Forensic Potentials of Solid State Drives

Zubair Shah(✉), Abdun Naser Mahmood, and Jill Slay

School of Engineering and IT, University of New South Wales, Canberra, Australia
Zubair.Shah@student.afda.edu.au, {A.Mahmood,J.Slay}@adfa.edu.au

Abstract. Extracting useful information from Solid State Drives (SSD) is a challenging but important forensic task. However, there are opposing views [14,15,22] that (1) SSDs destroy the forensics evidences automatically and (2) even after sanitization of SSDs, data can be recovered. This paper investigates this issue and reports experimental findings that identify the reason why certain SSDs seem to destroy forensic evidences while other SSDs do not. The experiments provide insight and analyses of the behaviour of SSDs when certain software components, such as Background Garbage Collector (BGC) and Operating System functions, such as TRIM, are executed on the SSD.

Keywords: Forensics · Solid state drives · SSD

1 Introduction

In recent years, more and more criminal investigations have centered on finding digital evidences extracted from computing devices, such as Computers, Mobile Phones and Notebooks. The evidences of crimes in physical dimensions are in tangible form; however the evidence of cyber-crimes exists electronically.

The investigation process of cyber-crimes often begins from the analysis of the storage media. Every computing device stores its data on the storage media and every activity of the computing device leaves some traces on the storage media. Meta-data of the electronic media can contain more useful information such as date, time, keys and often this meta-data have greater acceptability than paper based evidences [2,3]. However, if an inefficient recovery is performed then these evidences can be altered, therefore, would become erroneous. Consequently, any change in these evidences may impact court proceedings as well [4,5].

The evidence collected in recovery process requires confirmation to assess its reliability and integrity and it is really important to identify any loss and alteration that has happened in the recovery process [6]. If the data collected for the forensic purpose is altered or lost then it is the responsibility of the party submitting the evidence to prove the integrity of the data. If not, the opposing party can raise questions about the integrity of evidences [7]. Avoiding alteration or loss during the recovery process depends on the error free data recovery mechanism. Usually, write blocking along with bit stream copying process is used in the recovery process. This mechanism allows recovery of the data along with completeness, precision and reliability [8].

J. Tian et al. (Eds.): SecureComm 2014, Part II, LNICST 153, pp. 113–126, 2015.
DOI: 10.1007/978-3-319-23802-9_11

To reduce loss or alteration, it is necessary for the recovery process to thwart overwriting of data on the relevant drive. For example all the processes need to be stopped by shutting down the system before creating forensic image of the disk in order to minimize the chance of alteration or loss of data [9] by processes in memory. Also, hash value is calculated for collected forensic image in order to check the integrity of the forensic data. This hash value can validate if the forensic image is created multiple times or if the forensic image is placed in some place where alteration is possible in the forensic image [9].

Hard Disk Drives are magnetic storage devices that have well known forensic properties. Most computing activities that rely on disk access, including illegal activities, leave traces that can be later identified through forensic investigation. SSD is a newer technology and a superior alternative to HDDs that offers many benefits over HDD [10] such as read/write speed, durability against shock vibration and temperature. However SSD has some limitations such as life time of a cell in terms of writing data on it (10,000-100,000 times) and need of erasing the blocks before rewriting on the same block [11].

Wear leveling [1] technique is used to randomly select the pages for rewriting the data which prohibit the blocks from approaching the critical failure conditions due to overuse. To solve the problem of erasing the pages before writing, BGC [12] and TRIM command [13] are proposed. Background Garbage Collection (BGC) is a mechanism used in current SSD controllers to improve the write speed of data by deleting/zeroing the unused/garbage pages.Similarly TRIM is a command in modern operating system to inform the SSD controller that particular blocks of data are no longer required or not in use and should be wiped internally. BGC and TRIM commands are the two sources that could destroy the evidences which otherwise could be available for the recovery.

From existing literature it is evident that SSDs destroy forensics evidences and there is no chance to recover the deleted data by any means [14]. However, some research also points that existing data sanitization techniques available for HDDs are not useful or not sufficient for SSDs and new techniques are required specifically for SSDs [15]. The term Data sanitization has different meanings such as nulling out, masking data, shuffling records, encryption and censorship etc. In our context data sanitization means nulling out data to prevent its recovery by any means.

