International Journal of Hybrid Information Technology
Vol.8, No.2 (2015), pp.379-388
http://dx.doi.org/10.14257/ijhit.2015.8.2.35

Review of Live Forensic Analysis Techniques

Shuaibur Rahman and M. N. A. Khan

Shaheed Zulfikar Ali Bhutto Institute of Science and Technology, Islamabad
akhunzada.shoaib@gmail.com, mnak2010@gmail.com

Abstract

The widespread availability and extensive use of Internet across the world has caught attention of the criminals and digital crimes are occurring at an epidemic scale nowadays. The field of digital forensics is constantly evolving by employing new tools and technique to counter novel approaches employed by the criminals as well as to investigate the nature of the criminal activity and bring the culprits to justice. Traditionally, the static analysis was used to investigate the digital incidents. But due to advancement in technology and the fact that hackers are developing malware that do not leave footprint on the hard disk, the need for performing live digital forensic analysis in addition to the static analysis has become imperative. Live forensic analysis techniques have evolved during the last decade to analyses the memory content to get a better picture of the running application programmers, processes and active binaries. In this study, we look into different techniques of live analysis and critically review them by identifying their benefits and limitations. The key areas focused in this study pertain to virtualization, pagefile extraction and identifying the encryption keys.

Keywords: *Live Forensic Analysis, Digital Forensics, Volatile Memory Forensics*

1. Introduction

The computer crimes through Internet are increasing nowadays and digital forensics plays an important role to reduce their rampant and unchecked usage. Forensic investigation is used to find out the digital proof or evidence using different tools and it is inherently a difficult and complex process. Digital investigations take place in three main phases. The first phase is acquisition. In this phase, the investigator takes snapshot or images of digital device and replicates these images from the target device to some other device for in-depth analysis. The second phase is called analysis. In this phase, the investigator identifies the digital evidence using different types of techniques such as recovering the deleted files, acquiring information of user accounts, identifying information about the attached devices like USB, CD/DVD drives, external hard disks etc. The third phase is called reporting in which the investigator reconstructs the actual scenario based on the sequence of activities happened on the target system.

Digital forensic analysis is divided into two main categories. The first one is dead or the static forensic analysis. During this analysis, all the target devices that are required in the analysis are shutdown. The second category is live analysis. During this type of analysis, the system stays in the boot mode and is kept alive [2] to acquire pertinent information from the physical memory content.

Live analysis aims at gathering evidence from systems using different operations and techniques related to primary memory content. Live forensic is the most challenging kind of digital forensic investigations. To perform the live forensics, it is vital to understand the basic techniques and tools used in digital forensics. The investigator needs to acquire the complete image of a computer usage history as well as the current

ISSN: 1738-9968 IJHIT

state through live forensic analysis tools. Though static analysis is kind of a developed part of digital forensics, but other techniques related to live analysis need to be developed to mitigate its weaknesses.

Some legal issues about live forensics include inconsistency of result when compared to different analysis techniques as the state of raw data is inconsistent. If an independent investigator produces certain analysis result then after sometime another investigator may come up with bit different results. So, the issue of inconsistency of analysis is inherently linked to the live analysis. Another key issue is how we can auto-configure a system to detect the attached devices [1]. The present live forensic approaches suffer from some issues such as credibility, fidelity and integrity which are difficult to verify. On the other hand, anti-forensic techniques might change the static data when acquired by the investigators using different tools. In live forensic analysis, both the evidence gathering process and the analysis itself take place at the same time, so it might be difficult to recognize whether the acquired data values are legal or otherwise [5].

Though live analysis might not generate reliable outcome, nevertheless it is helpful in many cases. For instance, if several computers are engaged in an attack and an investigator wants to identify the state of each system then the live analysis is the most suitable way [3]. Wang et al. [5] propose a model for live analysis by breaking it into different stages such as gathering evidence, examining it, analyzing it and finally generating a report. This model is based on the physical memory or memory image/dump.

