Time Aspect of Information Stored On Windows Application Memory

Funminiyi Olajide
School of Engineering,
University of Portsmouth,
United Kingdom
funminiyi.olajide@port.ac.uk

Richard Trafford
Business School,
University of Portsmouth,
United Kingdom
richard.trafford@port.ac.uk

Galyna Akmayeva
Infonomic Society,
Republic of Ireland
gakmayeva@
infonomic-society.org

Charles Shoniregun
Infonomic Society,
Essex, United Kingdom
cshoniregun@
infonomic-society.org

Abstract- **In today digital investigation, forensic time aspect of information has become an important part of digital forensic research. Vital user input information can be gathered from the physical memory of computer systems. Digital forensic community demand accurate data collection, preservation, examination, validation, data analysis and presentation. This process has become the most important aspect of digital investigation. The extraction of forensically relevant evidence from the physical memory can reveals users' actions. This research will report the evidence that can be extracted and how that evidence changes with the length of time that the system is switched on when the application is still opened. In this experiment, the qualitative assessment of user input on some commonly used applications will be discussed.**

Keywords-Application; time; memory;Windows

I. INTRODUCTION

The development of digital investigation techniques focused mostly on evidence contained within the hard disks; little has been done on the volatile memory analysis of Windows application. Of recent time, the digital forensic research workshop [1] demand for more tools and techniques to be developed for capturing memory images and analyzing the memory content**.** As a result, there has been much progress with regards to forensic evidence gathering from the RAM of Windows and Linux systems. However, little research has been done on the analysis of the acquired evidence from the physical memory contents of computer systems.

A research work of [2], discusses the amount of sensitive evidence that are dispersed in the application memory. Limited efforts have been made into the formalization of the time aspect of application level information stored in the Windows application memory. Application level information is the extracted user input information from the physical memory of the most commonly used Windows applications [3]. Application level information can be defined as information which indicates how the user is (or in the case of terminated process, was) using an application. This research work details a qualitative assessment of evidence stored on the application memory and how that evidence can be reconstructed to determine the time aspect of forensically relevant information while the application is still opened and images were captured at set interval of 30 minutes hence, the system remains switched on. The approach of memory imaging and the extraction of forensically relevant data can be useful during digital investigation of user input. The forensic investigator can determine the quality information stored over time in the physical memory of the applications. This information can be used as information assurance when investigating crime related evidence and for tracing fraud on computer-based systems or digital devices such like the new emerging mobile technology. These actions will also reveals how user input activities can be recovered and reconstructed based on the event of user input on the applications. For the purpose of this research some commonly used Windows applications will be identified and the extracted user input from the physical memory of the applications will be investigated to reveals various user's actions. Thus, the extraction of forensically relevant evidence from Windows application relies on investigator's skill and experience. The approach taken in this research is pertinent towards what useful information can be recovered as evidence, to support the crime or fraud cases in the court of law.

In digital investigation, the investigators are determined to find out the exact likelihood of users action and this information can be reconstructed to further investigate the various event of user input on the applications. This process of investigations can help the investigators to carry out the analysis of time memory aspect of relevant information that are stored in the physical memory of the Windows application. For the evidence to be acceptable in the court of law, the evidence must be forensically sound and must be validated to be accepted for presentation and used as evidential document in the law court.

II. RELATED WORK

A method of [4] laid emphasis on the importance of forensic live response and event reconstruction methods. The extension of this research work in [5] focused on the application level evidence**.** This approach identified the important aspects of memory analysis and proposed an approach for the volatile memory analysis. The research method in [6] presented the hardware-based memory acquisition. This approach change memory contents as little as possible by using a PCI extension card to dump the memory content to an external device. There are open source software-based tools that have been developed recently for memory acquisition and for memory analysis.

978-1-908320-09/4/$25.00©2012 IEEE

A research of [7] focused on memory acquisition. This is a command line tool that captures and reconstructs the virtual address space of the system process and other processes.

A method of [8] is a tool that is capable of revealing hidden and terminated processes and threads whereas, Nigilant32 [10] are tools that can capture the physical memory contents of computer systems. In recent time Win32dd and Win64dd [9,7] have also been used to capture memory and perform memory analysis. In addition to these tools, MemParser [11] and the Volatility Framework [12] are examples of other tools that can perform full memory analysis on Windows application. Of these two, the Volatility Framework is more extensive. This tool is capable of performing the analysis on a variety of memory image formats such as DD format, crash dump and Hibernate Dumps. Volatility is able to list OS kernel modules, drivers, open network socket, loaded DLL modules, heaps, stacks and open files.