The purpose of this paper is to study the forensics potentials of SSDs of different manufacturers and to experimentally verify the availability or unavailability of the data after deleting or formatting the SSDs.

We have experimentally verified that SSDs destroy forensics evidences only if either the firmware of SSDs has BGC functionality enabled or if TRIM command is supported by SSD's firmware and configured properly in operating system and associated software. It has been our finding that in the absence of BGC and TRIM command support, SSDs do preserve data and live acquisition is possible like traditional HDDs. Sometimes, data can also be recovered from an SSD even after it has been formatted.

Rest of the paper is organized as follow, Sect. 2 presents some preliminaries about SSDs and forensics, related study is elaborated in Sects. 3 and 4 presents our methodology, experiments and results. Conclusion is given in Sect. 6.

2 Preliminaries

2.1 Solid State Drive (SSD)

SSD [16] is an emerging technology for storing data persistently, and slowly replacing the leading HDD storage technology. SSDs are quite different from HDD. For example, SSDs don't have electromechanical component and thus are much faster than traditional HDD. SSD stores data in microchips just like USB flash drive. They store data or retrieve files instantly and do not need to wait for moving parts to position on required sector of magnetic platter. However SSD suffer from a problem which does not exist in HDDs. They first need to erase a block before a new data can be written into it [17,18]. This obviously causes problems for successfully retrieving forensic information from the drives.

2.2 SSD and Forensics

With the emergence of SSD technology computer forensics faced newer challenges than traditional HDD. The SSD devices are usually based on flash memory such as battery backed SRAM or DRAM which includes flash backing storage. These types of memories include some key features which complicate forensics analysis [11,19]. For example;

- Flash memory is divided into pages of 2 KB, 4 KB or larger instead of 512 bytes blocks as in HDD.
- Flash memory pages must be erased before performing write operation instead of just writing in a single pass as in HDD.
- Rewriting a block does not necessarily rewrite on the same page because of wear leveling mechanism employed in SSDs.
- Each page of SSD has a number of write and erase cycles typically 10,000 to 100,000.
- Before storing the data on SSD it is often encrypted, erasing the encrypted data is done by deleting the older encryption key and generating a new one and marking those as garbage.

The SSD controllers are considerably more complex in performing the task of reading and writing data on to media as compared to HDD, with the following distinguished features [11,19].

- *Wear Leveling:* It is a mechanism which is used to avoid a block to physically wear out quicker than other blocks by spreading the data eventually. Using wear leveling technique, the firmware of SSD uses all the blocks evenly instead

of using few blocks repeatedly and reducing their life. SSDs have Flash Translation Layer (FTL) which is used to perform wear leveling. It maps logical sectors to physical pages. FTL is contained within SSD and are not accessible to end users.

– *Read, Modify, Relocate+Write:* When a partial page is required to modify, the firmware first reads the entire page into a cache built inside SSD, then it modifies blocks being written and writes the new page in a new location. The older page is marked for garbage collection.

These features are very good from forensics point of view because a block being modified might be available in cache or in its previous location if it is still not wiped internally by SSD. However, there are three other issues that complicate forensics evidence gathering because they make the data recovery almost impossible.

1. *Garbage Collector*
   SSD uses garbage collection mechanism to improve its write speed [12]. Write performance is improved by eliminating the need of erasing before writing. The erasing operation is performed in background and during free time when controller is not busy. GC accumulates data blocks which are marked unused by erasing it and reclaim blocks for reuse for later write operations [17,18]. However GC has implications on computer forensics. It operates independently without the need of intervention from the operating system. After about 150 s of power on, GC starts erasing the garbage blocks previously marked by the file system [14,17,18]. Therefore, there is a risk that the GC may delete the content of the media even during performing forensic copy in the lab.
2. *TRIM*
   TRIM is a command in modern operating system to inform the SSDs controller that particular blocks of data are no longer required or not in used and should be wiped internally. In the absence of BGC, TRIM command is an alternate to improve write performance of SSDs. It enables the controller to handles the garbage collection overhead in advance, which could otherwise significantly slow down future writes. In order for the TRIM command to work, the SSDs firmware, operating system and associated software must be properly configured. Usually modern operating systems such as Windows 7 have built-in TRIM command utility that can be configured in BIOS settings. Since this command if configured properly completely purges the data, therefore, the data recovery will becomes impossible.
3. *Encryption and Compression*
   Modern SSD controllers perform compression and encryption on data before saving them on the disk. Compression increases the speed of writing data on SSD and also allows more data to be stored on SSD. The encryption of data before writing to SSD's cells has two advantages. First it improves security and secondly this technique enables controller to erase entire SSD disk. Rather than wiping the entire media, deleting the encryption key leads

to the inability to recover or read the data. So in the forensic analysis even if the data is recovered without knowing or recovering its encryption key, it is usually impossible to read the recovered data and it may cause difficulty in the way of forensic analysis.

3 Related Work

In this Section we discuss the literature on recovery of data from flash based memories. In [20] Luck et al. recommended a three stage approach to retrieve files in general and video files in particular from a mobile phone (Containing NAND Flash Memory). During the first stage, the authors illustrate the method of renewing FAT and distillation of extant files by building version table which includes all available versions of logical sectors. In the first stage the authors have further described a six step approach which contain (i) Building Version Table, (ii) Rebuilding File Allocation Table (FAT) volume, (iii) Analyzing Volume Boot Record (VBR), (iv) Extracting directory, (v) Extracting extant files and (vi) Recovering lost chains and lost files. The main goal of all those six steps is building a data structure or rebuilds a file system that maps the logical data abstracted to physical location.

In the next stage authors' aim was to find again a chain of clusters and files. They described that although the directory entry is overwritten in many cases but cluster chain is still in the phone memory and need to secure all chains that exist in the memory, including all lost and partial fragments of lost chains [20]. The authors have described MPEG-4 3 GP file format and suggested that it is important for the forensic examiner to understand MPEG-4 3 GP file system as it helps in reconstructing deleted videos. In the third stage of their approach they used a technique called "Xtractor". The purpose of Xtractor is to play incomplete video by playback software like Apple QuickTime 7. They showed that as defective sectors can be recognize and replaced with null sectors (0×00) and using Xtractor they could still be played.

Although the research by Luck et al. is very useful for data recovery from NAND flash, however, it is related to the memories of the mobile phones. First memories installed in mobile phones do not apply "garbage collection" and as an end result the deleted files may still be present in memory and could be recovered by the approach suggested by the author. Second the approach is well elaborated and tested for video data (i-e MPEG-4) only. Therefore, the approach has very limited application in the SSDs forensics and data recovery. The only link that could be established is the process of rebuilding the FAT volume by building a version table containing all available versions of logical sectors. But it is still limited to FAT12 or FAT16 in mobile phones where first entry point is VBR rather than Master Boot Record (MBR).

In [15] Freeman et al. tested possible available tools and procedures for securely deleting data from SSDs. They found that all tools except GNU core utility dd left some file information which was recovered, but none of the recovered files were workable.

Authors started their explanation from the fact that SSDs store files in 4 KB page, yet data can only be deleted in 512 KB blocks. The procedure stores pages in disk controller cache as the file is being deleted, the disk controller remove all the pages from the block. Once the pages are removed from the block, the required authentic data is fetched from the cache and reallocated on an available block. The reset block is added to the SSDs free space [21]. Every 32 GB SSDs have 2.2 GB space which is used as cache and it is not visible to operating system. The controller of SSD uses this additional free space to save files, that reduces the need for the deletion of blocks that keeps the drive at best performance [15]. Authors uses 32 GB PQI SATA II 2.5 in. SSD. They have used the drive to connect to secondary SATA port. They formatted SSD as NTFS and for experimental purpose they saved and deleted data of varying size and file type. dd (GNU core utility), Eraser (version 5.8.7), Wipe and SDelete were the tools they have tested for data deletion/sanitization and Scalpel was used for file carving purposes.

The approach and findings of Freeman et al. proposed that there is no (except dd of GNU) available tool that can guarantee completion deletion of data from SSDs. Authors note that "Even after employing eraser tools to delete the data from SSD there is still remnant data in SSDs that could be recovered".

In the article [22] Wei et al. have discovered the inability or difficulty of deleting/purifying data from the SSDs. The authors have conducted a number of experiments with the aim of finding any remnant data after applying (1) built-in ATA or SCSI commands for sanitization and (2) software based sanitization. Authors conducted several experiments and showed results of experiments and the percentage of data they had recovered after applying different techniques. They claimed that none of the existing hard drive-oriented techniques for individual sanitization are effective on SSDs [22]. They showed that the sanitization of the SSD with currently available tools is extremely difficult and the tools available for sanitizing the HDD cannot be used to sanitize the SSD. Using these tools to sanitize the SSD will leave data in the SSDs which can be recovered by sophisticated software.

