A Survey On Various Methods To Detect Forgery And Computer Crime In Transaction Database

Pratik Patel, Shailendra Mishra

Abstract: A computer forensic method can be used for detecting the different types of forgeries and computer crime. Forgeries and computer crime are the most major concern of the digital world. Lots of techniques and methods have been used to find a proper solution to these problems. Nowadays, digital forensics are an important topic for research articles. In this paper a general survey has been carried out for different methods used in computer forensics to track the evidences which can be useful for detecting the computer crime and forgery. Forensic tools can be used for making any changes to data or tampering of data. Different rules sets or methods are defined to detect the various errors regarding the changes and the tampering of the data in different windows file system. Digital evidence can also be used to detect forgery or computer crime.

Index Terms: Computer forensics, Digital forensics, Evidence, FAT file system, Forensic tools, NTFS file system, $Log file.

1 INTRODUCTION

In the modern era, providing security of database is the most important part. It is the kind of method that prevents the database from the intentional or accidental threats. Theft and fraud of data, loss of confidentiality, loss of integrity etc are the major areas in which data can be lost. So, securing and detecting of data tampering needs investigation. Forensics is the method or technique used in the investigation of crimes related to digital or computer. Digital forensics is a branch of forensic science encompassing the recovery and investigation of material found in digital devices, often in relation to computer crime [1]. Computer forensic, Network forensic, Mobile forensic and Data analysis forensic are the branches of digital investigation. Hence to detect forgery and crime, related computer devices issues like chat logs, database, calendar, emails, internet browser history, video or audio files, timestamp etc, need evidence and related forensics tool. Current digital forensics[DF] tool produces result successfully but can't analyze the format of data management issues like hiding of data, tampering of data[date & time, chat logs, emails etc]. This can be done by formatting incompatibilities, designing, encryption or lack of training and it will not be created with digital science needs. For that they provide the investigator with access to evidence, but it is not clear whether the evidence used is reliable or not and also no method is provided to identify or analyze that evidence. Basically digital evidence is probative information stored or transmitted in digital form which a party can produce in court case during a trail. Before accepting digital evidence a court will determine if the evidence is relevant, whether it is authentic, if it is hearsay and whether a copy is acceptable or the original is required [2]. For this method, abstraction layer can be used and the layer is analyzed by digital forensic tools and also that digital analysis tools can be used as properties of abstraction layer [3]. DF tools can be used on daily basis to examine and analyze the data. So if we can't improve the efficiency of tools and our research process like computer crime and forgery then our data can be lost in future years [4]. Hence, we need to improve in research process and create or manage the existing forensics tools which help in computer crime and forgery to avoid data management issues. Using file time change tools and computer forensics method for detecting timestamp forgery is presented [5]. The paper is organized as follows: Section 2, gives a review of different related research work in the literature. In Section 3, Result and Analysis of techniques used and Section 4, concludes the paper.

2 RELATED RESEARCH WORK

Lots of research has been carried out to attain better enhanced data. Hence, different methods or forensics tools can be used and also to define rule set to detect forgery and computer crime. In 2002, Eoghan Casey et.al [6], suggested uncertainties in network related evidence that can be compounded by data corruption, loss, tampering, or errors in interpretation and analysis. Methods of estimating and categorizing uncertainty in digital data are introduced. In 2003, Brian Carrier et.al [3], suggested a technique based on digital forensics examination and analysis on tools using abstraction layers. Analysis tools can translate data from one layer of abstraction to another. Tools can also be used to identify data format and evidence. Abstraction layer is used to analyze large amount of data in manageable format. This is a necessary feature to design digital system because all data can be in terms of zero or one format.

