JCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-6, Issue-10, Oct 2018 E-ISSN: 2347-2693

Result Analysis of Hash Value Generation Using Security Algorithm for Device Forensics

Arvind S. Kapse[1*], Vilas M. Thakare[2], Avinash S. Kapse[3]

[1]Singhania University, Pacheri Bari, Dist:-Jhunjhunu, Rajasthan, India
[2]Dept.of CSE., Sant Gadge Baba Amravati University, Amravati.Maharashtra, India
[3]Dept. of CSE Anuradha Engg. College, Chikhli, Dist:- Buldhana, Maharashtra, India

[*]*Corresponding Author: arvind.kapse@yahoo.com,Mob.8446671321*

Available online at: www.ijcseonline.org

Accepted: 22/Oct/2018, Published: 31/Oct/2018

Abstract- Digital forensics tools are often used to calculate the hash value of the digital test unit. The MD5 and SHA hash function is used in digital forensics tools to calculate and verify that a dataset has not been altered, due to the application of multiple collection and analysis tools and procedures of evidence. In addition, because of the impact on the personal life of the subject of the survey, the verification of the proper functioning of the tools and procedures is crucial. This article discusses the importance of hashing value in digital forensics for digital evidence. The search uses six different possible cases as an experiment to generate and verify the hash value of the test drive by using a forensic tool to demonstrate the importance of the hash value in digital forensics. In addition, unreliable results can be obtained due to incorrect use of the Tools application.

***Keywords*—**SHA; MD5; hash function ; digital forensic.

I. INTRODUCTION

Digital forensics has grown rapidly in recent years as the use of forensic computing has proved invaluable in a wide range of court proceedings. Digital forensics is used not only to investigate computer crimes, such as network intrusion, data fabrication and the distribution of illegitimate material through digital services, but also to investigate crimes in which Evidence is stored in any format. Digital on any digital device [1]. One of the most important steps in a digital forensic investigation is the data acquisition stage, which is "collecting digital evidence of electronic media" [6]. During this step, the investigator creates an exact copy of the record or evidence file to produce a forensic copy. To avoid destroying the evidence, the investigation is conducted on the forensic copy instead of the original evidence data. As a result, any damage to data that occurs during the investigation process can be repaired using the test disk to create a new forensic copy that will be used to continue the investigation. Since a digital survey often produces results that are used in criminal or civil proceedings that may radically affect a person's life, the investigator must be absolutely sure that the forensic copy is an exact copy of the evidence. The hash value plays an important role in forensic investigations to test the accuracy of digital data in court. In this research article, we propose a real-time case study to demonstrate the importance of hash value in the digital forensic investigation process. The rest of this document is organized as follows. Section 2 provides an overview of digital forensics. Section 3 provides a brief summary of the hash function. Section 4 focuses on a case study of the hash value generated on the digital hard drive from a forenspoint of view. Finally, Section 5 concludes the work and our future work.

II. DIGITAL FORENSIC SCIENCE

Use of scientifically proven and proven methods for the preservation, collection, validation, identification, analysis, interpretation, documentation and presentation of digital evidence derived from digital sources to facilitate or promote the reconstruction of events considered as criminal or useful. . Anticipate unauthorized actions prejudicial to planned operations. In 2001, the DFRW [3] research workshop proposed a digital survey process with the following six steps. At this point, we are more concerned with the analysis phase; hash analysis is also part of all the numerical analyzes that we mentioned in the original DFRW model shown in Figure1.

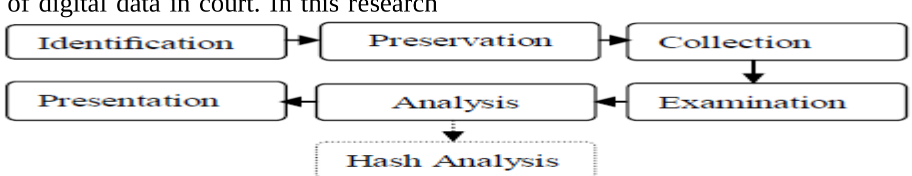

Fig. 1 Digital forensic investigation process

Digital forensics is the science that identifies, extracts, analyzes and presents digital evidence stored in digital electronic storage devices for use in court [1, 4, 5].

