

Deconstruct *and* Preserve (DaP): A method for the preservation of digital evidence on Solid State Drives (SSD).

I. Mitchell, T. Anandaraja, S. Hara, G. Hadzhinenov, & D. Neilson

Middlesex University

Abstract. Imaging SSDs is problematic due to TRIM commands and garbage collectors that make the SSD behave inconsistently over time. It is this inconsistency that can cause a difference between images taken of the SSD. These differences result in unmatched hash number generation and would normally be attributed to contamination or spoliation of digital evidence. DaP is a proposed method that ensures all images taken of the SSD are consistent and removes the volatility normally associated with these devices. DaP is not focused with the recoverability of deleted data, however DaP does stabilise the device to prevent unintentional contamination due to garbage collection. Experiments show that the DaP method works on a range of devices and consistently produces the hash-identical images. The conclusions are to consider DaP as a new Standard Operating Procedure (SOP) when imaging SSDs.

1 Introduction

A principle throughout all stages of any forensic investigation is the preservation of evidence. Digital Forensics is not exempt from this principle and has been identified at an embryonic stage in McKemmish [6], in digital forensic guidelines [1,14,11] and digital forensic frameworks [2,4]. Beebe & Clarke [2] state that digital evidence preservation is a fundamental principle and should be considered in all phases of a digital forensic investigation and continue to suggest that spoliation of digital evidence should be kept minimal and extensive contemporaneous notes are to be taken in such cases. These are cases where in order to extract the evidence it is necessary for the competent investigator to alter that evidence, this is not the case with SSDs. It has been demonstrated in [5] that with the investigator following SOPs and imaging the SSD, the digital evidence on the SSD changes. These changes are due to garbage collection and are independent of write-blockers, meaning that an SSD can change over time. Such “covert” changes yield inconsistencies between third party images. It is becoming common knowledge among practitioners that the practicality of producing two or more exact byte-for-byte images (hash-identical) from SSDs for all parties is becoming problematic [3]. Generally, the probability of producing hash-identical images is diminished as the time between seizure and data acquisition stage is increased. Nisbet *et al* [8] showed that for some TRIM enabled file systems deleted

data is unrecoverable after 1 hour. This includes time between different data acquisitions and is due to the volatility of SSDs, TRIM enabled file systems and the variance of aggressiveness of the manufacturer's garbage-collector. Furthermore there has been an increase in manufacturing of SSDs [13] that indicate SSDs will become ever more present in Digital Forensic Investigations. The problem is that there is a preconceived idea that these are storage media and therefore treated as traditional HDD. This treatment of digital evidence could jeopardise the evidential integrity and therefore raise questions about its admissibility in court. In addition TRIM enabled file systems can wipe clean SSDs in less than a minute. This is ideal for a quick sanitisation and permanent removal of all data without the prospect of recovery. The unrecoverability is attractive to criminals, who would rather lose their data than have it seized, examined and analysed for prosecution or as a result of e-Discovery requests.

McKemmish [6] states, "meaning of the data accessed by such change has not been unduly compromised". As long as the change can be accounted for, completed with competence and contemporaneous notes taken results in the potential digital evidence not being unduly compromised then change to digital evidence can be allowed in a court of law. In Carrier & Spafford's DFF [4] digital Crime Scene Preservation and Documentation sub-phase it states, "... preserve the state of as many digital objects as possible by reducing the number of additional events that may occur...". The proposed DaP procedure does exactly this by prohibiting the activation of the garbage collector and the non-execution of TRIM commands on the SSD.

The need for a systematic way of producing hash-identical images for SSDs would be advantageous to all parties and may include: reduction in time spent on questioning the evidential integrity of digital evidence obtained from SSDs; and reduce administrative and cognitive burden on Digital Forensic Analyst to ensure that minimal steps taken to reduce the risk of contaminating the data on SSDs. The proposed Standard Operating Procedure (SOP), Deconstruct and Preserve (DaP), is explained in §3, experiments on DaP are described in §3, results of experiments are described in §4 and finally conclusions are in §5.