The research work of [13] addresses the need for more sophisticated tools on physical memory acquisition and analysis. This is data carving method of research that can be used for a recovery purpose. This approach is frequently used during digital investigations. Moreover, it is essential that a new development tools should be integrated on other different approaches. Therefore, a research model of [14], presented the graphics extraction that is contained in a memory dump. A paper [15], identified the most commonly used application. This approach provides prospective evidence regarding the application of memory analysis on Windows computer systems.

III. Investigation Approach

To investigate the forensic time aspect of user input on Windows applications, a normal working environment in business organizations was replicated for capturing volatile memory images. As shown in Table I, the computer would be turned on at the start of the day and then turned off at the end of the day. When the computer is first turned on, the applications will be opened and the user will interact with the applications and images will be captured at set interval of 30 minutes. Series of tests will be carried out for days until 100 images were captured on each application. The physical memory in the computer was 2 Gigabytes (GB) and this resulted in 200 GB of images captured. After volatile imaging, copies of the images captured were made for preservation purposes. The aim of this research is to determine how a user is interacting with the computer systems when user input were made on these applications. This process will identify how user is using the application while reconstructing the event of user input stored on the memory allocated to these applications when the computer is still switched on.

In this experiment, the memory content of the applications was investigated and it was discovered that the extracted evidence that was stored in the physical memory are dispersed. By pattern searching techniques that was developed, relevant user input was extracted and converted into strings. User input was searched through the application memory while the extracted evidence was matched with the original user input.

TABLE I. Approach

Applications	User Actions (Open Application)
Excel 2007	List a set of numbers and some data texts, or texts of paragraph only. Draw a graph of the numbers and texts. Input may contain alphanumeric, character 0-9, brackets. Save document
PowerPoint 2007	Write a slide, slides of texts with commas, semi-colon, brackets, full stop. User input may contain or type alphanumeric, character 0-9, brackets close or open. . Long sentences or short sentences. Save document.

The pattern searching techniques was developed using the system composition program like python. In this case, an automated executable program was used to perform pattern matching of memdump strings of the applications and this was also used to search for memory evidence. This evidence information was reconstructed to determine the time memory aspect of the forensically relevant data based on what the user was doing on the application, what the user has been doing and what the user was using the application for. This approach can reveal the sensitive information pertinent to user activities on the application memory. By reconstructing this evidence information, this approach can lead to further investigation, when forensic investigators are analyzing the user input on the basis of other forensic questions of why, who, how, when and where the user input are stored in the memory.

IV. Qualitative Assessment

In this section, the qualitative assessment of forensically relevant data from the physical memory content of Excel and PowerPoint applications was investigated and analysed. This investigation describes how evidence was stored over time in the physical memory contents of these applications. As it can be seen from Figures 1, the evidence information was dispersed in the memory and as extracted from the physical memory, the line number was allocated to the evidence found on this application. In some occasions, evidence information occurs repeatedly and consistently over time at different location in the memory and this can be traced with the line numbers allocated. The extracted memdump strings of user input on these applications have been reconstructed for evidential purposes. There are partial fragment of the evidence information and also, in some cases, whole fragment of evidence were discovered in continuous block of the application memory. For example, in Figure 1, the extracted evidence of user input on Excel application was found dispersed in the memory. The user input contains both data texts strings and numeric characters. It is evident that this evidence is repeated and appears consistently in the memory. In another example, the extracted user input of data texts like *"Three Major parties in the UK"* was first found in line number "94" and was repeated in line number "75474". This information can be termed as whole fragment of user input found in continuous block of

evidence whereas, in line number "100948", the partial fragment of this information was found as *"rties in the UK"*.

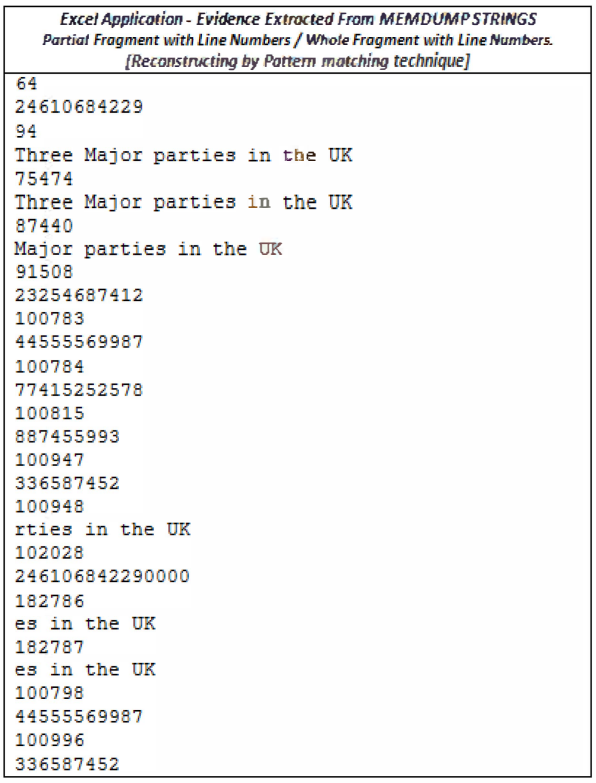

Figure 1. The reconstructed evidence of Excel Application

Also in Figure 1, another partial fragment of user input information was found in the allocated line number of "100948", "182786" and 182787". By reconstructing the user input, it was evident that the information are partially related to the information stored in line number "94", "75474", and "87440". This information may be in form of graph or table but the question of what the user was doing on this application can be traced to these line numbers. Further investigation into the numeric data of user input on Excel indicates that relevant data was found as repeated in the physical memory. For example, the extracted user input information like the numeric data of "4455569987" was stored in the allocated line number of "100783". This information was found repeated in another line number "100798". As shown in Figure 1, the extracted evidence of user input can be reconstructed and the forensically relevant data can be traced to the allocated line numbers, based on what the user was doing, what the user has been doing and what the user was using the applications for.

In another example, Fig. 2, describe the reconstructed application level information from the PowerPoint applications. In this experiment, the case was reversible; because the user input stored on this application was found dispersed more in whole fragment than the partial evidence information that was recovered. Although, the relevant evidence was extracted from the memdump strings, but more of the evidence information was recovered in continuous block of the application memory. Moreover, the evidence information that was dispersed in the memory content of this application can be reconstructed as partial or whole fragment of user input. By reconstructing the user input, more evidence information can be revealed. For example, the partial fragment of information was found stored in line number "4010" and also, in the allocated line number "4049". By reconstructing the two line numbers, the evidence stored in line number "7899"can be termed as the whole fragment of user input found in continuous block of evidence. This information represents the full sentence of forensically relevant data with full stops and commas. However, the partial fragment of evidence information was also found stored in line numbers "4010", "4049", "4050", "4051", "4052", and "4053" respectively. As shown in Figure 2, the user input evidence can be reconstructed to determine what the user was doing on the application, what the user has been doing and what the user was using the application for.

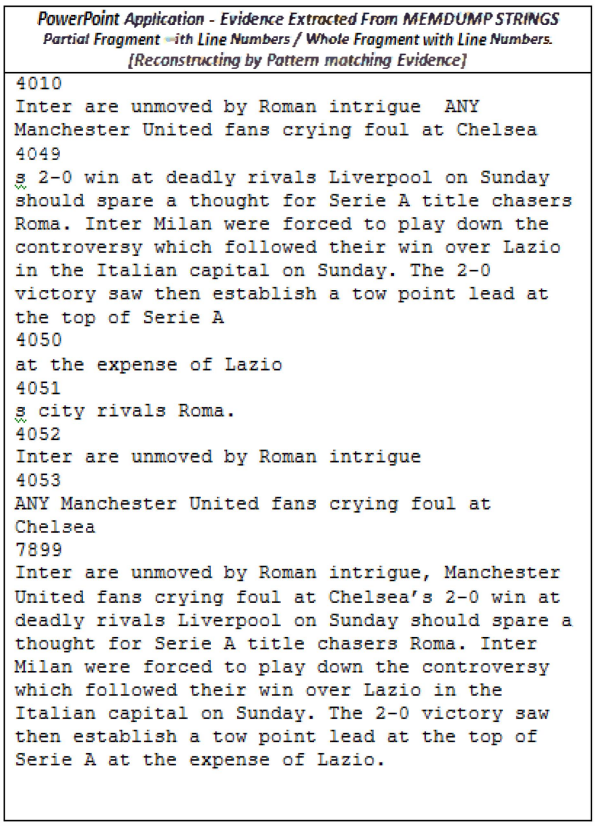

Figure 2. The reconstructed evidence of Excel Application

V. ANALYSIS

The purpose of this research experiments is to found out the forensically relevant data on physical memory and how this evidence can be assessed based on the quality of user input that has been reconstructed. Figure 1 indicates the extracted application level information recovered from Excel application. The extracted information was found stored in the memory allocated to the application with the associated line numbers. Some of this information was found repeated and dispersed. The qualitative assessment technique was adopted to assess the quality of the

recovered user input that could be used as evidence for forensic investigators.

As shown in Fig. 1, the character strings of the application level information contain reconstructed information that can be used as evidence information based on what the user is doing on this application, what the user has been doing and what the user is using the applications for. In this investigation, it can be said that there are text entered by the user and the numeric data; could be used to describe the items of user input information. This means that both numeric values and data text of information can be used to describe the application level information. It can be said that this information forms complete worksheets of Excel application. The numeric values with the associated line numbers may be computed, but it was difficult to link those of the numeric values stored in line numbers of the memory together. Thus, the application level information found was partially related to the original user input made on Excel application.

Figure 2 illustrates the user input recovered from the PowerPoint application. The user input found represents the extracted application level information from the volatile memory of the application. This information can be reconstructed to find out what the user was typing, what the user has been doing and what the user was using the application for. As illustrated in the figures above, this research experiment has clarified the important aspect of user activities on some commonly used Windows applications.

As earlier discussed above, the user input consists of alphanumeric data. This user input was made once while this application remains opened and user is interacting with other applications when images were captured at set interval of 30 minutes. The user input was investigated by pattern matching techniques that were developed. This evidence information is crucial to forensic investigators and it can be used to augment the qualitative assessment of user input stored over time in the physical memory. This is the sample application level information that was extracted from the memdump strings of the application memory. This evidence has been recovered for forensic analysis purposes.

Also, the extracted user input can be reconstructed to further investigate other forensic questions of what, who, when, where and how the user input was stored over time in the memory. For example, the question of *"what"* the user was doing on the application, can be traced to different allocated numbers of the reconstructed evidence in Figures 1 and 2. Also, the questions of "*who"* can be described as the identity of user that entered information on these applications. The questions of *"where"* can be termed as where user input was made. In this experiment, it can be said that the user input was made on Excel and PowerPoint applications. This is the applications that the user was using to input data. The question *of "when"* can be answered during the investigation of process listings and scanning of the applications using the Volatility tools. The question of *"how"* can be answered as the way user input information on these applications. It can be said the user input was dispersed in the application memory and as stored over time with the allocated line numbers.

VI. Conclusion

In this research, the qualitative assessment of user input was discussed on two most commonly used Windows applications. Specific emphasis has been laid on the quality of evidence stored over time in the allocated line number of the application memory. This approach describes the process of securing digital evidence and analysing the forensically relevant data on Windows computer systems. This experiment involves memory dumping, extraction of relevant data, strings conversion of evidence, finding the evidence and reconstructing the extracted evidence of user input. This approach may become part of forensic analysis in digital investigation.

VII. Future Work

In the future, forensic investigation and more practical experiments on other applications will be performed. The qualitative assessment of user input activities will be fully explored based on the assumptions and questions of how we can reconstruct this evidence to form a block chain of evidence in continuous format.

References

[1] Digital Forensic Research Workshop, DFRWS. (2007) http://www.dfrws.org/2007/challenge/index.shtml. [Online].

[2] F. Olajide. N. Savage, "Dispersal of Time Sensitive Evidence in Windows Physical Memory C, , June 2011.," in *yberforensics, International Conference on Cybercrime, Security & Digital Forensic*, The University of Strathclyde, Glasgow, UK, 2011, p. 27–29.

[3] F.Olajide, "A Study of Application Level Information From The Volatile Memory of Windows Computer Systemns ," PhD Thesis, University of Portsmouth, Portsmouth, UK, 2011.

[4] F. Olajide. N. Savage, "Forensic Live Response and Events Reconstruction Methods in Linux Systems," in *PGNET The Convergence of Telecommunications Networking and Broadcasting*, Liverpool, Dec. 2009, pp. 141-147.

[5] F. Olajide. N. Savage, "Application Level Evidence From Volatile Memory," *Journal of Computing in Systems and Engineering*, vol. II, no. 3, pp. 40-48, Jan. 2010.

[6] G. L. Garcia, "Forensic Physical Memory Analysis: an overview of tools and techniques," in *TKK T-110.5290 Seminar on Network Security*, Helsinki, Finland, 2007.

[7] Msuiche. (Accessed 2008) Msuiche.net at:, [Online]. http://www.msuiche.net/2008/06/14/capture-memory-under-win2k3-orvista-with-win32dd.

[8] ManTech. Memory. (2010) Memory dd. [Online]. http://www.mantech.com/msma/MDD.asp

[9] F. Cohen, "Challenges to digital forensic evidence.," in *Cybercrime Summit 06. Retrieved from: http://all.net/Talks*, Washington, 2006.

[10] Nigilant32, Agile Risk Management. (2006.) Agile . [Online]. http://agilerm.net/publications.4.html

[11] C. Betz, "Mempaser analysis tool.," in *DFRWS 2005 Forensic Challenge: http://www.dfrws.org/2005/challenge/memparser.shtml*, MA, 2005, pp. 100-115.

[12] Volatile. Systems. (2009) The Volatility framework: volatlile mwmory artifact extraction utility framework. . [Online]. http://www.volatilesystems.com/default/volatility

[13] D. Kleiman H. Carvey, "Windows Forensic Analysis Incident Response and Cybercrime Investigation Secrets," *1st ed. Syngress*

Publishing; , