Empirical Analysis of Solid State Disk Data Retention when used with Contemporary Operating Systems

By

Christopher King and Timothy Vidas

From the proceedings of

The Digital Forensic Research Conference

DFRWS 2011 USA

New Orleans, LA (Aug 1st - 3rd)

available at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/diin

Digital Investigation

Empirical analysis of solid state disk data retention when used with contemporary operating systems

Christopher King [a,*], *Timothy Vidas* [b]

[a] *CERT Program/Software Engineering Institute, Carnegie Mellon University, USA*
[b] *ECE/Cylab, Carnegie Mellon University, USA*

Keywords:
Solid state disks
Digital forensics
Data recovery
Hard drive technology

ABSTRACT

Data recovery techniques for platter-based disk drives have remained rather static due to the dominance of the hard disk for the last two decades. Solid State Disk drives have differing storage and recall functionality from platter-based disks and require special care when attempting data recovery. Manufacturers have varying implementations of garbage collection in each drive, which affects the amount of data retained on the disk. This paper presents an analysis of solid state disk data retention based off of empirical evidence of 16 different disks. It also discusses the data recovery problem faced by forensic examiners due to the ATA8 TRIM command, which can sanitize disks in seconds. The experiment shows that without TRIM, nearly all data is recoverable, but with TRIM enabled only up to 27% of blocks were recoverable dependent on the controller manufacturer.

1. Introduction

Since the introduction of Solid State Disks (SSD), they have been more expensive and have had less capacity than mechanical hard drives. Only recently have they approached the price and capacity of traditional disks. The read and write speed increase and low power usage over traditional hard drives have made them more desirable in laptops, portable devices, and in high performance desktop systems. SSDs have a read speed increase on the order of 10 times greater than a platter-based hard drive, and 5 times greater for write speed (Gray and Fitzgerald, 2008; Lee et al., 2008).

With expected growth rates of 54% from 2008–2013, the rapid adoption of SSDs requires forensic investigators to be cognizant of this new technology, as they may encounter it in the field (IDC, 2010). Unlike hard disk drives, flash-based disks have no moving parts and have different storage and recall functionality. Little research has been conducted that examines the impact of these new types of disk will have on data recovery. Hardware manufacturers use unique firmware and garbage collection algorithms that reduce the effectiveness of existing data recovery techniques. The variations in garbage collection implementation increase the analysis load for investigators due to the greater time required to analyze a disk, and thus any research that can provide them with guidance revealing which drives are likely to have recoverable data can save valuable time. This paper empirically analyzes the retention of deleted data on 15 different SSDs and one traditional Hard Disk Drive (HDD) to provide researchers and investigators with a comprehensive list of the drives with the most and least data retained. We show that on a TRIM-enabled SSD, using an Operating System (OS) that supports TRIM, that in most cases no data can be recovered. On a non-TRIM-supported OS, manufacturer garbage collection implementation appears to affect the amount of data recovered with results varying from 100% blocks recovered to 0% on the same test using different disks.

[*] *Corresponding author.*
E-mail addresses: cking@cert.org (C. King), tvidas@cmu.edu (T. Vidas).

1742-2876/$ – see front matter © 2011 Carnegie Mellon University and Tim Vidas. Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.diin.2011.05.013

2. Background

SSDs are made up of NAND flash memory chips, with data read and written in parallel from multiple chips on the disk (not unlike a RAID array) (Pavan et al., 1997). NAND flash is a type of metal-oxide semiconductor memory that has evolved from the same family as DRAM, EEPROM, and NOR Flash. It is non-volatile (unlike DRAM or SRAM) and provides sufficient density and speed for use as a primary storage device (Bez et al., 2003; Pavan et al., 1997). The basic physical properties of NAND flash present several problems to SSD designers. Drive sizes were small so SSD designers increased density by improving the sensitivity of the read/write mechanisms, resulting in Multi-Level Cell (MLC) Flash (the most common type of flash used on SSDs) (Bauer et al., 1995). Drive longevity was also an issue; wear on the flash cell as it is written and erased limits the lifetime of the devices. To increase this lifetime to the standard useful life of 5 years,[1] manufacturers have implemented the practice of "wear leveling." Wear leveling ensures that each block on the device is only written to once before writing to that block again. With a traditional hard disk, this would impact performance significantly because magnetic disks perform better when similar data is located in close proximity to each other (Akyürek and Salem, 1995). Since a SSD reads data across all the chips simultaneously, and requires no moving head, there is no performance loss for this operation. Wear leveling has proven to be effective; Intel estimates that their drives can support 20 GB of writes per day for five years (Intel, 2010).