III. FUNCTION OF HASH

Definition: An algorithm that converts a variable amount of text into a fixed-size output (hash value). Hash functions are used to create digital signatures, hash tables, and short text condensations for analysis purposes. Hash functions are also called "cryptographic hash functions". A hash function H is a transformation that takes an input of variable size 'm' and returns a string of fixed size, called hash value h (i.e., h = H (m)). Hash functions with only this property have a variety of general computing uses, but when used in cryptography, hash functions are typically chosen to have additional properties. The basic requirements of a cryptographic hash function are:

- The entry can be of any length,
- The output has a fixed length,
- H (x) is relatively easy to calculate for a given x,
- H (x) is unidirectional,
- H (x) is free of collisions.

It is said that a hash function H is unidirectional if it is difficult to invert, where "hard to reverse" means that, given a hash value h, it is impossible to find an entry x such that H (x) = h. If, given a message x, it is impossible in computation to find a message and not equal to x such that H (x) = H (y), then we say that H is a collision-free hash function. A hash function H without collision is a function for which it is not feasible to find two messages x and y such that H (x) = H (y). The hash value concisely represents the message or longest document from which it was calculated. you can imagine a message summary as a "fingerprint" of the larger document. Perhaps the main function of a cryptographic hash function is to provide digital signatures. Since hash functions are generally faster than digital signature algorithms, it is common to calculate the digital signature in a document by calculating the signature in the hash value of the document, which is small compared to the document itself. In addition, a summary may be made public without revealing the content of the document from which it is derived. This is important for digital timestamping, where, using hash functions, a timestamp of the document can be obtained without revealing its contents to the timestamp service.

A. CONVENIENT HASH FUNCTION

This section covers the hash functions that are most likely used in forensic / software tools: MD5 and SHA-1. For a detailed description, we refer the reader to the documents published by the standardization bodies. MD4 and MD5: MD4 was proposed by Ron Rivest in 1990 and MD5 [7] soon followed as the most powerful version. Its design had a great influence on the subsequent constructions of the hash function. The letters "MD" mean "message digest" and the numbers refer to the functions corresponding to the fourth and fifth designs of the same family of hash functions. SHA-0 and SHA-1: The Secure Hash Algorithm (SHA) was originally approved for use with the Digital Signature Standard (DSS) in 1993 [2]. Two years later, the standard was updated to become what is now called SHA-1 [8]. The first version of SHA is called SHA-0 in cryptographic literature, although it has never been officially designated by it. SHA-1 differs from SHA-0 exactly by an additional instruction, which is, however, extremely important from the point of view of cryptography. Since there was no reason to prefer the initial version of the standard, SHA-1 replaced SHA-0 in all but the most obsolete applications. The details of these hash functions are briefly illustrated in Table 1 below.

Table 1. Practical Hash function

Name	Block Size(bits)	Word Size(bits)	Output Size(bits)	Rounds
MD4	512	32	128	48
MD5	512	32	128	64
SHA-0	512	32	160	80
SHA-1	512	32	160	80

A. HASH VALUE GENERATION IN DIGITAL FORENSIC

Generally hash value is used to check the integrity of any data file but, in digital forensic it is used to check the integrity of evidence disk data. The image of a disk is created in digital forensic for analysis so, it is necessary the image have exactly or replica of evidence disk. The hash value generated during imaging should match when that image of evidence disk is extracted for detail analysis. In digital forensic hash value is generated for whole disk data not only single or multiple files. The hash value generated using forensic tools in the form of hexadecimal notation. Here we are giving an example to convert it in too two easily understandable form for forensic practitioner who don‟t have enough knowledge about computer system. Using the hash value generated of case1: 79EAB87F0D3A3B45954779A72F79AE63

Table 2 shows the binary form of the given: Hexadecimal value

Table 2 Binary code for Hash value

0111	1001	1110	1010	1011	1000	0111
7	9	E	A	B	8	7
1111	0000	1101	0011	1010	0011	1011
F	0	D	3	A	3	B
0100	0101	1001	0101	0100	0111	0111
4	5	9	5	4	7	7
1001	1010	0111	0010	1111	0111	1001
9	A	7	2	F	7	9
1010	1110	0101	0011			
A	E	6	3			

2. The following steps involve to convert the given hexadecimal hash value into decimal form:

Step1. Use Hexadecimal to Decimal conversion process as given below: 7*1631 + 9*1630 +E*1629+A*1628 +…………6*161 +3*160

Step2. Substitutes the equivalent numerical value in place of alphabet in hexadecimal hash value as given below. A=10, B=11, C=12, D=13, E=14, F=15 After substitution: 7*1631 + 9*1630 +*14**1629+*10**1628 +…….6*161 +3*160

Step3. Calculate hash value in decimal form.

