Research Journal of Applied Sciences 11 (6): 400-405, 2016
ISSN: 1815-932X

Mobile Forensics Triage for Damaged Phones Using M_Triage

Yusoof Mohammed Hasheem, Kamaruddi Malik Mohamad Ahmad
Nur Elmi and Rashid Naseem
Department of Information Security, Faculty of Computer Science and Information Technology,
University Tun Hussein Onn Malaysia, 86400 Parit Raja, Johor, Malaysia

Abstract: The increasing amount of information from damage mobile phones deemed useful for investigative lead to high demand for the timely identification, scrutiny and clarification of digital facts is becoming more crucial. In countless investigations critical data is needed as at the scene or inside a short time of period which is measured in hours as challenged to days. Most of mobile forensic triage tools were able to handle damaged mobile phones due to the significant difficulties in the field of mobile forensics. Both academic researchers around the world and the manufacturers of mobile forensics software are trying to see they came up with a solution to deal with issues regading damaged mobile. And it is evident that from 2006-2015 many mobile forensics triage framework have been proposed and developed to handle different kind of mobile phones but yet none of the tool consider handling damaged mobile phones. In this research, M_Triage framework is introduced to address the issue regarding triage examination on damaged Android-based mobile phones, with the ability to backup evidence in case of any error or loss of data.

Key words: Clarification, mobile forensic, M_Triage framework, regarding triage, android-based mobile phones, evidence

INTRODUCTION

Physical removal is one of the techniques for retrieving evidence data from damaged or passworded mobile phones that have been used for crime activities; the techniques assist crimes investigations especially in solving a kidnapping, illegal arms deal and fraud cases where manual and logical methods fail to extract evidence on damaged mobile phones. However, mobile forensics is the most difficult and challenging in the field of digital forensics (Owen and Thomas, 2011). Nevertheless, the major difficulties in the field of mobile forensic are the general lack of hardware, software and interface standardization within the industry (Hong, 2013; Lessard *et al*., 2011). This fact makes forensic processing a massive job, especially for integrated research.

Furthermore, mobile phone forensics is challenging field due to quick changes in engineering technology (Owen and Thomas, 2011). Various examples of mobile phones exist in the universe today. Manufacturers lack standardized methods of storing information. Most of the mobile phones use closed operating systems and has proprietary interfaces. To overcome this challenge, there is always a need for the development of new forensics tools and techniques (Barmpatsalou *et al*., 2013). In this research, M_Triage framework is introduced to address the issue regarding triage examination on damaged android-based mobile phones. The framework consist of nine steps namely; Generate data input, generate SHA-256 Hash value, imaging extracted data, NAND dual segmentation, pre-processing, block hash filtering based on irrelevant data, inference, image/multimedia carving, post-processing.

M_Triage is an extension of Decode and Lifter to handle data retrieval challenges regarding damaged android OS mobile phones. The tool is capable of generating SHA-256 Hash value and performing triage examination on damaged Android-based mobile phones. The primary aspect of M_Triage development is to acquire physical memory image from damaged mobile phones that could not be done by Decode and Lifter. The tool can manage bad blocks, analyze and obtain evidence quickly and accurately (i.e., valid address-book, call logs, SMS, images and videos, etc.) on android OS mobile phones.

M_Triage analysis is performed prior a full examination. M_Triage aims to obtain high-quality results, even for phones that have not been previously encountered by the tool. On the other hand, M_Triage can reduce the high number of false positive results generated by Decode using one of its components called block Hash.

Corresponding Author: Yusoof Mohammed Hasheem, Department of Information Security,
Faculty of Computer Science and Information Technology, University Tun Hussein Onn Malaysia,
86400 Parit Raja, Johor, Malaysia

