IJoFCS (2013) 1, 35-44

e-forensic PRESS

The International Journal of FORENSIC COMPUTER SCIENCE

www.IJoFCS.org

DOI: 10.5769/J201301005 or http://dx.doi.org/10.5769/J201301005

Triage in Live Digital Forensic Analysis

Muhammad Shamraiz Bashir(1) and M. N. A. Khan(2)

(1) Department of Computing, Shaheed Zulfikar Ali Bhutto Institute of Science And Technology (SZABIST), Islamabad, Pakistan. Email: chaudharyshamraiz@gmail.com

(2) Department of Computing, Shaheed Zulfikar Ali Bhutto Institute of Science And Technology (SZABIST), Islamabad, Pakistan. Email: mnak2010@gmail.com

Abstract - Due to frequent use of Internet and with technological advancements, cyber and malware attacks over the digital devices have increased manifold. Activities performed electronically can be investigated by means of digital forensic analysis methodologies. Live digital forensic tools are used for digital evidence collection and investigations of malicious activities that occurred on a standalone system or networks. Since compromised system remains active while using these tools, some serious issues relating to malicious functionalities and policy violations could lead to serious damages like data theft or data loss. In this paper, we present a critical review of the triage in live forensic. This paper discusses several techniques being used for performing live forensic analysis and critically evaluate their efficacy in terms of their applicability and reliability. A brief anecdote about the pros and cons of these techniques are also discussed. We present the findings of our study in the critical section.

Key words: Digital forensic, digital evidence, forensic analysis, Triage, live analysis.

I. Introduction

Digital communication techniques such as email, SMS, blogs etc. have progressed rapidly during the last two decades. Email is one of the most commonly used communication technique. Emails can be sent/received on laptops, computers and mobile phones as well as on some other digital devices such as PDAs and Notebooks. Despite its benefits, it can also lead to malware or cyber attacks in the digital society through different means. Generally, such attacks occur over the Internet and result in serious damages like misleading functionality or data theft. To prevent or suppress these kinds of attacks, pertinent procedures must be defined to identify such threats and respond to them quickly and appropriately.

Digital forensics, also known as forensic computing, is a controlled and systematic investigation that entails collecting, analyzing and validating digital evidence. Data residing on a computer or an IT device is gathered in the first step. The gathered data might be in the shape of messages, logs or email etc. Then log analysis indentifies part of the system infected by the

Paper submitted on June 4th, 2013.

malware or cyber attack. Afterwards, bit-by-bit memory content analysis is done by means of hardware and software methods. A hardware system identifies the infected drive and loads it into the memory, which usually alters content and memory state. Subsequently, the memory analyzation can be done using software methods.

Static or traditional analysis method is used for analyzing evidence from the targeted computer after turning it off which obliterates many running processes including network ports. Resultantly the opened connections become unavailable and memory contents are lost; therefore, a meticulous analysis becomes less effective. In addition, shutting down the systems can be unaffordable for some organization depending upon the nature of their business.

In the modern era of computing, live forensics is complementary to static analysis. Live forensics allows recovering and analyzing memory content, processes and data without shutting down the system. Live digital forensics plays a vital role during system examinations due to the potential availability of the digital evidence in the volatile memory such as running processes, network connections, opened ports and encryption keys etc.

Live forensic analysis primarily targets the volatile data which can only be collected from a running system; hence, the term "live" is coined for such type of examinations as otherwise such information cannot be extracted from a "dead" system whose power cord is pulled out. Performing live forensics has become necessary in the modern era because to hack data from a target machine, the malware authors now write Trojans that do not depart any footprint on hard disk. For example, Witty, SQL Slammer and Code Red worm are such type of malware whose existence can only be detected by analyzing the physical memory. Aljadid et al [16] highlight the significance and limitations of live response with respect to memory image analysis and report that in contrast to the former technique, the latter technique help avoid risk of trailing volatile evidence such as suspended processes. That is why memory image analysis due to the very specific reasons highlighted above has become an essential part of the live incident handling.

Live forensics, sometime referred as live incident response, is a methodology to extract memory, system processes and network related information – even the terminated and cached processes. However, live forensics is simply not a pure forensic response as it does not reveal underlying operating state of the system. Despite this, it helps understand the impacts of the running system and network processes on the overall system's state. It would be more appropriate to say that live forensics can be considered as the first step towards an incident response scenario.