IV. PROPOSED FRAMEWORK FOR HASH VALUE CALCULATION

The experimental model / framework for calculating the hash value in digital forensics is shown in Figure 2. With this model, test data was generated on the digital hard disk. You should also look for evidence of system tampering, data concealment or utility removal, unauthorized system changes, and so on. Detecting and recovering hidden or hidden information is a tedious task. Data must be searched carefully to recover passwords, find unusual hidden files or directories, file extension errors and signatures, and so on. When searching for the above mentioned information on a test disk, the forensic software also creates the hash value of the entire unit to check for integrity of the disc. During the forensic data acquisition phase, the hash value is generated when viewing the test disk and comparing this hash value when examining or copying the contents of the disk. If the hash value is the same as that of the forensic expert, suppose everything is fine, otherwise a certain type of manipulation is bound to the test disk. Here, the case study focuses on the importance of the hash value in the forensic examination that has been explored.

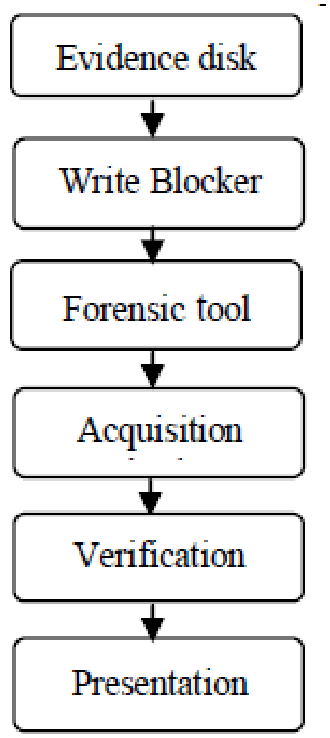

Fig. 2 Hash calculation model

The template is used to create an image of the test hard disk connected to the write blocker (for example, Fast Blok), to prevent any vulnerable program running on the system from writing anything. We can use any investigative tool to create a disk image with a hash value. Encase foren-sics [9] is a simple but concise tool used in this case study. Save an im-age of a hard disk to a file or segments that can be rebuilt later. Calculates the MD5 hash values and confirms the in-tegrity of the data before closing the files. The raw image created by the Encase program is now used for analysis and review purposes. When extracting data from the raw image, the Encase program also checks the previously generated hash value and creates a summary / report for the Judicial Validation and Presentation that searches for the match. The importance of the hash value in digital forensics is illustrated in six different cases that have been generated and analyzed below.

CASE 1. Calculate the hash value in the Original:

Here we generate the hash value of the original test disk, which contains suspicious files / data for forensic analysis and also checks the hash value after acquisition / visualiza-tion, report presented below in Figure 3.

Name:	Hash value test drive		
Description:	Physical Disk, 156301488 Sectors, 74.5GB		
Logical Size:			
Physical Size:	512		
Starting Extent:	0S0		
File Extents:	1		
Physical Location:	0		
Physical Sector:	0		
Evidence File:	Hash value test drive		
Full Path:	hash value test drive\Hash value test drive		
File Extents			
Start Sector	Sectors	Start Cluster	Clusters
1			
Device			
Evidence Number:	Hash value test drive		
File Path:	F:\research\case 1\Hash value test drive.E01		
Examiner Name:	kk		
Actual Date:	01/23/12 03:22:28PM		
Target Date:	01/23/12 03:22:28PM		
Total Size:	80,026,361,856 bytes (74.5GB)		
Total Sectors:	156,301,488		
File Integrity:	Completely Verified, 0 Errors		
EnCase Version:	6.2		
System Version:	Windows XP		
Acquisition Hash:	79EAB87F0D3A3B45954779A72F79AE63		
Verify Hash:	79EAB87F0D3A3B45954779A72F79AE63		
Partitions			
Code Type	Start Sector	Total Sectors	Size
07 NTFS	0	156,296,385	74.5GB

Fig. 3: Case 1 report

CASE 2. Add any file in the test unit and check the hash with the original: In this case, the experiments generated the report shown in Figure 4. The impact of adding a file to the test unit by mistake or by concern, correspondingly, the hash value is checked with the original. The hash value differs from the original / actual proof unit.