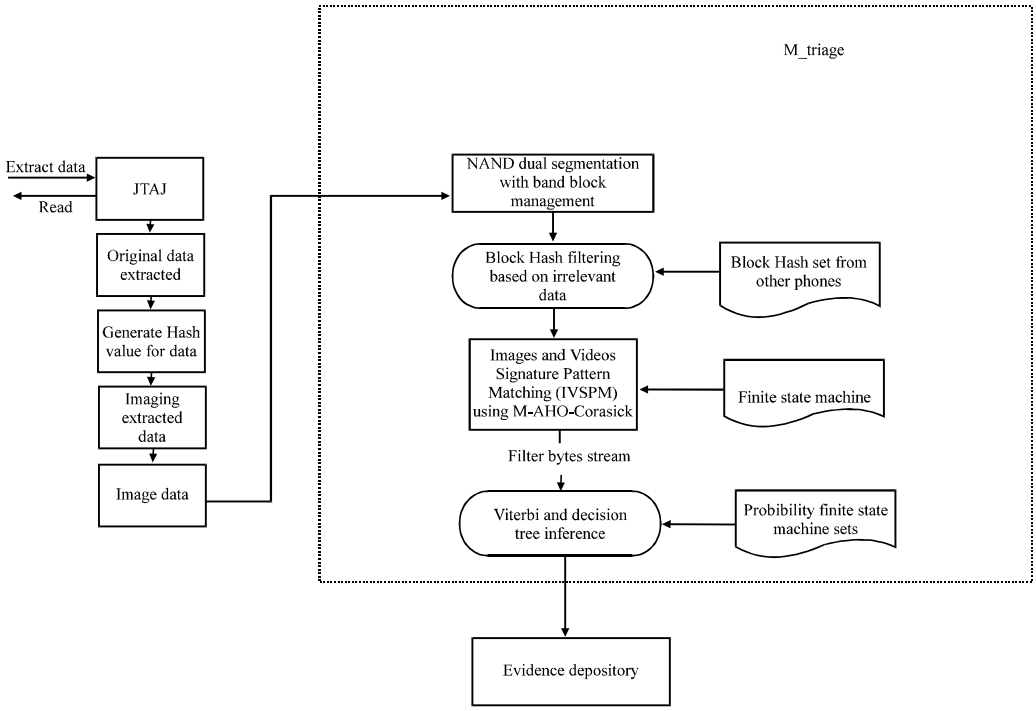

Fig. 1: Proposed framework of M_Triage (extended from Decode) Walls *et al*. (2011)

Filtering based on irrelevant data. Also, M_Triage leverages on the Windows OS by using one of its folder known as "AppData" to store Backup file and the generated SHA-256 Hash value for integrity checking. Hash and file backup capability makes the tool more reliable because the investigators can share the Hash sets of phones and the data found within each phone without worrying about data loss or data modified before and after triage examination

Proposed framework: The framework of M_Triage which is one of the contributions if this study is shown in Fig. 1 The framework is extended from Decode with the consideration of developing damage mobile forensics triage tool. There are two main phases in M_Triage application; the first step solves the challenges of acquiring the physical memory image from damage mobile phones using Joint Test Action Group (JTAG). The hardware is a commercial device integrated into M_Triage application with the intention to provide access to a damage phone memory. According to Klaver (2010) accurate physical acquisition is recognized as any means of physically removing memory from the mobile phone, via hardware techniques like chip off or JTAG to extract information from the device or use an (adapted) boot loader to reach low-level access to the device. These kinds of methods remain not only technically challenging and involve partial to full disassembly of the device but they require significant post-extraction analysis to reassemble the file organization.

Generate input data: To begin any examination on damage mobile phone using M_Triage, there are some few steps need to be considered. First disassemble the phone. Second locate the JTAG test access port in other to initiate a successful connection. The JTAG pin comprises of, Test Data In (TDI) the pin will transmitting serial data from RIFF Box to damage the mobile phone. Test Data Out (TDO) the pin are sending serial data from damaged mobile phone to RIFF Box. Test Clock (TCK), the pin detect clock testing. It also synchronizes the internal state of damage mobile phone operation. Test Mode Selection (TMS), the pin controls the TAP controller state transitions. After a successful connection is established between the RIFF box and the damaged phone the final step is to extract dump file from damaged mobile phone.

Generate SHA-256 hash value: One of the principal components in M_Triage application is the component that creates the SHA-256 hash value, the SHA-256 is an iterated hash function that processes 512-bit input message blocks and produces a 256-bit hash value (Lamberger and Mendel). The component generates the hash value of extracted dump file in other to maintain the

data integrity. And make a copy of the extracted dump file for future use, to avoid error loss of data on the original dump file.