Professionals responsible for incident of computer security and digital forensic analysis need to continually update their procedures, skills and forensic tools due to ever changing technology. Examiners capture an image of the memory content to analyze it by using various tools.WFT (Windows Forensic Toolchest) is one such a tool to automate live response on a Windows system. It collects information from the Windows system for system examination.When a forensic tool is invoked on a running system, it causes a minor change in original memory content. When a forensic tool is run from any removable media, causes alteration in volatile data. Likewise, remote forensic tools require network connections, command for execution in memory and make other alternations on the affected machine.

In this paper we provide an insight into the systematic set of procedures required to perform live forensic analysis of a running computer system. The triage is based on the different techniques and methodologies proposed in the contemporary literature.This paper is organized into five sections. After describing an introduction to live forensic analysis in the first section,

we discuss the related wok in section II. The next section describes merits and demerits of the various live forensics and memory imaging analysis techniques. Section IV highlights the grey areas and key challenges that are faced while performing live forensic analysis. This section also point outs the perspective dimensions to this research. Finally, we conclude in the last section

2. LITTRETURE REVIEW

The purpose of this literature review is to highlight the significance of live digital forensic analysis. In this section we analyze and identify the strength and limitation of live digital forensic analysis using various methods.

Decryption keys play a vital role in forensic analysis as they can reveal the true nature of the action performed by certain processes particularly the malware. Forensic analysis of a disk cannot be done when a system is switched off because the decrypted content is no longer available. Bolagh and Pondelik [1] proposed a technique to recover the decryption keys from the dump of the live image of a volatile memory. Proposed approach works on windows and Linux with TrueCrypt – a free open source tool that performs on-the-fly disk encryption. The authors also suggested a method to decrease the size of dump image considerably, especially in case when TrueCrypt is used for encryption, the size can be limited to 1-2 MB only. However, the proposed technique bears a limitation that the image should be present locally for forensic analysis. In addition, decryption keys are located through content search, and if certain data degradation happens in a disk then it becomes impossible to extract keys.

Advances in data encryption technologies have made the job of cyber investigators really tough. Dija et al. [2] proposed a technique to unseal encrypted drives. The proposed solution can merely decrypt only those sealed drives which are encrypted through BitLocker using its USB-only mode feature. The proposed solution is primarily based on discovering ".BEK" or 48-digit password file during live forensic analysis or image dump to recover BitLocker drive. For this purpose, brute force attack can also be used to recover the drive whether on a live forensic analysis or physical memory dump file.

Digital evidence can also be extracted from the data structure residing in memory by using different tools. Chan et al. [3] proposed a tool named as 'Cafegrid' which can be used for deep analysis and recovery of data structure of programs from memory in Widow and Linux. This tool also build a map of object systematically and track the use of memory structure during program execution. Summary statistics produced by this tool contain details of the memory accesses and variations made on the structure of data by intercepting the allocation and deallocation requests of data structure during program execution. This all is done by this tool after monitoring the running program and tracing the allocation of dynamic memory. This tool can help forensic analysts in evidentiary analysis process as the data structure information can be equally used for offline or live analysis in digital forensics.

Balaz and Hlinka [4] described different steps to acquire memory contents for forensic analysis Linux system which is compromised. For ease to professionals responsible for responding to the security incidents, the authors have also suggested the set of different tools required for each step of the investigative process for searching and analyzing the information acquired from the victim system. Linux command line tool 'grep' can also be used to analyze and parse data from the physical memory so that data gathered for analysis is more targeted, lesser in size and relevant. Sifting of the pertinent data assists forensic analysts in identification and preparation of evidentiary data in live and static forensic analysis.

Professionally sound and highly focused work is required for the collection of digital evidence.

XOVAL [5] which is an extension of OVAL (Open Vulnerability Assessment Language) and is an XML language, used to collect the digital data automatically. XOVAL framework keeps track of the specified data collected from target system and presents in a declarative way. Definition and identification of forensic primitives, which can be reused within multiple procedures, is a key feature of XOVAL.