Software access to physical blocks on a SSD is almost impossible without knowing the proprietary manufacturer ATA commands due to a feature called the Flash Translation Layer (FTL). The FTL runs in the SSD controller and translates the block commands from the OS to commands that can be performed on flash. It also maps the logical block addresses from the OS to the physical block addresses on the chip. The translation serves to mask the intricacies of flash hardware to the OS (Regan, 2009). Investigators should note that the SSD may be physically relocating data on the drive even though the OS is reporting that the blocks are unchanged. This may be relevant in drive-independent data recovery or other attempts to recover data without the SSD controller.

A NAND flash chip consists of cells, pages, blocks, and planes (Fig. 1), each with their own unique physical properties. A flash cell must be completely erased before a write can be committed (a delete-before-write operation), and erase operations take much longer than a write, so designers needed a way to minimize the impact of this operation while maintaining wear leveling (Agrawal et al., 2008). A solution was to implement garbage collection across blocks on the disk. The controller needs to maintain a number of active (or available) pages for writing, so the garbage collector runs in the background finding blocks with the most inactive pages. The garbage collector moves the active pages out of the block and erases blocks as necessary. Wear leveling, in general, requires that all cells on the disk are written to at least once before writing over the same cell again. As the disk writes data to new pages each time, garbage collection attempts to create free pages to be written. The opposing constraints of garbage collection and wear leveling have the unfortunate side effect of performance degradation as the disk becomes filled. As each block on the SSD is written, the garbage collector needs to move data between pages to keep the allocation pool filled (Hu et al., 2009). The loss in performance due to garbage collection led to the update of the ATA8 specification to include the TRIM function in the DATA SET MANAGEMENT command (Stevens, 2006).

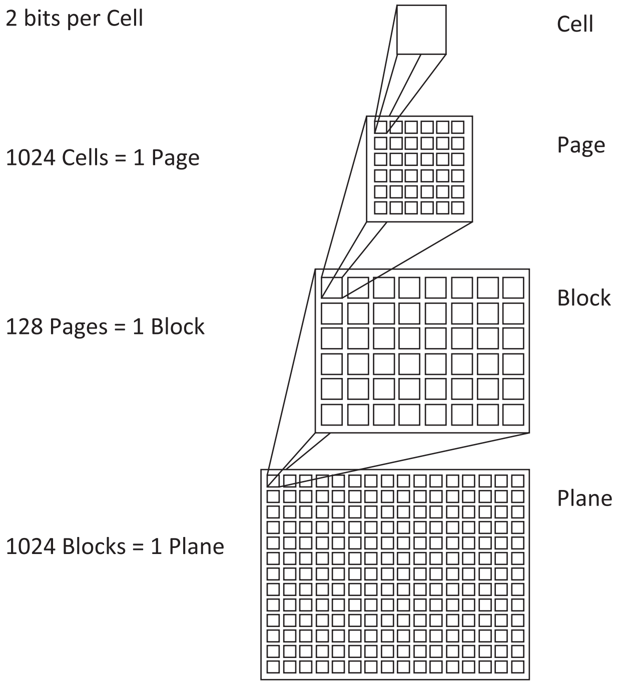

Fig. 1 – Sample hierarchy of flash chip architecture.

The ATA TRIM command was proposed as part of the DATA SET MANAGEMENT commands in 2007 (Shu, 2007). It is

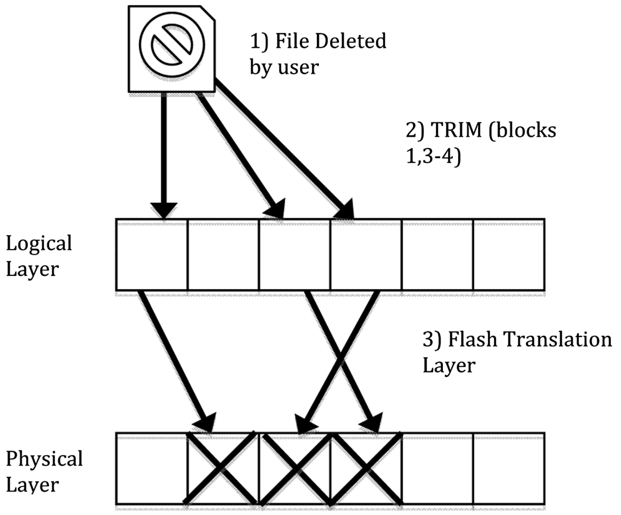

Fig. 2 – Depiction of a TRIM operation.