In [14] Bell et al. reports about "self-corrosion" which is actually caused by "garbage collection" mechanism employed in entrenched controllers of modern SSDs. The authors used only 64 GB P64 Corsair SSD directly connected to the secondary SATA channel on the motherboard. Authors tested the data to see what portion of the sampled bytes were "zero bytes". The experiment shows that almost all the data were zeroed within 300 s.

After a single run of GO program the authors managed a forensic analysis of the SSDs. They were able to recover 1090 files out of 316,666 files, none of which could be used to reconstruct the original file. They conducted various experiments on the same SSD and found that the SSD is able to delete the data automatically even during construction of forensic image.

At the end the authors provided a list of guidance for forensics of SSDs and claimed that the "golden age for forensic recovery and analysis of deleted data

and deleted meta-data may now be ending" [14]. From the literature review two opposite and interesting facts are revealed.

- "Even after applying sanitization techniques on SSDs there are still remnant data" [15,22].
- "Golden age for forensic recovery and analysis of deleted data and deleted meta-data may now be ending" [14].

The first view is that even if someone tries to remove the data in any possible way then there is still chance of leftover part of the data. In other words, it is not easy to accurately delete data from SSDs using the conventional techniques [15,22]. The second view is that the SSDs controller removes almost all the data and hardly any data could be recovered, for example, even during a quick format which does not require erasing the data [14]. This has motivated us to conduct further experiments and possibly find support for either view.

4 Proposed Method

As the results and conclusions from [14,15,22] had gone into the opposite directions. So it seems that there is a gap that needs to be filled. This is the main motivation of our research and that's why we aimed to conduct the SSDs forensics analysis further under a number of possible assumptions that these authors might had missed and possibly fill the gap between their results. Our experimental setup and assumptions are different than those employed by [14,15,22].

- First of all, previous research is conducted by attaching the SSD to a secondary channel of the motherboard. No experiment is conducted having SSDs as primary drive and an operating system installed on it. We believe that in reality when SSD is attached as a primary drive this may change its behavior because of the operating system. Since the operating system maintains the primary and secondary drive differently and garbage collector may behave differently as well. Even if the garbage collector deletes the data automatically we are interested to find out when the GC comes into action. Thus we want to conduct experiments both using SSD as primary drive as well as secondary drive.
- In the experiment conducted in [14], the SSD is filled entirely with data and then they have applied quick format. It is possible that if the controller is unable to find free space for incoming data then it activates the garbage collector. In other words it is possible that garbage collector's behavior changes with the amount of available free space or amount of space marked for throwing away (i.e., Garbage Collection)
- As the garbage collector from different manufacturers will behave differently, therefore, we conducted the experiments over SSDs from different vendors.

Table 1. Specifications of the Computers

Category	Description
Dell Laptop	
Manufacturer	Dell Inc
Model	INSPIRON 1545
Operating System	Microsoft Windows 7 Home Premium
RAM	4 GB
Hard Disk Drive	500 GB
Processor	Intel(R) Core(TM)2 Duo CPU T6600 @ 2.20 GHz, 2200 MHz, 2 Core(s), 2 Logical Processor(s)
Dell Desktop	
Manufacturer	Dell Inc
Model	OPTIPLEX 755
Operating System	Microsoft Windows 7 Professional
RAM	4 GB
Hard Disk Drive	250 GB
Processor	Intel(R) Core(TM)2 Quad CPU Q9300 @ 2.5 GHz, 2500 MHz, 2

5 Experiments and Results

We have performed three sets of experiments using three types of SSDs on two types of computers. The specifications of the computers used in the experiments are given in Table 1. The three types of SSDs used in our experiments are given in Table 2. We have selected Microsoft Windows 7 Professional and Microsoft Windows 7 Home Premium as the experimental operating systems. Windows 7 has native support for the TRIM command. For the TRIM to work, it is necessary that the underlying SSD support TRIM command and TRIM must also be enabled on Windows 7. To enable TRIM command on Windows 7 the following three options must be configured.

