and gather any evidences if present on the storage media.

## Hardware and Software Requirements

FTK is the primary software that is used to perform this research. The other details regarding the software and the hardware requirements are clearly presented in the table 5. They are used to test the effects of deleted data on the flash memory devices.

Table 5: *Hardware and software requirements*

Hardware requirements: <ol style="list-style-type: none"><li>1. Laptop – HP envy m6 laptop with core i5 6200u processor.</li><li>2. USB – Sandisk cruzer blade 16GB with USB interface</li><li>3. SSD – Lexar 512GB hard drive with USB interface</li><li>4. SD card – SanDisk ultra 16GB</li></ol>
Software requirements: <ol style="list-style-type: none"><li>1. Operating system – Windows 10 running on HP envy laptop</li><li>2. FTK toolkit compactible on windows machine</li><li>3. FTK imager compactible on windows machine</li></ol>

Computer forensic experts only try to retrieve information that exists on the device when it is powered down. This information is related to as “persistent data”. In-depth inspection of computer memory storage can reveal any important information that can be presented as a proof of evidence in a court of law. But with the usage of flash devices, retrieving in depth information has become one of the major challenges for the forensic investigators. To study the cause for this problem, an forensic experiment is conducted on flash memory devices, and necessary

conclusions are drawn from its results. This chapter digs deep into how the data is collected from the experiment and analyzed to gain insights from them.

### **Summary**

To investigate the key factors that prevents deleted data to be extracted as s evidence in an event of a crime, a forensic experiment is conducted to test the effects of deleted data on flash storage. The devices for the experiment are identified and dummy data that resembles a criminal case used to perform this experiment. The case data is loaded on to each of the device and forensic analysis is performed on them. FTK tool kit is one of the primary software that is used in the experiment. FTK imager is the software that is used to extract images from the devices before it is used for analysis. Image extraction is performed prior and after deletion of data on each of the devices. The extracted images are compared and analyzed for extracting insights them.

## **Chapter IV: Data presentation and Analysis**

### **Introduction**

Computer forensic experts only try to retrieve information that exists on the device when it is powered down. This information is related to as “persistent data”. In-depth inspection of computer memory storage can reveal any important information that can be presented as a proof of evidence in a court of law. But with the usage of flash devices, retrieving in depth information has become one of the major challenges for the forensic investigators. To study the cause for this problem, an forensic experiment is conducted on flash memory devices, and necessary conclusions are drawn from its results. This chapter explores into how the data is collected from the experiment and analyzed to gain insights from them.

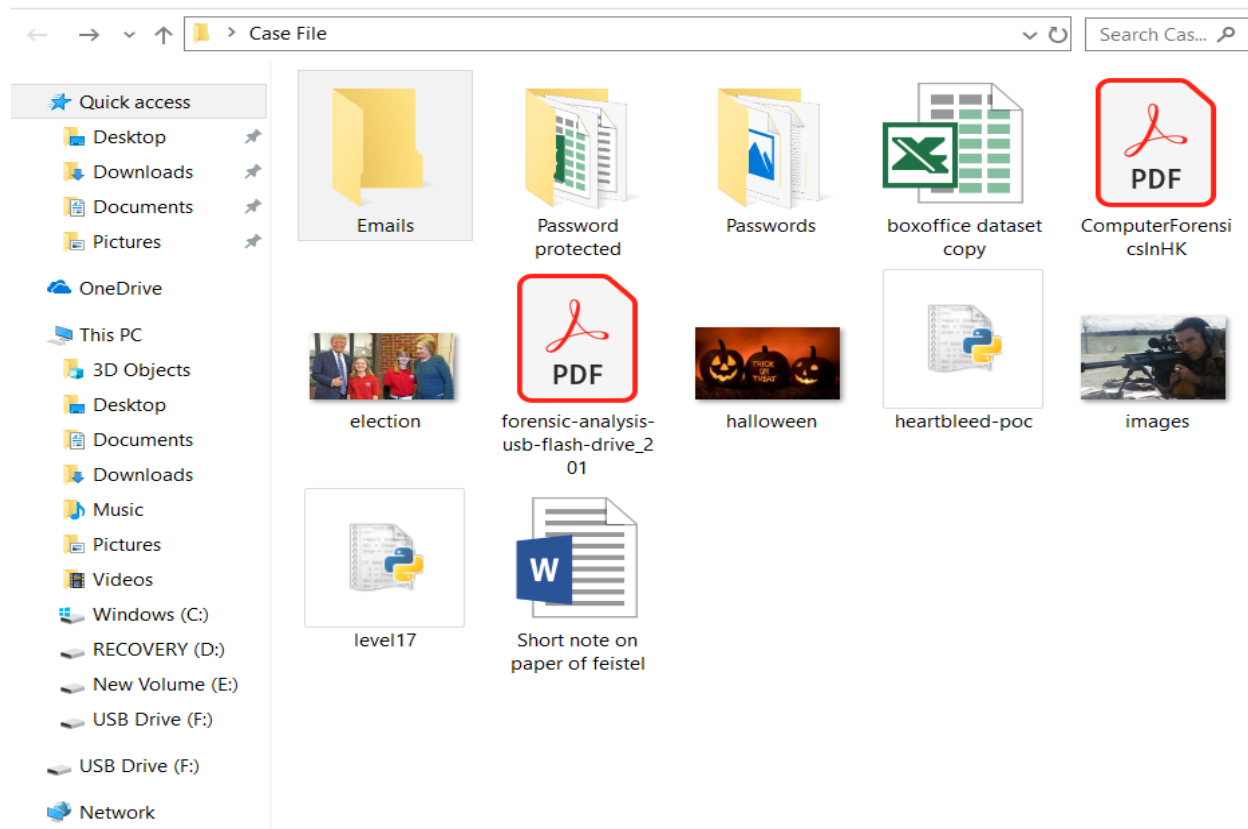
### **Data presentation**

For the purpose of this experiment, some data is created that consists of different types of files. These files include images, pdf, word documents, etc. that holds some dummy data which resembles like a case. Some of these files are marked hidden to see if it is revealed in any of the devices when it is undergone a forensic investigation.

#### **Creation of a case file**

The dummy data is created in a folder called Case Data and it is made to resemble a simple case. The contents of the folder include emails, passwords, password protected files, few images and some pdf files. The primary files that are related to the case are ‘emails’ folder, ‘passwords’ folder and ‘password protected’ folder. These folders are named explicitly to reveal what type of data it holds for easy identification of evidences instead of concealing them with

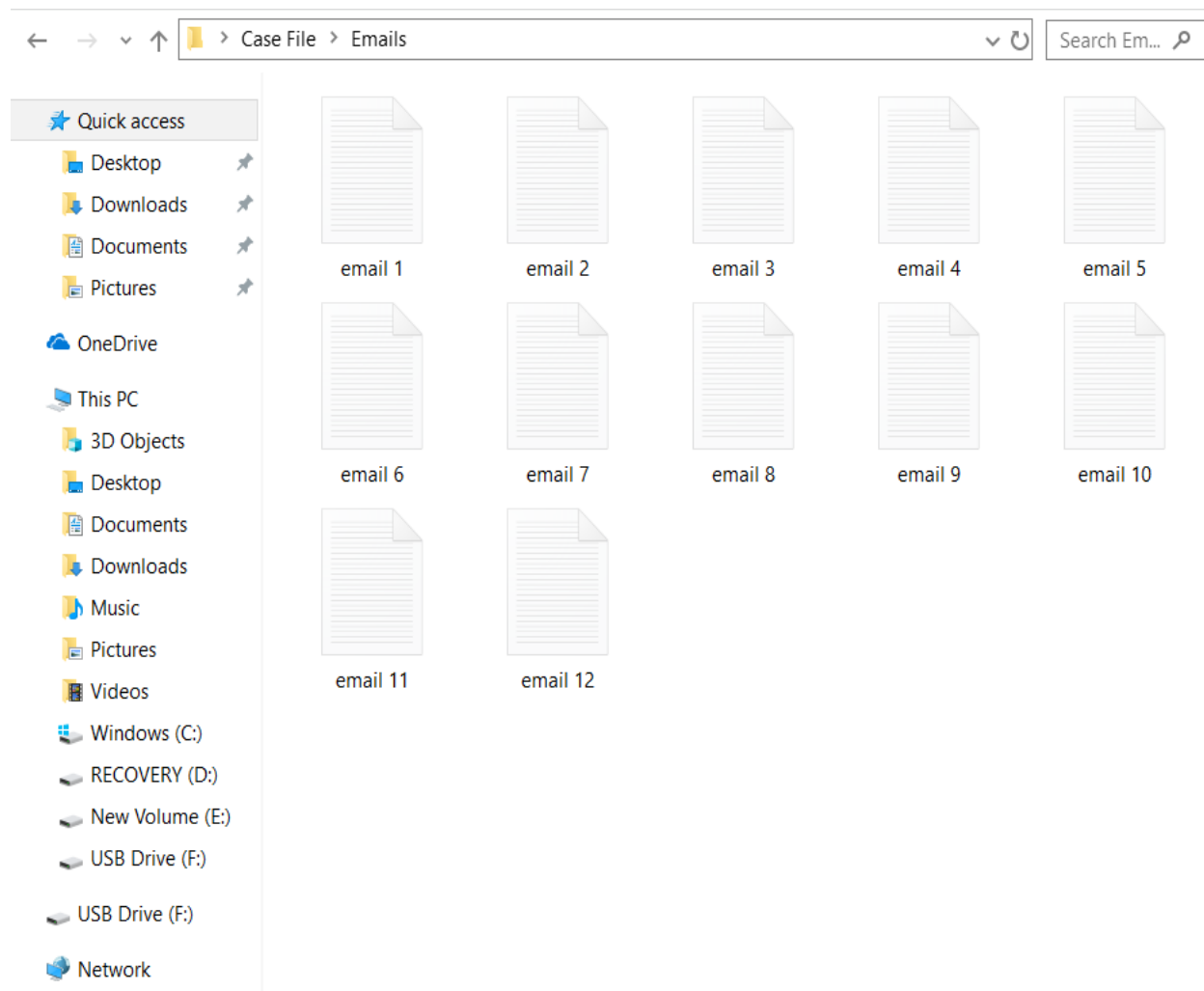
abstract names. The other files present in the case file are not useful for the case and remains as a dummy data for the experiment.



*Figure 13: Contents of Case Folder*

The case folder consists of three folders that is directly related to the case and other files which acts as a dummy data that is not related to the case. The other files consist of one excel file, two pdf files, two jpeg image files, a word document and a python script. These are just randomly created files with no resemblance to the case. The three folders inside the case file are directly related to the case and are named according to the type of files they hold inside them. For example, Emails folder contains emails text files, Password protected folder contains files that are protected using a password and passwords folder consists of files that hold the passwords

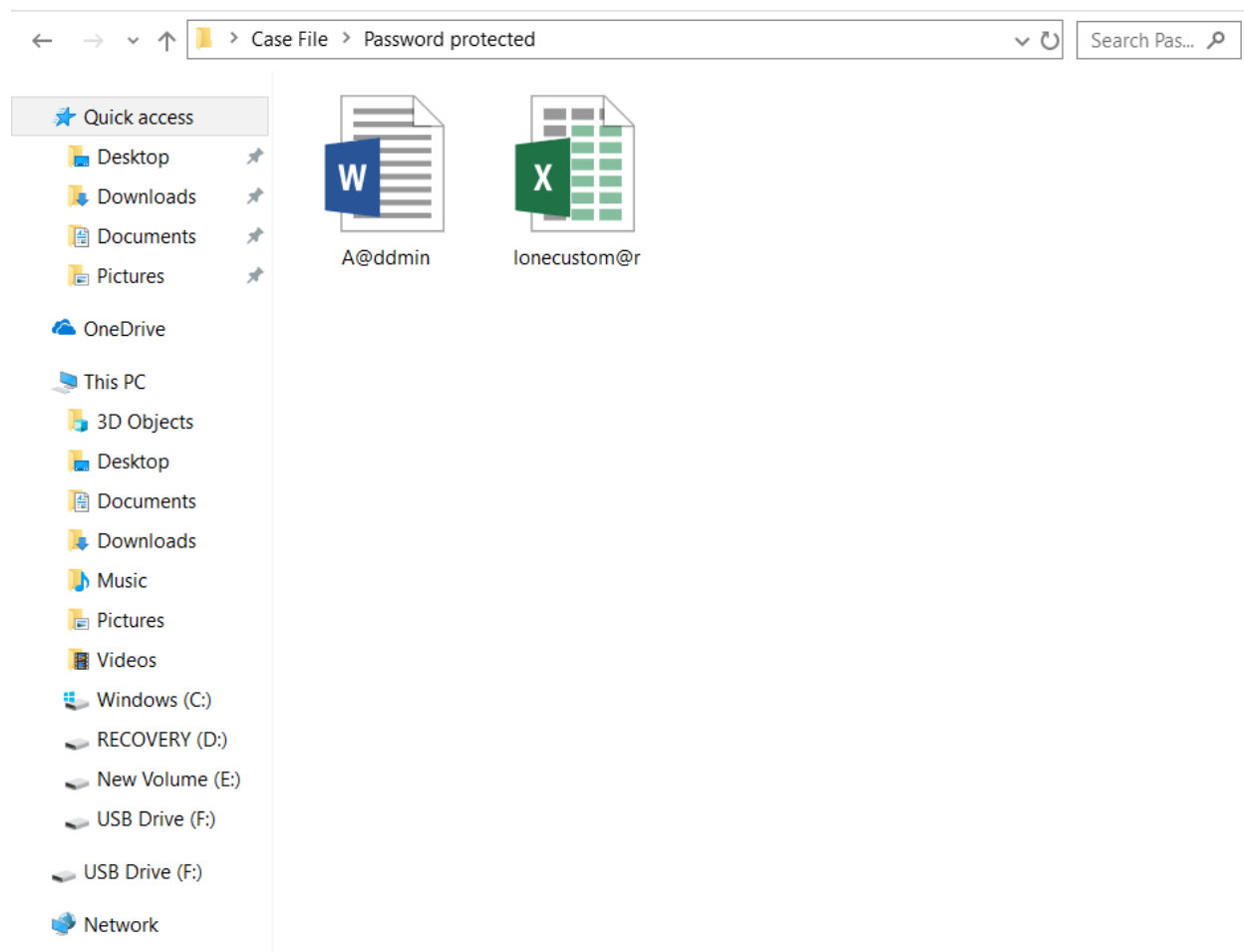
for the password protected files. The contents of each folder are briefly discussed in further part of this section.



*Figure 14:* Contents of Emails folder

The emails folder consists a set of text files which resembles emails that a person has been exchanging. This folder contains 12 text files that are hidden, and these files are named email 1 to email 12. All the emails represent a conversation that a prime suspect had with a person outside who helped the suspect commit a crime. All these text files are marked as hidden

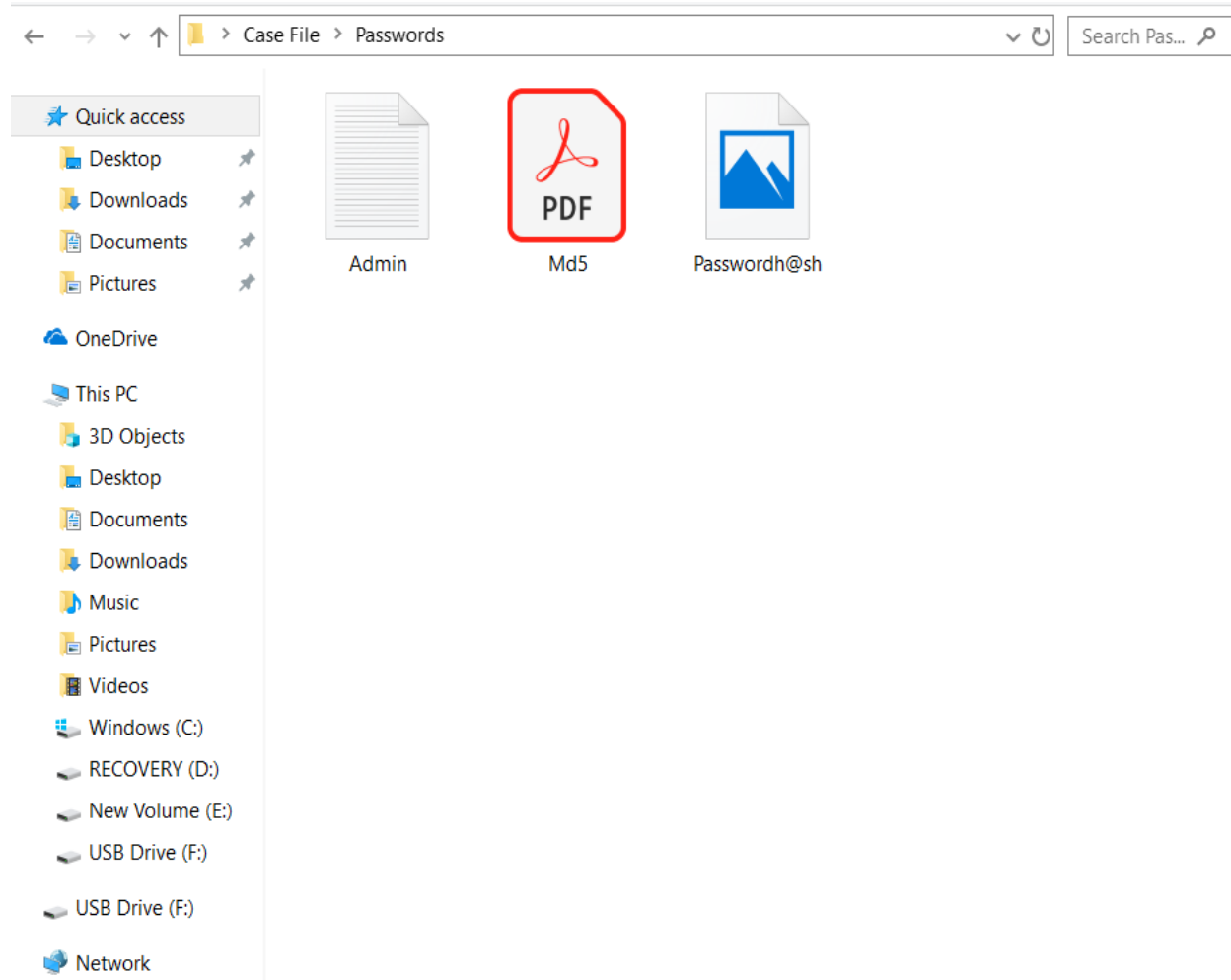
for the purposes of simulating an actual case. The emails are named in sequence according to the chronological order of the conversation for easy interpretation during the experiment.



*Figure 15: Contents of Password protected folder*

Password protected folder consists of two files which are protected using a password. The folder contains two files namely A@ddmin.docx and lonecustome@r.xlsx. The A@ddmin.docx is a word document that holds additional credential information relating to administrator bank account. The lonecustome@r.xlsx is a excel file that holds the information related to costumers in s bank. The passwords are concealed inside different files that are placed in password folder and can be tricky to be revealed.





*Figure 16: Contents of Passwords folder*

The passwords folder holds the files which contains the passwords for the password protected files. The folder contains three files namely Admin.txt, Md5.pdf, Passwordh@sh.png. The Admin.txt is a text file which hold the password for A@ddmin.docx file. Md5 is a pdf file which gives clues where to look for the password for the lonecustome@r.xlsx file. The Passwordh@sh.png is a image file that conceals a hash that is used in the md5 clue which reveals a string for that acts as a password for the lonecustome@r.xlsx file.

## Copying contents into flash devices

After creating a case file, the contents of the case file folder are copied directly into the three flash memory devices that is being used for the experiment. All the devices are made checked if there is any data present prior to copying the case data into them. The contents of each of the flash memory device are identical after copying.