[1] Five years is common among disk manufacturers but the number does vary.

slated for inclusion into the ATA standard, but has already been implemented by manufacturers. TRIM is a relatively new method SSDs use to deal with performance degradation as a result of the delete-before-write problem. TRIM changes disk garbage collection by allowing the operating system to mark blocks as deleted (Fig. 2). The disk controller still decides when to initiate and perform garbage collection. Other research (Hu et al., 2009) has shown that SSDs perform better when having more "scratch" space available on the disk. TRIM addresses both these issues. When a delete operation is performed, the OS sends the SSD a TRIM command along with a list of blocks to be deleted (up to 65,536 blocks (ANSI, 2007)). The controller then knows it can add those blocks to the free block pool (and thus runs garbage collection) (Shu, 2007). The use of TRIM has allowed SSDs to maintain their high performance even after heavy usage, and most SSD manufacturers have a TRIM-enabled drive model currently on the market.

3. Related work

Previous research into flash memory forensics has focused mainly on portable devices, such as thumb drives, phones, and PDAs. Flash chips have been present in those forms for years, so the existing research is much more comprehensive regarding specific single-chip implementation rather than on larger, complex flash arrangements such as an SSD drive.

Stokes (2008) shows how to recover files from NAND flash, focusing mainly on recovering media files from mobile devices. Breeuwsma et al. (2007) introduce flash memory, and show how it is possible to acquire data from USB memory sticks. Stokes and Breeuwsma's research provides methods of flash data recovery using traditional memory acquisition techniques, JTAG (direct memory access) acquisition, and physical extraction of the memory chip.

Harral and Mislan's (2007) small device forensics ontology reviews a number of different types of flash-based devices and provides some insight into how to approach analysis when encountering them in the field.

Phillips et al. (2008) experimentally tested data recovery of damaged flash drives. Phillips' findings show that physical destruction of the data on a flash chip is extremely difficult to do. Over-voltages, smashing, water, and incineration all proved ineffective in destroying all the data. Skorobogatov (2005) analyzed data remanence in flash memory devices. His research proved that data on a flash cell may still be extractable after a delete, but acknowledged that data recovery at that point is extremely difficult.

Regan's (2009) thesis on the forensic potential of flash memory provides the most applicable background for this research. His paper describes the operation of flash, literature survey, and how to recover and analyze data on flash file systems.

More recently, Bell and Boddington (2010) explored the effects of SSD garbage collection on data retention. They found that data is removed within minutes of the disk remaining idle. However, the limited sample size (one SSD on Windows XP) and analysis based off of small (10 kb) files reduce the usefulness of the results.

Perhaps most related to our work, Wei et al. (2011) tested common disk sanitization techniques on SSDs and found that some data, once deleted from a SSD, may still be resident on the disk due to wear leveling but inaccessible via software means. Using direct hardware connections to the disk, they were able to recover up to 16 copies of the original file in some cases. However, the authors did not address the TRIM command or the impact TRIM may have upon data recovery in general.

4. The forensics problem with SSDS

The large performance increase and power savings of SSDs over traditional hard disks may be predictors that SSDs will become the dominant disk type used by consumers and businesses in the future (Daim et al., 2008). More relevant for the digital forensics community is that TRIM and garbage collection on SSDs creates the equivalent of a sanitized disk[2] (Kissel et al., 2006). An erase operation on a SSD requires the controller to write ones to those blocks, which is equivalent to sanitizing that particular block. It must be noted, however, that wear leveling disk over provisioning can leave remnants of data not accessible through the FTL (Wei et al., 2011).

The quantity and diversity of different SSD manufacturers, each using a different disk controller, presents an issue of data recovery prioritization to the investigator. With each SSD, the performance can vary drastically due to physical design, component quality, and the custom algorithms for garbage collection used in each drive. The variance in aggressiveness of garbage collection algorithms between controller manufacturers also affects data recovery.

The ATA ERASE UNIT[3] command also presents a troublesome trend for examiners. As noted by Wei et al. (2011), the ERASE UNIT command can cause the disk to perform a secure wipe of the drive. However, variance in manufacturer implementation of this command also determines the effectiveness of the drive sanitization.

5. Experiments

We conducted a series of experiments to test the applicability of the data loss claims using three drive scenarios. *High Usage* replicates a heavily used drive with little space available for the user, *Low Usage* mimics a scenario where the drive has most of the free space available, and *Format* shows a standard scenario of disks formatted using the built-in operating system utility present on the install disc.