NAND dual segmentation: NAND dual segmentation is one of the key engines for speeding up the processing time during triage extraction in M_Triage application. During the extraction process, the dataset is read to know the actual size of the dump file, if the size is greater than 49 MB, then the algorithm will divide the memory into two blocks to reduce the size of the dump file. After the NAND dual segmentation is completed, the next step is to lavage the Microsoft. NET task-based parallel programming (Luo *et al*., 2013) which is used to achieve efficiency during triage extraction. The concept of task is introduced by The Task Parallel Library (TPL) build in Microsoft visual studio 2013. Running these tasks in parallel is known as task parallelism. A task is an independent unit of work which runs a program. The TPL utilized within the threads to execute these tasks in parallel.

Bad block management: After the NAND dual segmentation process is finished, then bad block management function will be called to handle the so-called bad block management module were the algorithm read for bad-block mark from the divided blocks. Bad block are identified and if bad-block mark found then add a block to bad-block table also if no bad-block mark found then the algorithm cheek is that the last block read from the actual size of the image file if it's correct and then finish creating a bad-block table. And, if no then go to the next block.

Bad block definition:

- Let B = {B1, B2} be the start pattern of block where
- B1 = FFh the blocks that are good
- While B2 = 00h which is the pattern for bad block, C is the bad block marker
- C = {C0,……,Ck} K is the number of total blocks

Hence,

- C0 = {B1, K} a block that is good
- C1 = {B1, B2, K, K} a group of blocks containing one bad block

Block Hash filtering based on irrelevant data: Block Hash filtering based on irrelevant data takes a stream of n bytes and produces a series of overlapping blocks of length b. The commencement of each block differs by d≤b bytes. Then it will build the signature database of all the target file of interest. Next stage is to search for any data that is not defined in the database by comparing the irrelevant data with the specified signature in the database and detect or mark their location when it's done then filter the irrelevant data including the mobile OS. The next stage is to create a hash library and then generate the SHA-1 value of each block if completed then captured and compare to the hash library. Any collision of the hash of a block with a block on another phone (or the same phone) is filtered out.

Images and videos signature pattern matching using M-Aho-Corasick: One of the main components that efficiently search for pictures and videos using multi-pattern signature marching is an M-Aho-Corasick algorithm. The algorithm is adapted and modified from the original algorithm known as Aho-Corasick where the failure links function is removed and replaced with a signature database that contain all the pattern and file structure that is relevant to investigator stored in it. For instance, a JPEG image can be searched in few minutes when the header-footer and the structure of the picture file ar e examined in the group. Also multimedia file such as 3GP and MP4 file can be searched in a few minutes when a File type box (Ftyp), Movie box (Moov) and Mdat (Media Data Box) are search in the group.

During the pre-processing in M_Triage application, a file pattern like JPEG, 3GP and MP4 are searching by building their signature database, then followed by creating the block tree and adding pattern ID's into the tree using automation. The signature is searching based on Finite State Machines (FSM) and If the pattern is searching within the dump file, the pattern will be compared with the once in the signature database for identifying the file of interest. If the signature is matched, then "go to" function will be called to mark the address of the valid signature and move to the next block. If the signature of interest is not found, then skip the block to the next block.

Inference: After block hash filtering based on irrelevant data and images/videos signature pattern matching has been performed, what remains is a reduced ad hoc data source about which only minimal information is usable. M_Triage obtains the maximum likelihood parse of the input stream, producing a ranked explanation of the information on the phone in the form of fields and records. More concretely, the output of M_Triage is a set of the call log, address book records, SMS, Images and videos. Each record consists of fields representing phone numbers, timestamps, strings and other structures extracted from the raw stream (Walls *et al*., 2011).

Post-processing: The inference process takes the set of records recovered by M-Aho-Corasick and Viterbi then

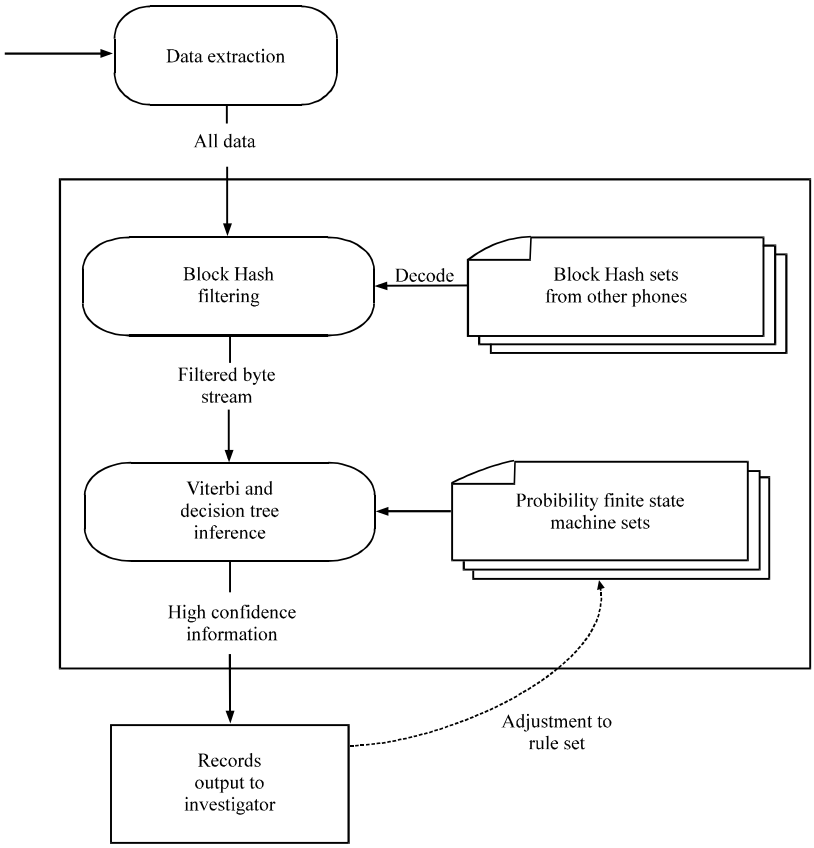

Fig. 2: Decode framework (Walls *et al*., 2011)

passes them through a decision tree classifier to remove again potential false positives. This step is referred to as post-processing. Decision tree classifier is applied because it's able to bring into account features that can be inefficient to encode in M-Aho-Corasick and Viterbi. For instance, the classifier considers whether a record was found in isolation in the byte stream or in proximity to other discs. In the prior case, the record is more likely to be a false positive.

Weka J48 Decision Tree is used is an open source implementation of a well-known classifier. In general, a decision tree can be used to determine whether or not an input is an instance of the object class for which it is prepared. The classifier is developed using a set of feature tuples representing both positive and negative examples (Walls *et al*., 2011).

In M_Triage application, the decision tree chooses whether a specified record and output from M-Aho-Corasick and Viterbi phase is valid. M_Triage application selected a set of features common to SMS, phone call log and phone address book records; number of existences from the average date; existence of phone numbers with the same area code; number of different patterns seen for the same number; number of references in the string; number of times the book appears in memory; distance to closest neighbor record (Walls *et al*., 2011). While the carefully chosen features common to images and videos are any component of graphic representation is resolute inside the memory block. M_Triage post-processing does not prevent the detective; it is a filter intended to make the detective's work easier. M_Triage can make both the pre and post-processing results available, ensuring that the investigator has as much useful information as possible.

RELATED WORK

Decode: In the year 2011, a novel mobile forensics and triage tool known as "Dec0de" was produced by Walls *et al.* (2011). The tool focuses on a data-driven approach to telephone triage. Their end is to allow detectives to remove evidence swiftly in 20 min from a phone, irrespective of whether that exact phone model has been come across previously. The authors highlighted that their assessment emphases on old phones, for instance mobile Phones with a lesser amount of ability than smartphones. The 20 min processing time claimed by decode was archived on 48MB mobile phones. Nevertheless, the researchers

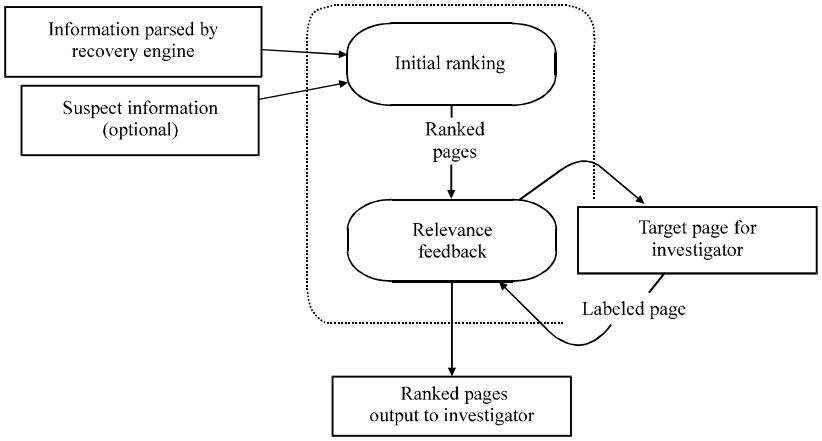

Fig. 3: Lifter Framework Walls 2014.

developed an algorithm called "Block Hash Filtering" a module within their too to reduce the amount of file to be processed during triage examination. Figure 2 shows more about the decode framework.

Lifter: In the year 2014 Decode is extended to support smartphones they came out with another version of a triage tool called "LIFTER." The researchers recommends the use of importance feedback to solve this problem known as a false positive result. A minor quantity of detective feedback can powerfully and correctly rank in order of significance, the results of a forensic triage tool. And again the toll lifter failed to concede filtering irrelevant data at the earlier stage (Walls and Levine, 2014) (Fig. 3).

CFFTPM: In the year 2006 a Cyber Forensic Field Triage Process Model propose an onsite or field way for granting the identification, analysis and clarification of digital facts in a short period construction, lacking the requirement of possessing to seize the system (s)/media back to the lab for an in-depth examination or obtaining a complete forensic image(s). The counseled model adheres to usually grasped forensic principles and does not negate the ability that after the new field triage is finished, the system(s)/storage mass media be transported back to a lab environment for an extra careful examination and analysis (Rogers *et al*., 2006).

However, the Cyber Forensic Field Triage Process Model (CFFTPM) is a formalization of real globe investigative ways that have distilled into a proper process model and the six main stages of the CFFTPM (planning, triage, usage/user profiles, chronology/time line, email and IM and case-specific evidence) are vital in such varied cases as financial fraud, individuality theft, cyber stalking and murder. Figure 4 shows more about the framework.

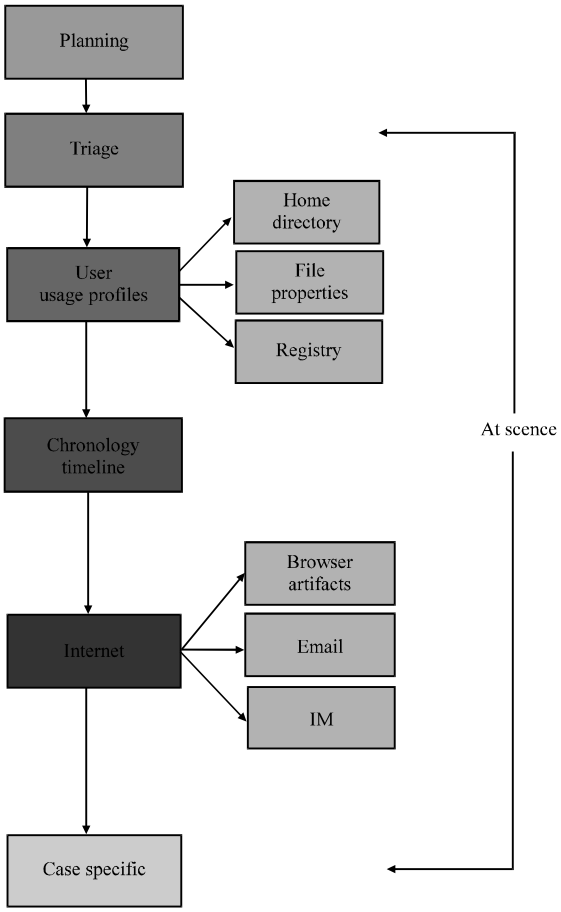

Fig. 4: CFFTPM framework (Rogers, 2006)

MIAT: MIAT is an open source software for the Mobile Forensics, designed to capture data with read-only access, directly from the internal memory of the device without the use of external hardware. This results in