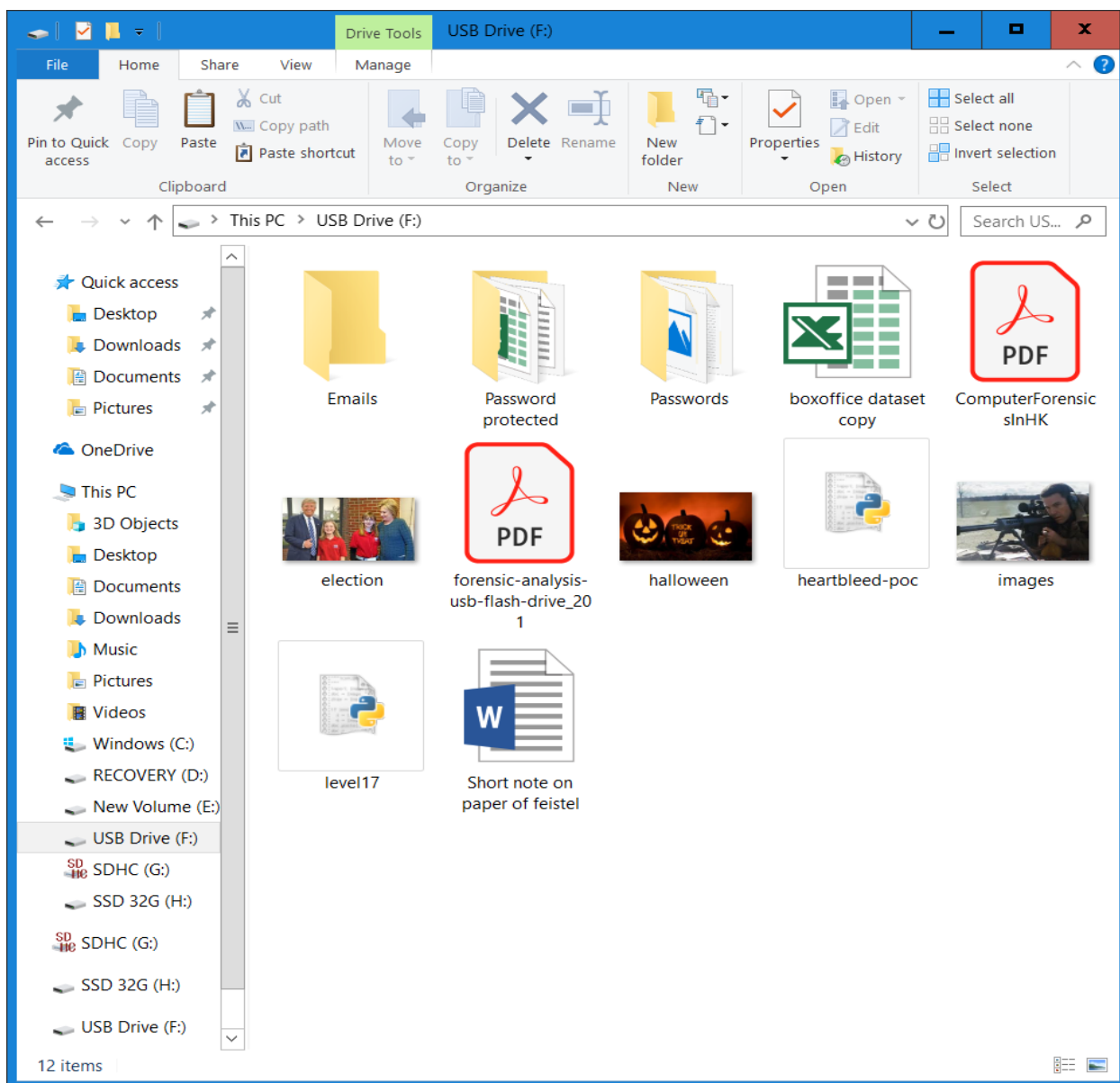


Figure 17: Copying contents of Case file into USB drive

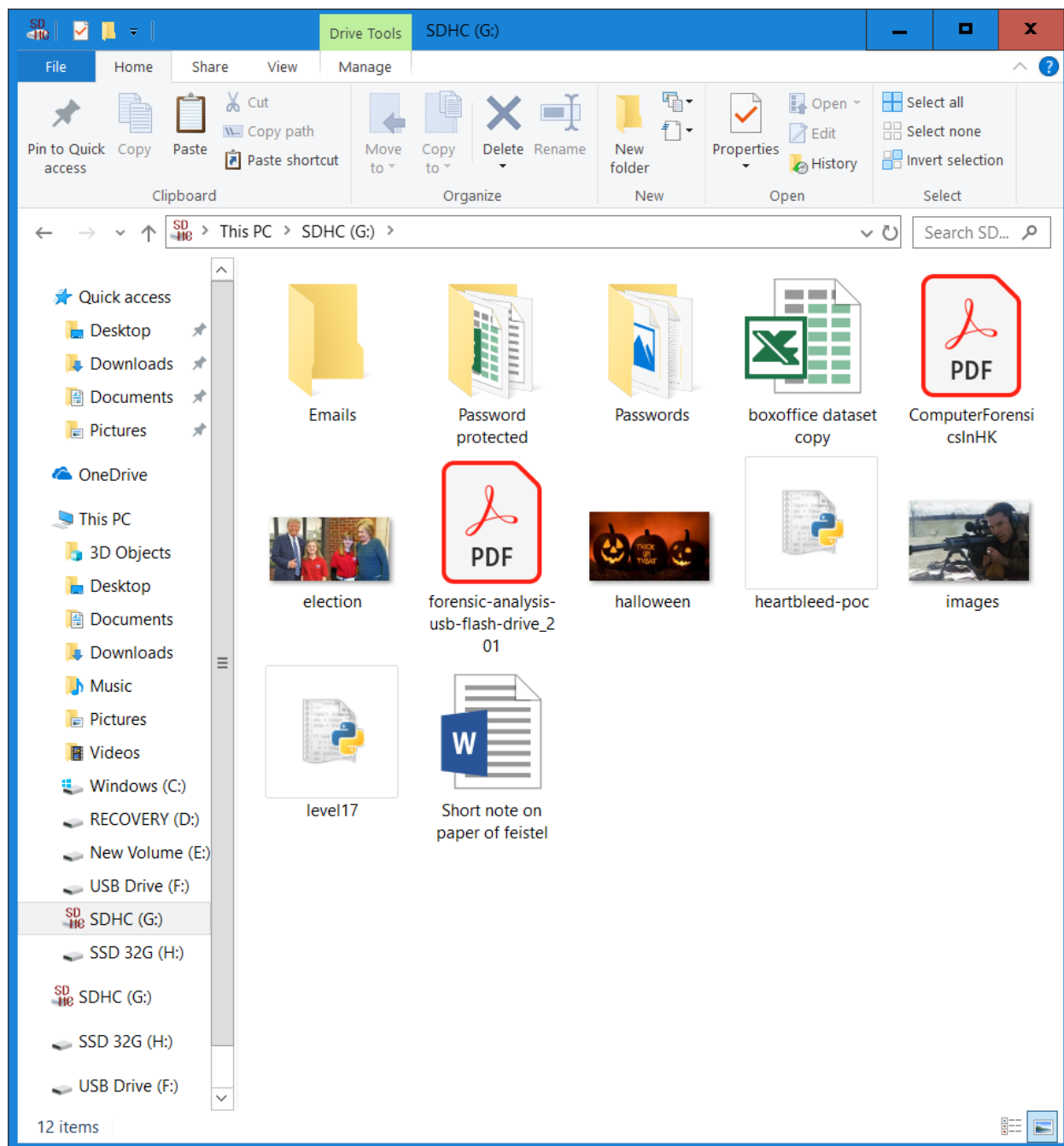
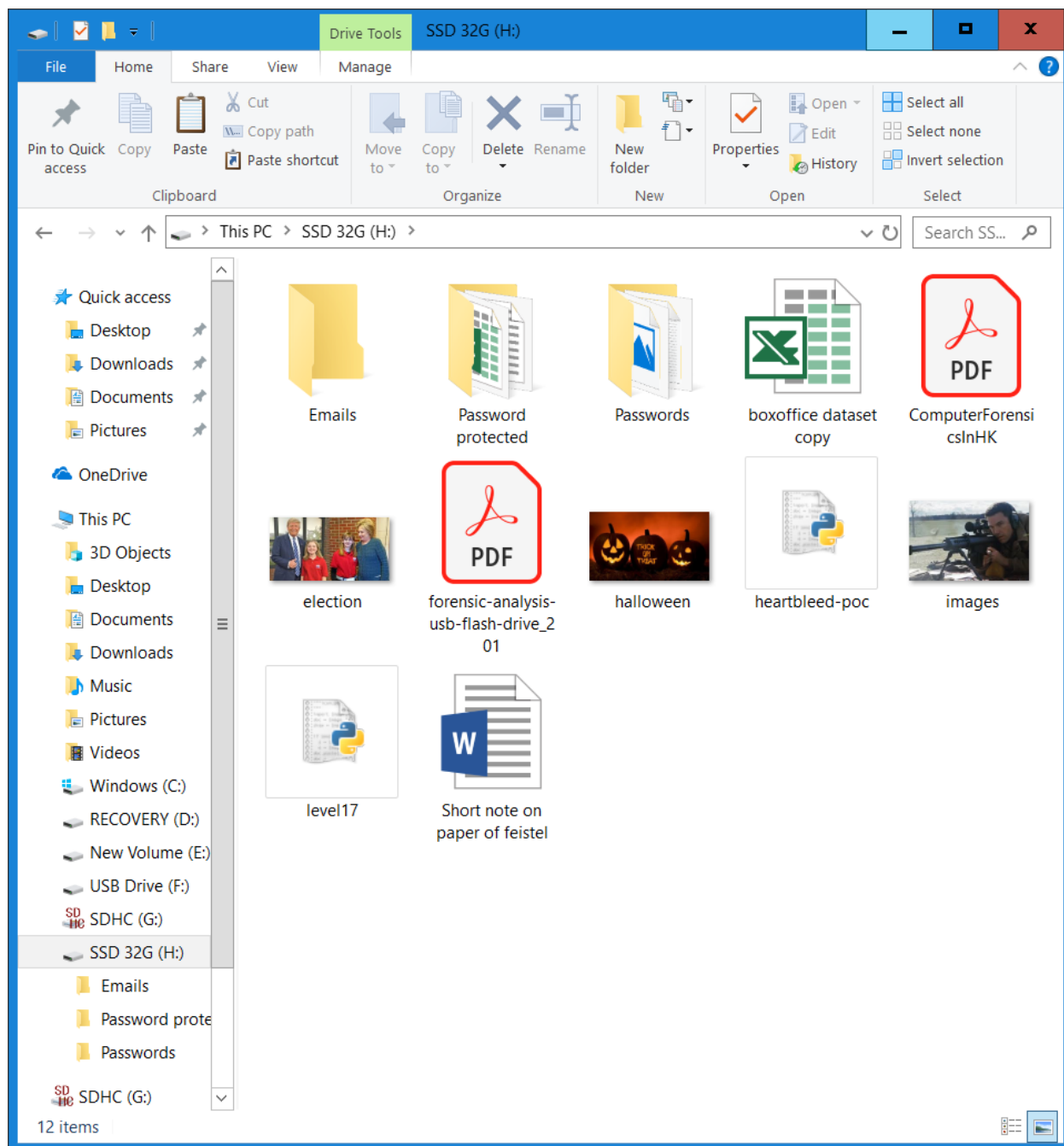


Figure 18: Copying contents of Case file into SD card



*Figure 19: Copying contents of Case file into SSD*

Once the case file contents are copied into all the flash devices, the flash memory devices are checked to see if they hold identical data in them. This is done by inspecting the folder structures in each of the device.

### Creating Images part 1 – After copying contents into devices

FTK imager is used to acquire the image of the USB, SD card and the SSD. In the first part the images are acquired from the devices right after copying the contents into them. These images are named USB\_IMG 01, SD\_CARD\_IMG 01, SSD\_IMG 01 respectively. The acquiring of the images are illustrated using the images below.

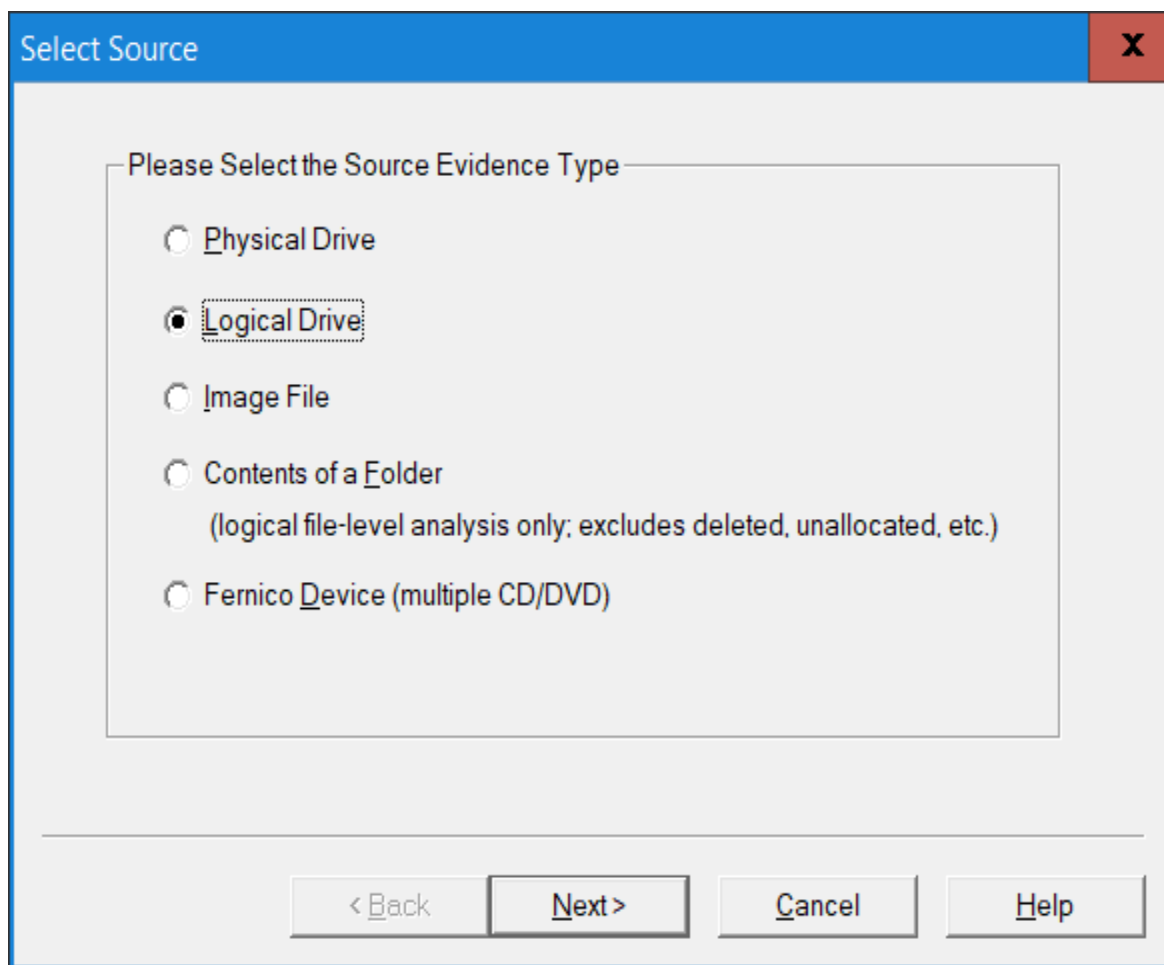
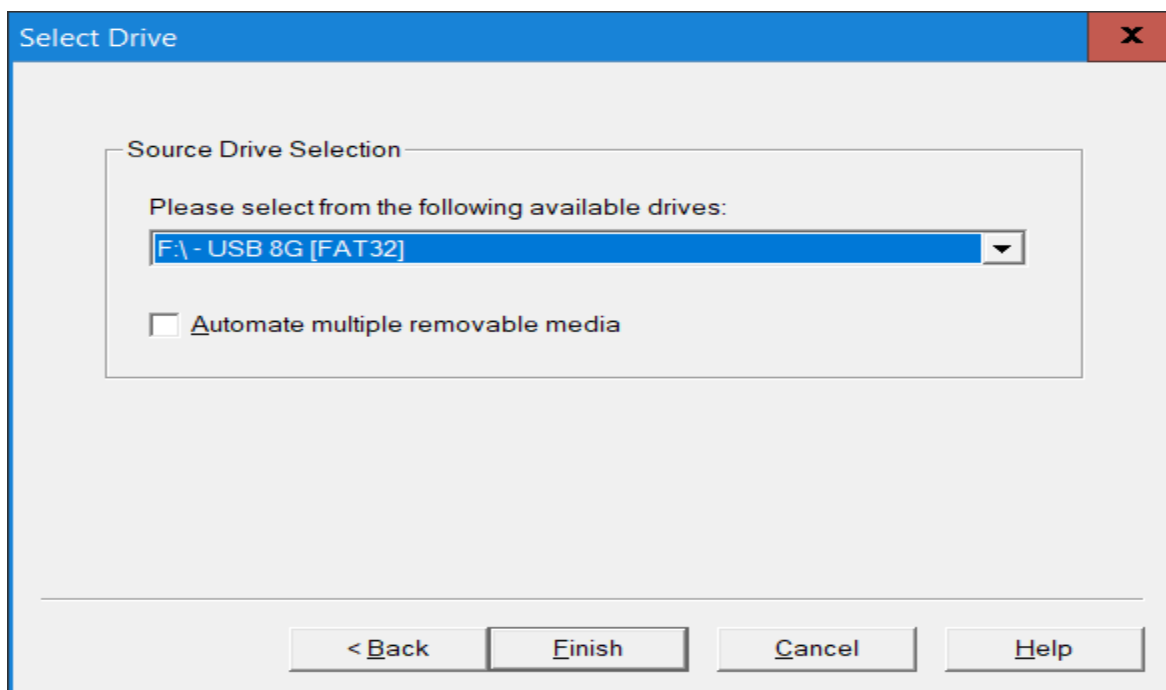


Figure 20: Selecting logical drive for all the devices

When imaging the flash memory devices, the logical drive is selected as the source for all the devices. This is the first step in acquiring the images from the devices. The first step is repeated for all the devices before extracting the image from them.



*Figure 21:* Select USB as source for image creation

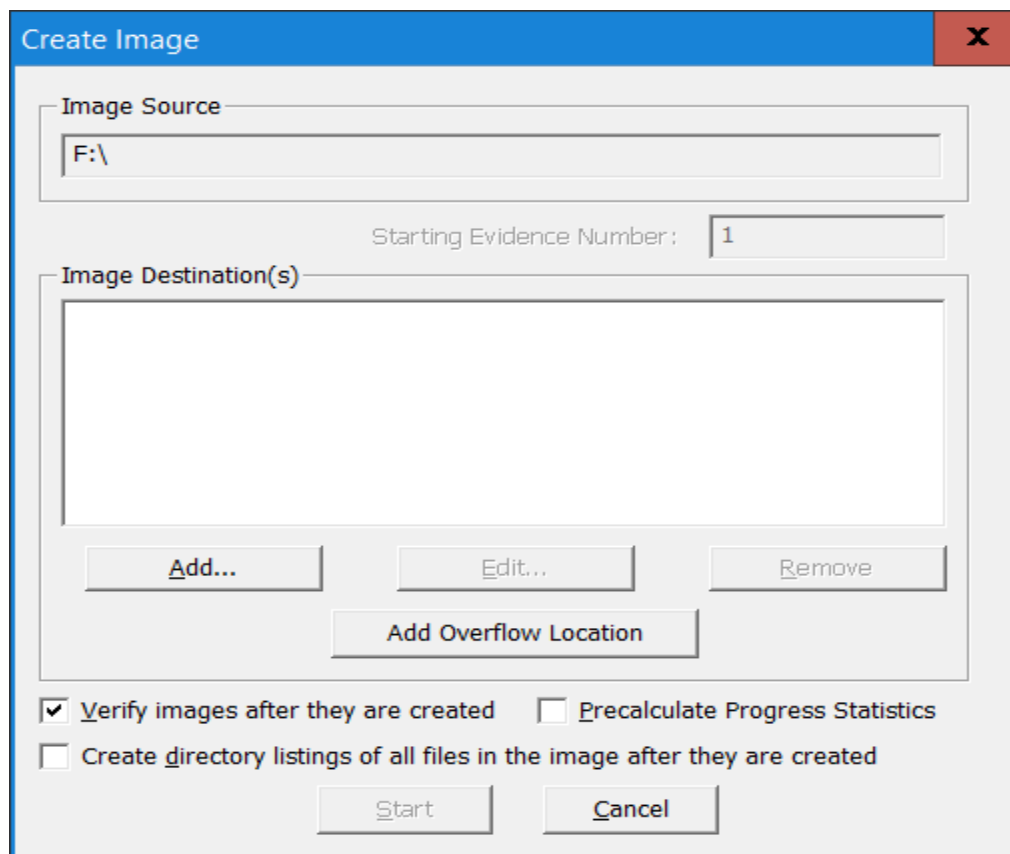


Figure 22: Dialogue box for adding options to USB image

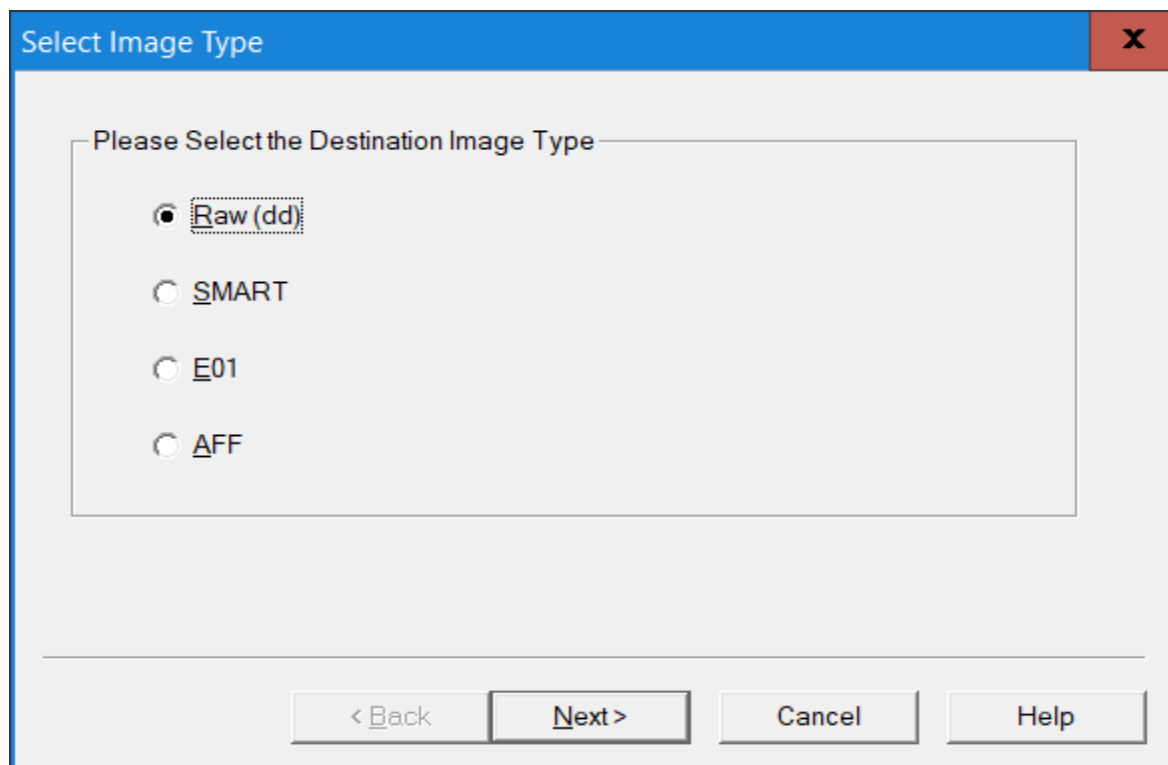
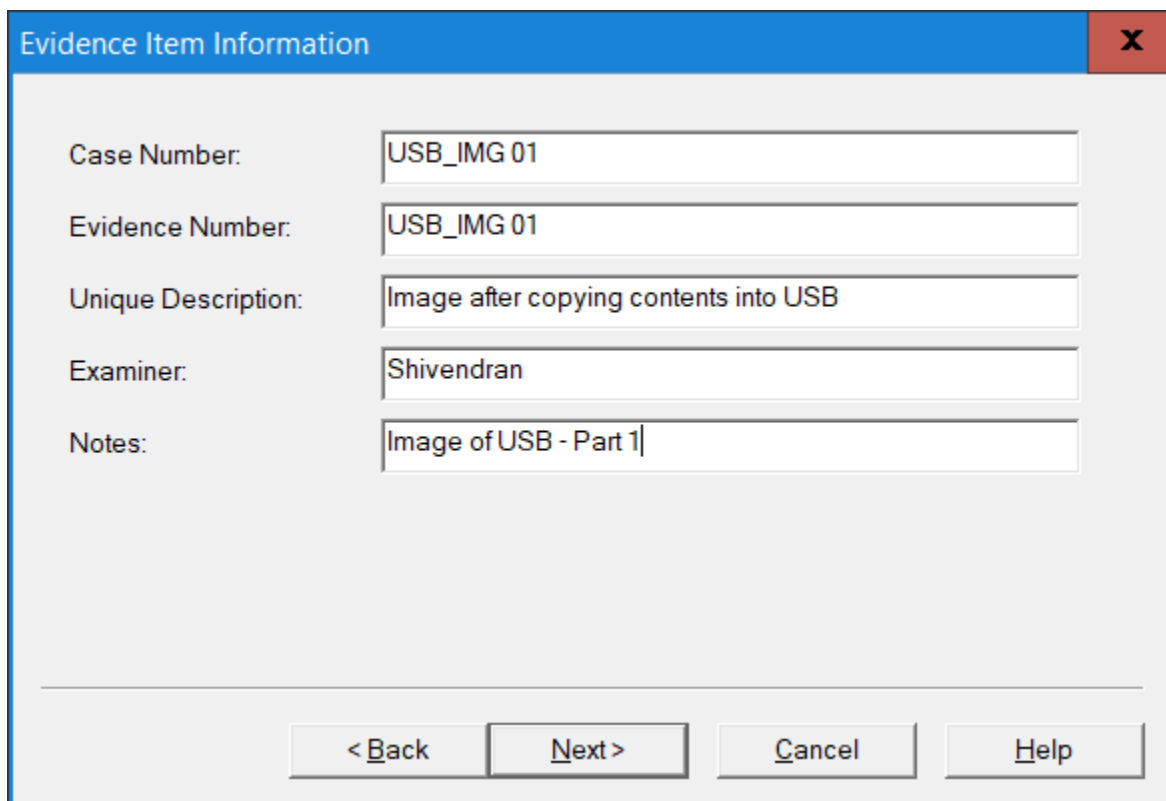