Li et al. [6] outlined a systematic procedure for reconstruction of malicious events to quickly identify the behavior and deduce the functionality of malware or suspicious code running over Android mobile operating system. Identification of malicious program on Android platform can be done by using Logcat which is a built-in API of Androids. Obfuscation, strings encryption and environmental configuration of the program code can be analyzed through decompilation and deobfuscation to help analyst to quickly locate the malicious events. Since code encryption in Android is always done through Data Encryption Standard (DES) algorithm; therefore the same procedure can also be done by PCAP (packet capture) analysis followed by strings decryption by rebuilding and decrypting the code with DES. However, these malicious events are reconstructed for Android-based live mobile malware forensic analysis. Nevertheless, to perform such type of malware analysis, the Android mobile users should be well educated and should have sufficient resources to perform such procedures for identification and analysis of malware forensics on their handsets.

In mobile forensics, examination of procedures for evidence identification, collection, and its documentation is deliberated. Marturana et al. [7] introduced the concept of triaging and adaptation of self-knowledge for identification of device, acquisition of data, and reporting the device-specific investigations after analysis. Triaging is normally done after the data acquisition, but before the detailed analysis. Triaging in mobile forensic consists of three stages, Firstly, the evidence is collected from victim's mobile. Secondly, the data related to forensic analysis is normalized and noise is removed from it. Third and the final step is called data classification and triaging. In this step, evidence classification is done by ordering and predicting similar data byusing knowledge management classification algorithms so that amount of data could be reduced for better performance in mobile forensic analysis. Such an approach limits the interest area and reduces the number of evidence by selecting the relevant and necessary evidence associated with each other for forensic analysis.

Yang and Yen [8] emphasize that live and dead forensic analysis can be carried out by saving the necessary scripts and different tools like Autopsy, FDumper, Scalpel, Fundl etc on a USB or DVD. Such an approach can help in performing live analysis of a running compromised system by plugging in the DVD/USB into the system. The script/tools stored on the DVD/USB when launched will collect the volatile information such as opened ports, user login history and active services etc. from the memory of victim system and stores it on the USB. Similarly, the static analysis can be done by using AIR (Automated Image and Restore) software. The information thus collected can be analyzed by using Scalpel and Fundl software stored on a DVD/USB. Fore forensic perspective, log on/log off, date and time, kernel level information and recently executed commands including processes and network status are of much importance for a forensic analyst in making appropriate decision about the significance of the forensic evidence.

HDFS (Hadoop Distributed File System) [9] was used to forensically analyze large amount of data by performing indexed searching on client-server architecture. In HDFS, the servers contain a master and a slave system, while client contains the web applications. Firstly, the forensic data is stored on a NAS (Network Access Storage), and then it is analyzed by ETL (Extract ,Transform and Load). In this process, data is extracted from NAS is transformed for the operational activities and then it is loaded into Hbase table in which multiple columns belong

to a particular column family. In this approach, the encrypted data is decrypted into plain text, and data filtration and searching through index patterns are applied to reduce the amount of data for analysis. The composition of Hbase database not only improves the data sharing rate, but also enhances ease for the analysts to perform digital forensic investigations.

LECT (Linux Evidence Collection Tool) [10] is capable to perform live forensic analysis on Linux systems. It collects forensic evidence in console mode which is stored on USB and generates reports in XML format which includes the relevant information about the investigation like timestamp and hashes. This proposed system automatically indentifies the target system, then collects the evidence automatically and also tools used in

this framework are so consistent that they do not change or makes very minor changes in the memory contents during the evidence collection phase, so that the integrity of evidence is not disturbed.

A framework proposed by Lempereur et al. [11] monitors the system behavior of heterogeneous computing environment at runtime. The main artifacts included in monitoring are hidden processes, files and network connection status within the system and between the host machines. This system also identifies the violence of security policies on standalone system as well as on network. In this system, actors and repositories was introduced to store digital forensic information. Actors facilitate information transformation from one channel to other, this flow of information is always from source to sink and records only events stream that perform successful operations. Afterwards, source and sink channels are combined to define the security policy, which are applied by traversing the graph from head to tail node of channel. Then the monitor agent is executed on each node to transform the review stream into information flow model.