Due to rapid increase in memory size, the forensic investigators strongly recommend the live response approach for acquisition of volatile evidence [4]. Through this technique the investigator can collect not only the information about live processes but also about the terminated and cache processes. Volatile memory analysis becomes an important piece of investigation because the physical memory could have potential evidence which investigator cannot find on the disk storage. To get hold of the incident, the volatile data acquisition is the initial step in digital investigation. Usually the investigator gathers the volatile data through live response, while the attacker might use different libraries to make the system calls to connect to the kernel and alter the volatile data.

Memory image analysis approach is also used to investigate the volatile data of a target system. This approach might serve as an alternate to live response. Using the administrative tools, the examiner can obtain all the volatile information in the memory including the terminated processes' information. Virtualization techniques are also used in digital forensics [7]. In virtualization approach, after obtaining memory image and imaging the hard disk boot sequence/mechanism of the target system, the duplicate copy of target hard disk is mounted on a virtual machine. In this way, the investigator may collect the important information which otherwise could not have been possible in the real environment. On the other hand, to boot the acquired image on a virtual machine, several changes are required to be carried out in the real environment. These virtualization techniques bring both the live analysis and static analysis closer to each other. In the static forensic analysis paradigm, Window registry plays an important role and forensic scientists obtain valuable information from Window registry and use it for extracting evidence [13].

2. Literature Review

Hay *et al.* [1] state that standard user interface technique is the simplest technique through which investigator interrelate with OS such as command shell or a telnet connection. Through this technique, the investigator can get the vital system information such as logged on users, open ports, network connections, list of processes etc. For Linux system, investigators use CD-ROM to statically link "ps" utilities rather than using system provided "ps" utilities to obtain the current list of processes. Microsoft renders a Computer Online Forensic Evidence Extractor (COFEE) just like a stick-base USB flash storage device. This drive is already loaded with a lot of automated common live analysis operation. Using the basic approach with system tools the investigator must logon to the target system. The actions performed by the investigator alter state of the investigated system, so the results might be different if replaying that process again. To achieve this goal it is also possible to modify the hardware of the target system for live forensics analysis.

Yadav [2] states that digital investigation is a difficult process and is used to find out the digital evidence using different tools. Digital forensics takes place in three main phases. The first phase is acquisition. In this phase, the investigator takes multiple snapshots or images of the digital devices to some external device to perform detail analysis. Second phase is called analysis. In this phase, the investigator identifies the proof using different types of techniques such as recovering the deleted files, acquiring information of user accounts, information about the attach devices like USB flash storage device. Third phase is called reporting where investigator reconstructs the actual scenario.

Carrier [3] focuses on the countermeasure tools and classifies rootkits into three categories: application level rootkits, library level rootkits and kernel level rootkits. The kernel level rootkit is difficult to counter and this will always be a challenge to live analysis. Live analysis might not generate reliable outcome but it is helpful in many cases. If many computers are engaged in an attack and an investigator wants to identify the state of each system then the live analysis is the most suitable choice.

Savold and Gubian [4] focus on extracting memory page file. The live analysis approach guarantees analyzing data more quickly. An important factor is that pagefiles have evidential data which is directly related to the processes that exist in the RAM dump, but the present analysis methods do not regard pagefile an important factor. For recapturing of pages which have been lost from the main memory, it might be needed at times to acquire pagefile for rebuilding of active processes. It might be difficult to collect the pagefile on a live system because of full control and security imposed by the operating system. To bypass the security and control of an operating system and to collect the pagefile, there are many tools available such as X-Ways forensic, FTK imager, and DiskExplorer. After collecting the pagefile and main memory, the analysis phase needs to be started. The investigator should identify which processes belong to which page in the pagefile. Investigators can get password of attack dictionary from the pagefile.