Original hash value: 79EAB87F0D3A3B45954779A72F79AE63

New hash value: DE9EAD6A3B7B02475ADB6EB83CCB2826

Name:	hash value case2		
Description:	Physical Disk, 156301488 Sectors, 74.5GB		
Logical Size:			
Physical Size:	512		
Starting Extent:	0S0		
File Extents:	1		
Physical Location:	0		
Physical Sector:	0		
Evidence File:	hash value case2		
Full Path:	hash value case 2\hash value case2		
File Extents			
Start Sector	Sectors	Start Cluster	Clusters
1			
Device			
Evidence Number:	hash value case2		
File Path:	D:\research\CASE2~1\HASHVA~1.E01		
Examiner Name:	kk		
Actual Date:	01/31/12 03:51:21PM		
Target Date:	01/31/12 03:51:21PM		
Total Size:	80,026,361,856 bytes (74.5GB)		
Total Sectors:	156,301,488		
File Integrity:	Completely Verified, 0 Errors		
EnCase Version:	6.2		
System Version:	Windows XP		
Acquisition Hash:DE9EAD6A3B7B02475ADB6EB83CCB2826			
Verify Hash:DE9EAD6A3B7B02475ADB6EB83CCB2826			
Partitions			
Code Type	Start Sector	Total Sectors	Size
07 NTFS	0	156,296,385	74.5GB

Fig. 4: Case 2 report

CASE 3. Remove any file from the test unit and compare the hash with the original:

In this case, the experiments show that if one of the files is removed from the evidence disk, the corresponding hash value of the unit is generated and compared to the original. The generated hash value difference report is shown in Figure 5.

Original hash value:
79EAB87F0D3A3B45954779A72F79AE63

New hash value:
ECB15214986D91DF876F2F773F9E0F4D

Name:	hash value test drive		
Description:	Physical Disk, 156301488 Sectors, 74.5GB		
Logical Size:			
Physical Size:	512		
Starting Extent:	1S0		
File Extents:	1		
Physical Location:	0		
Physical Sector:	0		
Evidence File:	hash value test drive		
Full Path:	hash value test drive\hash value test drive		
File Extents			
Start Sector	Sectors	Start Cluster	Clusters
1			
Device			
Evidence Number:	hash value test drive		
File Path:	F:\research\case 3\hash value test		
drive.E01			
Examiner Name:	kk		
Actual Date:	01/24/12 04:42:20PM		
Target Date:	01/24/12 04:42:20PM		
Total Size:	80,026,361,856 bytes (74.5GB)		
Total Sectors:	156,301,488		
File Integrity:	Completely Verified, 0 Errors		
EnCase Version:	6.2		
System Version:	Windows XP		
Acquisition Hash:	ECB15214986D91DF876F2F773F9E0F4D		
Verify Hash:	ECB15214986D91DF876F2F773F9E0F4D		
Partitions			
Code Type	Start Sector	Total Sectors	Size
07 NTFS	0	156,296,385	74.5GB

Fig. 5: Case 3 report

CASE4. Edit any file:

This case is totally different from the two previous cases of adding and deleting files from the unit, it is described in two cases below:

- **Case 4.1. Add content to any file and check the hash with the original:** Here, the case is displayed when a small amount of data is added to a file. Thereport generated with the hash value is shown in Figure 6. The comparison of the hash value with the original is also mentioned below.

Original hash value:
79EAB87F0D3A3B45954779A72F79AE63

New hash value:
E02365F1BFCCA37AAB5E62D6262EBADE

Name: hash value test deive
Description: Physical Disk, 156301488 Sectors, 74.5GB
Logical Size:
Physical Size: 512
Starting Extent: 1S0
File Extents: 1
Physical Location: 0
Physical Sector: 0
Evidence File: hash value test deive
Full Path: hash value test drive\hash value test deive
File Extents

Start Sector	Sectors	Start Cluster	Clusters
1			

Device
Evidence Number: hash value test deive
File Path: F:\research\case4_1\hash value test deive.E01
Examiner Name: kk
Actual Date: 01/25/12 10:53:48AM
Target Date: 01/25/12 10:53:48AM
Total Size: 80,026,361,856 bytes (74.5GB)
Total Sectors: 156,301,488
File Integrity: Completely Verified, 0 Errors
EnCase Version: 6.2
System Version: Windows XP
Acquisition Hash: E02365F1BFCCA37AAB5E62D6262EBADE
Verify Hash: E02365F1BFCCA37AAB5E62D6262EBADE
Partitions

Code	Type	Start Sector	Total Sectors	Size
07	NTFS	0	156,296,385	74.5GB

Fig. 6: Case 4.1 report

- **Case 4.2. Remove some contents from any file and compare hash with original:**

Here we are demonstrates the case when some portion of data is erased from any file. The generated report with hash value shown in fig. 7 below.

Original hash value:
79EAB87F0D3A3B45954779A72F79AE63
New hash value:
43D25B68F22A84CD95C5214F0414E511

Name: hash value test drive
Description: Physical Disk, 156301488 Sectors, 74.5GB
Logical Size:
Physical Size: 512
Starting Extent: 0S0
File Extents: 1
Physical Location: 0
Physical Sector: 0
Evidence File: hash value test drive
Full Path: hash value test drive\hash value test drive
File Extents

Start Sector	Sectors	Start Cluster	Clusters
	1		

Device
Evidence Number:hash value test drive
File Path: D:\research\case4_2\hash value test drive.E01
Examiner Name:kk
Actual Date: 01/27/12 11:13:09AM
Target Date: 01/27/12 11:13:09AM
Total Size: 80,026,361,856 bytes (74.5GB)
Total Sectors: 156,301,488
File Integrity: Completely Verified, 0 Errors
EnCase Version: 6.2
System Version: Windows XP
Acquisition Hash:43D25B68F22A84CD95C5214F0414E511
Verify Hash: 43D25B68F22A84CD95C5214F0414E511
Partitions

Code	Type	Start Sector	Total Sectors	Size
07	NTFS	0	156,296,385	74.5GB

Fig. 7: Case 4.2 report

CASE 5. Change the contents of one file to another and check the hash with the original:

The contents of one file are sometimes moved to another file instead of the entire file; the results of comparing the hash value with the originals are shown in Figure 8 below.

Original hash value:
79EAB87F0D3A3B45954779A72F79AE63
New hash value:
593026F4FB3437E7D47FC4178F22EC92

Name: hash value test drive
Description: Physical Disk, 156301488 Sectors, 74.5GB
Logical Size:
Physical Size: 512
Starting Extent: 1S0
File Extents: 1
Physical Location: 0
Physical Sector: 0
Evidence File: hash value test drive
Full Path: hash value test drive\hash value test drive
File Extents

Start Sector	Sectors	Start Cluster	Clusters
	1		

Device
Evidence Number: hash value test drive
File Path: D:\research\case 5\hash value test drive.E01
Examiner Name: kk
Actual Date: 01/30/12 04:09:08PM
Target Date: 01/30/12 04:09:08PM
Total Size: 80,026,361,856 bytes (74.5GB)
Total Sectors: 156,301,488
File Integrity: Completely Verified, 0 Errors
EnCase Version: 6.2
System Version: Windows XP
Acquisition Hash:593026F4FB3437E7D47FC4178F22EC92
Verify Hash: 593026F4FB3437E7D47FC4178F22EC92
Partitions

Code	Type	Start Sector	Total Sectors	Size
07	NTFS	0	156,296,385	74.5GB

Fig. 8: Case 5 report

CASE 6. Update some contents of the existing file and compare the hash: This is a case where the financial and accounting data are much more valuable than the other contents of the disk. Sometimes the suspect only changed the digital content of the data files. The experiments presented here are intended to verify whether the hash value generated in the forensic investigation tool differs from any previous case, as shown in Figure 9 below.