The physical memory of Windows application can reveal the applications run on a system and the nature of the user interaction with the system [12]. For extracting and investigating forensically relevant data, "Niglant32" tool was used to capture the image of memory allocated by the applications. Then text data was segregated from image using "strings". In the next step, pattern matching is performed to categorize the user activities instances and memory allocation to processes. In this process, original user input is used by the pattern matching to match it with partial fragments or with text information that was extracted from memory dump. This will reveal the user input stored in applications i.e., what activities a user carried out through launching a certain application and which memory was allocated to user process as well as the information retrieved is dispersed or in a whole fragment, also it is assured that information is valid for forensic relevant or not. This information could be more useful for forensic investigation.

A cryptographic model is presented by Law et al. [13] to protect data secrecy during digital investigation. In this model, investigators examine the bit stream image instead of examining the complete memory contents on storage media. Then encryption key is generated by the data owner. Indexes are built after scanning the image contents and are encrypted not to reveal the data for investigation. Subsequently, keywords are generated by the investigators to perform searching in digital evidence from images acquired by trap door. Data owner usually provides trap door in response of request from investigator. Then searching is performed over the encrypted index files by means of trap door during digital investigation process by the investigator.

Mrdovic et al. [14] proposed performing live and static digital analysis simultaneously, which enhances the understanding of events and provides ancillary insight into the present state of system for examination. Live digital analysis is done through virtualization while memory dump is used for static analysis. Memory dump of targeted system is taken before system shut down

so that it can be subsequently booted using VMware for performing live forensic analysis. Memory dump contains list of processes along with their startup time. The process list also includes hidden programs like rootkits which are generally hidden in live analysis. While live analysis is done after setting the system in hibernation mode. In this process volatile memory is preserved, and OS is started from disk image without changing image in virtual environment. The live and static combination assist digital investigators to analyze the image of system as well as memory contents while compromised system is running.

To resolve deficiencies in the current digital live forensic methods Wang et al. [15] proposed a physical memory analysis model for live forensic. For live forensic analysis, it is suggested to clearly separate different phases of live forensic analysis such as evidence collection, examination, analysis and report generation. The proposed model underlines some aspects to maintain the credibility of live forensic analysis. Firstly, authenticity tops the list as it pertains to identifying a key question about the data validity i.e., how much evidentiary data is affected by the evidence collection tools. Then integrity is considered to check whether the data gathered is complete or not. The next step is to verify that data is consistent and meet the requirement of forensic analyst besides verifying that analysis procedure is consistent. Then repeatability and applicability is assured. Finally, it is imperative to analyze that the method used for the forensic analysis is fault- tolerant i.e., method should not be interrupted if some evidentiary data is missing or tampered. A meticulous deliberation of the aforesaid aspects will assist and improve the credibility of forensic analyst.

3. CRITICAL EVALUATION

In this section we provide a critical review of different approaches used for live digital forensic analysis.

Ref.	Focused Area	Technique Used	Merits	Limitations/Caveats
[1]	Forensic analysis of encrypted drives	TrueCrypt	Recovers decryption keys from dump of the live image of a volatile memory.	In case data degradation happens on a disk, then it is impossible to extract decryption keys.
[2]	Live forensic analysis of BitLocker encrypted drives.	AES and Brute force attack algorithms. Step by step procedure to analyze BitLocker drive.	It can find ".BEK" file and password file for BitLocker drive to decrypt it by using brute force attack.	This feature pertains to Window 7 or NTFS drives. Other operating systems and FAT do not support bitlock drive encryption feature.
[3]	Data structure analysis for live and static forensic analysis.	Cafegrid	Cafegrid tool deeply analyzes and recovers data structure of a program from memory in Windows and Linux environment.	This tool only works on shared libraries and binaries in which compilation is done with debugging. Cafegrid does not work properly on common compilers which stores pointers in registers rather than the stack.
[4]	Forensic analysis of Linux systems.	Command line tool "Grep" has been used.	Classification and training of evidentiary data in live and static forensic analysis is done by using 'grep' tool. It reduces the amount of data by selecting the relevant evidence from data.	This is only specific to Linux system and requires client server architecture.

Ref.	Focused Area	Technique Used	Merits	Limitations/Caveats
[5]	Automatic collection of forensic data from digital devices.	XOVAL framework	Key feature of this framework is that it collects digital data automatically from a compromised system and presents it in a declarative way by defining data primitives for live forensic analysis.	The proposed technique collects data of only those objects which are defined in the XOVAL framework and ignores system objects like user login information.
[6]	Android mobile malware forensic analysis.	Logcat and PCAP (Packet Capture) analysis.	Identifies malware in mobiles and decrypts code.	This approach requires performing all the steps manually. Therefore, it is time consuming and entails expertise.
[7]	Mobile forensic	Triaging and self-knowledge.	This technique uses self-knowledge and triaging concept for identification, acquisition and reporting of device-specific investigations. It also reduces the amount of concerned data for forensic analysis by classifying the relevant and similar data.	It is only adoptable for cold or dead analysis. This technique also endures classification problems e.g., different input produces same output. It also lacks analysis reports and relevant data which is necessary for forensic analysis.
[8]	Digital forensic using USB/DVD.	Autopsy, FDumper, Scalpel, Fundl and AIR (Automated Image and Restore).	It supports both live and static forensic analyses and automatically collects volatile information such as opened ports, user login history, active services on Window and Linux systems.	It is not a centralized approach as it supports only Chinese language.
[9]	Using forensic as a service	HDFS (Hadoop Distributed File System) and ETL (Extract Transform, and Load)	It is a live analysis technique and supports remote services. It reduces amount of hardware required for forensic analysis. It also supports wired and wireless connections. Index matching is fast and reduces the amount of data for forensic analysis.	Effective ETL process is required for handling large amount of data.
[10]	Digital forensic analysis of Linux systems	LECT (Linux Evidence Collection Tool) framework.	Automatic identification of targeted system. Report generation based on hash values and timestamps. It also maintains integrity of evidence.	It works only for specific Linux versions such as Ubuntu, Debian and Fedora (Red hat).
[11]	Monitoring of policies violation on network.	Live monitoring of system behavior.	Identifies the violation of security policies and also automatically monitors the hidden processes and network connections on both local (standalone) and network systems.	Complexity in monitoring of network nodes and collaboration among them.
[12]	Forensic related data extraction from Windows system.	"Niglant32" and pattern matching tools.	Detects applications used and events performed by user and categorizes/ groups the relevant events.	Gathered information is incoherent and huge in size which is usually less forensic relevant.
[13]	Protection of data for digital investigation.	Cryptographic model.	Gathered data for forensic analysis is in encrypted form, and index file search is performed. Only the data requested by the investigators is decrypted.	Irrelevant data is also encrypted, due to which index file size increases, thus making the process execution slow.

Ref.	Focused Area	Technique Used	Merits	Limitations/Caveats
[14]	Performing simultaneous live and static digital analyses.	VMware and memory dump.	The system allows live and static combination of image of system as well as memory contents, and lists rootkits while system is in running mode.	The proposed approach suggest performing forensic analysis of the captured memory image on a virtual machine. However, advanced malware stop operating or sometime disguise themselves on a virtual environment. There is also a caveat that evidence extracted through performing forensic analysis on a virtual machine might not be acceptable in a court of law.
[15]	Physical memory analysis.	Live digital forensic model.	Identifies the completion and integrity of evidentiary data, and also verifies consistency of the analysis procedure. It enhances scientific creditability of digital evidence.	The proposed system requires employing manual procedure.

3.1 Tools Commonly Used for Digital Forensic Analysis

In this section we list different commonly used for live digital forensic analysis, their platform, description and availability in tabulated way.

Table I. below lists some of thekey forensic analysis tools used for perfuming live forensic analysis.

Table I. Live Forensic Analysis Tools.

Tool Name	Platform	Description	Availability
AIR (Automated Image and Restore)	Windows	A GUI to create live image dump of memory.	Freeware/Commercial
Autopsy	Window/Linux	It is used for disk analysis.	Freeware
PDUMP	Windows	This tool is used to take live memory dump.	Freeware
WFT (Windows Forensic Toolchest)	Windows	Used to automate the incident response and perform live digital analysis.	Freeware
SMART	Linux	Performs live digital forensic analysis.	Commercial
DEFT (Digital Evidence & Forensic Toolkit)	Linux	DEFT is used in live computer forensic systems.	Commercial
BinDiff	Windows	This tool is the extension of IDA pro. It identifies and highlights the changes and compares code of program before and after running.	Commercial
SIFT (SAN Investigation Forensics Toolkit)	Ubuntu	It is multipurpose tool to perform live digital forensic in operating system.	Commercial
IEF (Internet Evidence Finder)	Windows	This tool is used to collect evidence from the Internet.	Freeware
FTK (Forensic Toolkit)	Windows	It is multipurpose tool which is used to indexing of digital evidence.	Commercial
DFF (Digital Forensic Framework)	Windows / Linux/Mac	Digital evidence collection and analysis toolkit.	Commercial