2 Background

2.1 Solid State Drives, SSD.

SSDs cannot over-write like HDDs. This, along with wear-levelling [3], means that the equivalent of sectors have to be reset to zeroes, before being written to. This creates all kinds of problems that is the responsibility of the controller. One of the most notorious challenges to computer forensics is the garbage collector. To increase the efficiency of the reset/write process on SSDs the garbage collector will identify deleted files, and their associated blocks and reset them. This means that if the garbage collector is scheduled then any deleted data on the SSD will be lost. A write-blocker will not prevent the loss of data since the garbage collector is located in the controller.

To help this process further file systems have developed TRIM. When TRIM is enabled it allows the Controller to identify deleted/unallocated space on the SSD and start the reset process. TRIM has a scheduled time and initiates the garbage collector. Why is this an issue for Digital Forensics? There are two reasons: i) data loss; and ii) evidential integrity.

Data Loss. Wipe commands are more effective on SSDs. This allows devices to be wiped quicker, essentially this is the responsibility of the garbage collector and manufacturers have established aggressive ways of resetting gates to zero. Whereas to wipe a HDD may take two takes and many hours, SSDs can do this in one take and minutes. It will not be long before suspects can initiate a wipe of the SSD remotely. Once the SSD has been wiped then recovery of data is impossible; this data-loss is a threat to digital forensics and would change the SOP of seizure and collection of digital evidence.

Evidential Integrity. Evidence integrity forms the basis of certainty that evidence under investigation has not been changed upon seizure or when volunteered. All best practice guidelines [1] stipulate that evidence in the first instance should not be changed, or when change is necessary this process must be documented and be repeatable by third parties using the same steps.

Typically a digital forensic investigation of a device can be described in [6] and by the following notation in Fig.1. This shows the Acquisition stage and importantly the verification that the images, $I_i, \forall i > 0$, are byte for byte copies of the device, I_0 . The verification of evidential integrity is described in the acquisition stage by ensuring that the hashes match. When hashes do not match during an acquisition stage the evidential integrity has been compromised.

Seizure: $S(I_0)$

Management: $M(I_0)$

Acquisition: $D(I_0) = I_1$

– $D(I_0) = I_i$

where $D(I_0)$ is the duplication process

– $H_k(I_0) = H_k(I_i) \forall i$,

where

$i = 1, 2, 3, \dots, n$;

H_k is a hashing algorithm, e.g. SHA1 then H_{SHA1} ; and

n is the number of images of the device, I_0 , required.

Analysis: $A(I_1)$

Report: $R(I_1)$

Fig. 1. Nomenclature to describe five stages of Digital Investigation as indicated in [6]. This notation indicates the device as I_0 and any resulting image of that device as I_n , where $n > 0$. For example, the Acquisition stage yields I_1 and subsequent stages use I_1 , an image, and not I_0 , the original device.

Whilst other technology that is susceptible to change (such as GPS, mobile, network seeking technology) can be managed by investigators by using methods to physically secure those types of devices to prevent access, change to data and deliberate deletion, SSDs have no such measure.

SOP on mobile devices have adopted to best perfect techniques that can recover data from volatile devices, non-standard operating systems, hybrid devices, encrypted and locked devices. Peer acceptance from investigators together with pseudo-standardised processes offered by forensic vendors have led to a community practice where in mobile phone investigations it is an accepted practice that at times agents [7] may need to be installed on Android devices to extract data from devices and removed post extraction.

The proposed DaP SOP wants to create a peer accepted methodology for SSD technology regardless of manufacturer differences. This development would provide assurance to practicing investigators that evidence has not been altered and provides confidence that steps are repeatable by third parties. By ensuring evidential integrity suspects and defence teams cannot argue that crucial evidence has been lost that would aide in their defence.