Wang *et al.*, [5] propose a model for live analysis by breaking it into different stages such as evidence gathering, examining it, analyzing it and finally generating a report. This model is based on the physical memory or memory image. Live analysis based on the physical memory takes place on the running system. An external media toolkit is linked to the system so this will potentially change behavior of the target system as well as it might be difficult to verify the credibility of evidence. It should be important to

enable validation of live digital evidence. After acquiring an image of memory, the raw data of the toolkit should be available for the third party assessor so that they can make their own conclusions about the validation of live digital evidence. Under this model, the memory image collection is the only task to execute so the impact of evidence gathering can be reduced.

In [6], Aljaedi *et al.* described the volatile memory techniques, due to rapid increase in memory size, the forensic investigators strongly recommend the live response approach for acquisition of volatile evidence. Through this technique, the investigator can collect not only the information about live processes but also about the finished and cache processes. Volatile memory analysis becomes an important piece of investigation because the physical memory could have potential evidence which investigator cannot find on the hard drives. To get a good hold of the incident, the volatile data achievement is the initial step in the digital investigation. Usually the investigator gathers the volatile data through live response while the attacker might use different libraries to make the system call and connect the kernel and revise the volatile data. To investigate the volatile data of target system, a memory image analysis approach is also used. This approach might serve as an option to live response. With administrative tools, the examiner can obtain all the volatile information in the memory including the terminated processes information.

In paper [7], the authors focused on vitalization forensic technique, virtualization techniques are also used in digital forensics which after taking memory image and hard disk boot that copy of hard disk to some virtual machine and collect the important information which investigator may not access in the real environment. To boot image on a virtual machine, several changes needs to be done in the real environment. These virtualization techniques bring both the live analysis and static analysis closer to each other.

Using live view the image of hard disk can be booted in virtual machine before the system shut down acquire the memory dump and this can be restore after system booted in virtual environment so the memory dump will be the same or as closed to state when it was acquired and all the process will be start in same order as well as all this include the information about the process such as when it started etc. This contains open files, network port, and the programs which is hidden such as rootkits which are not noticeable during live analysis. Also encryption key can sometime be reestablished from the memory dump. The good aspect about this approach is that the image of memory is preserved and remains unaffected and may be acceptable in the court of law.

Gianni and Solinas [8] conducted live forensic analysis on two virtual machine having two different operating system window XP and windows 7. The authors focus only some general applications such as Google Talk, Skype and the Internet Explorer browser. They use FTK imager software for the acquisition of dump memory and save this dump memory image into connected USB device. The purpose of this work is to figure out the differences between window XP and window 7 during live forensic analysis so this shows that there are no major differences found during live analysis on both the operating system.

In paper [9], the authors focused on vitalization forensic technique, standard computer forensic or digital live forensic is carried out on a target system's memory dump. A main issue is how to perform live forensic based on virtual machine. Virtual machine can also operate just like a forensic tool that the investigator can do forensic analysis more easily and efficiently by using a copy or an image of the file of disk and

also acquire the virtual machine file form the target system because the host system also has some files of virtual machine.

In [10] the authors propose how to find encryption keys from the memory dump, in Linux they used TruCrypt software having AES encryption algorithms with XTS mode (two encryption keys) to acquire image of the memory so they get all the data except password. They analyze that the encryption keys is found in HEAP of the final process generated by the TrueCrypt, from the HEAP process. They also find PID and with PID final process, they find address of pages in the memory dump which is the access process, so using this method they reduce the size of image because they do not acquire image of all memory. For getting image in the window, they acquire image of memory which is 256MB and starting address 86000000 because the pool which created by TrueCrypt driver is placed in the address bigger than 86000000, where pool is a kernel mode just like a user mode heap. All the keys exist in kernel memory because all the encryption and decryption processes are implemented as kernel factor so the TrueCrypt driver assigns the kernel mode memory and store all the keys as non-paged pool.

The limitation of this work is for finding the location of keys. They have to install different package on a target system for different kind of a encryption package Second if the memory contain some a small amount of decomposition then the extract keys might be having some error and the big problem about this work is that it might be possible that the key is split across the memory in a consistent pattern. LiveView and VFC (Virtual Forensic Computing) are the common tools to adjust the virtual machine configuration to adopt the target OS settings.

In [11], the authors made an attempt to find out the important evidence from memory dump and about the modern browser sessions in form of the logged in users. Different web applications were chosen and a test was carried out on two browsers. The authors analyzed that the important information such as password and username can be found. They used Nigilant32 tool for capturing image for live memory. This tool needs to be installed on the target system and run from a CD or USB device. After acquiring the memory dump, the authors search watchfully to find out the related information about the concerned application. First of all, they extracted all the strings using Window Sysinternals utility and saved it in a text files for different images because they acquired different images of the memory. Then they searched to find the related information like username and password in the text file.

In [12], the authors suggest how to acquire files live from the operating system that are associated with the virtual machine. The paper also focuses on the files which are acquired live to evaluate and obtain data that is in raw form. The paper is helpful for forensic scientists and provides them important information from the raw data. The aim of the live forensic is to collect data from the systems that is running. It also offers the advantage to provide extra information about volatile memory and states of the systems that cannot be obtained through static forensic. Many techniques have been suggested to employ and use VM as a tool for forensic evidence. To use VM as forensic evidence, memory dump from host machine is taken live and files corresponding to the virtual machine are pointed out. The files which are obtained live are used as a starting point for VM to run in a system on which it is running. To use VM as a tool for forensic, the corresponding OS is loaded in the VM, which mocks a live system.

In [13] get pertinent information from registry for the purpose of analysis by listing concerns from the suspect's viewpoint. The paper also discussed that some predefined measures are required to explore the remaining of registry to find the uninstalled programs. Window registry is a central place that contains information about the

system's and applications' configuration, and is accessed by the users and various applications during their execution. It is a superb resource for getting data for evidence purpose. On the other hand, since it contains important credentials, a suspect can easily hide information in the registry hives. When a program is uninstalled, the credentials related to it are not removed and hence become a secure hiding place for the hackers as they can store specific data in these registry values. For this purpose, some registry keys are required to be examined to know about the criminal activities on the computer. Keys which have most recently been used are the key factors while examining to know what files and folders have been accessed recently. Most accessible files and folders are the best locations to investigate.

The registry values are in binary form and hackers can easily store a part or the whole binary file into the registry. Forensic analyst should be able to know about the relevant information in the registry. However, finding hidden data can be a tedious and time consuming task. Suspect may be able to hide information in one of the following way. They may consider the registry value which may not affect the whole system. They may keep in mind the priorities of forensic analysts while undertaking investigations. They may use value mostly related to least used programs or may use other value types which are least frequently used by the applications. First registry values are examined by removing an installed program. It is required that the analyst be able to examine the keys of the installed program as well as the one which have been removed and assign the priorities to them while find the related keys for the purpose of examination.

In [14] the authors proposed a solution for collecting the digital evidence; they develop a live USB/DVD, during analysis if the target system is alive they format a script program and saved it in simple USB device and through this script they collect the required information from the memory dump and stored that created files into USB device (live analysis). If the target system is turned off the reboot they system by live USB/DVD and produce an image file of the memory, live USB/DVD have some tool to produce the image of the disk such as AIR, image file producer etc. The propose system have its own self developed script stored in USB they assume the target system have linux operating system installed having USB port as well to gather the evidence, the propose system also show other information about target system such as current processes, network ports, freshly executed commands, information about kernel, date time and host name etc, the propose system also give a graphical user interface such as Xdialog to show the result of the target system analysis.