Figure 23: Dialogue box for selecting the type of image



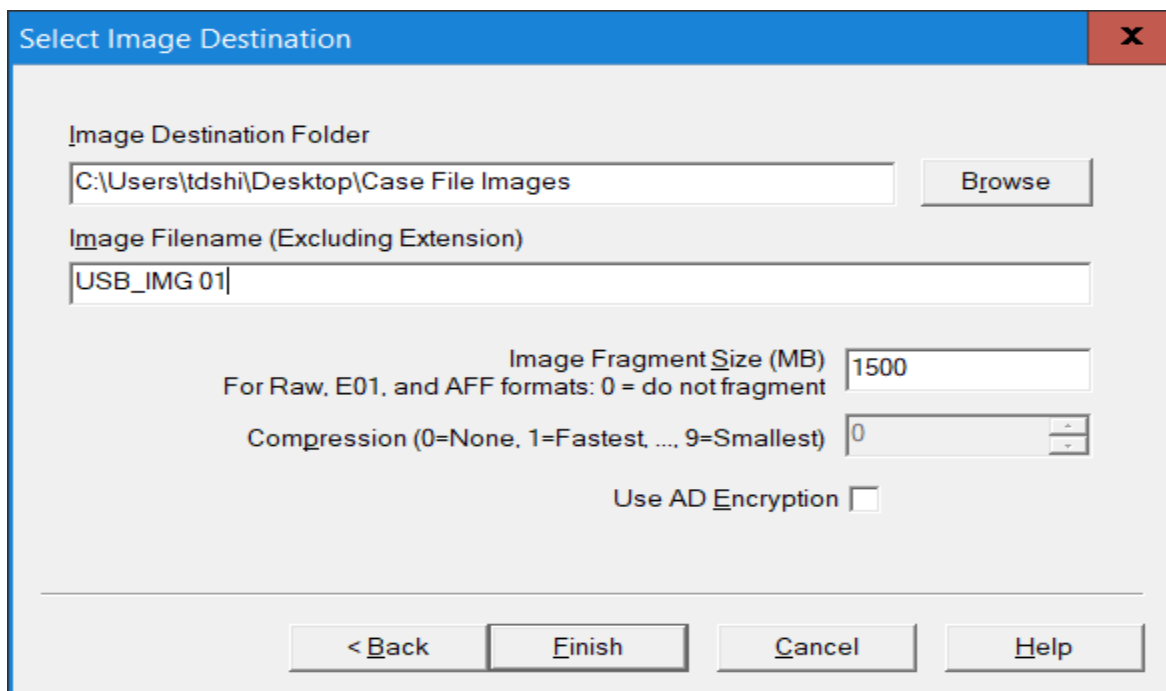


The dialog box titled "Evidence Item Information" contains the following fields and values:

Case Number:	USB_IMG 01
Evidence Number:	USB_IMG 01
Unique Description:	Image after copying contents into USB
Examiner:	Shivendran
Notes:	Image of USB - Part 1

Buttons at the bottom: < Back, Next >, Cancel, Help

Figure 24: Providing additional image information for USB



The dialog box titled "Select Image Destination" contains the following fields and values:

Image Destination Folder	C:\Users\tdshi\Desktop\Case File Images	Browse
Image Filename (Excluding Extension)	USB_IMG 01	
Image Fragment Size (MB)	1500	
For Raw, E01, and AFF formats: 0 = do not fragment		
Compression (0=None, 1=Fastest, ..., 9=Smallest)	0	
Use AD Encryption	<input type="checkbox"/>	

Buttons at the bottom: < Back, Finish, Cancel, Help

Figure 25: Providing destination and image file name for USB image

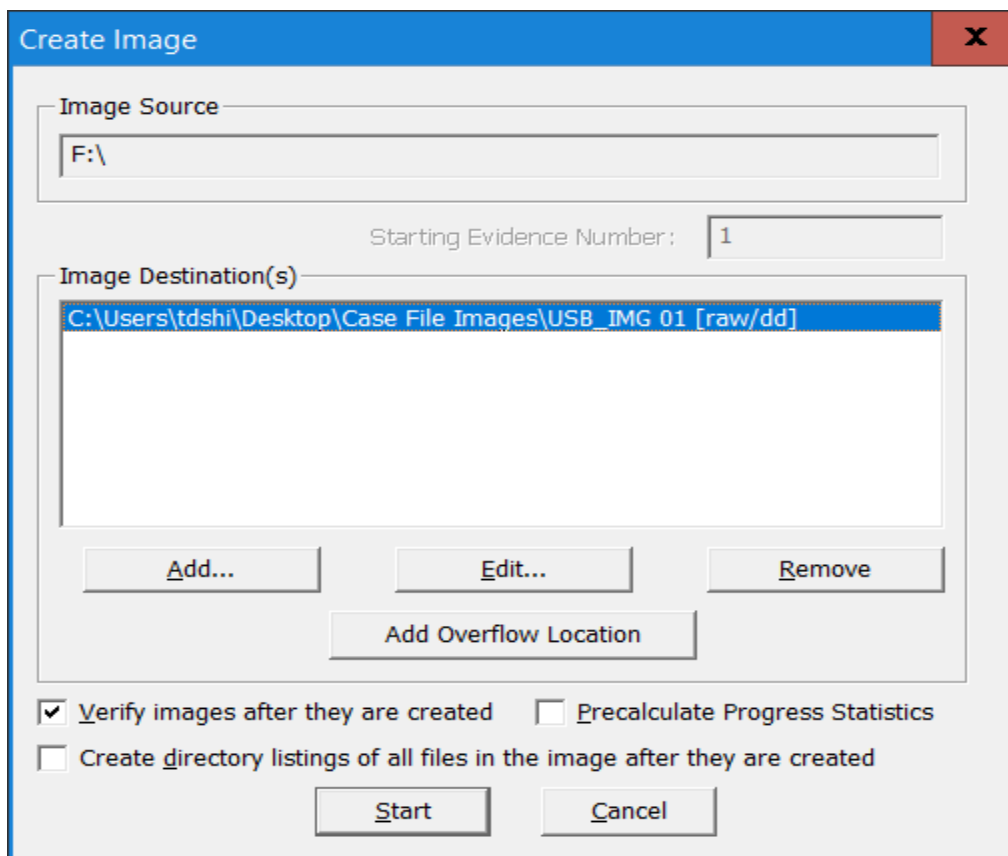


Figure 26: Dialogue box before starting image creation for USB

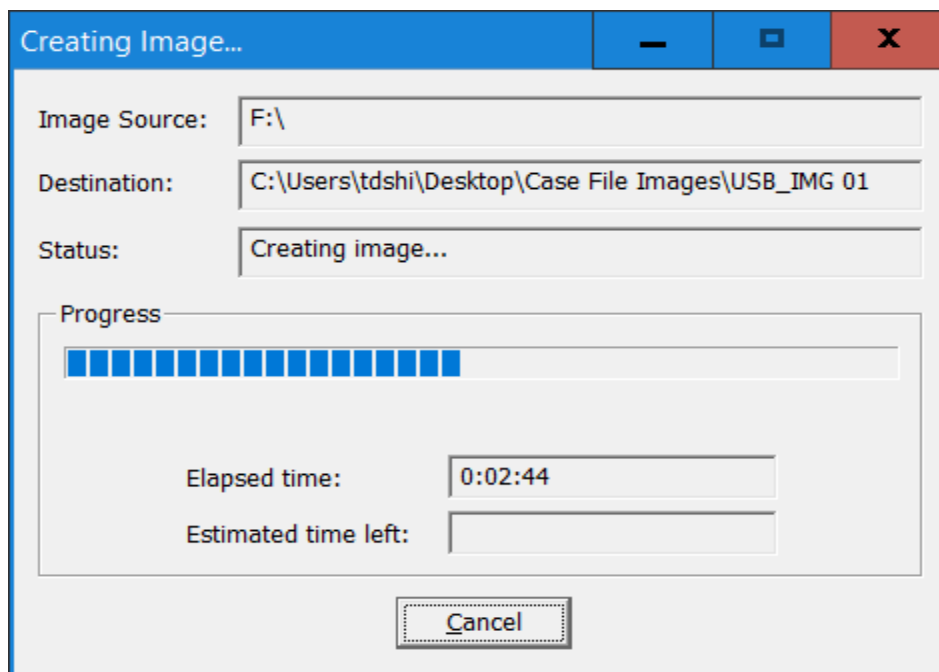


Figure 27: Image creation process for USB

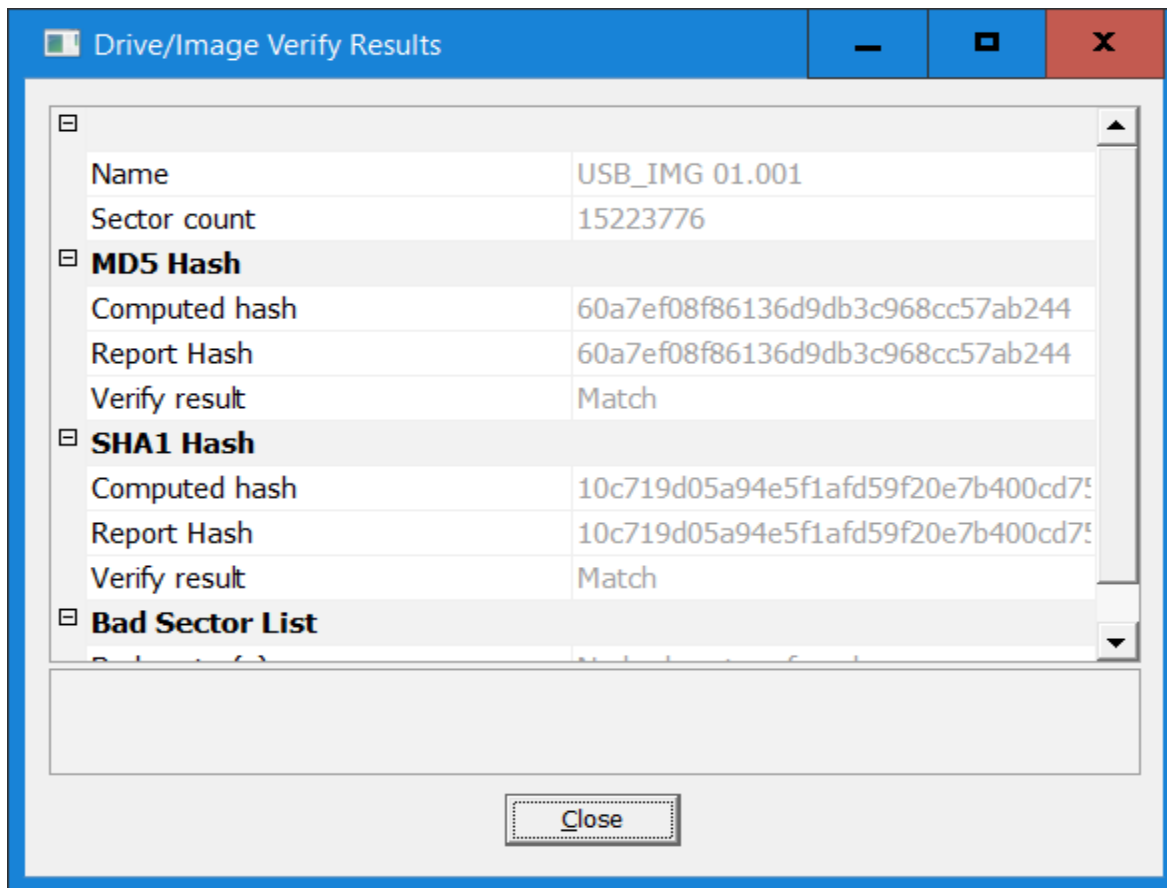


Figure 28: USB image creation completion

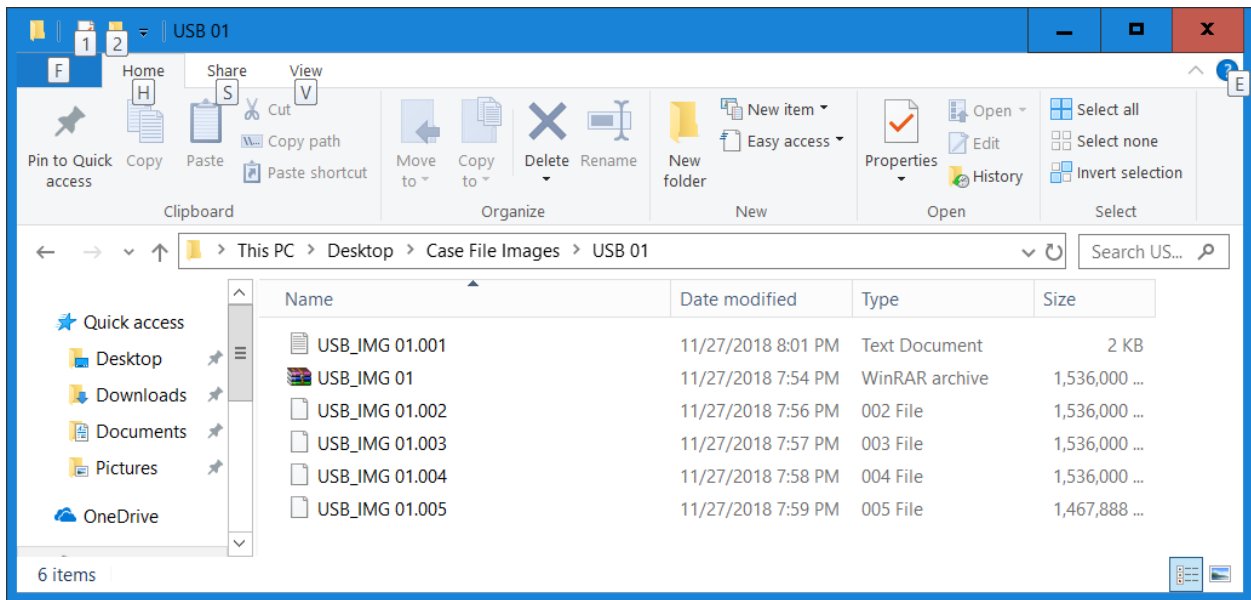
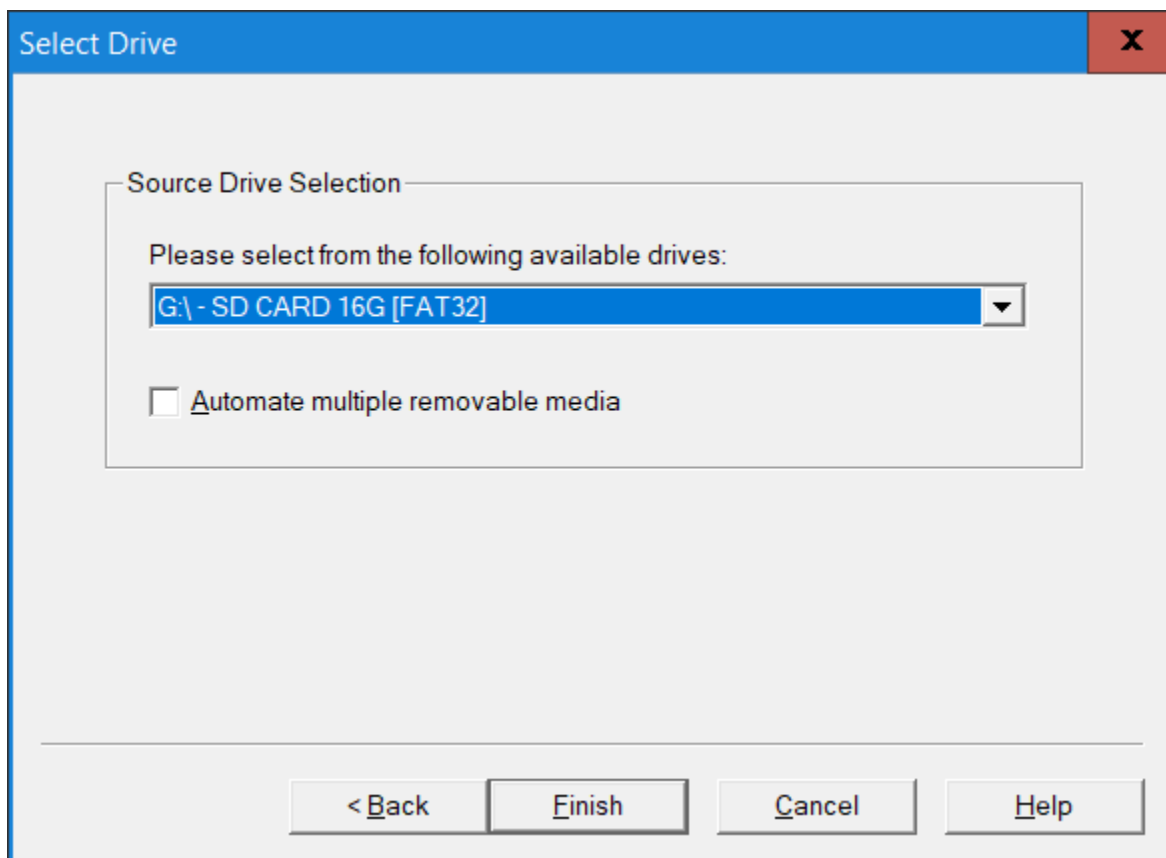


Figure 29: Images of USB drive part 1



*Figure 30: Select SD card as source for image creation*

Evidence Item Information

Case Number: SD\_CARD\_IMG 01

Evidence Number: SD\_CARD\_IMG 01

Unique Description: Image after copying contents into SD Card

Examiner: Shivendran

Notes: Image of SD card - Part 1

< Back   Next >   Cancel   Help

Figure 31: Providing additional image information for SD card

Select Image Destination

Image Destination Folder  
C:\Users\tdshi\Desktop\Case File Images Browse

Image Filename (Excluding Extension)  
SD\_CARD\_IMG 01

Image Fragment Size (MB) 1500  
For Raw, E01, and AFF formats: 0 = do not fragment

Compression (0=None, 1=Fastest, ..., 9=Smallest) 0

Use AD Encryption

< Back Finish Cancel Help

Figure 32: Providing destination and image file name for SD card image



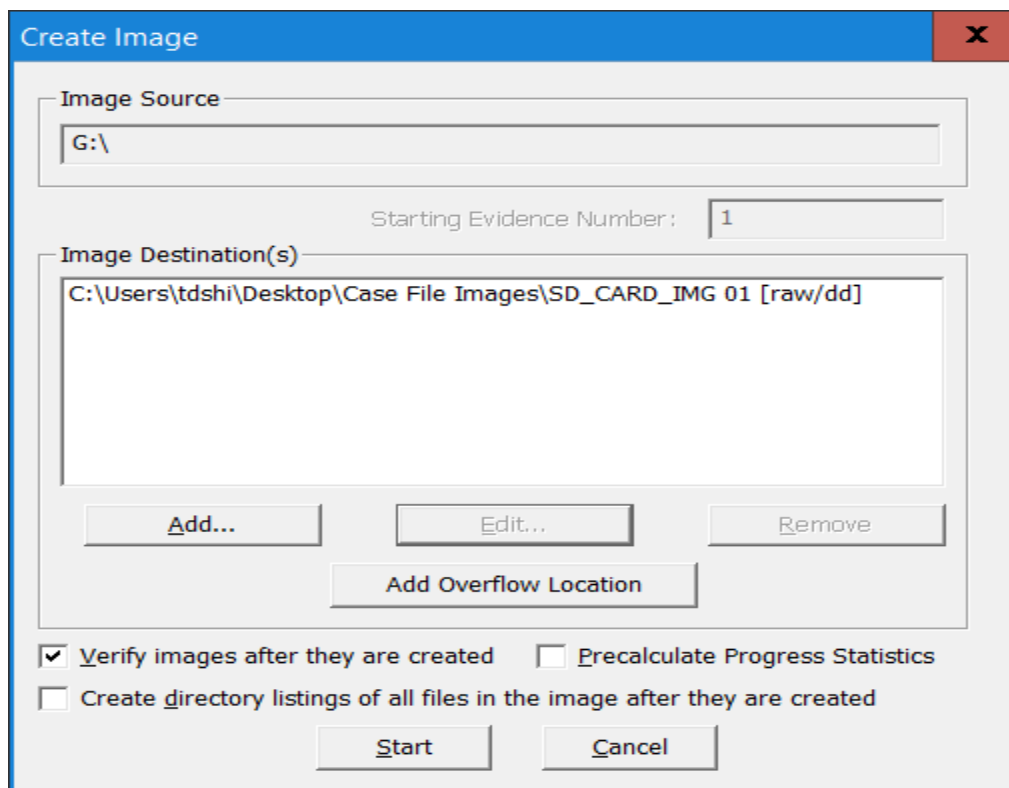


Figure 33: Dialogue box before starting image creation for SD card

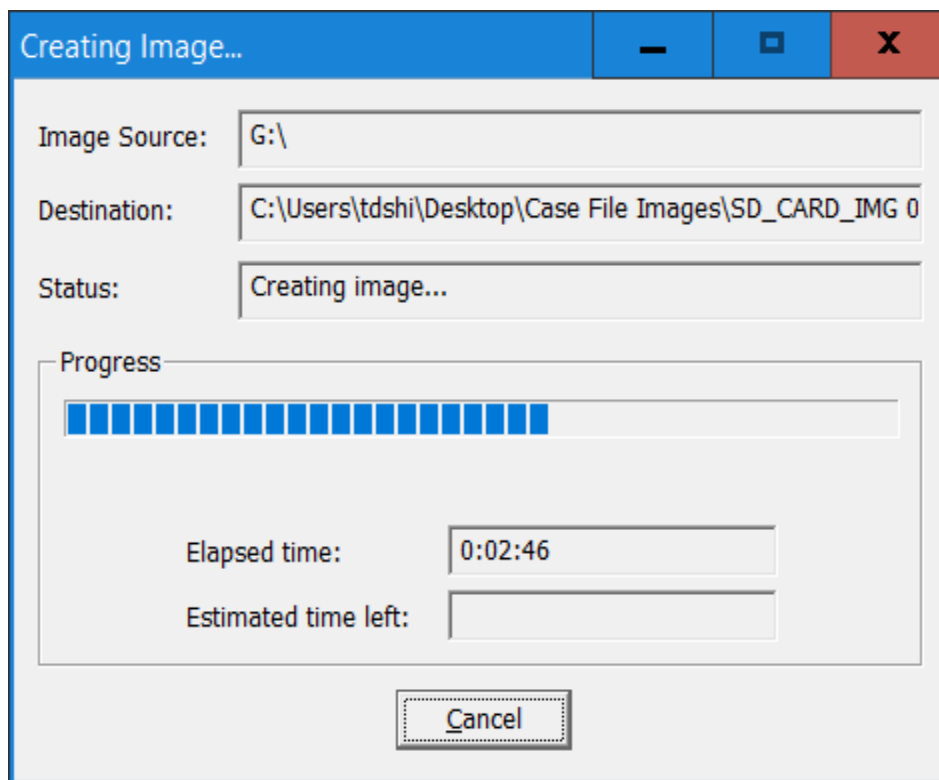


Figure 34: Image creation process for SD card

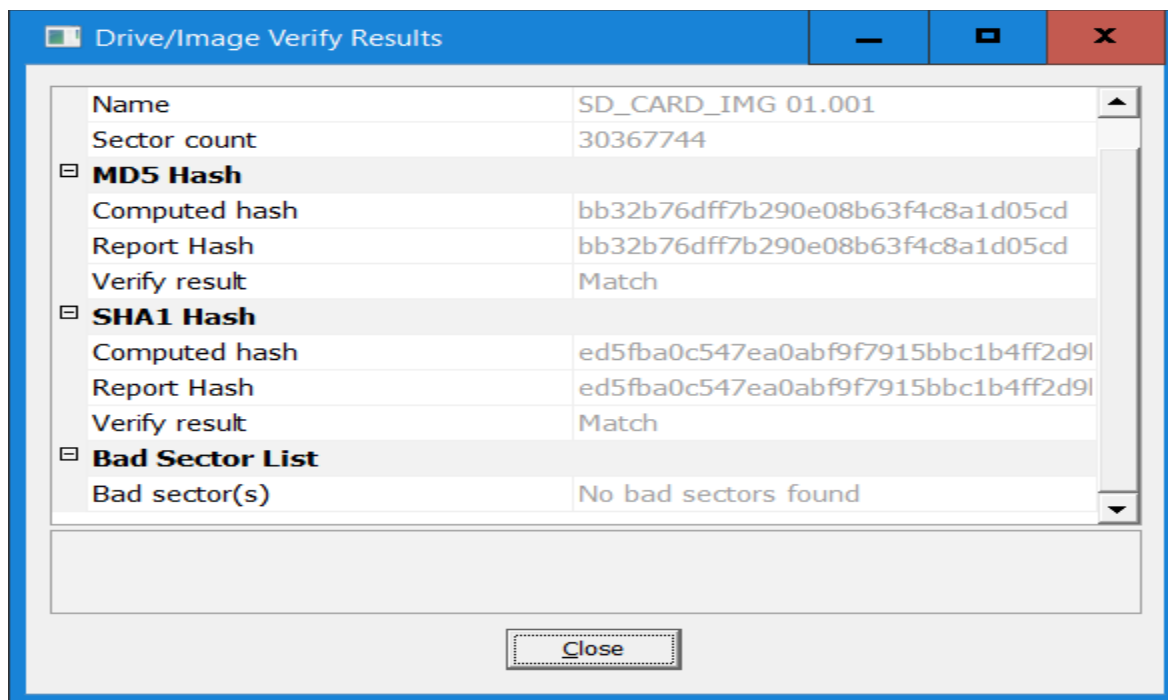


Figure 35: SD card image creation completion

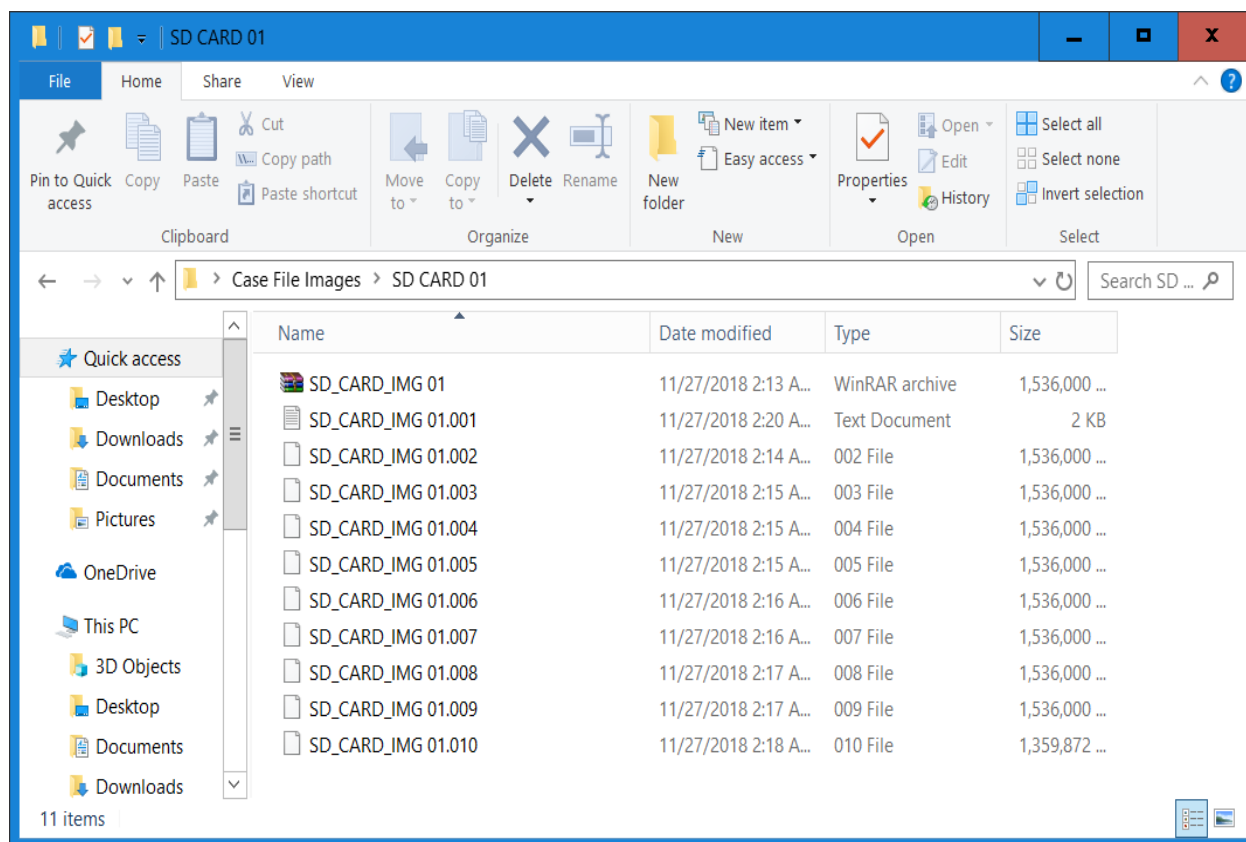
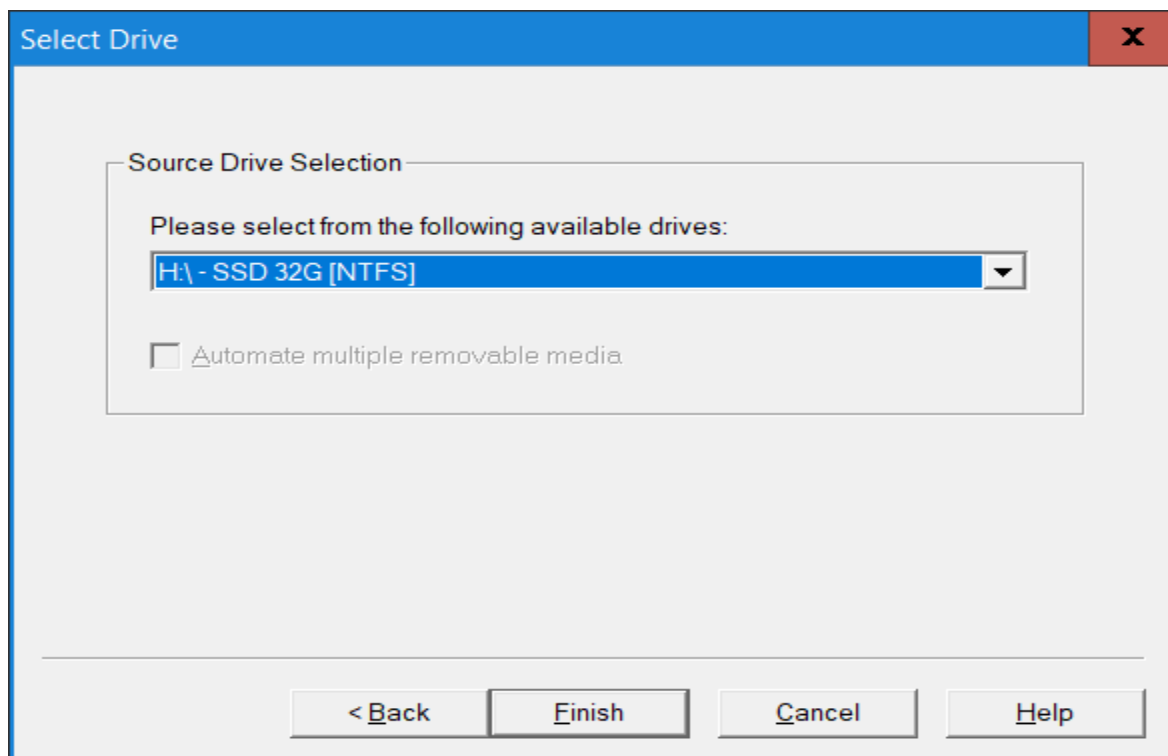
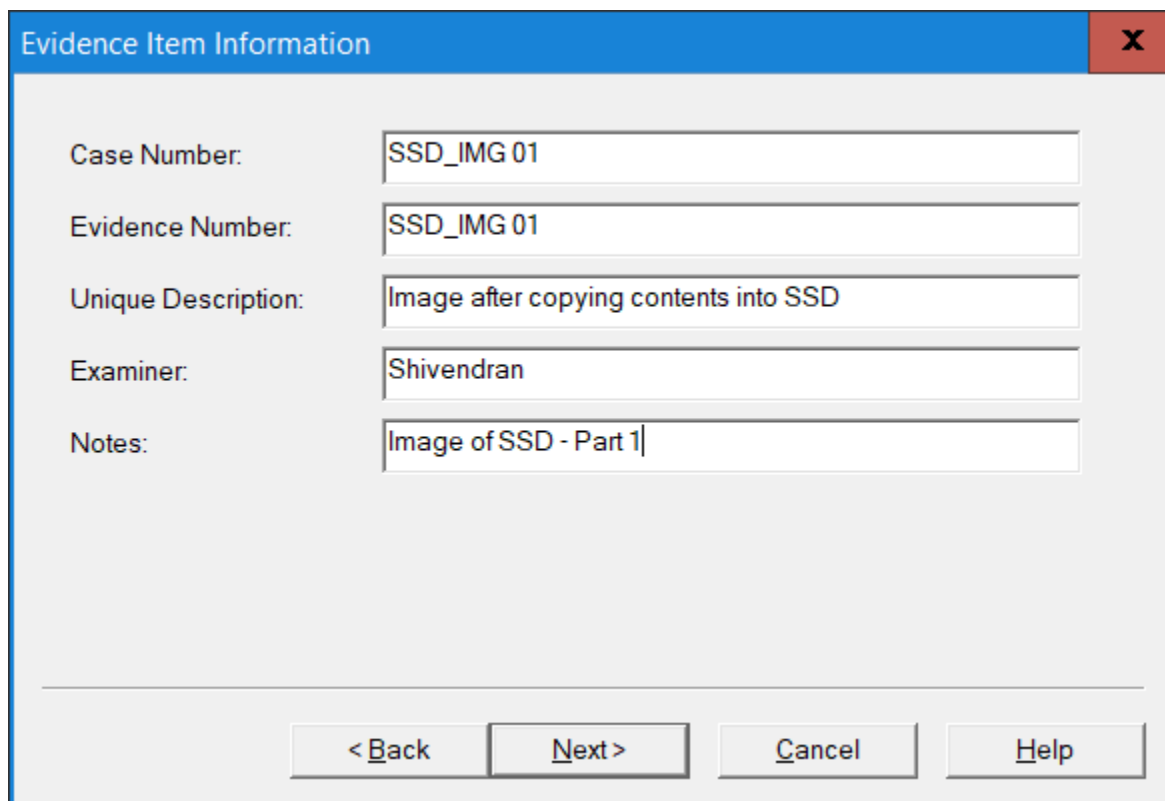


Figure 36: Images of SD card part 1



*Figure 37: Select SSD as source for image creation*

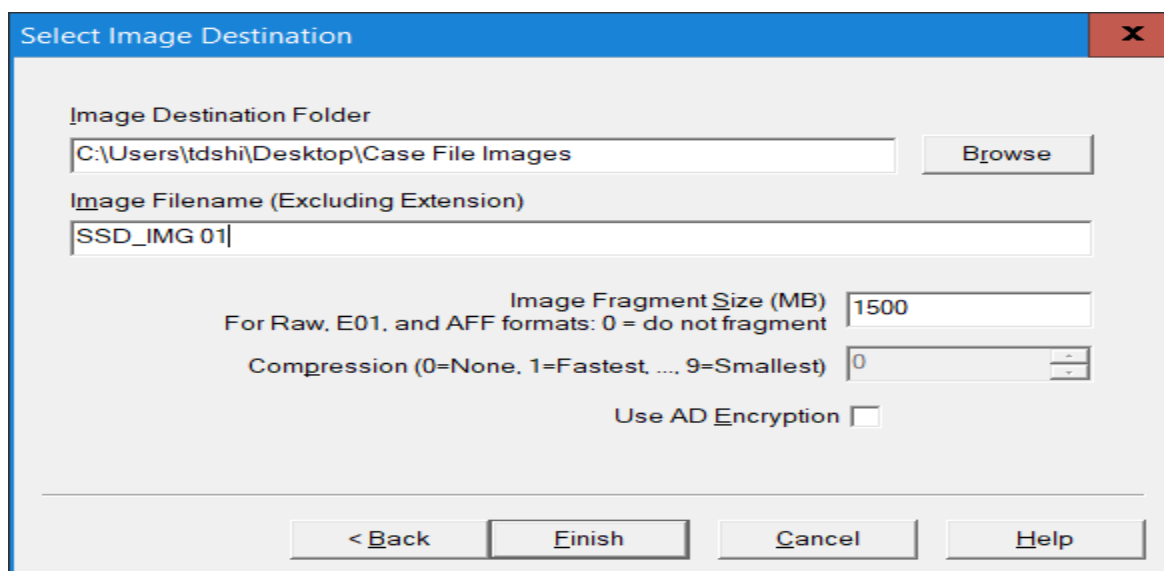


The dialog box titled "Evidence Item Information" has a blue header bar with a close button (X) on the right. It contains five text input fields:

- Case Number: SSD\_IMG 01
- Evidence Number: SSD\_IMG 01
- Unique Description: Image after copying contents into SSD
- Examiner: Shivendran
- Notes: Image of SSD - Part 1

At the bottom, there are four buttons: "< Back", "Next >", "Cancel", and "Help".

Figure 38: Providing additional image information for SSD



The dialog box titled "Select Image Destination" has a blue header bar with a close button (X) on the right. It contains the following fields and controls:

- Image Destination Folder: C:\Users\tdshi\Desktop\Case File Images (with a "Browse" button to the right)
- Image Filename (Excluding Extension): SSD\_IMG 01
- Image Fragment Size (MB): 1500 (with a note: "For Raw, E01, and AFF formats: 0 = do not fragment")
- Compression (0=None, 1=Fastest, ..., 9=Smallest): 0 (with a spinner control)
- Use AD Encryption:

At the bottom, there are four buttons: "< Back", "Finish", "Cancel", and "Help".

Figure 39: Providing destination and image file name for SSD image

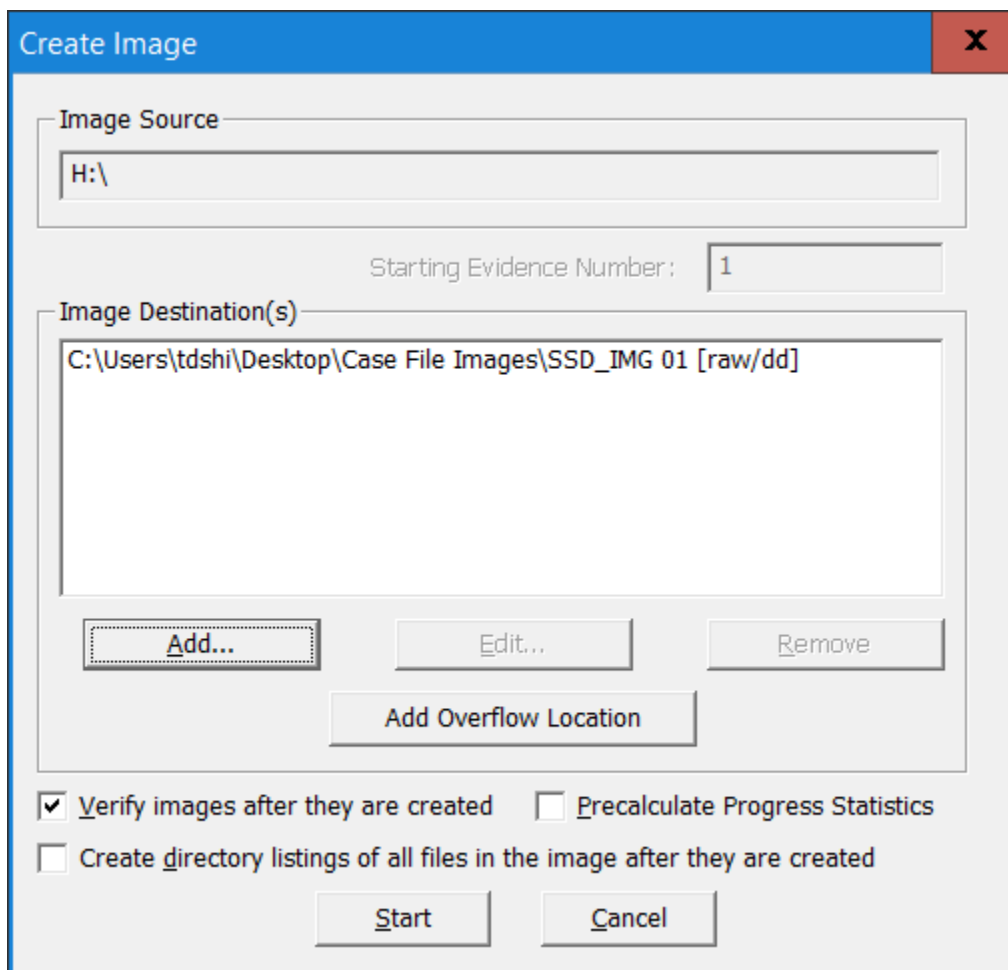


Figure 40: Dialogue box before starting image creation for SSD

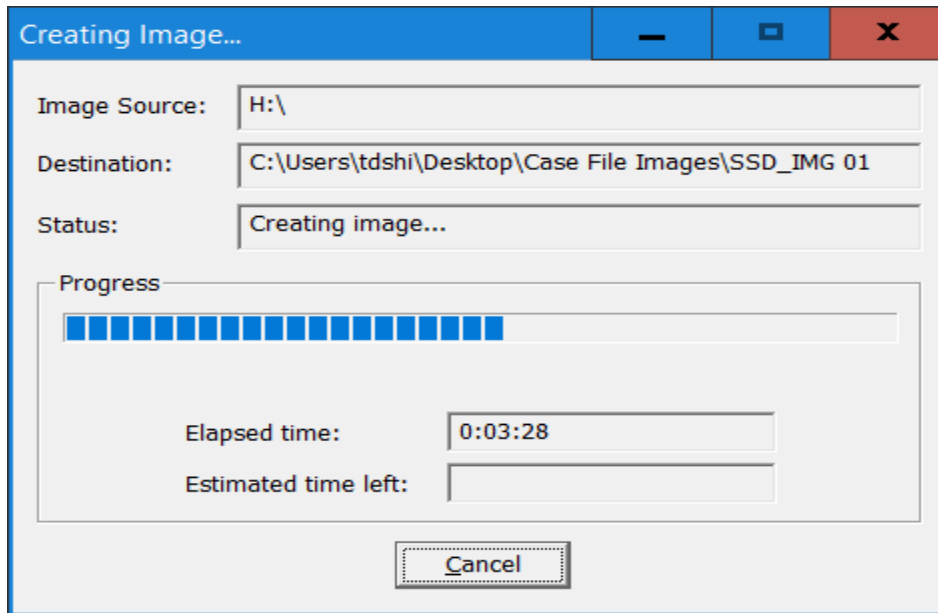


Figure 41: Image creation process for SSD

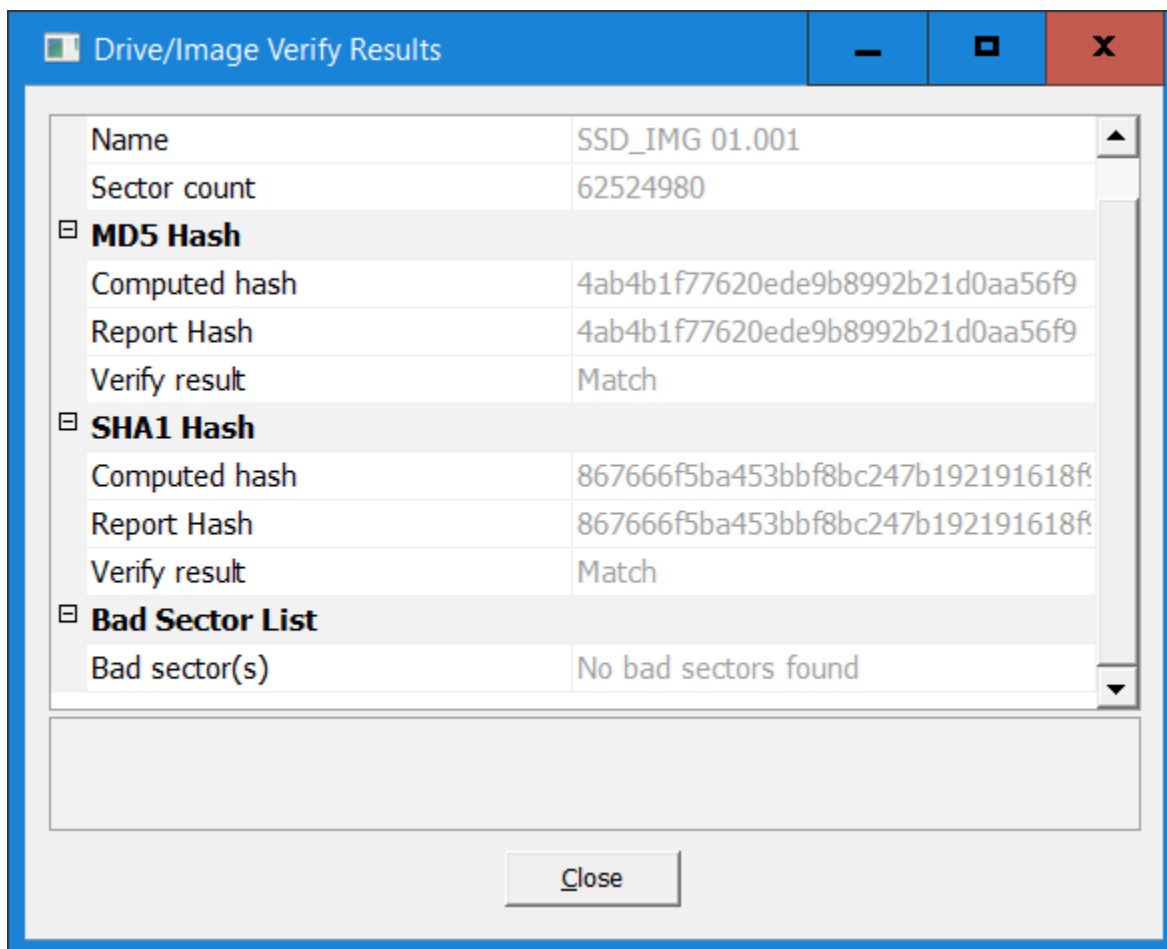


Figure 42: SSD image creation completion



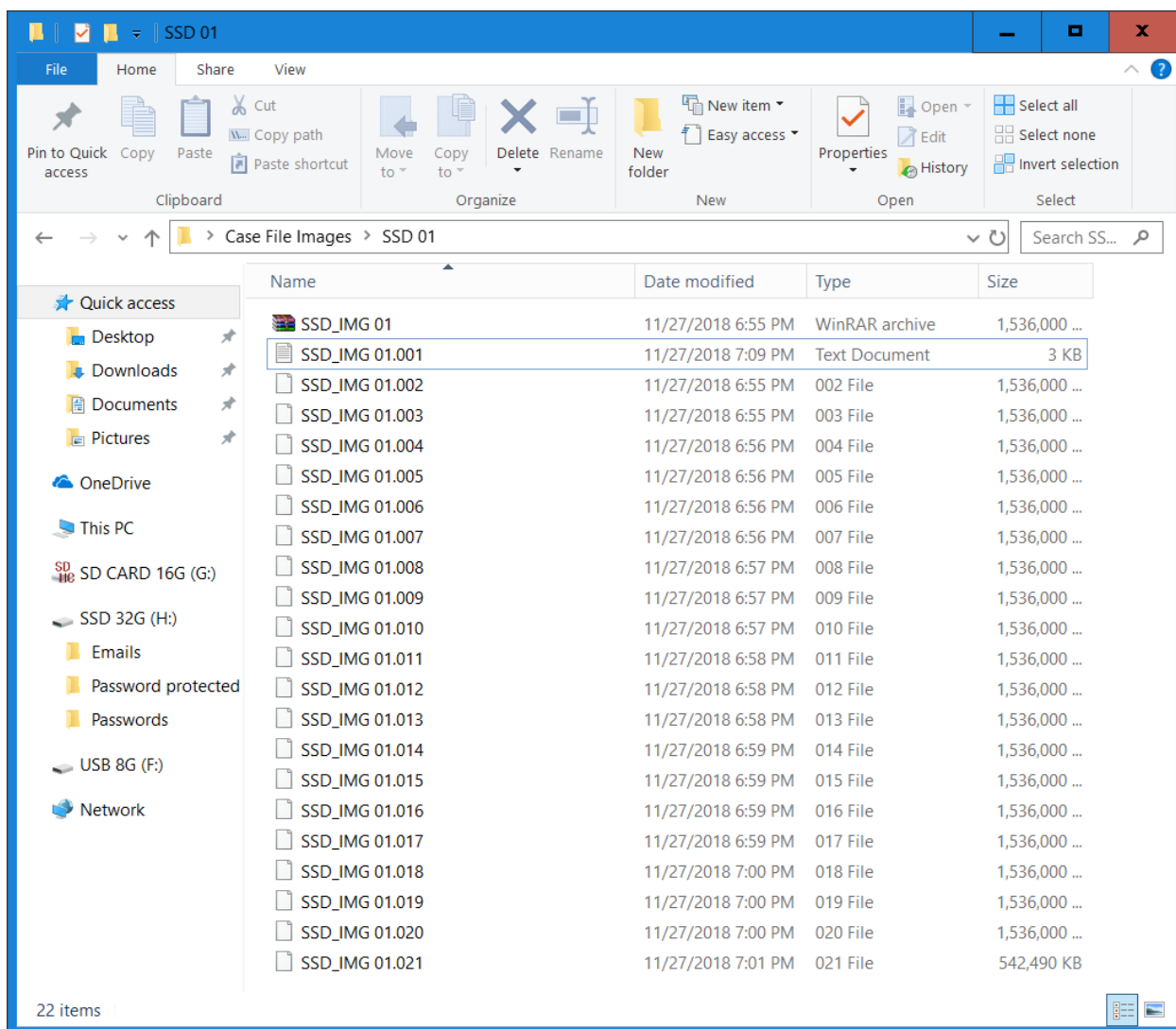


Figure 43: Images of SSD part 1

### Creating Images part 2– After deleting certain contents from devices

Some of the files from the flash memory devices are deleted for part 2 of the experiment. The files that are deleted will be the critical files that are related to the case plus a few dummy files that are not related to the case is also deleted. Passwords folder which is directly related to the case is deleted on all three devices. This folder is one of the important piece for finding evidence in the case. The other files that are deleted are election.jpg, images.jpg, heartbleed-

poc.py, forensic-analysis-usb-flash-drive\_201.pdf. These files are not related to the case and are not useful for finding evidences relating to the case. All the devices are checked for same contents of file after deletion process.

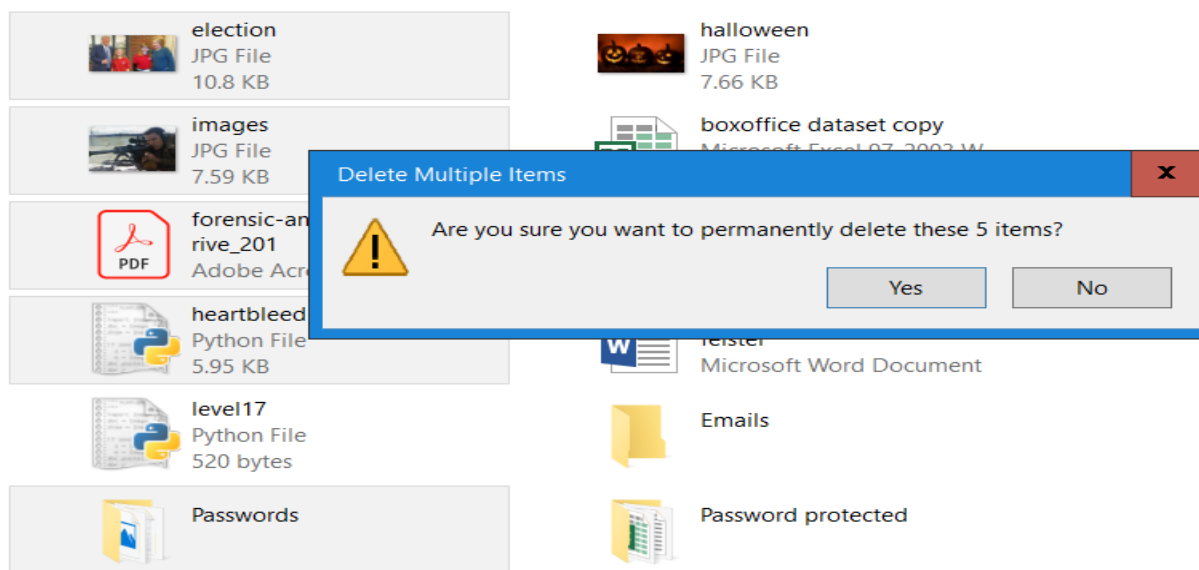


Figure 44: Files that are deleted on each device

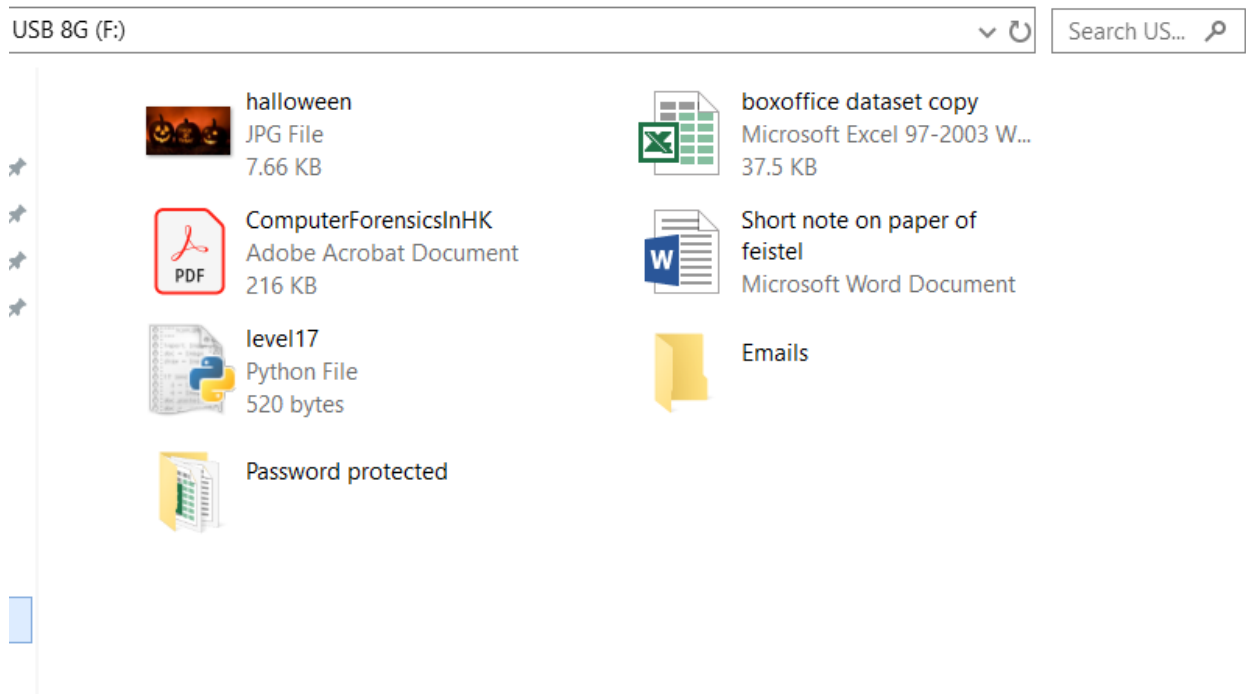


Figure 45: Contents of USB drive after deleting

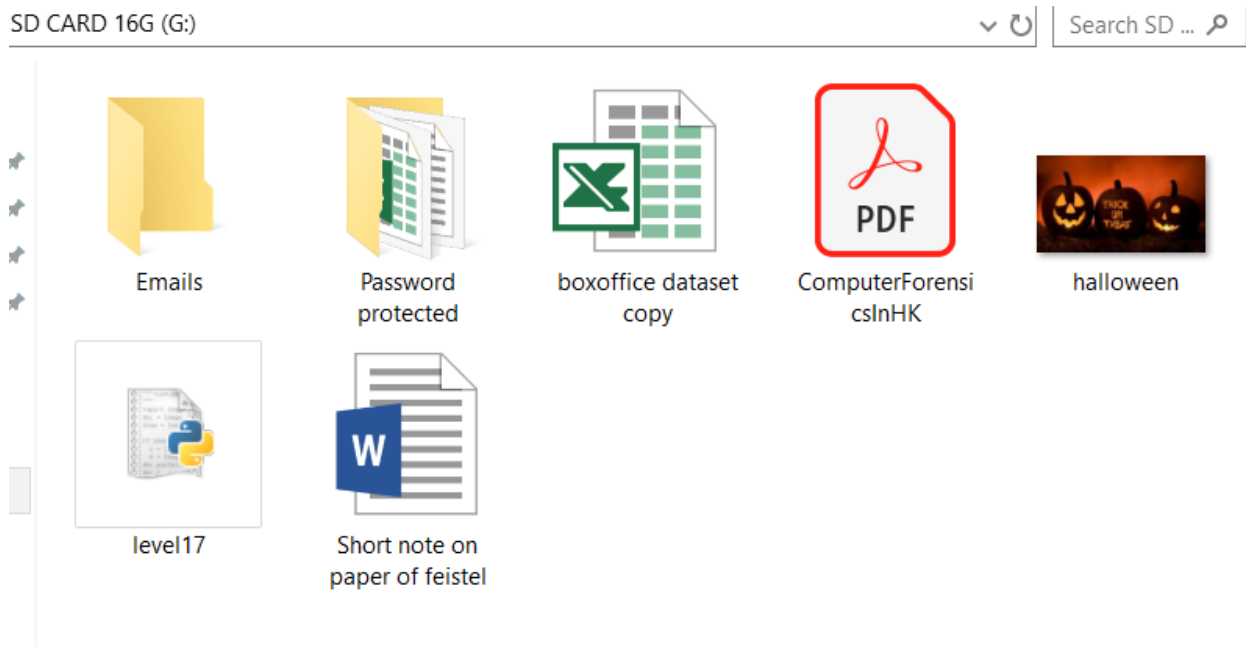
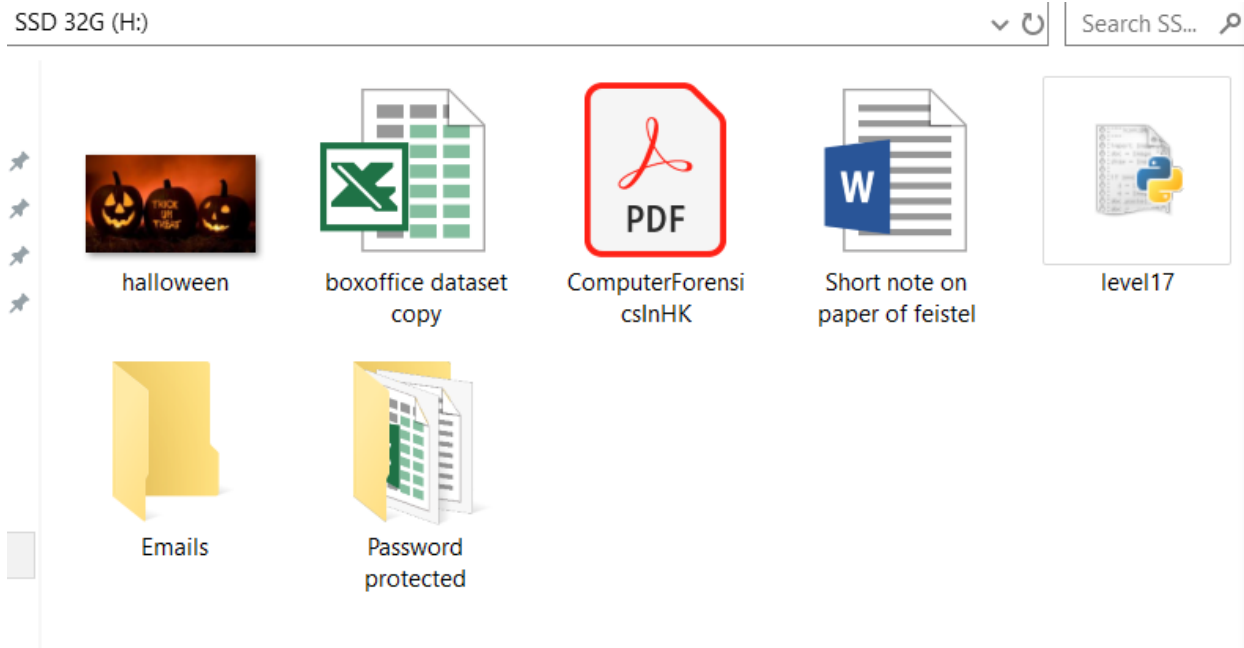


Figure 46: Contents of SD card after deleting



*Figure 47: Contents of SSD after deleting*

After deleting the essential contents from all the devices, images are extracted from all the three devices. The image creation process is repeated. These images will be saved as IMG part 2 for each of the flash memory device.

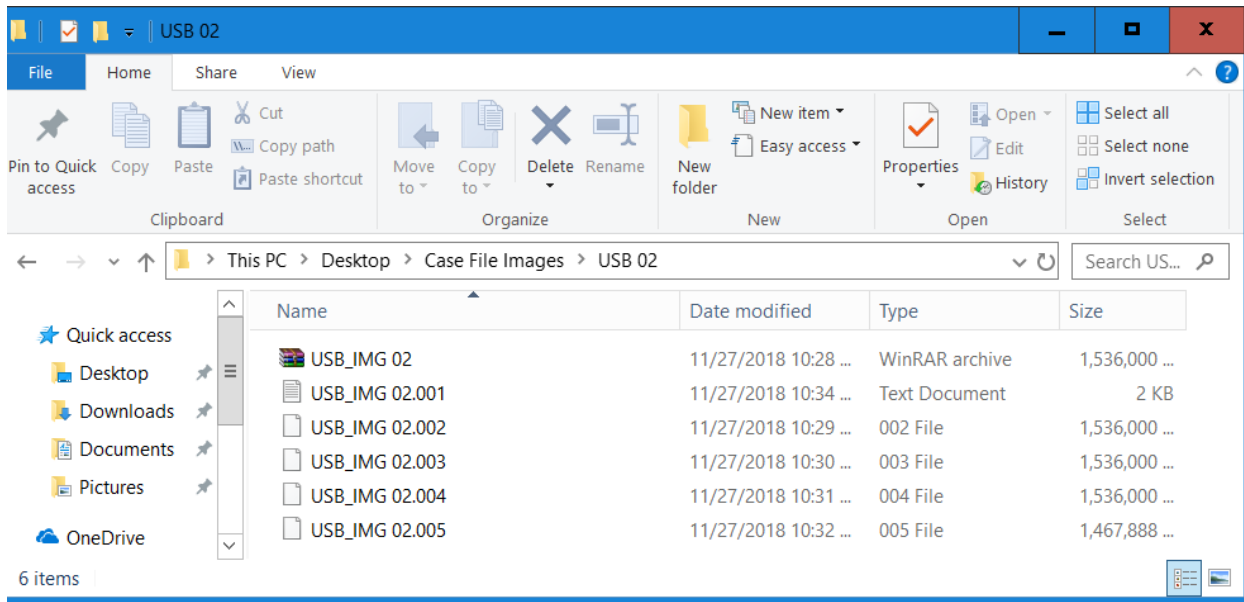


Figure 48: Images of USB drive part 2

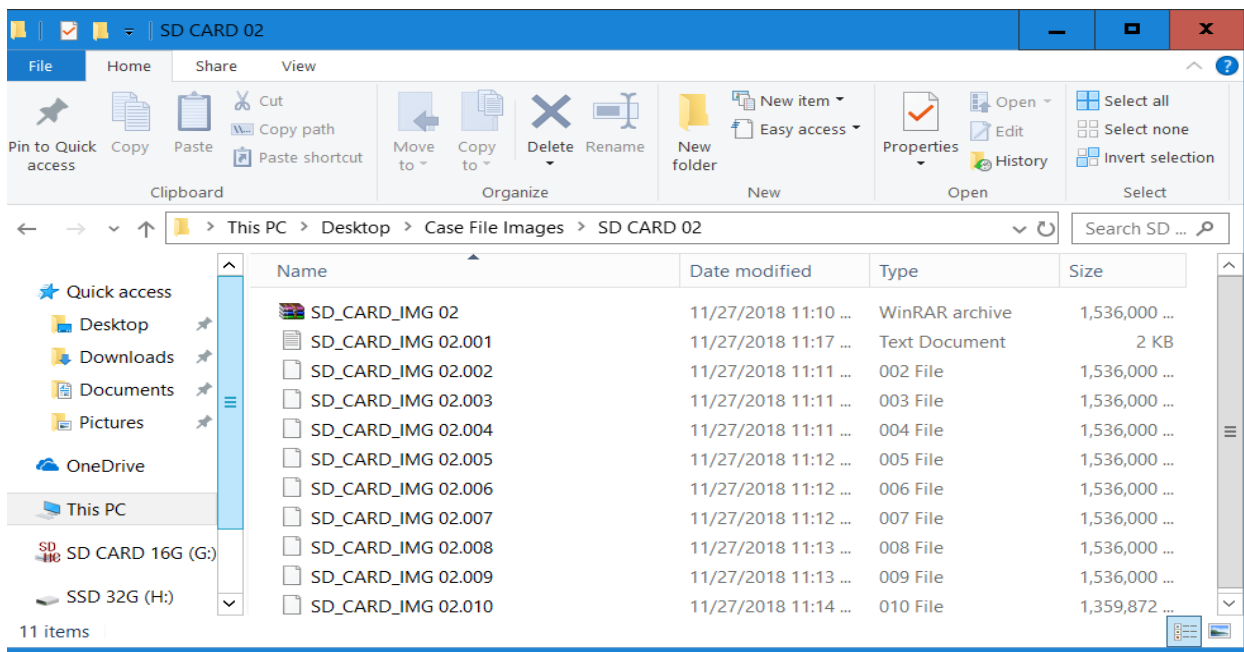


Figure 49: Images of SD card part 21

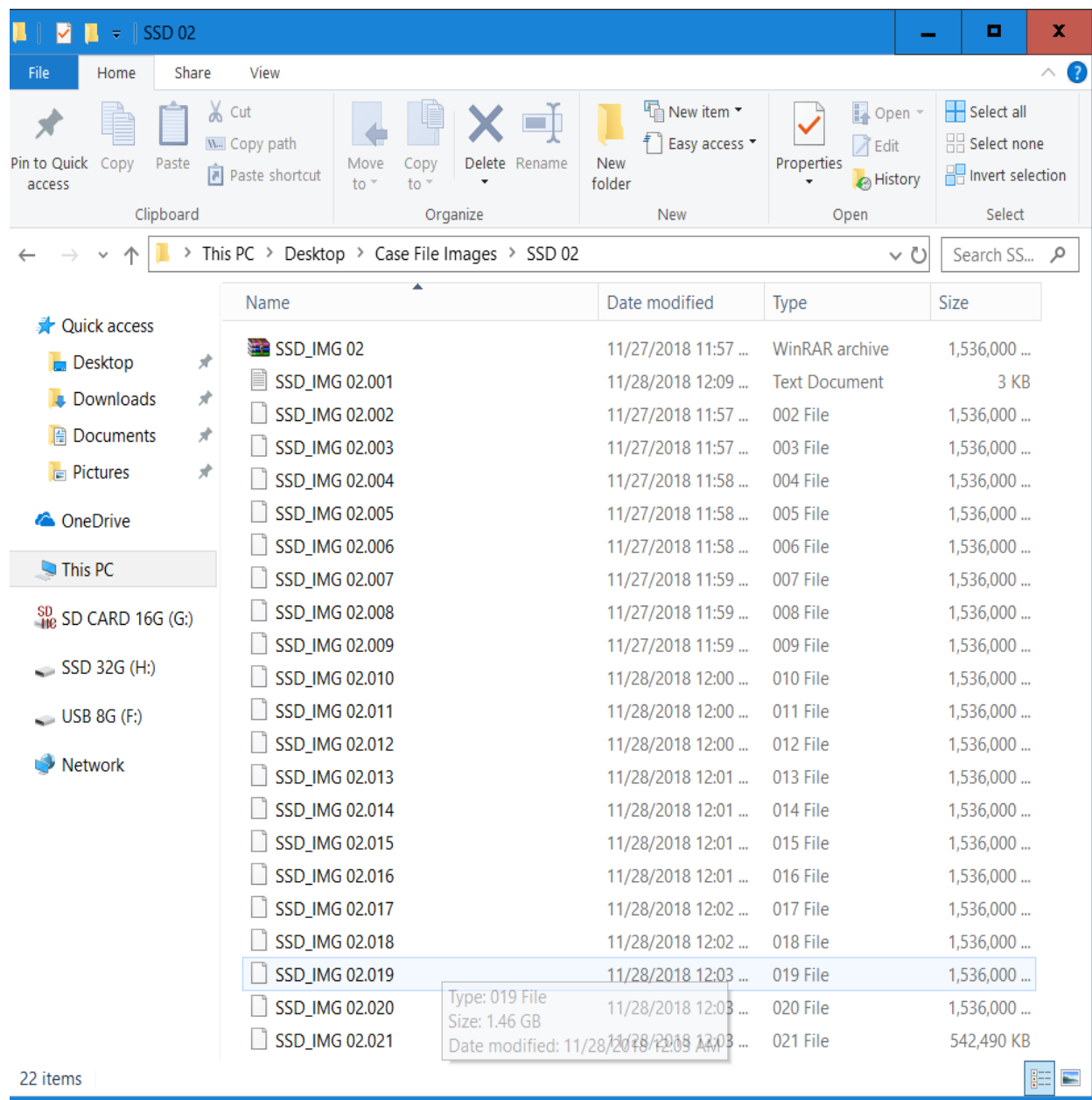


Figure 50: Images of SSD part 2

Figure 48, 49 and 50 shows the image files for USB, SD card and SSD respectively. The process of creating image is similar as in Figure 20 through Figure 43. The two images from each device will undergo analysis to reveal if deleted evidences and other concealed items can be

extracted from them. The next section describes in brief about the process of analysis of the image files.

### **Data analysis**

For the purpose of this experiment, a dummy case file is created, and the contents of the case file is copied into all the three devices. After copying the contents into the memory devices, an image of each device is extracted as IMG 01. After this process, some of the essential files related to the case are deleted on all the three devices and an image of each device is extracted as IMG 02. In this section IMG 01 and IMG 02 will be subjected to inspection to find if all the concealed and deleted items are recoverable using FTK toolkit.

#### **Analyzing USB\_IMG 01**

FTK toolkit software is used for the purpose of analyzing the images of the flash memory devices. The specific search will be done to find if all the hidden items and the concealed evidences are extracted from them.

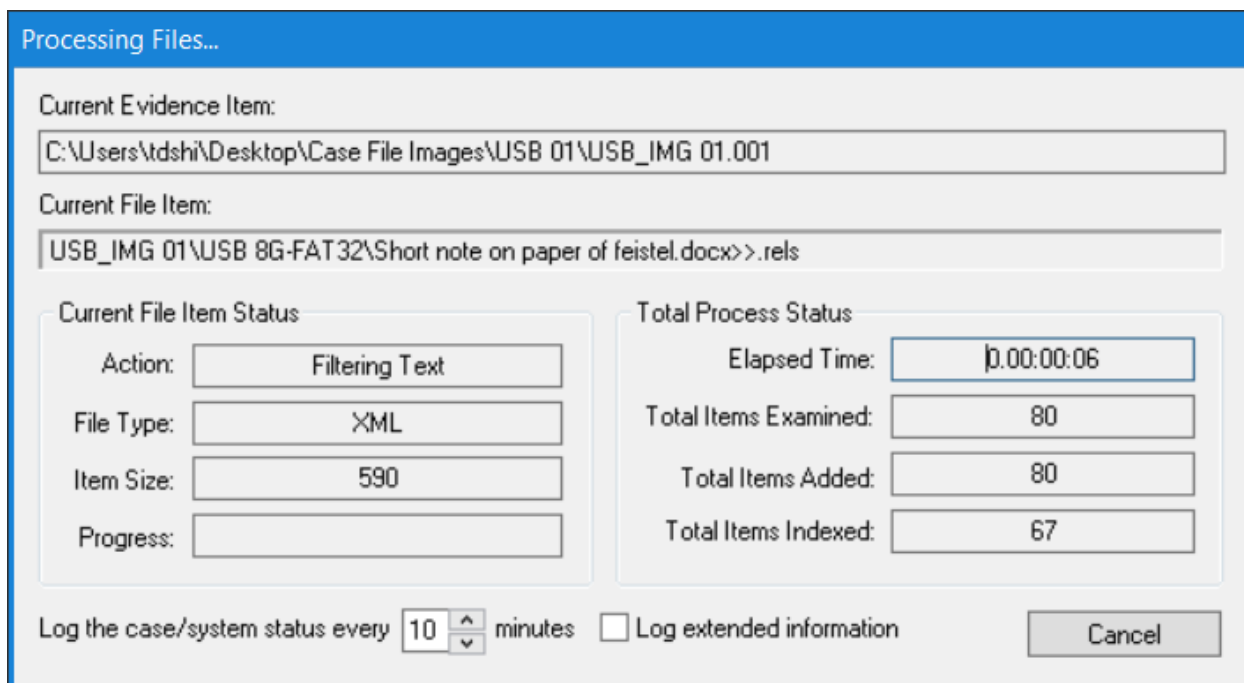


Figure 51: FTK toolkit processing image file USB\_IMG 01

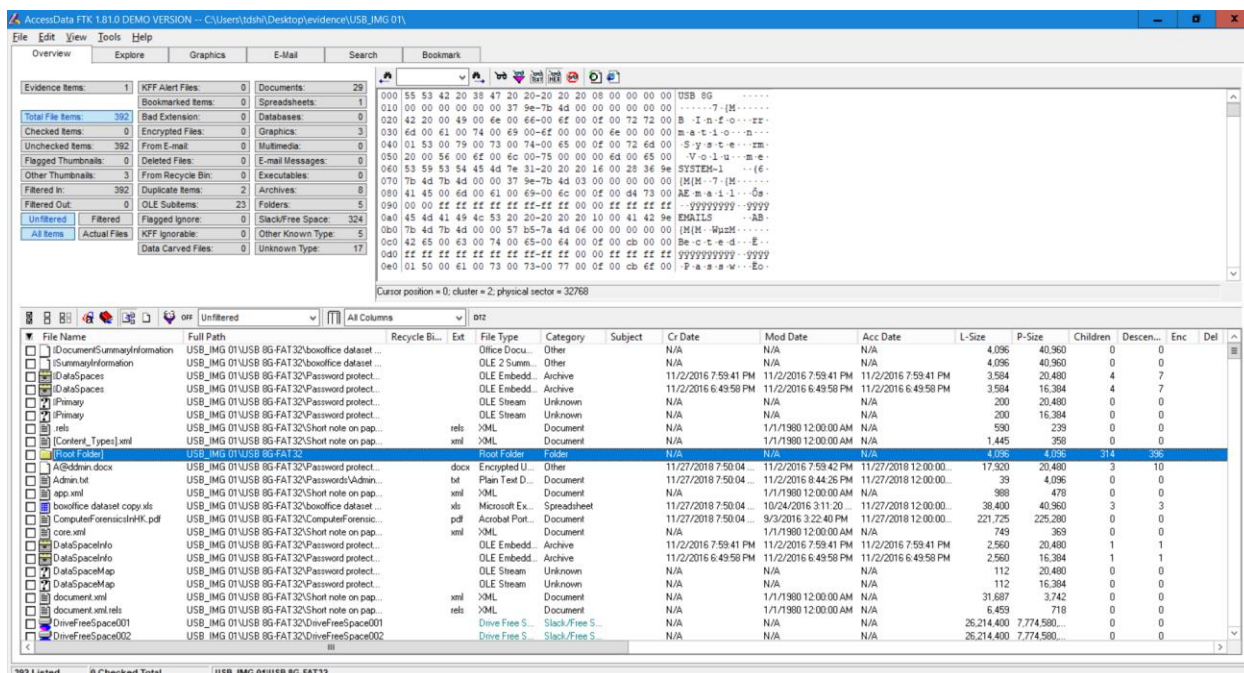


Figure 52: Reading contents of the image file



Once the image is processed, a search is performed to get the hits for emails, passwords and password protected files. Figure 53 shows the number of hits for the keywords. This search ensures that there is no loss of hidden items and concealed items in USB.

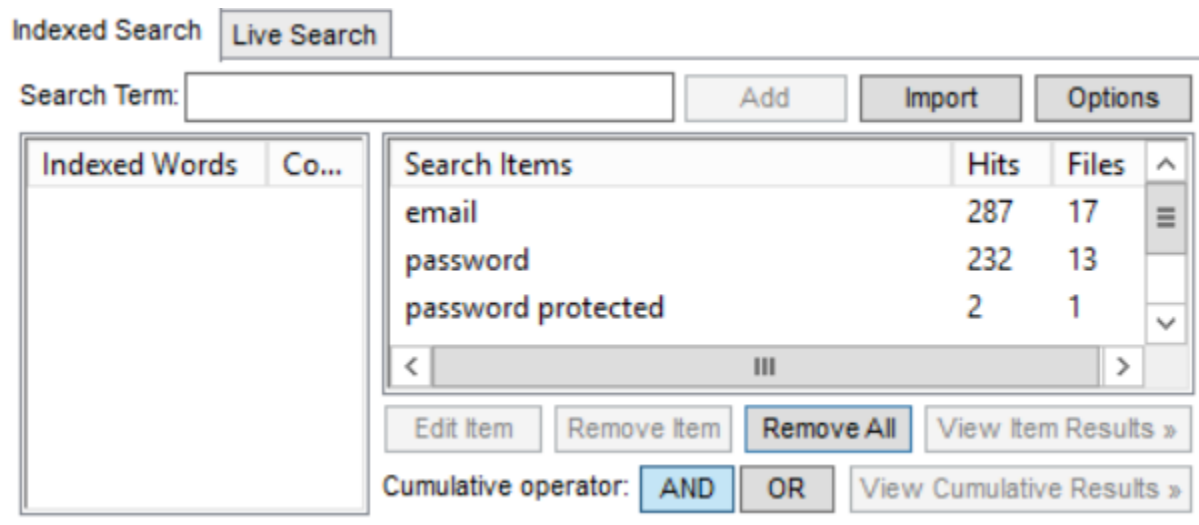


Figure 53: Search for specific files in USB

### Analyzing USB\_IMG 02

The same process is repeated to investigate the image of the USB that was created after deleting certain files on it.

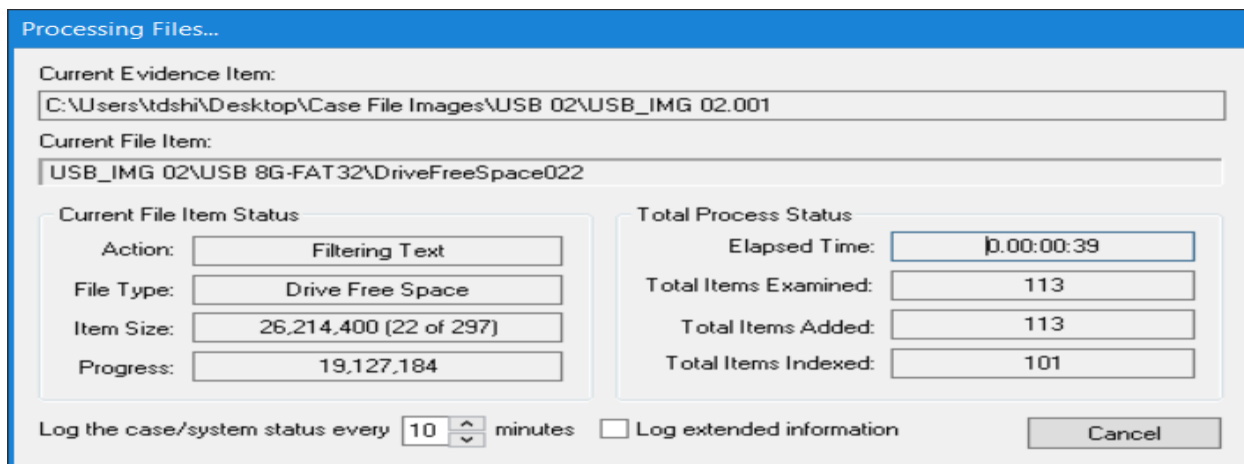


Figure 54: FTK toolkit processing image file USB\_IMG 02

After processing the image file, the image is investigated for deleted items and searched for the specific keywords. From Figure 55 it is evident that all the files are recoverable from the USB after deletion. This is because the search results from USB\_IMG 01 (Figure 53) matches the search results from the USB\_IMG 02. Figure 56 reveals all the deleted content present in the USB drive,

The screenshot shows a search interface with the following components:

- Buttons: Indexed Search, Live Search
- Search Term: [Empty text box]
- Buttons: Add, Import, Options
- Table with columns: Indexed Words, Co...
- Table with columns: Search Items, Hits, Files
- Buttons: Edit Item, Remove Item, Remove All, View Item Results »
- Cumulative operator: AND, OR
- Button: View Cumulative Results »

Search Items	Hits	Files
email	287	17
password	232	13
password protected	2	1

Figure 55: Search results from USB\_IMG 02

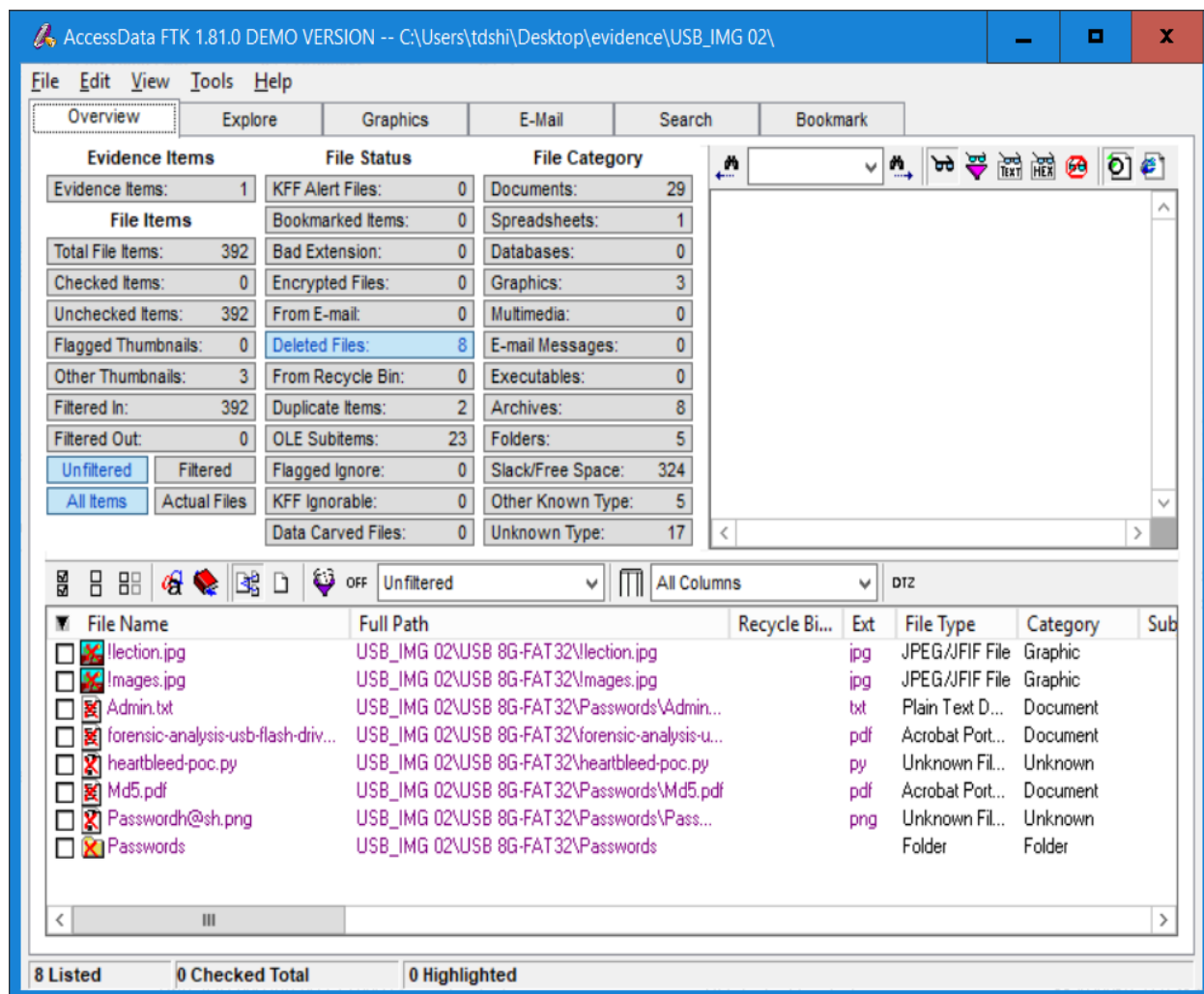


Figure 56: Deleted items on USB

### Analyzing SD\_CARD\_IMG 01

In the next step, SD card image is investigated for concealed and hidden items.

SD\_CARD\_IMG 01 is processed by the FTK toolkit. Once the image is processed, search function is performed to find the hits of specific items on the SD card.

Figure 57 shows the progress of processing the SD\_CARD\_IMG 01. Figure 58 shows the search results obtained from the SD\_CARD\_IMG 01. From the search results it is evident that

the hidden files and concealed items are present on the SD card. These results varies from the results obtained from USB\_IMG 01.