3 Method

The set of experiments show how evidence can be deliberately deconstructed in order to preserve evidence. The experiment demonstrates how this would work on traditional devices without loss or effect to content. This paper then continues the experiment on SSD and provides a procedure for preserving evidence via deconstruction. Table 1 outlines the stages in DaP, this is further described in the list below and would be encapsulated in standard operating procedures for seizure, transportation, storage, analysis, reporting and presentation of digital evidence.

- 1. Pre-verification** : non-essential, but strongly recommended and required to prove preservation of potential digital evidence and no contamination has occurred. Also benefits Digital Forensic Investigators to confirm and verify the data acquisition stages, see stages 5-6. Using a hashing algorithm, hash the SSD to yield H_{I_0} .
- 2. Deconstruction** : Completed by competent practitioner trained in DaP. Deconstructs the SSD/partition and identifies the component responsible for management, e.g. MBR or VBR. Once identified the start and end of the offset address of the component are recorded, it is then extracted from I_0 and stored as I_1 in a secure location by the custodians.
- 3. Preservation** : Completed by competent practitioner trained in DaP. Preserving the SSD partition requires inserting, \oplus , the component identified in the deconstruction stage with zeroes, Z_{y-x} . The size of Z is determined by $y - x$ and the start location is at offset x . This stage preserves the SSD by putting it in a stable state and therefore prevent any risk of contamination. It is recommended to Hash the device that can be matched in stage 4 and thus ensure that the acquisition stage is completed correctly.


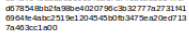







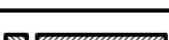
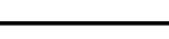
Stage	Input Visual	Process	Output	Output Visual
1. Pre-verification		$H_k(I_0)$	H_{I_0}	
2. Deconstruction		$E_x^y(I_0)$	I_1	
3. Preservation		$Z_{y-x} \oplus_x^y I_0$	I_0	
4. Acquisition		$D(I_0)$	I_j	
5. Reconstruction		$I_1 \oplus_x^y I_j, j \geq 2$	I_j	
6. Verification		$H_k(I_j), j \geq 2$	H_{I_j}	$H_{I_0} = H_j, \forall j \geq 2$

Table 1. Stages for Deconstruct and Preserve, DaP. Explanation of stages as follows: 1. H - hashing function e.g. SHA256, to uniquely identify device; 2. E_x^y - deconstructs and extracts VBR between offset x, y ; 3. Z_{y-x} - zeroes of length $y - x$; \oplus_x^y - inserts the first operand, Z_{y-x} , into the second operand, I_0 . 4. $D(I_0)$ - data acquisition producing an exact byte-for-byte copy, I_j . 5. \oplus inserts the output from deconstruction into the image, I_j between offset x, y . 6. H - complete hashing on reconstructed image, the result equals the original device for all images.

4. **Acquisition** : Using write-blockers and NIST approved software duplicate the original device, I_0 . This stage can be repeated many times to yield I_j , where $j \geq 2$.
5. **Reconstruction** : Request I_0 from the custodians along with the start and end offset addresses, x, y . Insert, \oplus , the original component, I_0 , into the image of the preserved SSD, I_j , to yield a new image that can be read and analysed.
6. **Verification** : Non-essential but strongly recommended. Using a hashing algorithm, hash the image, $H_k(I_j)$, and test for match with H_{I_0} .

All stages are fully automated using code as described in experiment. DaP is formally described as in Fig. 2.

4 Results & Evaluation

4.1 Control Experiment.

The following list explains the process of the experiment, that is not too different from [3]. The minor differences are: i) Python code, instead of batch code; ii) the replacement of 'EVIDENCE' with '12345678'; and iii) the size of the files. The

Pre – Verification : $H_k(I_0) \rightarrow H_{I_0}$
Deconstruction : $E_x^y(I_0) \rightarrow I_1$
Preservation : $Z_{y-x} \oplus_x^y I_0 \rightarrow I_0$
Acquisition : $D(I_0) \rightarrow I_j, j \geq 2$
Reconstruction : $I_1 \oplus_x^y I_j \rightarrow I_j, j \geq 2$
Verification : $H_k(I_j) = H_{I_0}, \forall j \geq 2$

Fig. 2. Description of DaP as a generalised formal process. The operation $Q \oplus_x^y P$ indicates insert Q in P between the offset address x, y

major difference is the **Deconstruct and Preserve** stage that backs up data from the VBR (FAT file system was used here), thus rendering the file system useless and disabling the ability of any file system dependent information being altered. Then by reconstruction of the device as an image so that software can be used to analyse and produce reports.