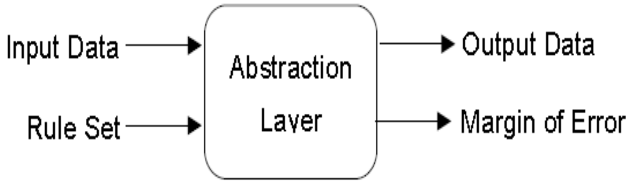


Fig 1: Abstraction Layer Inputs and Outputs [3]

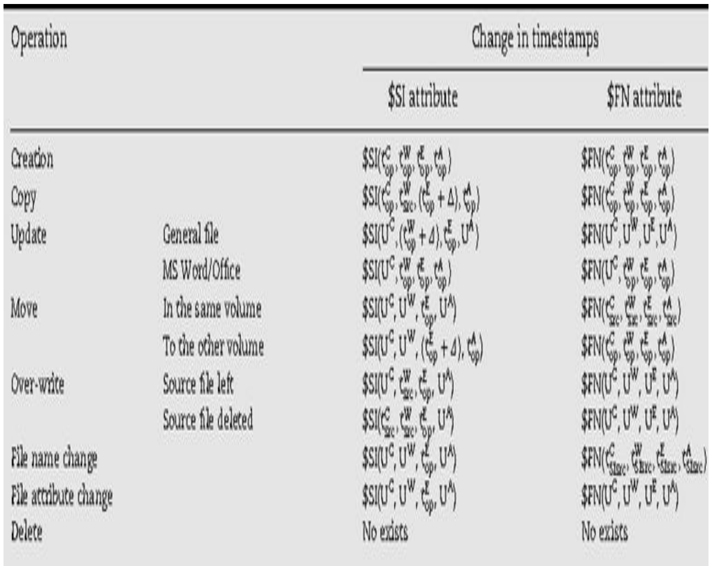
In fig 1, abstraction layer contains inputs, rule set and outputs. Inputs are data and translation rules. The rule set defines how the data should be processed, and output is derived from input data and margin of errors. So according to this technique, the tools analyzed the abstraction layer and provided security at all levels. Finally using abstraction layer, the author identified

- *Pratik Patel is currently pursuing masters degree program in Computer Science & Engineering in Gujarat Technological University, India, E-mail: pratikpatel02411@yahoo.com*
- *Shailendra Mishra is currently working as Assistant Professor in Computer Science & Engineering Department in Parul Institute of Technology, Gujarat, India. E-mail: shailendrabemtech@gmail.com*

where the error was generated and helped to determine the result. In 2010 Simson L. Garfinkel et.al [4], carried out digital forensic research: the next 10 years. It summarized the current forensic research direction and argues to move forward as the community needs for different approaches. Coming digital crisis like growing size of storage devices, requirement and complexity of tools also cost tool development, malware and trojen horse can't detect so needs RAM forensics etc. Today's tools cannot work with computer crime cases and forgery, hence uses "visibility, filter, & report" model for extracting and displaying information. For more accurate research process or higher level abstraction used, alternative analysis model like stream based disk forensics, stochastic analysis, and prioritized analysis. So if we make digital forensics research more efficient then we can create a new abstraction for data representation that will help for future research forensic processing. Many forensic computing examinations are a fundamental part of date and time evidence. For that in 2004 Chris Boyd, Pete Forster et.al [7], referred detail about forensic issues such as time/date evidence in a court and there is no guarantee, so that the forensic software alone will correctly interpret the raw data. In 2013, Gyu-Sang Cho et.al [5], enhanced a computer forensic method for detecting timestamp forgery in NTFS. The idea was that timestamp forgery by using file time change tools leaves evidence in the log records of $logfile. So here $logfile is used to propose a computer forensic model to detect a timestamp forgery in windows NTFS. $logfile contains the record of sequence of operations performed in file. Whenever the system crashes and maintaining consistency at that time using a redo and undo operation information, it can be recovered. $logfile also contains a restart area and logging area for how to start the recovery after the system failed and transaction record. The transaction record like Master file table [MFT] contains $STANDARD_INFORMATION and $FILE_NAME. $STANDARD_INFORMATION contains basic metadata for file or directory and four time values i.e. creation time, write(modified) time, MFT entry modified time, and accessed time. $FILE_NAME stores the file's name and parent directory information. It can have multiple file name attribute to support MS-DOS based file names. It contains same timestamp as in $SI but different time values because windows doesn't change time values the same way as in $SI. Now in $logfile all the transaction record saves automatically, like if we perform some operations like copy, update, move, over-write, file name change, file attribute change etc., then according to that operation changes are recorded in $logfile. Both past and present time stamp can be found in log records of $logfile. In the table, changes in timestamp by different file operations are mentioned. To define changes in timestamp operations, some definitions are mentioned below:

1) Operation execution time t_{op}
2) Original timestamp before an operation performed tsrc
3) Delta time Δ
4) U means the time value is unchanged
5) $SI(t^C_{op}, t^W_{op}, t^E_{op}, t^A_{op})$ and $FN(t^C_{src}, t^W_{src}, t^E_{src}, t^A_{src})$
6) "Any" means any arbitrary time can be taken.

Here the author has defined some rules to detect timestamp forgery. On the basis of the rules, it was possible to detect the time stamp forgery.



Operation		Change in timestamps	
		$SI attribute	$FN attribute
Creation		$\$SI(t^C_{op}, t^W_{op}, t^E_{op}, t^A_{op})$	$\$FN(t^C_{op}, t^W_{op}, t^E_{op}, t^A_{op})$
Copy		$\$SI(t^C_{op}, t^W_{src}, (t^E_{op} + \Delta), t^A_{op})$	$\$FN(t^C_{op}, t^W_{op}, t^E_{op}, t^A_{op})$
Update	General file	$\$SI(U^C, (t^W_{op} + \Delta), t^E_{op}, U^A)$	$\$FN(U^C, U^W, U^E, U^A)$
	MS Word/Office	$\$SI(U^C, t^W_{op}, t^E_{op}, t^A_{op})$	$\$FN(U^C, t^W_{op}, t^E_{op}, t^A_{op})$
Move	In the same volume	$\$SI(U^C, U^W, t^E_{op}, U^A)$	$\$FN(t^C_{src}, t^W_{src}, t^E_{src}, t^A_{src})$
	To the other volume	$\$SI(U^C, U^W, (t^E_{op} + \Delta), t^A_{op})$	$\$FN(t^C_{op}, t^W_{op}, t^E_{op}, t^A_{op})$
Over-write	Source file left	$\$SI(U^C, t^W_{src}, t^E_{op}, U^A)$	$\$FN(U^C, U^W, U^E, U^A)$
	Source file deleted	$\$SI(t^C_{src}, t^W_{src}, t^E_{op}, U^A)$	$\$FN(U^C, U^W, U^E, U^A)$
File name change		$\$SI(U^C, U^W, t^E_{op}, U^A)$	$\$FN(t^C_{SIsrc}, t^W_{SIsrc}, t^E_{SIsrc}, t^A_{SIsrc})$
File attribute change		$\$SI(U^C, U^W, t^E_{op}, U^A)$	$\$FN(U^C, U^W, U^E, U^A)$
Delete		No exists	No exists

Table 1: Changes in timestamp by operation [5]

The investigator does not need to check the log record in $logfile. According to authors knowledge, this is the first research on utilizing a NTFS journaling file i.e. $logfile. New technology file system (NTFS) is more secure than file allocation table (FAT). NTFS is designed to be more stable and reliable when compared to FAT system and it also used transaction logs or master file table during crashes to recover data [8].

3 RESULT AND ANALYSIS

In 2002, Eoghan Casey et.al [6], suggested a method for uncertainity of data and also result that if errors or loss of data than used method to solved that problem. In 2003, Brian Carrier et.al [3], presented an easier approach of abstraction layers, which can be used to analyze large number of data in more manageable format. File system abstraction layer is example of lossless layer so that it has zero margin of abstraction error. Here in File allocation table [FAT] file system, an example on abstraction layer is used to give brief overview of the file system and also to describe the proposed layers of abstraction. FAT32 is basically used because it is simpler than FAT12 and FAT16. The FAT file system has seven layers of abstraction. The result and analysis of file system introduced. A digital forensics analysis tool for FAT file system would provide the investigator with inputs and outputs to each of the seven abstraction layers. In 2010, Simson L. Garfinkel et.al [4], advised that tools are especially important when they are used for activities such as computer crime and forgery. In today's year, the DF tools implement the same conceptual model for finding and displaying information in terms of "visibilities filter and report" model. In this model following steps were taken:

- All the data collected and analyzed were made visible in a user interface.
- Individual user data object were explored.
- Users were able to apply filters to short the display.
- Users were able to perform activities like searches for keywords, names, phone number and others specific contents.

- Finally, users were able to generate report and also able to follow to find the report.

This model does not follow the parallel processing because of which the delays have increased with each passing years. Hence nowadays only five data abstractions are used widely like disk images, packet capture files, files, files signature and extracted Named Entities. Efforts to develop new format and abstraction have failed, so DF community needs to develop abstraction and thinking about file metadata, file system metadata like timestamp, application profiles, user profiles etc. So DF research is more efficient through a creation of new abstraction. In 2013, Gyu-Sang Cho et.al [5], presented a computer forensic method to detect timestamp forgery in NTFS. To obtain it, the author defined seven rules and by using those rules, timestamp forgery was detected. Here author presented results and showed the analysis on rule 1 i.e. Future Time. According to future time rule, timestamp forgery can be generated if $t^{E}_{op} << t_{ANY}$. Here by using the tools named "changes Files Date and Time tool" changes are made and also arbitrary time (t_{ANY}) is obtained which is always greater than entry modification time (t^{E}_{op}).

Suspected file : MemoNote.txt
Size : 785 bytes
Timestamps : 2011.6.2. 01:23:59 (Creation)
(GMT) : 2011.6.2. 01:45:02 (Write/Modified)
2011.5.27. 01:25:29 (MFT Entry Modified)
2011.6.2. 01:45:02 (Access)

The investigator proves that there is inconsistency with the timestamp by using "rule 1" and the investigator does not need to check log record in $logfile. Author proposed three cases of txt file for notepad, docx for MS Word 2010 and pdf for Adobe Acrobat using file time change tools.

4 CONCLUSION AND FUTURE WORK

In this paper, we have surveyed the various methods to detect computer crime and forgery. This existing method is considered as topic of research to introduce how forgery and computer crime can be detected. A better approach of abstraction layer was discussed where the data was analyzed from a larger database. We have also seen that forgeries were efficiently detected on a NTFS file system. In future, the forgery detection method can be deployed on any different file system. Here we can also add new and better approaches which can help to detect the computer crime and forgery in an efficient manner. A variety of tools might also be used to obtain efficient results.

REFERENCES

[1] http://en.wikipedia.org/wiki/Digital_forensics

[2] http://en.wikipedia.org/wiki/Digital_evidence

[3] Brian Carrier, "Defining Digital Forensic Examination and Analysis Tools Using Abstraction Layers" International Journal of Digital Evidence (2003) pp.1-12.

[4] Simson L. Garfinkel, "Digital forensics research: The next 10 years", Digital Investigation 7(2010) S64-s73.

[5] Gyu-Sang Cho, "A computer forensic method for detecting timestamp forgery in NTFS", computer & security 34(2013) 36-46

[6] Eoghan Casey. Error, Uncertainty, and Loss in Digital Eidence. International Journal of Digital Evidence, 1(2), Summer 2002.

[7] Boyd C, Forster P. Time and date issues in forensic computing a case study. Digital Investigation Feb.2004;1(1):18-23.

[8] http://www.adrc.com/ckr/ntfs_fat.html