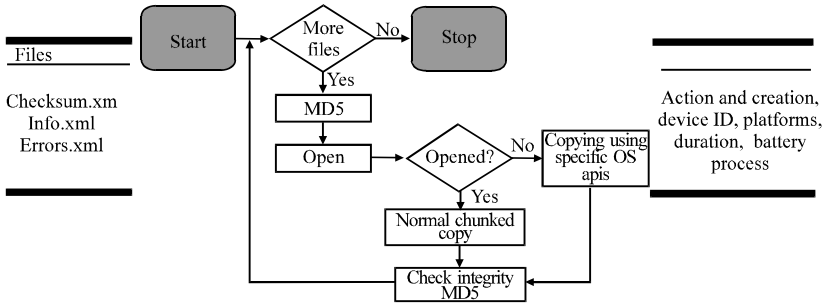

Fig. 5. MIAT framework MIAT

enormous advantages for Forensic workers who do not have necessity to equip themselves with cables and devices each type and size , depending on your phone to be analyzed but hold a sufficient Micro SD forensics, on which' the content and source of MIAT and which are then stores the information acquired ("MIAT (Mobile Internal Acquisition Tool) Gianluigi Me Summary", Production). MIAT calculate an MD5 hash first and after the copy of each file, to highlight possible corruption. Moreover, unlike other software for mobile forensics, MIAT does not require a source people intermediary between the forensic workstation and mobile device to be tested; this can imply a degree of assurance of the integrity and verifiability of information higher. The MIAT was developed only for Symbian and Windows operating systems (Fig. 5).

CONCLUSION

Based on the literature findings, it is evident that from 2006-2015 many mobile forensics triage framework have been proposed and developed as a mobile forensics triage tool, but none of the tools were able to handle damaged mobile phones due to the significant difficulties in the field of mobile forensics. In this research, M_Triage framework is introduced to address the issue regarding triage examination on damaged Android-based mobile phones with the ability to backup evidence in case of any error or loss of data.

REFERENCES

Barmpatsalou, K., D. Damopoulos, G. Kambourakis and V. Katos, 2013. A critical review of 7 years of mobile device forensics. Digital Invest., 10: 323-349.

Hong, I., H. Yu, S. Lee and K. Lee, 2013. A new triage model conforming to the needs of selective search and seizure of electronic evidence. Digital Invest., 10: 175-192.

Klaver, C., 2010. Windows mobile advanced forensics. Digital Invest., 6: 147-167.

Lamberger, M. and F. Mendel, 2011. Higher-order differential attack on reduced SHA-256. Cryptology ePrint Archive: Report 2011/037, January 20, 2011. https://eprint.iacr.org/2011/037.

Lessard, J., G. Kessler and G.C. Kessler, 2011. Android forensics: Simplifying cell phone examinations. Android Forensics: Simplifying Cell Phone Examinat., 4: 1-12.

Luo, Z., Q. Zheng and X. Hei, 2013. Parallel programming based on microsoft. Net Tpl, pp: 505-507.

Owen, P. and P. Thomas, 2011. An analysis of digital forensic examinations: Mobile devices versus hard disk drives utilising ACPO and NIST guidelines. Digital Invest., 8: 135-140.

Rogers, M.K., J. Goldman, R. Mislan, T. Wedge and S. Debrota, 2006. Computer forensics field triage process model. Proceedings of the Conference on Digital Forensics, Security and Law, January 2006, Association of Digital Forensics, pp: 27-40.

Varma, S., R.J. Walls, B. Lynn and B.N. Levine, 2014. Efficient smart phone forensics based on relevance feedback. Proceedings of the 4th ACM Workshop on Security and Privacy in Smartphones and Mobile Devices, November 3-7, 2014, Scottsdale, AZ, USA., pp: 81-91.

Walls, R.J., 2014. Inference-based forensics for extracting information from diverse sources. Ph.D. Thesis, University of Massachusetts Amherst, Amherst, MA.

Walls, R.J., E.G. Learned-Miller and B.N. Levine, 2011. Forensic Triage for mobile phones with DEC0DE. Proceedings of the USENIX Security Symposium, August 12-14, 2011, Washington, DC. -