The control experiment was completed on a USB using FAT16. The size of the USB was small at ≈ 256 Mbytes. This experiment demonstrates an implementation of DaP and its principles, albeit on a small scale. The experiment was repeated 5 times and results shown in Table 2. The hashing algorithm used was SHA with 512Byte option, reducing hash collision [10] to an insignificant probability. These set of results had 100% matches and therefore the practical supports the theory with non-volatile media. In the next sub-section the aim is to apply this to SSDs.

Test	Device SHA512	Repatriation SHA512	Recovery Percent
1	match	match	100
2	match	match	100
3	match	match	100
4	match	match	100
5	match	match	100

Table 2. Control experiment and yields identical SHA512 hashes for Device and Repatriation. Results were conducted on a ≈ 256 MByte USB, a non-volatile data storage medium.

4.2 SSD Without DaP

Section 4.1 shows that DaP works on traditional media. To see if DaP works on SSDs (file system: EXT4, OS: Linux, Ubuntu 14.04 LTS) an experimental framework was set up to test the hypothesis: does the introduction of DaP preserve evidence? To test this two experiments are required: i) to prove without DaP evidence is destroyed; and ii) to prove with DaP evidence is preserved. The following steps were repeated 5 times on various partitions, all code written in

python and available on request from first author. The following steps outline the experiment:

Format: Create a ≈ 15 Gbyte TRIM enabled partition. External Partition connected SATA to eSATA, since TRIM commands can be executed through SATA [12].

Generation: Generate a template/file with string '12345678'

Populate: Fill (98%) the partition with the file.

Deletion: Delete Files

Hash: Hash the partition, H_1

TRIM: Issue TRIM command, `fstrim -v /media/user/sdb1`. This is similar to pseudo-activation of cron, or likewise in other operating systems, to instantiate and identify deleted files.

Hash: Hash the partition, H_2

Test	Device	TRIM Enabled	$H_1 = H_2$
1	SanDisk	Yes	False
2	OCA-Agility	Yes	False
3	Micron M510	Yes	False
4	Kingston V Series	No	n/a
5	SK Hynix SC210	Yes	False

Table 3. SSD without DaP. All hashes before and after TRIM do not match.

This experiment demonstrates without DaP the evidence on the SSD is changed and thus the two images before and after differ.

4.3 SSD With DaP

The above results in section 4.2 show that after issuing the TRIM command the evidence changes. In this section the experimental framework introduces DaP, all other conditions remain, e.g. OS. The experiment is described in the list below:

Format: Create a ≈ 15 Gbyte TRIM enabled partition. External Partition connected SATA to eSATA, since TRIM commands can be executed through SATA [12].

Generation: Generate a template/file with string '12345678'

Populate: Fill (98%) the partition with the file.

Deletion: Delete Files

Hash: Hash the partition, H_1

DaP: Complete stages 1-2 of DaP.

TRIM: Issue TRIM command, `fstrim -v /media/user/sdb1`

DaP: Complete stages 3-4 of DaP.

Hash: Hash the image partition, H_2

Test	Device	TRIM Enabled	$H_1 = H_2$
1	SanDisk	Yes	True
2	OCA-Agility	Yes	True
3	Micron M510	Yes	True
4	Kingston V Series	No	True
5	SK Hynix SC210	Yes	True

Table 4. SSD with DaP. All hashes before and after TRIM command match.

The results show that the partitions before and after TRIM are preserved and therefore future images of this partition can match. The content can be retrieved once the image is reconstructed. In all cases this produced a match with the original content and thus preserves digital evidence (DaP also worked on TRIM disabled SSD).