5.1. *High usage scenario*

This scenario attempts to replicate the use case of a heavily used drive. A common scenario for this would be a user who has filled their disk with movies or music, leaving little

[2] The definition of "media sanitization" used here is based off of the NIST 800-88 definition of "clearing" a drive.

[3] The word "UNIT" is not addressing SSDs specifically, it is a general term used in the ATA specification.

available free space for the OS to use. This is intended to stress the garbage collection of each disk by limiting the amount of blocks available for cleaning. The disk was filled with hundreds of binary files to simulate this high usage scenario.

5.2. *Low usage scenario*

The low usage scenario intends to replicate the use case of a new system for a user or a disk with little changes made. An example of this would be a brand new system with only a handful of non-OS files resident on the disk. This case tests how well garbage collection performs in its ideal scenario – a disk with many cleaned blocks available. This was tested using a disk with a fresh OS install and only the two reference files available on the disk.

5.3. *Format scenario*

The OS format scenario shows a standard use case of a freshly formatted disk. An example of this is a user selecting only the default format options. This scenario was tested by placing the reference files on the disk before formatting the drive with the default settings. Note that the Windows 7 and Ubuntu default format option is a “quick format” that merely erases partition table metadata compared to default format option in Windows XP, which overwrites all sectors on the disk.

The experiments were designed to test three hypotheses:

1. Deleted data on high usage disks will have a lower chance of permanent loss due to increased stress of garbage collection. Disk firmware will have a greater effect on recovering data due to the number of pages the garbage collector has to prioritize for deletion.
2. Deleted data on low usage disks will have a greater chance of permanent loss due to focused garbage collection.
3. Deleted data recovery from OS formatted disks will be dependent on disk firmware.

Fifteen different drives, from ten different manufacturers, and one HDD control were used to test each of these cases. Six of the drives supported TRIM and all used MLC flash. The experiment was performed using Windows 7, Windows XP, and Ubuntu Linux 9.04 on each drive. Windows 7 was the only OS that supported TRIM during the course of the experiment. Each drive was secure wiped by writing zeros to the drive before installing the OS. After initial OS install, two reference files – a large binary file (650 MB) containing a non-repeating alphanumeric pattern and a small text file (900 kB) containing the full text of *The Saga of Burnt Njal* were copied to the disk. These files were used for each test and were recovered from each drive after a deletion. The starting and ending block locations of each file were obtained using the Sleuth Kit series of tools. Blocks were extracted from the disk using a Perl script[4] and Sleuth Kit and the recovered blocks were then compared to the original files and analyzed for data loss by performing a byte-wise comparison.

Table 1 – Percent blocks recovered by test and operating system type.

Test	Control	Imation1	Corsair1 (TRIM)	Crucial1 (TRIM)	PQI1	RiData1	OCZ1 (TRIM)	OCZ2	Patriot1	Patriot2	Kingston1	Intel1 (TRIM)	Intel2 (TRIM)	Intel3 (TRIM)	Intel4	Transcend1
Large File, Low Usage, Win7	100.00%	100.00%	0.00%	0.00%	71.23%	100.00%	0.00%	100.00%	100.00%	100.00%	100.00%	0.00%	0.00%	0.00%	100.00%	70.49%
Large File, Low Usage, WinXP	99.99%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
Large File, Low Usage, Linux	99.99%	99.98%	100.00%	99.99%	99.98%	100.00%	100.00%	100.00%	100.00%	99.98%	99.99%	99.98%	99.98%	100.00%	99.98%	100.00%
Large File, High Usage, Win7	100.00%	100.00%	0.00%	0.00%	88.60%	100.00%	0.00%	100.00%	100.00%	100.00%	100.00%	0.00%	0.00%	0.00%	100.00%	100.00%
Large File, High Usage, WinXP	99.99%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
Large File, High Usage, Linux	100.00%	99.98%	100.00%	100.00%	99.98%	99.98%	100.00%	100.00%	99.99%	34.78%	100.00%	99.99%	0.00%	99.98%	99.99%	100.00%
Large File, Format, Win7	100.00%	100.00%	0.00%	0.00%	100.00%	100.00%	0.00%	99.87%	100.00%	0.00%	0.00%	0.00%	0.00%	0.00%	100.00%	100.00%
Large File, Format, WinXP	99.99%	100.00%	0.00%	0.00%	0.00%	99.67%	100.00%	0.00%	100.00%	100.00%	0.00%	100.00%	0.00%	0.13%	99.40%	99.67%
Large File, Format, Linux	99.86%	100.00%	0.00%	77.88%	0.00%	3.83%	99.87%	100.00%	12.37%	7.36%	3.83%	83.52%	100.00%	100.00%	0.00%	100.00%
Small File, Low Usage, Win7	99.98%	0.00%	25.53%	27.54%	0.00%	99.98%	25.53%	99.98%	99.98%	0.00%	99.98%	0.00%	0.00%	0.00%	99.98%	99.98%
Small File, Low Usage, WinXP	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%	0.00%	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%
Small File, Low Usage, Linux	99.98%	99.98%	99.98%	99.98%	99.98%	99.98%	99.98%	86.20%	99.98%	99.98%	99.98%	99.98%	99.98%	99.98%	96.97%	99.98%
Small File, High Usage, Win7	99.98%	99.98%	27.54%	26.28%	99.98%	99.98%	25.53%	99.98%	99.98%	99.98%	99.98%	0.00%	0.00%	0.00%	99.98%	99.98%
Small File, High Usage, WinXP	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%
Small File, High Usage, Linux	99.98%	99.98%	99.98%	99.98%	99.98%	99.98%	99.98%	99.98%	99.98%	98.71%	99.56%	99.98%	99.98%	99.98%	98.27%	99.98%
Small File, Format, Win7	0.00%	26.96%	24.46%	0.00%	0.00%	24.63%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	24.61%	0.00%
Small File, Format, WinXP	96.57%	57.21%	24.69%	0.00%	0.00%	54.27%	49.03%	53.05%	46.84%	0.00%	37.73%	59.84%	86.20%	46.76%	58.45%	48.20%
Small File, Format, Linux	99.98%	99.98%	0.00%	99.98%	99.98%	0.00%	99.98%	99.98%	99.98%	0.00%	99.98%	27.41%	99.98%	99.98%	0.00%	99.98%
Average	93.86%	87.62%	49.74%	56.93%	69.60%	81.97%	66.28%	85.12%	86.23%	57.63%	74.12%	59.10%	54.40%	57.77%	81.71%	89.52%

[4] Source code and full results available at www.ece.cmu.edu/~tvidas/.

Table 2 – Percent blocks recovered on a TRIM-enabled disk with Windows 7

Test	Control	Intel1	Intel2	Intel3	OCZ1	Corsair1	Crucial1
Large File, Low Usage	100.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Large File, High Usage	100.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Large File, Format	100.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Small File, Low Usage	99.98%	0.00%	0.00%	0.00%	25.53%	25.53%	27.54%
Small File, High Usage	99.98%	0.00%	0.00%	0.00%	25.53%	27.54%	26.28%
Small File, Format	0.00%	0.00%	0.00%	0.00%	0.00%	24.46%	0.00%

This chart shows only those disks that support TRIM (and the control), which is 6 of the 16 tested.

Table 3 – Percent blocks recovered on a TRIM-enabled disk with Windows XP.

Test	Control	Intel1	Intel2	Intel3	OCZ1	Corsair1	Crucial1
Large File, Low Usage	99.99%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
Large File, High Usage	99.99%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
Large File, Format	99.99%	100.00%	0.00%	0.13%	100.00%	0.00%	0.00%
Small File, Low Usage	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%
Small File, High Usage	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%	96.57%
Small File, Format	96.57%	59.84%	86.20%	46.76%	49.03%	24.69%	0.00%

This chart shows only those disks that support TRIM (and the control), which is 6 of the 16 tested.

6. Results

6.1. *Data recovery test*

In total, 144 tests were conducted, revealing some interesting trends (see Table 1). Data Recovery using TRIM-enabled disks on Windows 7 is practically impossible (see Table 2). All TRIM-enabled drives showed close to 0% data recovery for the large file. For small files, the difference between manufacturers is pronounced. The Intel drives showed 0% data recovery with the small file but it was possible to recover 25–30% of the data on the other drives. The difference in recovery percent is most likely due to the type of controller used by the drive. Intel uses their own proprietary controller software while the other three drives licensed their controllers from Indilinx. The exact cause is unknown and remains future research.

Using the same drives, and comparing them to an OS without TRIM support (Windows XP in this case), the difference in recovery is significant (see Table 3). Both the large and small file deletion tests had near 100% recovery, while the format tests had low levels of recovery. The format test can be explained due to the way Windows XP conducts formats by default. The OS goes through an actual deletion of the files on the disk and rebuilds the NTFS structure. Windows 7 does the equivalent to a Windows XP "quick" format, which simply removes the $MFT and marks all files as deleted. Investigators

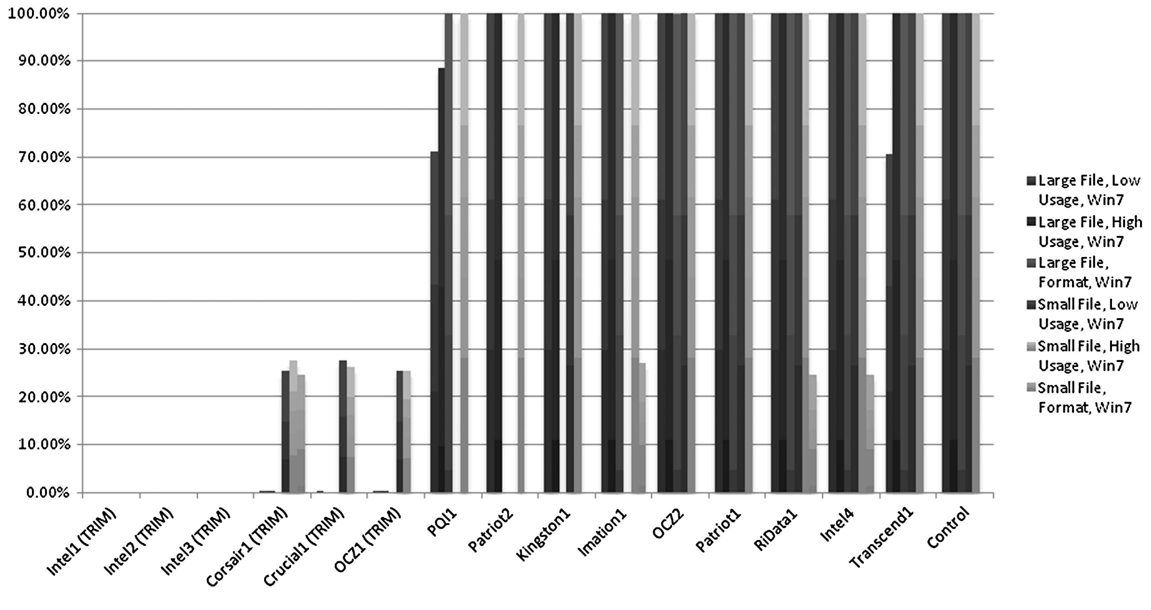

Fig. 3 – Percent blocks recovered in Windows 7 across all tests.

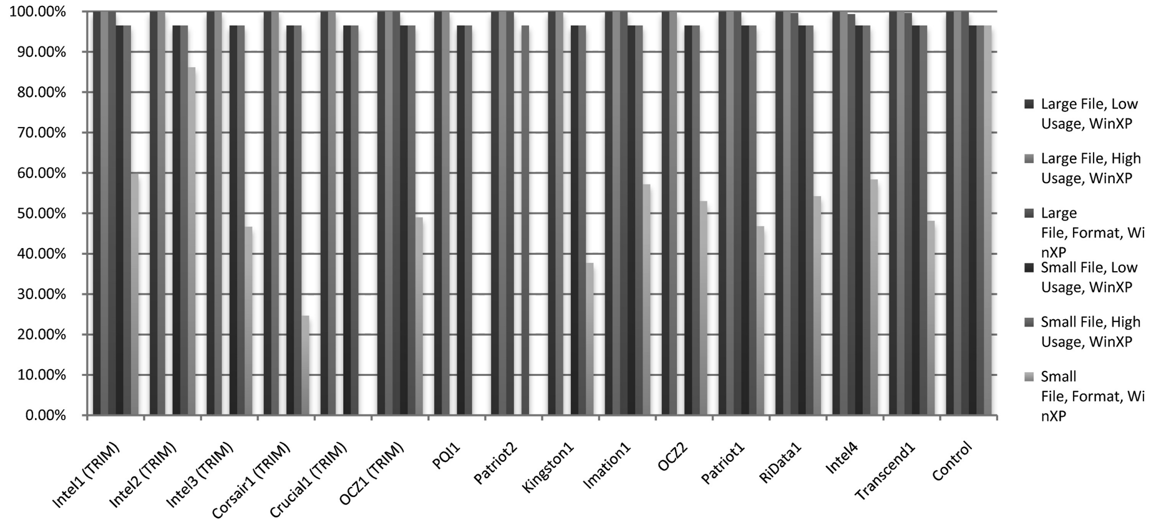

Fig. 4 – Percent blocks recovered in Windows XP across all tests.

should be aware that for SSDs that support TRIM, the OS type is important for deciding whether to attempt recovery.