Tool Name	Platform	Description	Availability
OS Forensics	Windows	This tool is used to perform forensic analysis on web browsers, emails, Files, and Images.	Freeware/ Commercial
CAINE (Computer Aided Investigative Environment)	Linux	Toolkit for live computer forensics on Linux.	Freeware/ Commercial
COFEE(Computer Online Forensic Evidence Extractor)	Windows	A generic toolkit for live digital forensic analysis.	Commercial
CMAT (Compile Memory Analysis Tool)	Windows	Memory analyzer tool.	Freeware
Wire Shark	Windows/Mac/Linux	Captures and analyzes packets on network.	Open source
Network Miner	Windows/Linux	Extracts files, images and other metadata from PCAP files on the network.	Freeware
Hash Keeper	Windows	Database application for storing file hash signatures.	Freeware

4. Key Challenges for Performing Live Digital Forensic

Device diversity is one of the key challenges in live digital analysis, which leads to time constriction to locate and acquire relevant data. Since examiners have to look into the large volumes of data gathered for digital analysis; therefore, it is imperative to reduce data size by filtering the unnecessary information so that it becomes handy to perform analysis. In this study, we have observed that digital evidence is very diverse and distributed in nature; hence relevant event should be categorized to perform effective analysis. Anti-forensic techniques also pose many difficulties while gathering and analyzing digital data. The potentially incomplete view of system (because of presence of rootkits etc.) during the live forensic analysis is a major problem. Therefore, the extracted significant events should be protected and handled carefully. We have also found in this research that mostly open source tools are used for gathering and analysis of digital evidence which may affect in testing and validation process.

5. CONCLUSIONS

Modification of memory content is unavoidable while performing live forensic analysis to collect evidence. In live forensic analysis, the sophisticated forensic tools are not only required to collect and analyze data, but are also need to resolve any ambiguity or conflicts introduced due to their execution. For example, memory image capturing tools can swap or reallocate the memory addresses. It is very difficult to calculate impact of memory alteration due to execution of forensic tool, but it is not practically impossible. So it is more important to measure the content alteration of volatile memory due to execution of forensic tool; therefore, data acquired by the memory analysis tools should be consistent with the actual data and correspond to the real live system status at that instant. Memory alteration occurs due to many reasons; one of them is loading and running of memory contents gathering tools, which affect the key traces.

Due to rapid increase in the number of Internet users across the world, the frequency of

digital attacks has increased manifold. Therefore, there is a need to devise effective methodologies and develop efficient tools to detect these attacks timely and triage the appropriate procedures without disturbing the functionality of the running system. Considerable amount of work is required to develop pertinent triage for live digital forensic analysis. In this paper, we have critically examined different techniques for performing live forensic analysis. This research produces a comparative study of the tools and techniques regarding digital forensic analysis.

6. FUTURE WORK

There are still many feasible and optimal ways to conduct live digital forensic analysis. The current live forensic analysis tools do not produce the desired results, due to which suitability of the process or technique cannot be determined. So, there is a need to devise effective methodologies and develop efficient tools to detect these attacks timely and triage the appropriate procedures without disturbing the functionality of the running system. Considerable amount of work is required to develop pertinent triage for live digital forensic analysis. As a future dimension to this research, we intend to propose a new framework to triage the procedures in live digital analysis.

REFERENCES

[1] Stefan Bolagh, Matej Pondelik. "*Capturing Encryption Keys for Digital Analysis*". In *Proceedings 6th International Conference on Intelligent Data Acquisition and Advanced Computing Systems(IDAACS)*, pages 759--763, Prague, 15-17 September 2011.

[2] Dija S, Balan C, Anoop V and Ramani B. "*Towards Successful Forensic Recovery of BitLocked Volumes*". In *Proceedings 6th International Conference on System of Systems Engineering (SoSE)*, pages 317--322 , Albuquerque, NM, 27-30 June 2011.