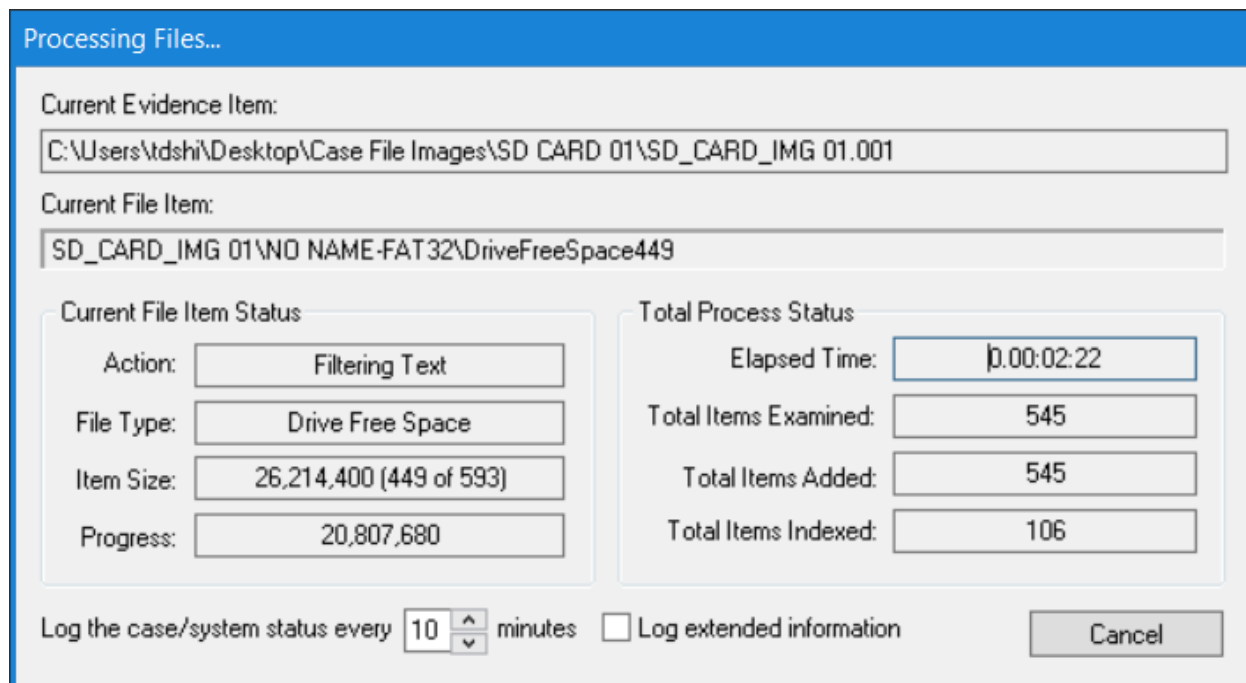


Figure 57: FTK toolkit processing image file SD\_CARD\_IMG 01

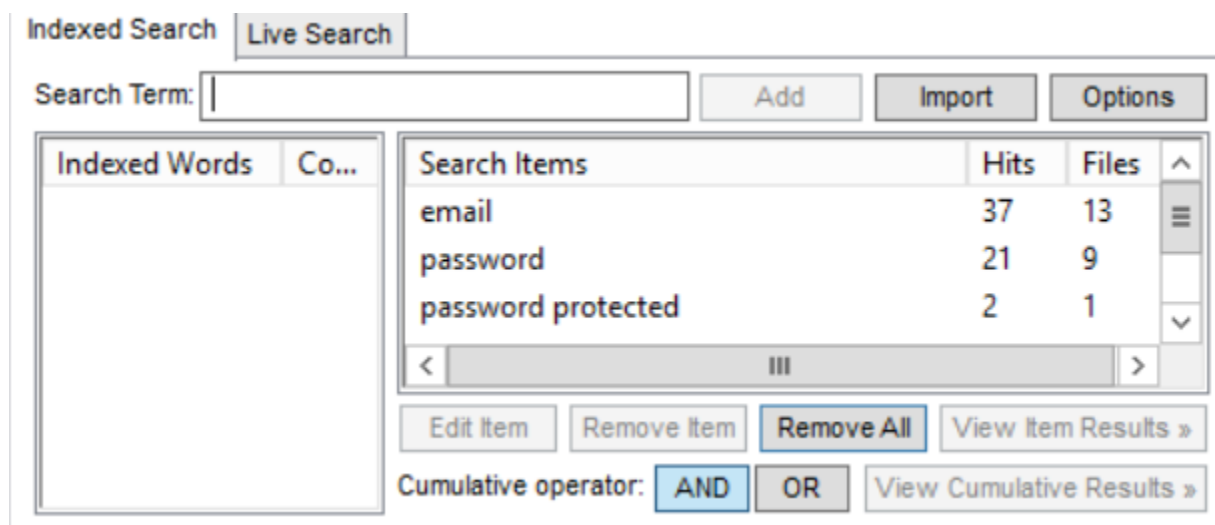


Figure 58: Search results for SD\_CARD\_IMG 01

## Analyzing SD\_CARD\_IMG 02

Image SD\_CARD\_IMG 02 consists the snapshot of the SD card after few essential contents are deleted from it. This image is analyzed using the FTK toolkit to investigate if any deleted items can be extracted from the SD card.

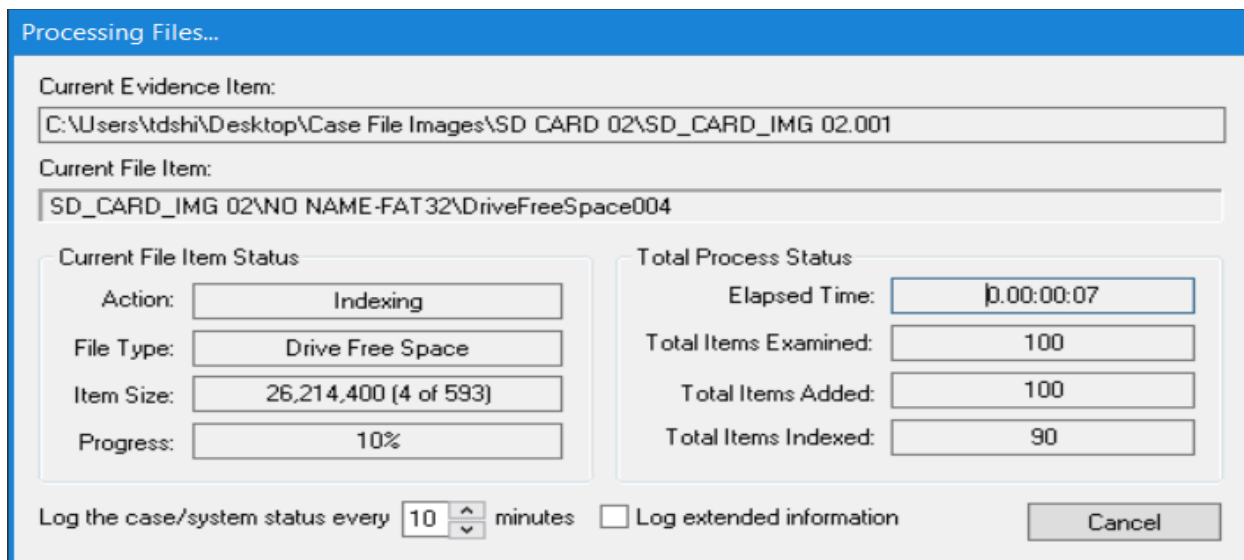


Figure 59: FTK toolkit processing image file SD\_CARD\_IMG 02

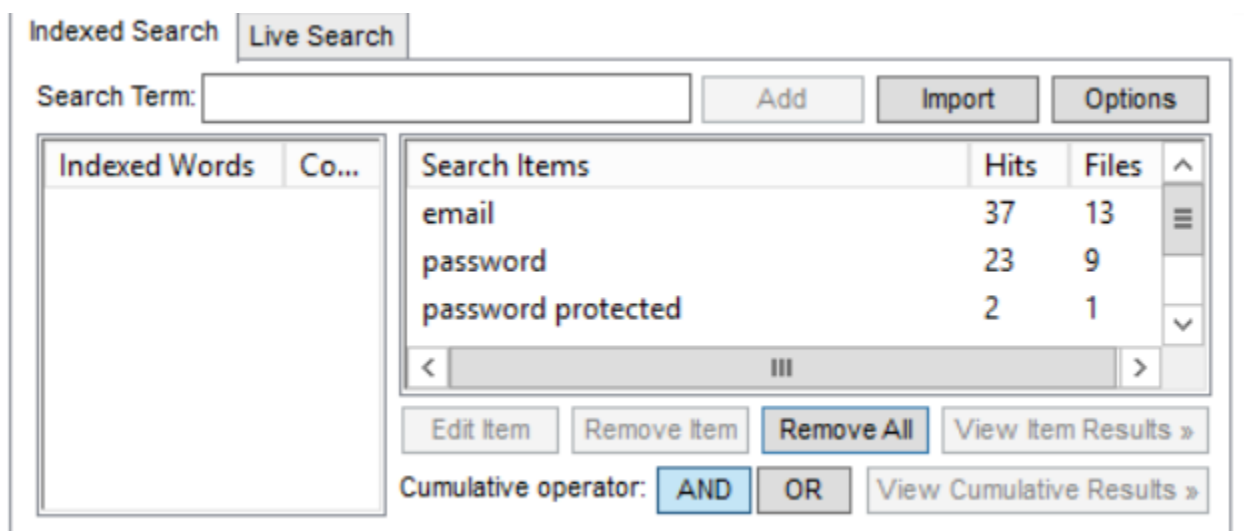


Figure 60: Search results from SD\_CARD\_IMG 02

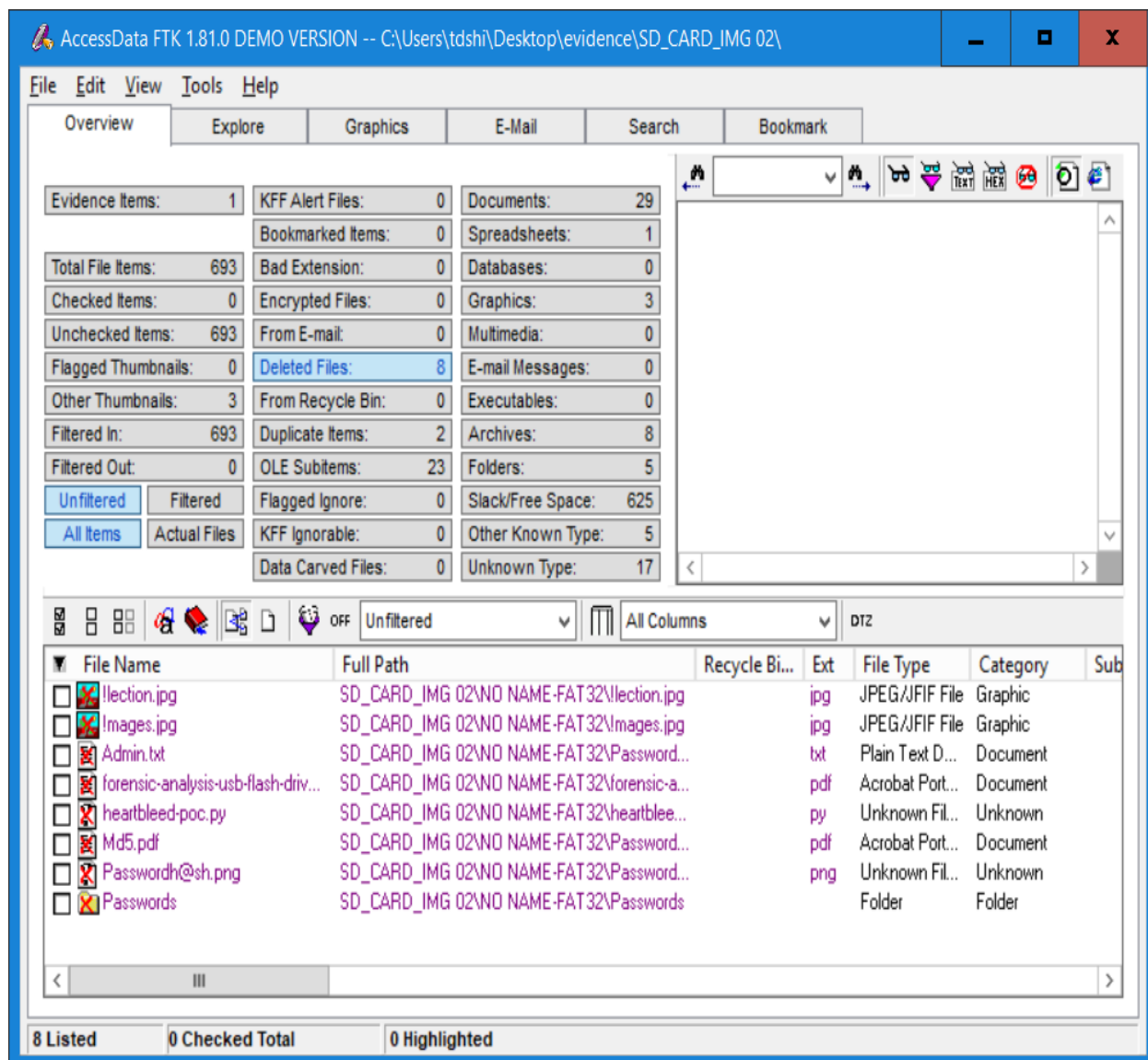


Figure 61: Deleted items on SD card

From Figure 60 it is evident that all the files are recoverable from the SD card after deletion process. This is because the search results from SD\_CARD\_IMG 01 (Figure 58) matches the search results from the USB\_IMG 02. Figure 61 reveals all the deleted content present in the USB drive, However all the deleted items can be recovered from SD card, there is a variation in the results obtained from USB and SD card.

## Analyzing SSD\_IMG 01

In the next step, SSD image is investigated for concealed and hidden items. SSD\_IMG 01 is processed by the FTK toolkit. Once the image is processed, search function is performed to find the hits of specific items on the SSD.

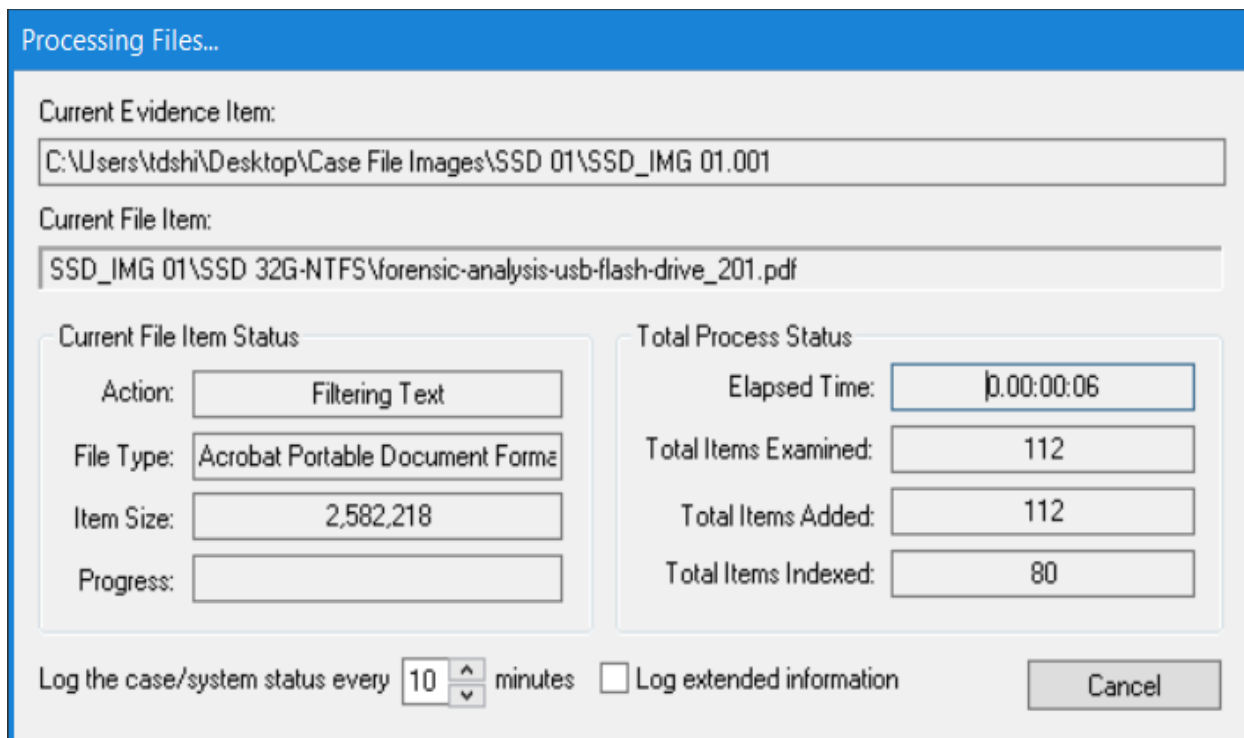


Figure 62: FTK toolkit processing image file SSD\_IMG 01

Figure 62 shows the progress of processing the SD\_CARD\_IMG 01. Figure 63 shows the search results obtained from the SD\_CARD\_IMG 01. From the search results it is evident that the hidden files and concealed items are present on the SD card. These results vary from the results obtained from USB and SSD images.

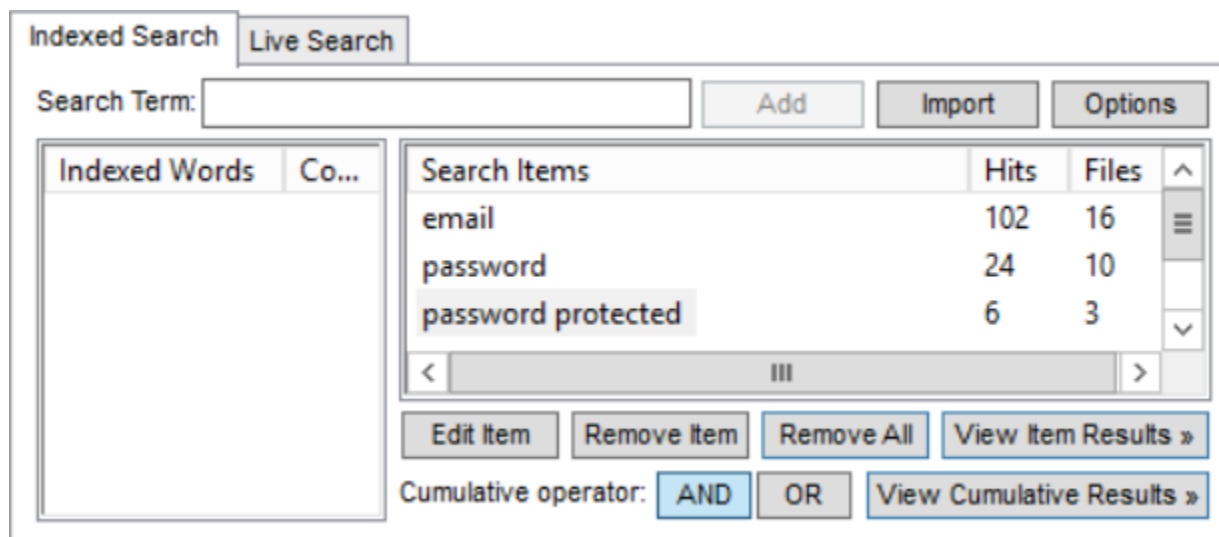


Figure 63: Search results from SSD\_IMG 01

### Analyzing SSD\_IMG 02

Image SSD\_IMG 02 consists the snapshot of the SSD after few essential contents are deleted from it. This image is analyzed using the FTK toolkit to investigate if any deleted items can be extracted from the SD card.

