SSD: New Challenges for Digital Forensics

P. M. Bednar[1,2] and V. Katos[3]

Abstract ICT changes continuously and we are used to look at IT in a slightly different way every year. Things are developed and manufactured to be smaller and faster but few changes are truly technologically revolutionary. Some changes creep up on us as they arrive under cover of previously known technology. Solid State Disks (SSD) is such a technology. The use of SSD is simple enough and for many purposes it can be used as if it was a normal hard disc but many times faster and with a very much lower power consumption. But, SSD is not an evolution of hard disc technology, it is a completely new technology which imitates the behaviour of a hard disc. There are major underpinning differences which have serious consequences for security and for digital forensic. Due to how the SSDs work it is not always certain that deleted data are purged from the disc. On the other hand SSD's can sometimes purge data all by themselves even if they are not connected to any interface with only the power on. This means that normal guidelines aimed at hard discs for how to preserve digital forensic evidence are not just inappropriate but could if followed result in potential evidence being lost, destroyed or deemed unvalid as evidence. This paper gives an overview of some of the principal and unexpected challenges that SSDs have brought with them for Digital Forensics investigations.

Key words: Digital Forensic, Solid State Disk, Investigatory Guidelines.

Introduction

In recent years, manufacturers striving to improve performance for consumers of computing equipment have been frustrated by problems with traditional hard disk drives (HDD). Difficulties have included reliability, speed of data access and high power consumption which have all combined to inhibit the desired improvements

[1] Lund University, Lund, Sweden, peter.bednar@ics.lu.se
[2] University of Portsmouth, Portsmouth, UK, peter.bednar@port.ac.uk
[3] Democritus University of Thrace, Xanthi, Greece, vkatos@ee.duth.gr

[1]. These difficulties arise from the mechanical qualities of the HD drives, which rely on multiple plates rotating on a spindle within a protective casing and store data magnetically. Solid State Disks (SSD), based on flash memory, have been adopted as a solution to these problems. They are built from semiconductor chips and hence have no rotating parts, are compact in size and more economical of power consumption. Furthermore, performance for random data access compares extremely favourably with that of traditional hard disks [9, 10, 12]. Often HDD and SSD share a common interface (both logical and physical, e.g. SATA, PCI-Express or Thunderbolt etc) and so are an attractive and realistic substitute, allowing for backward compatibility of products. There are also some hybrid drives which combine both HDD and SSD technology in one package (e.g. such as Seagate Momentus XT which is a 500GB HDD - incorporating 4GB NAND flash memory).

However, there are many differences in operation between the two types of storage device when looking beyond this common interface. HDDs, based on magnetic storage, can be written and re-written many times. Data which is no longer required can be removed simply by overwriting with new data. However, beneath the superficially similar interface, the internal operation of flash memory is very different. The disk offers an array of logical block addresses to the host but the internal organization depends upon complex algorithms. One disadvantage of the SSD technology over its predecessor is that existing data must be erased before blocks can be reused (e.g. they cannot simply be overwritten). While read-latency of SSD compares very favourably with HDD, and write-latency is comparable, speed of erasure is relatively slow. There is an added problem in that erase cycles are limited; within 100,000 cycles (or in some types of drive even fewer) flash memory cells cease to be able to hold data.

These difficulties are addressed mainly in three ways, by ‘wear levelling’ and ‘house-keeping’ (‘garbage collecting’). The problem of ageing of memory cells is mitigated by chip manufacturers in with three main strategies, a) to write on different memory blocks in each re-write, b) to occasionally move files that are not used very often to other memory blocks and c) to have extra memory blocks (invisible to the external interface) to compensate the limited lifespan of each physical memory cell [e.g. 7, 8, 9, 10]. In SSDs, disk operation is controlled by complex algorithms, separating slower erasure cycles from read-write operations. Wear levelling algorithms [2] distribute writing operations randomly in order to minimise the impact of cell ageing. A mapping scheme is therefore needed to map logical addresses of data to physical locations on flash memory. This includes the re-distribution of existing data also. Housekeeping algorithms are needed to manage erasure of data that is no longer valid, to prepare cells for reuse. These ‘garbage collectors’ are independent of read-write cycles [3]. All of these factors raise challenges in respect of Digital Forensics, since established practice in both safeguarding and investigation of evidence reflect the ways in which data cells are located and addressed in traditional HDD technology. This paper will discuss the challenges posed by SSD in this respect.

Challenges

Recently, new issues have been presented in separate research the consequences of which result in significant challenges for the Digital Forensics community. These relate to the characteristics of SSDs in two respects.