[3] Ellick Chan, Shivaram Venkataraman, Alejandro Gutierrez, Roy H. Campbell. "*Characterizing Data Structures for Volatile Forensics*". In *Proceedings of 11th International Workshop on Systematic Approaches to Digital Forensic Engineering(SADFE)*, pages 1--9, Washington, DC, USA, 2011.

[4] A. Balaz, R. Hlinka. "*Forensic Analysis of Compromised Systems*". In *Proceedings of 10th Emerging eLearning Technologies & Applications (ICETA)*, pages 27--30, Stara Lesna, 8-9 November. 2012.

[5] Martin Barrere, Gustavo Betarte and Marcelo Rodriguez. "*Towards machine-assisted formal procedures for the collection of digital evidence*". In *Proceedings of 9th Annual International Conference on Privacy, Security and Trust (PST)*, pages 32--35, Montreal, QC, 19-21 July 2011.

[6] Juanru Li, Dawu Gu, Yuhao Luo. "*Android Malware Forensics: Reconstruction of Malicious Events*". In *Proceedings of 32nd International Conference on Distributed Computing Systems Workshops (ICDCSW)*, pages 552--558, Macau, 18-21 June 2012.

[7] Fabio Marturana, Gianluigi Me, Rosamaria Berte and Simone Tacconi. "*A quantitative approach to Triaging in Mobile Forensics*". In *Proceedings of 10th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pages 582--588, Changsha, 16-18 November 2011.

[8] Chung-Huang Yang , Pei-Hua Yen. "*Fast Deployment of Computer Forensics with USBs*". In *Proceedings of International Conference on Broadband, Wireless Computing, Communication and Applications (BWCCA)*, pages 413--416, Fukuoka, 4-6 November 2010.

[9] Jooyoung Lee and Sungyong Un. "*Digital Forensic as a Service : A Case Study of forensic Indexed Search*". In *Proceedings of International Conference on ICT Convergence (ICTC)*, pages 499--503, Jeju Island, 15-17 October 2012.

[10] Joonho choi,Antinio Savoldi, Paolo Gubian, Seokhee Lee and Sangjin Lee. "*Live Forensic Analysis of a Compromised Linux System Using LECT (Linux Evidence Collection Tool)*". In *Proceedings of International Conference on Information Security and Assurance (ISA)*, pages 231--236, Busan, 24-26 April 2008.

[11] Brett Lempereur, Madjid Merabti, Qi Shi. "*Information Flow Monitoring: Model, Policy, and Analysis*". In *Proceedings of International Conference on Developments in E-systems Engineering (DeSE)*, pages 227--232, Dubai, 6-8 December 2011.

[12] Funminiyi Olajide, Nick Savage, Galyna Akmayeva and Charles Shoniregun. "*Extracting Forensically Relevant Information from Windows Applications*". In *Proceedings of International Conference on Information Society (i-Society)*, pages 423--428, London, 25-28 June 2012.

[13] Frank Y. W. Law, Patrick P. F. Chan, S. M. Yiu, K. P. Chow, Michael Y.K.Kwan,Hayson K.S.Tse, Pierre K.Y.Lai. "*Protecting Digital Data Privacy in Computer Forensic Examination*". In *Proceedings of 6th International Workshop on Systematic Approaches to Digital Forensic Engineering (SADFE)*, pages 1--6, Oakland, CA, 26 May 2011.

[14] [14] Sasa Mrdovic, Alvin Huseinovic, Ernedin Zajko. "*Combining Static and Live Digital Forensic Analysis in Virtual Environment*". In *Proceedings of 12th International Symposium on Information, Communication and Automation Technologies (ICAT)*, pages 1--6, Bosnia, 29-31 October 2009.

[15] Lianhai Wang, Ruichao Zhang, Shuhui Zhang. "*A Model of Computer Live Forensics Based on Physical Memory Analysis*". In *Proceedings of 1st International Conference on Information Science and Engineering (ICISE)*, pages 4647--4649, Nanjing, 26-28 December 2009.

[16] Aljaedi A., Lindskog D., Zavarsky P., Ruhl, Almari F. "*Comparative Analysis of Volatile Memory Forensics: Live Response vs. Memory Imaging*". In *Proceedings of 3rd International Conference on Privacy, security, risk and trust (passat)*, pages 1253--1258, Boston, MA, 9-11 October 2011.