Comparing each drive across all tests, we see that there is a change in recoverability of data between low and high usage (see Fig. 3), in accordance with the first hypothesis. In Windows 7, up to 17% more data can be recovered from large files in high usage disks and up to 100% more data can be recovered from small files in high usage disks. This difference may be explained by the theory of garbage collection struggling to clean blocks on the disk in the high usage scenario. The second hypothesis also seems to hold true in all cases except for in Windows 7, where two non-TRIM-enabled drives have 100% data loss.

Comparing data recovery across operating systems can provide information on how SSDs operate in different usage environments that an investigator may encounter (see Fig. 4). Contrasted with Fig. 3, Windows 7 shows the least amount of data recovery over the other two operating systems due to its TRIM support. Ubuntu proves to be the easiest to recover data, since it uses a combination of a quick format and has a lack of TRIM support.

6.2. *Manual TRIM test*

TRIM may also be used for anti-forensics. Many manufacturers have created tools that allow the user to initiate TRIM or garbage collection on the disk at will. Since ATA8 supports a TRIM of up to 65,536 blocks in a single command (ANSI, 2007), it is possible that TRIM can sanitize an entire disk in seconds.

To test this theory, we used the open source tool hdparm and a short bash script to sanitize a drive in about 20 s. Due to caching and in-memory execution, this script also functions on the OS boot drive. After running the test on the boot disk and rebooting, the disk failed to mount and further analysis of the disk showed it to be completely sanitized. The execution of such commands could be automated and permit individuals to clear disks rapidly on demand or to trigger the action based on a series of events. Such automation may result in first responders encountering a completely sanitized disk.

7. Conclusion

SSDs should not be considered the same as an HDD when encountered in the field. The advent of the TRIM command has made data recovery all but impossible using current techniques. Our data shows that when TRIM is enabled, in most cases no data is recoverable. Without TRIM enabled, the blocks recovered vary, but averages close to 100%. Both these statistics appear to be manufacturer dependent – for example Intel drives have the lowest recoverability. The performance increase from TRIM has resulted in widespread adoption among SSD manufacturers and eventually all SSDs will support the command. As SSD adoption grows as well as use of Windows 7 and other TRIM-supported OS's, traditional data recovery will no longer be a viable option for investigators.

Forensics investigators should take note of the OS type and whether the system is using a SSD before removing power from a system. In some cases, volatile memory analysis may be more useful for evidence collection due to lack of deleted file material in static analysis. In cases that the OS or drive does not support TRIM, typical organizationally defined procedures apply.

As demonstrated here, SSDs can be used for anti-forensics. A relatively short series of ATA commands (potentially in a script) can cause the drive to TRIM itself, destroying evidence. It is possible that a virus or worm with administrative privileges would be able to issue TRIM commands to wipe entire drives in seconds.

8. Future research

There are several avenues of future research available with solid state disks and their operation. While this paper examined data recovery between high and low usage disk scenarios, the effect of high disk activity on data recovery was not studied. A comparison of data recovery in idle versus high disk activity scenarios may provide some insight into manufacturer garbage collection efficiency. Observations of the TRIM command may also prove interesting. Understanding when different OS's send TRIM commands to the disk is important for understanding when deletes occur. Finally, SSD data recovery on Linux ext4 file systems and TRIM-supported kernels may provide some comparisons to the research presented here.

Acknowledgments

The authors wish to acknowledge Alex Nicoll, David Zubrow, and Rhiannon Weaver for their advice and support of this project. We also wish to thank the anonymous reviewers, whose remarks were very helpful in revising this paper. This work is supported in part by Heinz College, CyLab at Carnegie Mellon under grant DAAD19-02-1-0389 from the Army Research Office, and NSF IGERT award DGE-0903659.

REFERENCES

Agrawal, N., Prabhakaran, V., Wobber, T., Davis, J., Manasse, M., Panigrahy, R. Design tradeoffs for SSD performance 2008, pp. 57–70.

Akyürek S, Salem K. Adaptive block rearrangement. ACM Transactions on Computer Systems (TOCS) 1995;13(2): 89–121.

ANSI. Data set management commands Proposal for ATA8-ACS2; 2007.

Bauer M, Alexis R, Atwood G, Baltar B, Fazio A, Frary K, et al. A multilevel-cell 32 Mb flash memory. In: Solid-State Circuits Conference; 1995. p. 132–3.