First, when a user attempts to delete data stored on an SSD, this does not necessarily purge data from the disk, even if traditionally stringent methods are adopted [11]. This is because data mapping does not necessarily relate logical structures to physical locations on the disk, the relationship being managed in practice by a complex algorithm known only to the manufacturer (and so is model, version, firmware and manufacturer dependent etc). Intel for example has recently made an effort to address this problem and provides a 'Solid-State Drive Toolbox' (downloadcenter.intel.com) that works (only) with its own SSDs. One of the features in the Toolbox is to allow the secure erase of drive content, but it only works with XP, Vista and Windows 7 and not with drives in RAID configuration).

Secondly, controller software in many of the latest SSDs actually purges redundant data automatically whenever the disk is powered on. This background 'garbage' collection mechanism happens completely independently of instructions from the operating system, and should not be confused with the TRIM command that enables operating systems to indicate to a drive which data blocks are no longer required (TRIM is a hybrid technology intended to allow for SSDs to work more efficiently).

This is quite a new problem and there are currently only a few articles on the subject. While this topic does require some technical understanding, we do not intend to create a technically-oriented discussion. In this paper we choose to focus not on the well-known benefits and challenges of SSD, but on some of the more unexpected consequences of currently-available SSD technology. This is also rather difficult task as the quantity of openly available data and information about SSD technology is limited at present. In this paper we start by drawing upon two key articles which introduce the two on-the-surface contradictory issues mentioned above, and elaborate further on the influence of SSD firmware, controllers and directions for future technology development. From the point of view of Digital Forensics, these challenges necessitate a focus on inquiry into context of incident response, and development of guidelines for forensic investigations. There is a need for analysis and awareness of differences between type of control behavior and SSD "housekeeping" strategies, as the direction of development.

The main issue can be described simplistically as:

"SSDs are really hard to erase AND really hard to recover."

In a personal note, one of the key authors [3] describes the issue in these terms:

> *"Drive data was traditionally purged manually, by having the computer tell the drive to write something else over the top of the old data. In the absence of such an overwrite, magnetically stored data persists. However, if you try that trick on an SSD, it may not work. The logical address you try to overwrite may be remapped on the fly, so that your 'overwrite' goes to some other physical cell rather than the one which stored the data. From a logical viewpoint, it looks like the overwrite worked - you can't access the data any more through your computer's OS. But from the drives point of view, the data is still there, lurking in some physical cell that is presently out of use as far as the logical sector list is concerned".*

The author goes on to point out the advantage of this arrangement for enterprising hackers using clever firmware, or even a direct physical approach. He describes how SSD drives have been developed to optimize their performance through controllers that 'preemptively wipe' data cells containing data no longer referenced by the file system. The need to enhance performance of future writes requires a suitable pool of unused blocks. The logical map 'seen' by the computer does not reflect the physical layer 'seen' directly by the controllers of the disk itself. The controllers have access to the filesystem's metadata indicating which cells should be reset for optimum performance. This means that data may be eradicated which would traditionally have been recoverable by forensic experts accessing an HDD. In summary, data that the computer instructs the drive to delete may or may not be removed physically from the drive. However, redundant data may be eradicated without a previous instruction via the host computer, by controllers in the SDD itself, while a TRIM command from the operating system may trigger resetting of cells containing data which is no longer required [3]

Is this an artificial problem for digital forensics? Perhaps one possible solution to this problem for forensic investigators could be to disassemble the SSD and read the memory chips independently of the built in controller. Or perhaps to disassemble the controller and to exchange it with one especially designed for forensic investigations. However it is not an easy task and it is doubtful if it would be possible to switch off the SSD controller and bypass it in any other way. In any case, validation, certification and forensic compliance would be a non-trivial exercise because the acquisition process will involve interference and tampering with the hardware of the storage device and as such the approach will need to be product specific. This does not address the additional complication introduced by the incorporation of controllers who encrypt / decrypt input/output data on the fly (such as is available in for example some of the recent SSDs made by Samsung etc).

Obviously, this is in all probability an artificial problem mainly due to lack of widespread understanding of the implication of this new technology and the current lack of maturity of the SSD technologies (compared to HDD). For example, although there is limited expectation of finding data in RAM from a computer that

is switched off, it was shown in a recent study [4] that it was possible to retrieve data from the volatile memory (such as passwords associated to web applications) even when the computer was found to be switched off. This was possible because the computer was a desktop model and was plugged into mains so that the motherboard had constant current. We should note that there are some motherboards that allow us to boot by pressing keys from the keyboard, or the mouse led may not switch off. This is very crucial for RAM forensics because the operating system does not zero out unused RAM; this is simply too time consuming. On the other hand software developers do not care to protect the sensitive variables storing passwords. For example, in Firefox you could send a password encrypted through HTTPS but this is only applicable on the network level. In RAM it is in plaintext, and can therefore be harvested.

Basically, the problem is not necessarily that SSD technology may or may not allow data to be retrieved in the future - but rather that, due to the under developed memory controllers (and controller software) today, (some) SSDs may contain data which is retrievable. We could argue that SSD introduces more uncertainty to the acquisition process and this additional amount of uncertainty requires a greater and greater range of different responses and measures.

It is logical that an SSD should be designed to optimize efficiency and speed by means of controllers which automatically purge unused data from its memory cells [e.g. 9, 10, 12]. The interesting question is not why they are organized in this way, but rather why all this was not accomplished at an earlier date in development of SSDs. We suggest that there is only one reason, and that is that SSD technology has not yet reached maturity. The rush to incorporate the TRIM feature in operating systems provides an example; TRIM should not really be needed at all, as most of the existing features of TRIM could be dealt with in the background by the onboard SSD controller. It is probably only a matter of time before SSD is truly 'plug and play', without a need to be nursed or maintained by any external manipulation. In the meantime, however, everyone from manufacturers, technologists, and customers to forensic investigators (including researchers) are still novices in their understanding of SSD as an ubiquitous technology.

The apparent paradox introduced above, and issues with SSD memory configuration, represent true challenges for forensic teams. However, while the engineering challenge is a moving target, the principles of the underpinning problems are quite predictable (with some effort). For example, when a crime occurs investigators are given the right to collect computers from a suspect. This can mean that they physically go and collect the hardware. In such cases, the hardware will often be unplugged; as happens when computers are handed over to the police and put in a 'safe' (contained) environment for forensic investigations to be made.

Obviously, there are some instances when forensic investigation is made immediately 'on site' but this depends on the characteristics of the particular engagement - often the investigators demand the hardware to be handed over to them immediately and the investigation is done elsewhere. This is quite common practice, as for example in the Sony Playstation 3 case in the USA, and also in the Pi-

rate Bay case in Sweden, and to name but two. In each case the hardware was taken from site and investigation was done elsewhere.

It is rather easy to unplug a computer from a wall socket; this is the work of only an instant, compared to the work involved in deleting files from traditional HDDs, which takes some time to do thoroughly. This may have great significance for court proceedings. It may be significantly more plausible to argue that a computer was switched off 'by accident' - or even as part of a normal police procedure - than to put forward the rather feeble argument that a hard disk was wiped 'by accident', after investigators requested the hard disk to be handed over, or entered the premise with a search warrant. However, this is exactly that what could happen in the Digital Forensic laboratory – because the original SSD could be changed 'all by itself' without any intervention by investigators. The carbon forensic bitstream copy of the SSD will then no longer be same as to the original, and the process of applying one-way cryptographic functions would be inappropriate and ineffectual. This would create opportunities for a suspect to develop a Trojan Defense in court and claim that the evidence was 'planted' after the hardware concerned had been seized.

The integrity of the whole device (i.e. that not even one bit, including the metadata have been modified after the device seizure) is ensured by cryptographic means and more specifically by cryptographic one way hash functions. These functions have the property to produce a digest of a whole disk as large as a few bits (typically 160 to 256 bits) in a manner that even a single bit change on the suspect disk will produce a completely different digest output, in an unpredictable fashion (that is, we cannot "guess" what the output of a hash function a bit change in input will create if we do not perform the actual function). Again, in the case of the SSD, the hash output will change over time and therefore cannot be trusted nor used to prove the integrity of the SSD's contents.

Conclusion: a complex situation with added uncertainty

The Association of Chief Police Officers (ACPO) has recently released a new version of the 'Good Practice Guide for Computer-Based Electronic Evidence' [5], which is considered to be the point of reference for a first responder and a forensic analyst. Although that the guide is a very carefully written document, and a valuable resource, the information and advice contained in it cannot be used for handling of SSD devices. The updated version has a significant focus on live forensics, suggesting that if a system is switched on the first responder should consider capturing data residing in the volatile memory prior to switching it off. However such recommendation is not suitable for an SSD device. Due to the device's internal garbage collection functionality, when a file is deleted it is gradually purged by the device itself. As such, if the device is connected to a live system, the responder should detach the device from the system in order to preserve any deleted

files. On the other hand, depending on the situation, the system should typically remain working in order to allow the responder to capture the contents of the RAM. Apart from the obvious situation that there may be usernames, passwords, user activities and history in the volatile memory, there is the scenario of an encrypted SSD partition or container in general. In such a scenario, the encryption key is expected to reside in RAM in plaintext form.