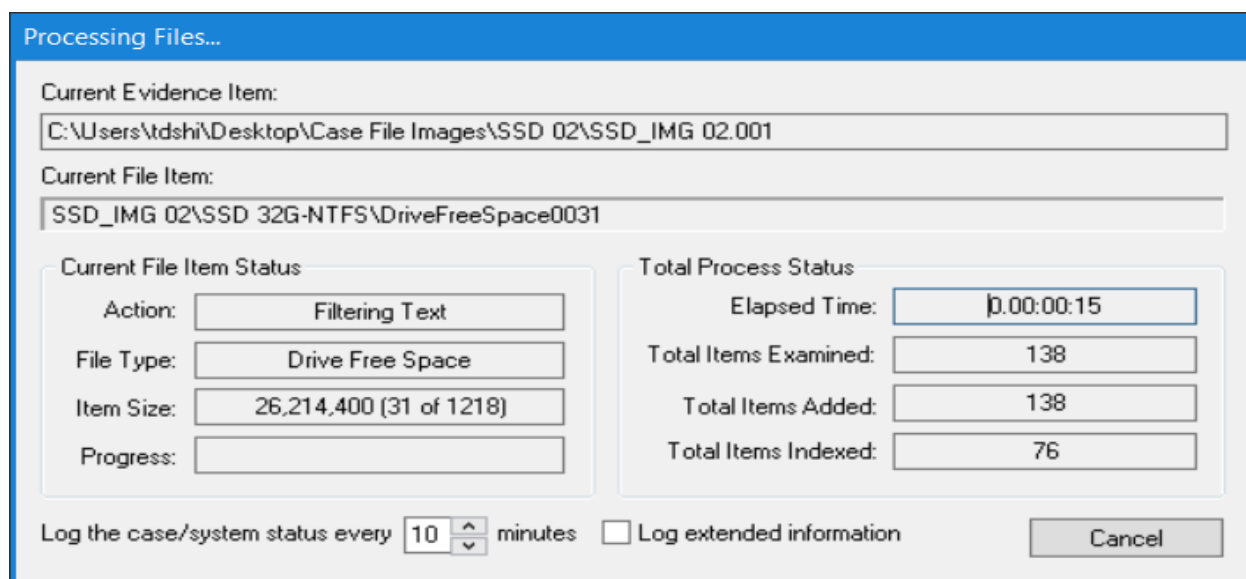


Figure 64: FTK toolkit processing image file SSD\_IMG 02



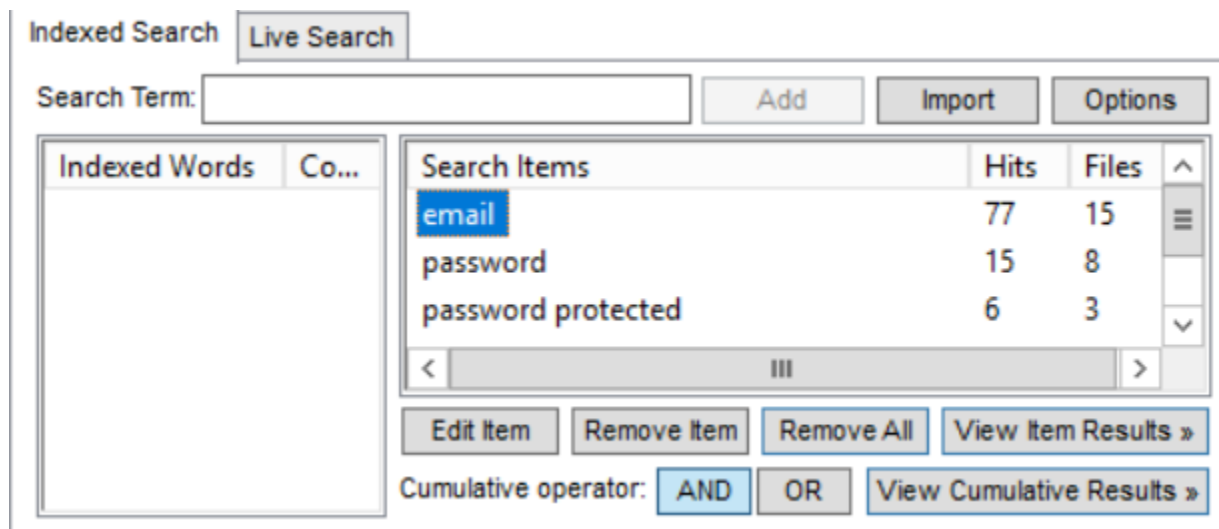


Figure 65: Search results from SSD\_IMG 02

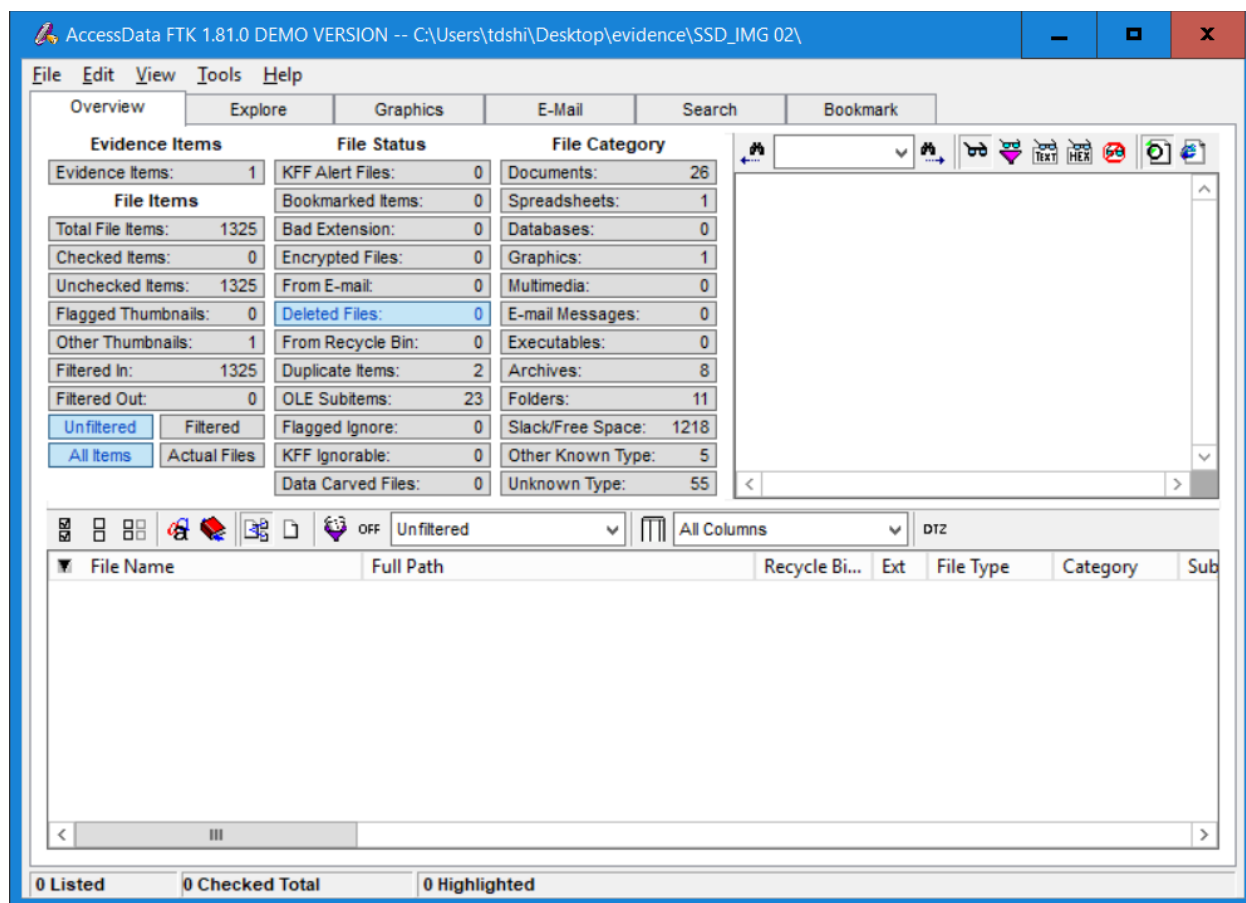


Figure 66: Deleted items on SSD

From Figures 65 and 66 it is evident that the deleted items are not present in the SSD. This proves that SSD doesn't hold any deleted content which leads to loss of evidences from the SSD.

### **Summary**

This chapter discusses briefly about the steps taken to conduct the experiment and collecting the results and analysis of the results. Dummy set of data is created that resembles like a criminal case for the purpose of this experiment. The case is copied on each of the flash device. A snapshot image of each device is extracted using FTK imager before deletion and after deletion. These images are analyzed using FTK toolkit to see if the deleted items are recoverable from all the three devices. From the analysis it is evident that SSD did not reveal any essential information that is related to the case. The next chapter briefly describes the results obtained and analyzes the explores the behavior of each type of flash device when it is undergone forensic analysis.

## **Chapter V: Results, Conclusions and Recommendations**

### **Introduction**

Digital forensics face a huge problem in retrieving deleted information from flash memory devices. To explore the reason for this problem, an experiment is conducted on different types of commercially available flash memory devices. In this experiment, the flash memory devices undergo a forensic investigation after creating and deleting data from them. From the experiment, it is evident that solid state drives do not reveal any deleted data when compared to USB and SD card. This chapter briefly explains the results obtained and the reasons behind the results obtained from them.

### **Results**

The experiment is conducted in two parts. The first part involves creation of a file that resembles an actual case to be investigated using the forensic methods. After creation of these files, these files are copied on all the three flash memory devices. After copying the case files, a snapshot image of the drive is extracted as IMG 01 for all the devices. Figure 67 shows the structure of the case file that is created for the purpose of this experiment.

The second part of the experiment involves deletion of few files from all the three devices that is directly related to the case. All the devices are expected to have identical files and folders in them after deletion. After deletion, a snapshot image of each device is obtained which is named as IMG 02. Figure 68 shows the structure of the case file after deleting certain evidences that is related to the case.

```
C:.\n  boxoffice dataset copy.xls\n  ComputerForensicsInHK.pdf\n  election.jpg\n  forensic-analysis-usb-flash-drive_201.pdf\n  halloween.jpg\n  heartbleed-poc.py\n  images.jpg\n  level17.py\n  Short note on paper of feistel.docx\n\n—Emails\n—Password protected\n  A@ddmin.docx\n  loncustom@r.xlsx\n\n—Passwords\n  Admin.txt\n  Md5.pdf\n  Passwordh@sh.png
```

Figure 67: Structure of the created case file

```
H:.\n  boxoffice dataset copy.xls\n  ComputerForensicsInHK.pdf\n  halloween.jpg\n  level17.py\n  Short note on paper of feistel.docx\n\n—Emails\n—Password protected\n  A@ddmin.docx\n  loncustom@r.xlsx
```

Figure 68: Structure of case file after deletion

After the creation of pre-deletion and post-deletion snapshot images of the drive, the images are analyzed using FTK toolkit. Keyword search is used to query the contents of the drives. These results are compared to find if all the evidences are obtained from all of the drives even after deleting the contents inside them.

Table 6: *Number of files associated with each keyword search*

Number of files	USB		SD card		SSD	
	IMG 01	IMG 02	IMG 01	IMG 02	IMG 01	IMG 02
Email	17	17	13	13	16	15
Password	13	13	9	9	10	6
Password protected	1	1	1	1	3	3

Table 6 summarizes the search results obtained from each of the image files of all the three devices. IMG 01 is the image obtained before deleting contents from the device. IMG 02 is the image obtained after the contents are deleted from the device. The numbers indicate the number of files present that matches a keyword. From the table it is evident that deleted files are recovered in USB and SD card using forensic analysis. SSD does not reveal the deleted files with the standard forensic investigation tools.

Table 7: *Number of hits associated with each keyword search*

Number of hits	USB		SD card		SSD	
	IMG 01	IMG 02	IMG 01	IMG 02	IMG 01	IMG 02
Email	287	287	27	37	102	77
Password	232	232	21	23	24	15
Password protected	2	2	2	2	6	36

Table 7 shows the number of hits associated with each keyword search. It is evident that the number of hits from IMG 01 is almost identical with the number of images from IMG 02 for

both SD card and USB drive. There is a huge difference in hits of IMG 01 and IMG 02 of SSD. This proves that SSD loses the data is lost and unrecoverable along with the files upon deletion of evidences.

The difference in the number of hits and number of files for all the three devices is caused due to the storage behavior of the flash devices. USB and SSD creates a set of additional files that has meta data and logs related to the content saved on the devices. These meta data is not accessible on a standard operating system directory list. In case of SD card, there is no meta data created by the device. Figure 69 and 70 shows the differences in the additional meta files created in SSD but not in SD card.

File Name	Full Path	Recycle Bi...	Ext	File Type	Category	Sub
\$I30	SSD_IMG 02\SSD 32G-NTFS\Emails\I30			Unknown Fil...	Unknown	
\$LogFile_1	SSD_IMG 02\SSD 32G-NTFS\LogFile_1			Unknown Fil...	Unknown	
\$MFT	SSD_IMG 02\SSD 32G-NTFS\MFT			Unknown Fil...	Unknown	
email 1.txt	SSD_IMG 02\SSD 32G-NTFS\Emails\email 1.txt		txt	Plain Text D...	Document	
email 10.txt	SSD_IMG 02\SSD 32G-NTFS\Emails\email 10.txt		txt	Plain Text D...	Document	
email 11.txt	SSD_IMG 02\SSD 32G-NTFS\Emails\email 11.txt		txt	Plain Text D...	Document	
email 12.txt	SSD_IMG 02\SSD 32G-NTFS\Emails\email 12.txt		txt	Plain Text D...	Document	
email 2.txt	SSD_IMG 02\SSD 32G-NTFS\Emails\email 2.txt		txt	Plain Text D...	Document	
email 3.txt	SSD_IMG 02\SSD 32G-NTFS\Emails\email 3.txt		txt	Plain Text D...	Document	
email 4.txt	SSD_IMG 02\SSD 32G-NTFS\Emails\email 4.txt		txt	Plain Text D...	Document	
email 5.txt	SSD_IMG 02\SSD 32G-NTFS\Emails\email 5.txt		txt	Plain Text D...	Document	
email 6.txt	SSD_IMG 02\SSD 32G-NTFS\Emails\email 6.txt		txt	Plain Text D...	Document	
email 7.txt	SSD_IMG 02\SSD 32G-NTFS\Emails\email 7.txt		txt	Plain Text D...	Document	
email 8.txt	SSD_IMG 02\SSD 32G-NTFS\Emails\email 8.txt		txt	Plain Text D...	Document	
email 9.txt	SSD_IMG 02\SSD 32G-NTFS\Emails\email 9.txt		txt	Plain Text D...	Document	

Figure 69: Files results for keyword email in SSD

File Name	Full Path	Recycle Bi...	Ext	File Type	Category	Sub
email 1.txt	SD_CARD_IMG 02\NO NAME-FAT32\Emails\ve...		txt	Plain Text D...	Document	
email 10.txt	SD_CARD_IMG 02\NO NAME-FAT32\Emails\ve...		txt	Plain Text D...	Document	
email 11.txt	SD_CARD_IMG 02\NO NAME-FAT32\Emails\ve...		txt	Plain Text D...	Document	
email 12.txt	SD_CARD_IMG 02\NO NAME-FAT32\Emails\ve...		txt	Plain Text D...	Document	
email 2.txt	SD_CARD_IMG 02\NO NAME-FAT32\Emails\ve...		txt	Plain Text D...	Document	
email 3.txt	SD_CARD_IMG 02\NO NAME-FAT32\Emails\ve...		txt	Plain Text D...	Document	
email 4.txt	SD_CARD_IMG 02\NO NAME-FAT32\Emails\ve...		txt	Plain Text D...	Document	
email 5.txt	SD_CARD_IMG 02\NO NAME-FAT32\Emails\ve...		txt	Plain Text D...	Document	
email 6.txt	SD_CARD_IMG 02\NO NAME-FAT32\Emails\ve...		txt	Plain Text D...	Document	
email 7.txt	SD_CARD_IMG 02\NO NAME-FAT32\Emails\ve...		txt	Plain Text D...	Document	
email 8.txt	SD_CARD_IMG 02\NO NAME-FAT32\Emails\ve...		txt	Plain Text D...	Document	
email 9.txt	SD_CARD_IMG 02\NO NAME-FAT32\Emails\ve...		txt	Plain Text D...	Document	
forensic-analysis-usb-flash-driv...	SD_CARD_IMG 02\NO NAME-FAT32\forensic-a...		pdf	Acrobat Port...	Document	

Figure 70: Files results for keyword email in SD card

## Conclusions

From the results above it is concluded that different types of flash memory device respond differently when subjected to forensic investigation. The reasons behind the difference in behavior is elaborated in brief in this section.

In the case of USB and SD cards, the deleted data is completely recoverable. This is because when data is deleted on a USB or SSD, the data is not actually deleted. It is marked as unimportant. This is because the process of deleting the data will take more time when it is completely wiped off. When the operating system needs more space, the blocks are overwritten with the new data. Overwriting a used block is a time taking process because the block has to be deleted first before new data is written in to the flash cell. Forensic tools try to explore into the devices unused spaces to find out if there is any data that is marked unimportant and retrieves them.

In the case of solid-state drives, the deleted data is not recoverable using the traditional forensic analysis methods. This is because the solid-state devices work differently when

compared to SD card and USB drives. Before any data is written in a SSD flash cell, the flash cell must be emptied. New SSD's comes with empty cells. Therefore, writing data in them is faster with empty cells. But if the drive is full, overwriting new data is a time taking process for SSD's. To overcome this issue, the TRIM command was introduced.

### **The TRIM Command**

Trim command speeds up the process of writing the data into used space on a SSD. The latest versions of operating systems, the TRIM command is by default enabled for SSD's. The command automatically determines which data blocks is no longer usable and wipes them immediately upon further request from the operating system. TRIM is useful for the operating system to determine which blocks are unwanted and returns the addresses of the unwanted blocks back to the operating system. This provides the option of the garbage collection of SSD and skip the invalid blocks instead of rewriting the block itself. This functionality of the trim command provides higher performance during the write operation, by not waiting for the block to be deleted first before it is over written. When a delete command is issued, the data is completely and permanently wiped off from the solid-state device. This makes it almost impossible to recover any deleted files on an SSD.

### **TRIM on external SSD**

TRIM command is by default enabled for operating systems for internal SSD's. The operating system cannot perform TRIM operation on external SSD by default. This is because the TRIM command is a SATA command and operating systems can only send TRIM command over SATA connected SSD. USB interface does not support TRIM to be executed on external SSD.



Few SATA USB adapters gives the opportunity to run TRIM command for externally connected SSD. When a SSD is connected externally using SATA – USB interface, the read and write speeds gradually increase when compared to USB connected external SSD. To perform TRIM on the externally SATA-USB connected SSD, command “Optimize-Volume -ReTrim” is issued form windows PowerShell. Figure 71 shows the image of connecting an SSD using USB - SATA connector.



*Figure 71:* Externally connected SSD using SATA-USB device

Figure 72 shows how trim operation can be performed using “Optimize Volume - ReTrim” command. This command runs only through PowerShell with administrator privileges.

This command helps to perform optimization functions like defragmentation, trim, storage consolidation and storage tier processing. It helps creating more space on a hard drive.

```
PS C:\WINDOWS\system32> Optimize-Volume -DriveLetter H -ReTrim -Verbose
VERBOSE: Invoking retrim on SSD 32G (H:)...
VERBOSE: Performing pass 1:
VERBOSE: Retrim: 0% complete...
VERBOSE: Retrim: 100% complete.
VERBOSE:
Post Defragmentation Report:
VERBOSE:
Volume Information:
VERBOSE: Volume size = 29.79 GB
VERBOSE: Cluster size = 4 KB
VERBOSE: Used space = 73.84 MB
VERBOSE: Free space = 29.72 GB
VERBOSE:
Retrim:
VERBOSE: Backed allocations = 29
VERBOSE: Allocations trimmed = 28
VERBOSE: Total space trimmed = 26.95 GB
PS C:\WINDOWS\system32>
```

Figure 72: Performing trim operation on externally connected SSD

### Self-corrosion of SSD

SSD's also destroy evidence through the process of self-corrosion. Garbage collection process will be running in the background in most of the SSD's. This process collects addresses of unimportant blocks or sectors on a SSD. Garbage collection also destroys the data that is marked as deleted. Therefore, when a data is deleted, the background process automatically wipes off the data and which makes the data not recoverable in the future. This process cannot be prevented in any way. This behavior of SSD is known as self-corrosion. (Gubanov & Afonin, Why SSD Drives Destroy Court Evidence, and What Can Be Done, 2012).

### **Future Work**

With the help of this experiment, it is proven different types of flash device respond differently when it is subjected to forensic investigation. SSD makes it harder for the forensic investigators to extract deleted data using traditional forensic investigation methods. Some of the possible cause for this problem is discussed in this paper. There is no solution proposed that can overcome this issue. Future research can be focused on theorizing possible solutions and testing them so that forensic investigators can easily recover deleted evidences from SSD. This helps in creating a chance for less crime rates and easily find a possible suspect in a criminal activity.

## References

- Audrey. (2015, July 27). *EVERYTHING YOU NEED TO KNOW ABOUT SLC, MLC, & TLC NAND FLASH*. Retrieved from Digital Discount:  
<https://www.mydigitaldiscount.com/everything-you-need-to-know-about-slc-mlc-and-tlc-nand-flash.html>
- Bem, D., & Huebner, E. (2007). Analysis of USB Flash Drives in a Virtual Environment. *SMALL SCALE DIGITAL DEVICE FORENSIC*.
- Bez, R., Camerlenghi, E., Modelli, A., & Visconti, A. (2003). Introduction to Flash Memory. *IEEE*.
- Brant, T. (2018, March 26). *SSD vs HDD*. Retrieved from pcmag:  
<https://www.pcmag.com/article2/0,2817,2404258,00.asp>
- Breeuwsma, M., Jongh, M. d., Klaver, C., Knijff, R. v., & Roeloffs, M. (2007). Forensic Data Recovery from Flash Memory. *SMALL SCALE DIGITAL DEVICE FORENSICS JOURNAL*, 1-2.
- Bui, S., Enyeart, M., & Luong, J. (2003). *Issues in Computer Forensics* .
- Burr, G., Kurdi, B., Scott, J., Lam, C., Gopalakrishnan, K., & Shenoy, R. S. (2008). Overview of candidate device technologies for storage-class memory. *IEEE*.
- Deng, Y., & Zhou, J. (2011). Architectures and optimization methods of flash memory based storage systems. *Journal of Systems Architecture - Embedded systems design*.
- Fulford, B. (2002, 06 24). *Unsung hero*. Retrieved from Forbes:  
<https://www.forbes.com/global/2002/0624/030.html>

- Gal, E., & Toledo, S. (2005). *Algorithms and Data Structures for Flash Memories*. ACM computing surveys.
- Gibson, W. C., & Cohen, J. S. (2014, March 20). *The Deletion of Data is Often Key Evidence in Proving Facts of a Case*. Retrieved from Vorys: <https://www.vorys.com/publications-1219.html>
- Gubanov, Y., & Afonin, O. (2012). *Why SSD Drives Destroy Court Evidence, and What Can Be Done*. Retrieved from Belkasoft:  
<https://belkasoft.com/download/info/SSD%20Forensics%202012.pdf>
- Gubanov, Y., & Afonin, O. (n.d.). *Why SSD Drives Destroy Court Evidence, and What Can Be Done About It*. Retrieved from Belkasoft Ltd:  
<https://belkasoft.com/download/info/SSD%20Forensics%202012.pdf>
- Huang, P.-C., Chang, Y.-H., Kuo, T.-W., Hsieh, J.-W., & Lin, M. (2008). The behavior analysis of flash memory storage systems. *IEEE Symposium on Object Oriented Real-Time Distributed Computing (ISORC)*.
- J., P. B., D., S. C., & R., K. D. (2008). Recovering data from USB Flash memory sticks that have been damaged or electronically erased. *Proceedings of the 1st international conference on Forensic applications and techniques in telecommunications, information, and multimedia and workshop*. Adeliade, Australia.
- Kay, R. (2010, June 7). *Flash memory*. Retrieved from Computer World:  
<https://www.computerworld.com/article/2550624/data-center/flash-memory.html>
- Kruse, W. G., & Heiser, J. G. (2001). *Computer Forensics: Incident Response Essentials*. Boston: Addison.

- Luck, J., & Stokes, M. (2008). An Integrated Approach to Recovering Deleted Files from NAND Flash Data. *SMALL SCALE DIGITAL DEVICE FORENSICS JOURNAL*.
- Me, G., & Distefano, A. (2008). An Overall Assessment Of Mobile Internal Assessment Tool. *The Digital Forensic Research Conference*. Baltimore: DFRWS.
- Poole, I. (n.d.). *Flash Memory Wear, Reliability & Life*. Retrieved from Radio electronics: <http://www.radio-electronics.com/info/data/semicond/memory/flash-wear-reliability-lifetime.php>
- R, N., J, K., & R, K. (2015). NAND Flash Memory Organization and Operations. *Information Technology & Software Engineering*.
- Regan, J. E. (2009). *THE FORENSIC POTENTIAL OF FLASH MEMORY*. California.
- Roubos, D., Palmieri, L., Kachur, R. L., Herath, S., Herath, A., & Costantino, D. D. (2007). A Study of Information Privacy and Data Sanitization Problems. *Journal of Computing Sciences in Colleges* .
- Rouse, M. (2015, March). *flash memory*. Retrieved from SearchStorage: <http://searchstorage.techtarget.com/definition/flash-memory>
- Rouse, M. (n.d.). *TLC flash (triple-level cell flash)*. Retrieved from Tech Target: <https://searchstorage.techtarget.com/definition/TLC-flash-triple-level-cell-flash>
- Rouse, M., & Kranz, G. (2016, May). *SSD*. Retrieved from TechTarget: <http://searchsolidstatestorage.techtarget.com/definition/SSD-solid-state-drive>
- Sansurooah, K. (2009). A forensic overview and analysis of USB flash memory devices. *Australian Digital Forensics Conference*. Edith Cowan University.

Satti, R. S., & Jafari, F. (2015). Reviewing Existing Forensic Models to Propose a Cyber Forensic Investigation Process Model for Higher Educational Institutes. *I. J. Computer Network and Information Security*.

Skorobogatov, S. (2005). Data Remanence in Flash Memory Devices. *Cryptographic Hardware and Embedded Systems-CHES 2005*. Edinburgh.

Techopedia. (n.d.). *Digital Forensics*. Retrieved from Techopedia:

<https://www.techopedia.com/definition/27805/digital-forensics>

Woodford, C. (2017, June 29). *Flash memory*. Retrieved from ExplainThatStuff:

<http://www.explainthatstuff.com/flashmemory.html>