- Turn off system protection
- Enable AHCI mode in system BIOS
- Enable AHCI mode in window 7 registry.

For the recovery of files, PC Inspector [23] was used. PC Inspector is an open source software specially designed for the recovery of multimedia files from the camera memory or micro SD. It is open source and specifically designed for flash based memories. The drawback of PC Inspector is that it can only recover multimedia files of different formats. Paragon Partition Manager is used for partitioning and initial formatting of the SSDs in order to use it and view it in Windows operating system.

Table 2. Specifications of the SSDs

Category	Description
Crucial m4 64 GB	
Name	Crucial M4 SSD
Model	CT064M4SSD2
Capacity	64 GB
Form factor	2.5
Sequential READ	up to 500 MB/s
Sequential WRITE	up to 95 MB/s
Samsung 470 Series 64 GB	
Name	Samsung 470 Series
Model	MZ-5PA0641
Capacity	64 GB
Form factor	2.5
Sequential READ	up to 250 MB/s
Sequential WRITE	up to 170 MB/s
Kingston SSDNow V 100 64 GB	
Name	Kingston SSDNow V 100 SSD
Model	SV100S2N1646
Capacity	64 GB
Form factor	2.5
Sequential READ	up to 250 MB/s
Sequential WRITE	up to 145 MB/s

5.1 Experiment 1: Connecting SSDs to Dell Laptop Using USB Port

The purpose of this set of experiments is to check if the SSDs can preserve data after a quick format. Through experiments it was found that, data can only be preserved if there is no background garbage collector and the TRIM command is not performed, because these are the two possible causes that could delete the data from the SSD and no data will be recovered. In all other cases the data must be available for the recovery.

Experiment 1.1: Recovery from Crucial M4 SSD: In this experiment we connected the crucial M4 SSD to the USB port of laptop using Kingston USB case. The entire space of the crucial SSD is filled out by pasting a 3.44 MB JPEG image 17627 times. A free space of 272 KB is left over that could not hold any further image of the selected size. After this the SSD is quick formatted and system is restarted after 15 min. When the Five minutes after the system reboot,

the recovery software was started to recover the JPEG images. The recovery process completed 100 % in about 32 h and 15 min to complete.

From the recovered data it is observed that 17625 pictures were recovered and the software miss only two pictures out of 17627 pictures. From the result it is clear that the crucial M4 SSD does not have background garbage collector. It is also cleared that TRIM command also does not work under this experimental setup.

Experiment 1.2: Recovery from Samsung 470 Series SSD: The same experiment as conducted in Experiment 1.1 with the crucial SSD is repeated with the Samsung SSD. This experiment took almost the same time as that of Experiment 1.1. The result of this experiment also similar to Experiment 1.1. This SSD also does not have background garbage collector and not even TRIM command worked in our experimental setup.

Experiment 1.3: Recovery from Kingston SSDNow V 100 SSD: The same experiment as conducted in Experiment 1.1 and 1.2 is repeated with the Kingston SSDNow V 100 as well. This experiment took almost the same time as that of Experiment 1.1 and 1.2. Kingston SSDNow V 100 also did not have GC and TRIM enabled, and we were able to recover the same number of files from this SSD.

5.2 Experiment 2: Connecting SSDs to Dell Desktop Using Secondary SATA Port

As it is clear from the previous results that TRIM does not work with the USB port, the purpose of this experiment was to check the support of Windows 7 TRIM command for the SSD connected to SATA secondary port. Windows 7 was installed on a separate hard disk drive which was attached to the primary SATA port of the system. It was evident from previous experiments that Windows 7 cannot send TRIM command on USB port. Here we want to clarify the work of TRIM on SSD attach to secondary SATA port. The necessary preparations for this set of experiments are similar to Experiment 1.

Experiment 2.1: Recovery from Crucial M4 SSD: The Crucial M4 SSD was connected to SATA 1 on the dell desktop computer. As scanning of the large 64 GB SSD for files recovery is much time consuming, therefore, we decided to create two partitions in the SSD. One partition had a capacity of 5.85 GB and the second one had the remaining capacity of 59.6 GB. The 5.85 GB drive was filled by pasting a 3.48 MB JPEG image 1719 times. Only 2.89 MB space was free that could not hold any more image of the selected size. After filling the drive, it was quick formatted and the system was shut down after 15 min of the format operation and then restarted. When the system fully booted the PC Inspector was started for file recovery. The whole recovery process ran for about 3 h and 30 min.

The result of this experiment is similar to Experiment 1.1. It was found that connecting with USB port and connecting with the SATA secondary port does not make any difference. Almost all the files were recovered by the PC Inspector. TRIM command did not work with SATA secondary port as well.

Experiment 2.2: Recovery from Samsung 470 Series SSD: The same experiment as conducted in experiment 2.1 with the crucial SSD was repeated with the Samsung SSD as well. This experiment took almost the same time as that of Experiment 2.1. The result of this experiment was not different from the experiment 2.1. The TRIM command did not worked for this SSD as well while connecting it to SATA secondary port.

Experiment 2.3: Recovery from Kingston SSDNow V 100 SSD: The same experiment as conducted in Experiments 2.1 and 2.2 was repeated with the Kingston SSDNow V 100 as well. This experiment took almost the same time as that of Experiments 2.1 and 2.2.

The result of the Kingston SSD was similar to the other two SSDs. The TRIM command does not work either with the USB port or with the secondary SATA port. All these three SSDs were able to preserve data after quick format. If there is no background garbage collector in the SSD then the TRIM command never activates any garbage collection cycle in the SSDs if they are attached externally to the computer.

5.3 Experiment 3: Connecting SSDs to Dell Desktop Using Primary SATA Port

As it is clear from the previous results that TRIM does not work with the USB port and SATA secondary port. So in this setup the SSD is connected to the SATA primary port and operating system is installed on it. Windows 7 professional was installed on each of the three SSDs. During the installation we made two partitions (for all the three SSDs) were made one was labeled as C having size of 55.7 GB and the other one was labeled as D having a size of 3.90 GB. For each of the three SSDs, operating system was installed on the C partition. TRIM command was enabled by making the necessary changes in the OS and BIOS.

Experiment 3.1: Recovery from Crucial M4 SSD: In this experiment, the 3.90 GB of D drive was filled with 1522 JPEG images of size 2.59 MB. Only 1.31 GB space was left out as free. After filling the drive, it was quick formatted and the system was shut down after 15 min of the format operation and then started again. When the system fully booted PC Inspector was started for file recovery. The whole recovery process ran for about 2 h.

This time the result was completely different from all the previous experiments. The software scanned the entire D partition but could not find even a

single byte of data on the SSD. The TRIM command worked perfectly in this scenario of the experiment. We were not able to recover any single image from the drive.

Experiment 3.2: Recovery from Samsung 470 Series SSD: The same experiment as conducted in Experiment 3.1 with the crucial SSD was repeated with the Samsung SSD as well. This experiment took almost the same time as that of Experiment 3.1. Just like crucial SSD the Samsung SSD also erase all the data as the software was unable to recover any data. The Samsung SSD also shows that TRIM command works if the SSD is the primary drive. The experimental results shows that both crucial and Samsung SSDs cannot preserve data when connected on the SATA primary port.

Experiment 3.3: Recovery from Kingston SSDNow V 100 SSD: The same experiment as conducted in Experiments 3.1 and 3.2 was repeated with the Kingston SSDNow V 100 as well. This experiment took almost the same time as that of Experiments 3.1 and 3.2.

Just like Crucial and Samsung SSD, Kingston SSD also erases all the data and the software was unable to recover any data. The Kingston SSD also shows that the TRIM command works if the SSD is the primary drive. The experimental results show that all of the three SSDs, Crucial, Samsung and Kingston cannot preserve data when they are connected on SATA primary port.

6 Discussion and Conclusion

With the growth of emerging technology of SSDs in computers and other similar devices like cellular phones, tablets and netbooks, there are challenges for forensics analysis which are not experienced with traditional HDDs. Existing forensics analysis tools treat SSDs much like traditional hard disks drives. However, the technological difference between SSDs and HDDs requires new forensics tools designed specifically to address SSDs.

The purpose of wear leveling technique is to prevent blocks that contain frequently altering data from going bad faster than those which holds static data. Wear leveling techniques are usually implemented in Flash Translation Layer or in the Controller. It provides an opportunity to recover old data as well as metadata after a file is deleted or changed and new information is rewritten to a new physical location.