Another problem introduced by SSD technologies is that the use of write blockers is inappropriate. Consider, for example, the proposal set out for the evaluation of a hardware write blocker device [6, p.S5]: *'... a write blocker should block all write commands sent to a hard drive*'.

Typically a digital media acquisition (such as a hard disk) is performed by accessing it (mounting it) through a write blocker which is a hardware (or software) device that blocks any write attempt to the storage device. This is important since even a read file operation actually hides a write to the file's metadata (the access time in particular). Any change to the evidence-disk is not acceptable as it would render the evidence non-admissible in court. Since the write blocker is applied outside the storage device we can easily see that in the case of an SSD device a write blocker will not have the desired effect as the SSD internals will write to the storage area.

Clearly in the case of an SSD device this kind of requirement is misaligned, as the purpose of the write blocker is not fulfilled. ACPO's guide must explicitly exclude this particular category of memory devices. Or perhaps even more importantly make explicit that requirements specifically aimed at HDD may be fundamentally unsuitable for SSD – and possibly also in some way unsuitable for developing hybrid technologies which are changing fast. Useful guidelines are intrinsically important in the practice of Digital Forensic Investigation, but in cases such as this where technology not just changes fast - but the underpinning principles upon which the technology is grounded is completely different - those very same guidelines might sabotage the whole purpose for which the guidelines were created in the first place.

Finally, a long lasting dilemma a first responder may face is the decision to pull the plug when a computer is found to be switched on a crime scene. Although a computer found to be switched off should not be switched on with the hard disks connected to the rest of the system, in the case of a switched on system, the first responder must make the decision whether to literally pull the plug, or perform some other kind of action (such as performing a RAM dump, or any other kind of live acquisition). In the case of an SSD found to be connected on a live system it seems that the most suitable procedure would be to pull the SSD device from the system and perform a RAM dump in order to retain any potential encryption keys in case of an encrypted partition existing on the SSD filesystem.

References

1. Chen, F., Koufaty, D. and Zhang, X. (2009). Understanding intrinsic characteristics and system implications of flash memory based solid state drives. Proceedings of the 11th International Joint Conference on measurement and modeling of computer systems, 181–192 NewYork, ACM
2. Chang, L. (2007). An Efficient Management Scheme for Large-scale Flash-memory Storage Systems. Proceedings of the ACM Symposium on Applied Computing. Nicosia
3. Bell, G.B. and Boddington, R. (2010). 'Solid State Drives: the Beginning of the End for Current Practice in Digital Forensic Recovery? ' *Journal of Digital Forensics, Security and Law, 5*(*3*), 1-20.
4. Karayanni, S., and Katos, V. (2011). 'Practical password harvesting from volatile memory'. *7th International Conference in Global Security Safety and Sustainability*, 2011, in press.
5. Association of Chief Police Officers (2010). Good Practice Guide for Computer-Based Electronic Evidence v4..
6. Lyle, J. (2006). A strategy for testing hardware write block devices. Digital Investigation, 3S, S3-S9.
7. Takeuchi, K. (2009). Novel Co-Design of NAND Flash Memory and NAND Flash Controller Circuits for Sub-30 nm Low-Power High-Speed Solid-State Drives (SSD). IEEE Journal of Solid-State Circuits, 44(4), 1227–1234.
8. Chang, L. On Efficient Wear Leveling for LargeScale FlashMemory Storage Systems. SAC'07 March 11-15, 2007, Seoul, Korea.
9. Cooke, J. (2007). Flash Memory Technology Direction. Microsoft WinHEC 2007.
10. Ekker, N., Coughlin, T. and Handy,J. (2009). Solid State Storage 101. January 2009. SNIA, Solid State Storage Initiative White Paper. Storage Networking Industry Association, San Fransisco.
11. Wei, M., Grupp, L. M., Spada, F. E. and Swanson S. (2010). Reliably Erasing Data From Flash-Based Solid State Drives, University of California: San Diego http://www.usenix.org/events/fast11/tech/full_papers/Wei.pdf
12. Rizvi, S. and Chung, T. (2010). Flash SSD vs. HDD: High performance oriented modern embedded and multimedia storage systems, 2010, 2nd. International Conference on Computer Engineering and Technology.