In [15], the authors focus on different possible options to use the hibernation feature and allow the analyst to carry on the static analysis on disks. It mainly considers the hider data on disk. The paper also discusses the availability of data in the state of hibernation. A technique for hiding and showing the data is also discussed. The technique used tools that are of wide use. The tools used for this purpose using the keys and kept it in memory. The memory which contains these keys should be store to the disk using the process of hibernation. The analyst should keep in mind the corresponding decrypted tools that might be helpful in forensic analysis. The file of other hibernate state should also be used for this purpose. The paper also discuss on the possible future work to extract these keys from memory using other files.

Iqbal *et al.* [25,26] proposed performance metrics for software design and software project management. Process improvement methodologies are elaborated in [27-28] and Khan *et al.* [29] carried out quality assurance assessment. Amir *et al.* [30] discussed agile software development processes. Khan et al. [31] and Khan et al. [32] analyzed issues pertaining to

database query optimization and requirement engineering processes respectively. Umar and Khan [33-34] analyzed non-functional requirements for software maintainability. Khan *et al.* [35-36] proposed a machine learning approaches for post-event timeline reconstruction. Khan [37] suggests that Bayesian techniques are more promising than other conventional machine learning techniques for timeline reconstruction. Rafique and Khan [38] explored various methods, practices and tools being used for static and live digital forensics. In [39], Bashir and Khan discuss triaging methodologies being used for live digital forensic analysis. References [31-47] reviewed different techniques in different domains and reported their critical evaluations.

3. Critical Review

In this section, we provide a critical review of the surveyed literature.

Table I. Critical Review of Live Forensic Analysis

Ref #	Technique Used	Focus Area	Pros	Cons
[1]	Standard user interface, modify the hardware of the target system, Virtual Machine Interaction (VMI).	Live forensic analysis.	Live analysis techniques can give the investigator a more complete picture of computer past and current state.	Inconsistency, repeatability, configuration of system to automatic detects the attached device.
[2]	Live analysis, static analysis.	Digital forensic.	Live analysis can reduce the time and effort of investigator.	The real challenge of digital forensics is on mobile devices because different cell phone uses different operating system such as Android, Apple, Blackberry, Windows Mobile, Symbian.
[3]	Countermeasure.	Live forensic analysis.	Easily counter the application level counter rootkit, (modify or hide).	Difficult to counter the kernel level rootkit.
[4]	Extract invalid paagefile.	Live forensic analysis.	Pagefiles have evidential data which is directly related to the processes that exist in the RAM dump.	It might be difficult to collect the pagefile on a live system because of full control and security imposed by the operating system.
[5]	Physical memory analysis.	Live forensic analysis	The memory image collection is the only task to execute so the impact of evidence gathering can be reduced.	The issue of credibility of live forensics should be considered.
[6]	Live response, Memory imaging.	Volatile memory forensic.	The investigator can collect not only the information about live processes but also about the finished and cache processes.	Due to consistently modification in linux kernel this also a challenge to collect important data from RAM or memory dumps through memory image and live response analysis technique.
[7]	Virtualization or virtual machine.	Live forensic analysis.	These virtualization techniques bring both the live analysis and static analysis closer to each other, and the image of memory is preserved and remains unaffected and may be	Several changes needs to be done in the real environment, it may be effect the important evidence.

			acceptable in the court of law.	
[8]	Virtualization or virtual machine. (Window XP, window 7).	Live forensic analysis.	There are no major differences found during live analysis on both the operating system (window XP, window 7)	For the acquisition of image software must be installed on target system like FTK imager etc.
[9]	Virtualization or virtual machine.	Live forensic analysis.	Virtual machine can also operate just like a forensic tool that the investigator can do forensic analysis more easily and efficiently.	The damaging of the host system should be considered.
[10]	Identifying encryption keys in memory.	Live forensic analysis.	Reduce the size of acquire image, no need to acquire the whole image of the memory.	Need to install different package on a target system for different kind of a encryption package, if the memory contain some a small amount of decomposition then the extract keys might be having some error, it might be possible that the key is split across the memory in a consistent pattern.
[11]	Recover digital evidence from a system's RAM, about the most recent browsers.	Live forensic analysis.	User name and password can be recovered.	The tool use for acquired the memory image is so expensive like Nigilant32.
[12]	Virtualization or virtual machine.	Live forensic analysis.	The paper is helpful for forensic investigator and provides them important information from the raw data; focus on live internal data (file) acquisition.	There might be important evidence in other files of the target system.
[13]	Extract some registry entries associated to forensic analysis based on Windows XP.	Live forensic analysis.	Window registry plays an important role in forensic computing as it provides important information for forensic analysis	Finding hidden data in window registry can be a tedious and time consuming task for investigator.
[14]	Live USB/DVD	Live forensic analysis.	Open source and easy to use and friendly Graphical user interface.	System modification may be loose some of the evidence.
[15]	Extract hibernate file.	Live forensic analysis.	Hibernate might be contain important information about the target system.	Integrity of a compromise system should be compromised and Forensic analysis should be hard if the data been available is in encrypted form