We have experimentally verified that SSD behavior differs when it is attached to the secondary SATA ports and the primary SATA ports with the operating system installed on it. TRIM command only works in the latter case. And it is even worse than BGC and has the potentials to destroy forensics evidences instantly after the deletion is performed. It is also important to note that BGC is not implemented in all the SSDs available in market and those having BGC start erasing garbage blocks approximately 150 s after the deletion is performed [14].

The firmware does not clear or zero the SSD automatically. It requires an operating system that supports the TRIM command to erase the data permanently. Therefore, in the absence of BGC and inability of the TRIM command, live acquisition is still possible. If encryption is enabled then the data recovered during live acquisition without encryption key is almost useless for forensics analysis, since it is hard to understand or make sense of encrypted data.

References

1. Lofgren, K.M.J., Norman, R.D., Thelin, G.B., Gupta, A.: Wear leveling techniques for flash EEPROM systems, 8 May 2001 (U.S. Patent 6,230,233)
2. Flusche, K.J.: Computer forensic case study: espionage, Part 1 just finding the file is not enough!. Inf. Syst. Secur. **10**(1), 1–10 (2001)
3. Janes, S.: The role of technology in computer forensic investigations. Inf. Secur. Tech. Rep. **5**(2), 43–50 (2000)
4. Guide, N., Ashcroft, J., Electronic Crime Scene Investigation: A Guide for First Responders Series: NIJ Guide
5. Carrier, B., Spafford, E.H.: Getting physical with the digital investigation process. Int. J. Digital Evid. **2**(2), 1–20 (2003)
6. Boddington, R., Hobbs, V., Mann, G.: Validating digital evidence for legal argument (2008)
7. Berg, E.C.: Legal ramifications of digital imaging in law enforcement. Forensic Science Communications, 2(4) (2000)
8. Kenneally, E.E., Brown, C.L.: Risk sensitive digital evidence collection. Digital Investig. **2**(2), 101–119 (2005)
9. Carrier, B.: File System Forensic Analysis, vol. 3. Addison-Wesley, Boston (2005)
10. Kasavajhala, V.: Solid State Drive vs. Hard Disk Drive Price and Performance Study, Dell Technical White Paper, Dell Power Vault Storage Systems (2011)
11. Hu, X.Y., et al.: Write amplification analysis in flash-based solid state drives. In: Proceedings of SYSTOR 2009, The Israeli Experimental Systems Conference. ACM (2009)
12. Lee, J., et al.: A semi-preemptive garbage collector for solid state drives. In: IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS). IEEE (2011)
13. Seppanen, E., O'Keefe, M.T., Lilja, D.J.: High performance solid state storage under linux. In: IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST). IEEE (2010)
14. Bell, G.B., Boddington, R.: Solid state drives: the beginning of the end for current practice in digital forensic recovery? J. Digital Forensics Secur. Law **5**(3), 1–20 (2010)
15. Freeman, M., Woodward, A.: Secure state deletion: testing the efficacy and integrity of secure deletion tools on Solid State Drives. In: Australian Digital Forensics Conference (2009)
16. Olson, A.R., Langlois, D.J.: Solid state drives data reliability and lifetime. Imation White Paper (2008)
17. Agrawal, N., et al.: Design Tradeoffs for SSD Performance. In: USENIX Annual Technical Conference (2008)

18. Chen, F., Koufaty, D.A., Zhang, X.: Understanding intrinsic characteristics and system implications of flash memory based solid state drives. In: Proceedings of the eleventh international joint conference on Measurement and Modeling of Computer Systems. ACM (2009)
19. Garfinkel, S.L.: Digital forensics research: the next 10 years. Digital Invest. **7**, S64–S73 (2010)
20. Luck, J., Stokes, M.: An integrated approach to recovering deleted files from NAND flash data. Small Scale Digital Device Forensics J. **2**(1), 1941–6164 (2008)
21. Roberts, D., Kgil, T., Mudge, T.: Integrating NAND flash devices onto servers. Commun. ACM **52**(4), 98–103 (2009)
22. Wei, M.Y.C., et al.: Reliably erasing data from flash-based solid state drives. In: Proceeding FAST (2011)
23. Arthur, K.K., Venter, H.S.: An investigation into computer forensic tools. In: Proceeding ISSA (2004)