Bell GB, Boddington R. Solid state drives: the beginning of the end for current practice in digital forensic recovery? Journal of Digital Forensics, Security and Law 2010;5(3):1–20.

Bez R, Camerlenghi E, Modelli A, Visconti A. Introduction to flash memory. Proceedings of the IEEE 2003;91(4):489–502.

Breeuwsma M, Jongh M, Klaver C, Knijff R, Roeloffs M. Forensic data recovery from flash memory. Small Scale Digital Device Forensics Journal 2007;1(1):1–17.

Daim T, Ploykitikoon P, Kennedy E, Choothian W. Forecasting the future of data storage: case of hard disk drive and flash memory. Foresight 2008;10(5):34–49.

Gray J, Fitzgerald B. Flash disk opportunity for server applications. Queue 2008;6(4):18–23.

Harrill D, Mislan R. A small scale digital device forensics ontology. Small Scale Digital Device Forensics Journal 2007;1(1):242.

Hu, X., Eleftheriou, E., Haas, R., Iliadis, I., Pletka, R. Write amplification analysis in flash-based solid state drives 2009, ACM, pp. 1–9.

IDC. IDC outlook for solid state drives remains bright, despite market challenges and economic uncertainties 2010.

Intel. Intel® solid state drives. Features; 2010.

Kissel R, Scholl M, Skolochenko S, Li X. NIST SP800-88 Guidelines for media sanitization. NIST Spec Publ; 2006:88.

Lee, S., Moon, B., Park, C., Kim, J. and Kim, S. A case for flash memory SSD in enterprise database applications. SIGMOD '08, (Vancouver, 2008), ACM, pp. 1075–1086.

Pavan P, Bez R, Olivo P, Zanoni E. Flash memory cells-an overview. Proceedings of the IEEE 1997;85(8):1248–71.

Phillips BJ, Schmidt CD, Kelly DR. Recovering data from USB flash memory sticks that have been damaged or electronically erased. In: Proceedings of the 1st international conference on Forensic applications and techniques in telecommunications, information, and multimedia and workshop. Adelaide, Australia: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering); 2008. p. 1–6.

Regan JE. The forensic potential of flash memory computer science. Monterey: Naval Postgraduate School; 2009. 99.

Shu, F. Data SET Management commands Proposal for ATA8-ACS2 2007.

Skorobogatov S. Data remanence in flash memory devices. In: Cryptographic hardware and Embedded systems (CHES) 2005 7th International Conference. Edinburgh, UK: Springer-Verlag New York Inc.; 2005. p. 339.

Stevens, C.E. Working draft AT attachment 8 – ATA/ATAPI command set (ATA8-ACS) 2006.

Stokes M. An integrated approach to recovering deleted files from NAND flash data. Small Scale Digital Device Forensics Journal 2008;2(1):13.

Wei M, Grupp LM, Spada FE, Swanson S. Reliably erasing data from flash-based solid state drives. USENIX Association; 2011. 8–8.

Christopher King is a Member of the Technical Staff in the CERT Program, part of the Software Engineering Institute at Carnegie Mellon University. His recent work at CERT has focused around creating insider threat technical controls, developing insider threat assessments, and researching new techniques for insider threat prevention. Before coming to CERT, Chris worked at several government and industry organizations, working on information security architectures, systems engineering, and application development. His research interests are in a variety of topics, including digital forensics, network security, and organizational theory. Chris has a M.S. in Information Security Policy and Management from Carnegie Mellon University, a B.S. in Information Sciences and Technology, and a B.A. in Medieval Studies, both from Penn State University.

Timothy Vidas is an Electical and Computer Engineering PhD student at Carnegie Mellon University. His recent research interests revolve around mobile platform security and privacy, but his interest often strays into volatile data collection and analysis, reverse engineering, malware analysis, and the like. On occasion, Tim provides digital forensics mission support to several federal agencies. Prior to becoming a full-time PhD student, Tim has held research positions at CERT and at universities such as the Naval Postgraduate School and the University of Nebraska working on a variety of computer and network security projects such as a high assurance security kernel, collaborative reverse engineering, and RAM analysis from a forensic perspective. In addition to research, Tim likes to teach and has a wide set of IT-related interests. He maintains several affiliations like ACM, IEEE, HTCIA, and Infragard and has obtained several industry certifications. Tim has a B.S. and M.S. in computer science and is a DEFCON CTF black badge holder, a contest he currently organizes with two other talented individuals. In his free time he toys around with digital forensics competitions, CTF exercises, and any other interesting looking challenges.