The hypothesis is accepted, DaP preserves digital evidence on: TRIM enabled SSDs; TRIM disabled SSDs; and traditional non-volatile storage media.

5 Conclusion

DaP is a proposal to effectively resolve issues related to digital evidence preservation on SSDs and ensures the stability and consistency of current and future images taken from the SSD. To emphasise DaP is not going to increase the ability to recover data from SSD. Market share of SSD is increasing, along with that is knowledge by defendants that there are methods to wipe the information with 0% chance of no data recovery. Research [3,5,8] has backed this fact up and shows under certain circumstances how this can be achieved. Under e-Discovery requests, such violations are covered by the wilful destruction of evidence under the guise of a preservation order [9], however a preservation order does not exist at crime scenes. The defendant can exercise the deletion of data remotely, once the TRIM and deletion command is issued the garbage collector destroys any chance of recovering the data. This all happens whilst in the Digital Forensic Lab. DaP is able to preserve the data on the SSD and keep the data consistent between different images taken of the SSD. With the above experiments this has been shown to be stable and valid; each experiment was duplicated a few times to ensure the reproducibility and each time each image gave the same result, identical hash number. DaP would work on HDD as well and add a further layer of protection along with traditional hardware, such as write-blockers. With this in mind the introduction could change several SOPs for first response teams. Finally, there are some recommendations for the use of DaP.

5.1 Recommendations.

DaP has been presented as a SOP to handle the data acquisition stage of an SSD. The recommendations are as follows:

- Follow SWDGE, NIJ or other national [11,14] guidelines for Digital Evidence Sequestration for First Response Teams (FRT).
- complete the extraction stage as early as possible, and even consider this as part of the FRT’s procedures.
- use a custodian database to store the extraction of H_1 . This is future work and would involve chain of custody updates for other requests to image the device.

References

1. Association of Chief Police Officers (ACPO): Good practice guide for digital evidence (ver. 5) (March 2012), <https://www.7safe.com/research-and-insight/acpo-guidelines>
2. Beebe, N.L., Clark, J.G.: A hierarchical, objectives-based framework for the digital investigations process. *Digital Investigation* 2(2), 147–167 (2005)
3. Bell, G.B., Boddington, R.: 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 (2010)
4. Carrier, B., Spafford, E.H.: An event-based digital forensic investigation framework. In: *Digital forensic research workshop*. pp. 11–13 (2004)
5. King, C., Vidas, T.: Empirical analysis of solid state disk data retention when used with contemporary operating systems. *Journal of Digital Investigation* 8, S111–S117 (2011)
6. McKemmish, R.: What is forensic computing? *Trends and Issues in Crime and Criminal Justice* (118) (1999)
7. MSAB: XRY - Android basics: debugging and extractions (2015), available on XRY certification course
8. Nisbet, A., Lawrence, S., Ruff, M.: A forensic analysis and comparison of solid state drive data retention with trim enabled file systems. In: *Australian Digital Forensics Conference*. pp. 103–11 (2013)
9. Redgrave, J.M.: *The Sedona Principles: Best Practices, Recommendations & Principles for Addressing Electronic Document Production*. Pike & Fischer-A BNA Company (2007)
10. Rogaway, Shrimpton: Cryptographic hash-function basics: Definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance. *LNCS Software Encryption* (2004)
11. Scientific Working Group on Digital Evidence (SWDGE): Model standard operation procedures for computer forensics (ver. 3), <https://www.svgde.org/>
12. Shu, F., Obr, N.: Data set management commands proposal for ata8-acs2. *Management* 2, 1 (2007)
13. Statista.com: Global shipments of hdds and ssds in pcs from 2012 to 2017 (June 2016), <http://www.statista.com/statistics/285474/hdds-and-ssds-in-pcs-global-shipments-2012-2017/>
14. U.S. Department of Justice : *Electronic Crime Scene Investigation: An On-the-Scene Reference for First Responders*. National Institute of Justice (November 2009)