4. Conclusion

In this work, we review advantages and disadvantages of different techniques about live forensic analysis, we analyze that due to increase in cyber crime the live analysis is the best way to investigate the target system, also live forensic analysis have so many advantages over static analysis. Furthermore virtualization live analysis solved so many issues related to live analysis also pagefile extraction and identify encryption keys technique get investigator a more live picture of memory dump. Live USB/DVD and

free open sources are also developed to help investigator during investigation. But there are still some problem remain such as alter the target system might be corrupt some important evidence, integrity and repeatability of a compromise system etc.

References

[1] B. Hay, K. Nance, and M. Bishop, "Live Analysis: Progress and Challenges," IEEE Security and Privacy vol. 7, no. 2, (**2009**) March, pp. 30–37.
[2] S. Yadav, "Analysis of Digital Forensic and Investigation," VSRD-IJCSIT, vol. 1, no. 3, (**2011**), pp. 171-178.
[3] B. D. Carrier, "Risks of live digital forensic analysis," Communications of the ACM, vol. 49, no. 2, (**2006**), pp. 56-61.
[4] A. Savold., and P. Gubian, "Towards the virtual memory space reconstruction for windows live forensic purposes," In IEEE Systematic Approaches to Digital Forensic Engineering, 2008. SADFE'08. Third International Workshop on, (**2008**), May, pp. 15-22.
[5] L. Wang, R. Zhang., and S. Zhang, "A model of computer live forensics based on physical memory analysis," In IEEE Information Science and Engineering (ICISE), 2009 1st International Conference on, (**2009**), December, pp. 4647-4649.
[6] A. Aljaedi., D. Lindskog, P. Zavarsky, R. Ruhl, and F. Almari,, "Comparative Analysis of Volatile Memory Forensics: Live Response vs. Memory Imaging," In IEEE Privacy, security, risk and trust (passat), 2011 ieee third international conference on and 2011 ieee third international conference on social computing (socialcom), (**2011**0, October, pp. 1253-1258.
[7] S. Mrdovic, A. Huseinovic, and E. Zajko, "Combining static and live digital forensic analysis in virtual environment," In IEEE Information, Communication and Automation Technologies, 2009. ICAT 2009. XXII International Symposium on, (**2009**), October, pp. 1-6.
[8] F. Gianni, and F. Solinas, "Live Digital Forensics: Windows XP vs Windows 7," In IEEE Informatics and Applications (ICIA), 2013 Second International Conference on, (**2013**), September, pp. 1-6.
[9] L. Zhang, D. Zhang, and L. Wang, "Live digital forensics in a virtual machine," In IEEE Computer Application and System Modeling (ICCASM), 2010 International Conference on, vol. 4, pp. V4-328, (**2010**), October.
[10] S. Balogh, and M. Pondelik, "Capturing encryption keys for digital analysis," In IEEE Intelligent Data Acquisition and Advanced Computing Systems (IDAACS), 2011 IEEE 6th International Conference on, vol. 2, (**2011**), September, pp. 759-763.
[11] I. Mohanty, and R. L. Velusamy, "Information Retrieval From Internet Applications For Digital Forensic," arXiv preprint arXiv:1209.3590 (**2012**).
[12] V. Meera, M. M. Isaac, and C. Balan, "Forensic acquisition and analysis of VMware virtual machine artifacts," In IEEE Automation, Computing, Communication, Control and Compressed Sensing (iMac4s), 2013 International Multi-Conference on, (**2013**), March, pp. 255-259 .
[13] Y. Kim, S. Lee, and D. Hong, "Suspects' data hiding at remaining registry values of uninstalled programs," In ICST Proceedings of the 1st international conference on Forensic applications and techniques in telecommunications, information, and multimedia and workshop on. (**2008)**, January, p. 32.
[14] C. H. Yang, and P. H. Yen, "Fast deployment of computer forensics with USBs," In IEEE Broadband, Wireless Computing, Communication and Applications (BWCCA), 2010 International Conference on, (**2010**), November, pp. 413-416.
[15] S. Mrdovic, and A. Huseinovic, "Forensic analysis of encrypted volumes using hibernation file," In IEEE Telecommunications Forum (TELFOR), 2011 19th on, (**2011**), pp. 1277-1280.
[16] S. Iqbal, M. Khalid and M N A. Khan, "A Distinctive Suite of Performance Metrics for Software Design", International Journal of Software Engineering & Its Applications, vol. 7, no. 5, (**2013**).
[17] S. Iqbal and M. N. A. Khan, "Yet another Set of Requirement Metrics for Software Projects", International Journal of Software Engineering & Its Applications, vol. 6, no. 1, (**2012**).
[18] M. Faizan, S. Ulhaq and M N A. Khan, "Defect Prevention and Process Improvement Methodology for Outsourced Software Projects", Middle-East Journal of Scientific Research, vol. 19, no. 5, (**2014**), pp. 674-682.
[19] M. Faizan, M NA. Khan and S. Ulhaq, "Contemporary Trends in Defect Prevention: A Survey Report", International Journal of Modern Education & Computer Science, vol. 4, no. 3, (**2012**).
[20] K. Khan, A. Khan, M. Aamir and M N A. Khan, "Quality Assurance Assessment in Global Software Development", World Applied Sciences Journal, vol. 24, no. 11, (**2013**).

[21] M. Amir, K. Khan, A. Khan and M N A. Khan, "An Appraisal of Agile Software Development Process", International Journal of Advanced Science & Technology, vol. 58, (**2013**).
[22] M. Khan and M. N. A. Khan, "Exploring Query Optimization Techniques in Relational Databases", International Journal of Database Theory & Application, vol. 6, no. 3, (**2013**).
[23] M N A. Khan, M. Khalid and S. ulHaq, "Review of Requirements Management Issues in Software Development", International Journal of Modern Education & Computer Science, vol. 5, no. 1, (**2013**).
[24] M. Umar and M N A. Khan, "A Framework to Separate Non-Functional Requirements for System Maintainability", Kuwait Journal of Science & Engineering, vol. 39, no. 1 B, (**2012**), pp. 211-231.
[25] M. Umar and M. N. A. Khan, "Analyzing Non-Functional Requirements (NFRs) for software development", In IEEE 2nd International Conference on Software Engineering and Service Science (ICSESS), (**2011**), pp. 675-678.
[26] M. N. A. Khan, C. R. Chatwin and R. C. Young, "A framework for post-event timeline reconstruction using neural networks. digital investigation", vol. 4, no. 3, (**2007**), pp.146-157.
[27] M. N. A. Khan, C. R. Chatwin and R. C. Young, "Extracting Evidence from Filesystem Activity using Bayesian Networks", International journal of Forensic computer science, vol. 1, (**2007**), pp. 50-63.
[28] M. N. A. Khan, "Performance analysis of Bayesian networks and neural networks in classification of file system activities", Computers & Security, vol. 31, no. 4, (**2012**), pp. 391-401.
[29] M. Rafique and M. N. A. Khan, "Exploring Static and Live Digital Forensics: Methods, Practices and Tools", International Journal of Scientific & Engineering Research, vol. 4, no. 10, (**2013**), pp. 1048-1056.
[30] M. S. Bashir and M. N. A. Khan, "Triage in Live Digital Forensic Analysis", International journal of Forensic Computer Science, vol. 1, (**2013**), pp. 35-44.
[31] A. Sarwar and M. N. Khan, "A Review of Trust Aspects in Cloud Computing Security", International Journal of Cloud Computing and Services Science (IJ-CLOSER), vol. 2, no. 2, (**2013**), pp. 116-122.
[32] A. H. Gondal and M. N. A. Khan, "A review of fully automated techniques for brain tumor detection from MR images", International Journal of Modern Education and Computer Science (IJMECS), vol. 5, no. 2, (**2013**), p. 55.
[33] Zia, A., & Khan, M. N. A. (2012). Identifying key challenges in performance issues in cloud computing. International Journal of Modern Education and Computer Science (IJMECS), 4(10), 59.
[34] ur Rehman, K., & Khan, M. N. A. (2013). The Foremost Guidelines for Achieving Higher Ranking in Search Results through Search Engine Optimization. International Journal of Advanced Science and Technology, 52, 101-110.
[35] Khan, M., & Khan, M. N. A. (2013). Exploring query optimization techniques in relational databases. International Journal of Database Theory & Application,6(3).
[36] Shehzad, R., KHAN, M. N., & Naeem, M. (2013). Integrating knowledge management with business intelligence processes for enhanced organizational learning. International Journal of Software Engineering and Its Applications, 7(2), 83-91.
[37] ul Haq, S., Raza, M., Zia, A., & Khan, M. N. A. (2011). Issues in global software development: A critical review. Journal of Software Engineering and Applications, 4(10), 590.
[38] Zia, A., & Khan, M. N. A. (2013). A Scheme to Reduce Response Time in Cloud Computing Environment. International Journal of Modern Education and Computer Science (IJMECS), 5(6), 56.
[39] Khan, M., & Tariq, M. (2011). The Context of Global Software Development: Challenges, Best Practices and Benefits. Information Management & Business Review, 3(4).
[40] Shahzad, A., Hussain, M., & Khan, M. N. A. (2013). Protecting from Zero-Day Malware Attacks. Middle-East Journal of Scientific Research, 17(4), 455-464.
[41] Khan, A. A., & Khan, M. (2011). Internet content regulation framework. International Journal of U-& E-Service, Science & Technology, 4(3).
[42] Kaleem Ullah, K. U., & MNA Khan, M. K. (2014). Security and Privacy Issues in Cloud Computing Environment: A Survey Paper. International Journal of Grid and Distributed Computing, 7(2), 89-98.
[43] Abbasi, A. A., Khan, M. N. A., & Khan, S. A. (2013). A Critical Survey of Iris Based Recognition Systems. Middle-East Journal of Scientific Research, 15(5), 663-668.
[44] Khan, M. N. A., Qureshi, S. A., & Riaz, N. (2013). Gender classification with decision trees. Int. J. Signal Process. Image Process. Patt. Recog, 6, 165-176.
[45] Ali, S. S., & Khan, M. N. A. (2013). ICT Infrastructure Framework for Microfinance Institutions and Banks in Pakistan: An Optimized Approach.International Journal of Online Marketing (IJOM), 3(2), 75-86.
[46] Mahmood, A., Ibrahim, M., & Khan, M. N. A. (2013). Service Composition in the Context of Service Oriented Architecture. Middle East Journal of Scientific Research, 15(11).
[47] Masood, M. A., & Khan, M. N. A. (2015). Clustering Techniques in Bioinformatics. I.J. Modern Education and Computer Science, 2015, 1, 38-46.