Original hash value:
79EAB87F0D3A3B45954779A72F79AE63
New hash value:
C6D351D5F05CC6273EBD153FC25B5EB

```
Name:              hash value test drive
Description:       Physical Disk, 156,301,488 Sectors74.5GB
Logical Size:      0
Initialized Size:  0
Physical Size:     512
Starting Extent:   0S0
File Extents:      1
References:        0
Physical Location:                0
Physical Sector:   0
Evidence File:     hash value test drive
File Identifier:   0
Code Page:         0
Full Path:         hash value test drive\hash value test drive
Device
Name:              hash value test drive
Actual Date:       02/02/12 11:56:19AM
Target Date:       02/02/12 11:56:19AM
File Path:         O:\resarch\case 6\hash value test drive.E01
Case Number:       hash value test drive
Evidence Number:                  hash value test drive
Examiner Name:                    kk
Drive Type:        Fixed
File Integrity:    Completely Verified, 0 Errors
Acquisition Hash:c6d351d5f05cc6273ebd153fc25b5e7b
Verify Hash:        c6d351d5f05cc6273ebd153fc25b5e7b
GUID:               556f8c3a7bbeb647a99303a23c9a14f7
EnCase Version:6.2
System Version:Windows XP
Disk Signature:  F3418DFB
Partitions
Code               Type           Start Sector    Total Sectors    Size
07                 NTFS           0               156,296,385      74.5GB
```

Fig. 9: Case 6 report

V. CONCLUSION AND FUTURE WORK

The role of the hash value is demonstrated in this research work using different cases involved in the manipulation of analyzed and verified data. This search is heavily focused on the hash value of the entire digital disk, not on a single file. The purpose of this work is to show that even if a slight change occurs in the digital proof, it is detected in the hash value. Given a different heuristic, it would be interesting to apply this technique to other file systems than Windows in the future and to compare the results.

ACKNOWLEDGMENT

All through this research work, many people have provided me help and support. I would like to acknowledge them with great appreciation for their inspiration and kind support. First and foremost, I would like to express my deep gratitude to ***Dr. V. M. Thakare*** my supervisor, for the inspiration, encouragement and expert guidance that have directed me towards my research goals. His insistence in rigor and precise has also benefited me towards my research goal. He has patiently taught me how to think, an important asset as a researcher. He has always been a driving force for the successful completion of this work.

REFERENCES

[1]. S. Ardisson. Producing a Forensic Image of Your Client's Hard Drive? What You Need to Know, Qubit, 1, pp. 1-2, 2017.

[2]. "Digital signature standard," FIPS PUB 186-2, 2016. Available from: www.csrc.nist.gov/publications/fips/fips186-2/fips186-2-change1.pdf

[3]. G. Palmer. A road map for digital forensic research. Report From the First Digital Forensic Research Workshop (DFRWS), August 2016.

[4]. M. Alazab, S. Venkatraman & P. Watters. Digital forensic techniques for static analysis of NTFS images, Proceedings of ICIT2009, Fourth International Conference on Information Technology, IEEE Xplore, 2009.

[5]. M. Reith, C. Carr, & G. Gunsch. An examination of digital forensic models, International Journal of Digital Evidence, 1, pp.1-12, 2002.

[6]. Nelson, B. Phillips, A. Enfinger, F. Steuart, C. Guide to Computer Forensics and Investigations: Third Edition, Course Technology, 2008.

[7]. Ronald L. Rivest. The MD5 message-digest algorithm, IETF RFC 1321, 1992. Available from www.ietf.org/rfc/rfc1321.txt

[8]. Secure hash standard, FIPS PUB 180-1, 1995. Available from: www.itl.nist.gov/fipspubs/fip180-1.htm

[9]. Guidance Software Inc. Encase forensics. http://www.guidancesoftware.com.

AUTHORS PROFILE

Arvind S. Kapse, Research Scholar, Singhania University,Pacheri Bari, Dist:-Jhunjhunu, Rajasthan, India

Dr. Vilas M. Thakare, Professor and Head in Computer Science, Faculty of Engineeing & Technology, Post Graduate Department of Computer Science, SGBAU, Amravati University, Amravati, Maharashtra

Dr. Avinash S. Kapse, Associate Prof. CSE Dept. Anuradha Engg. College, Chikhli, Dist:- Buldhana, Maharashtra, India.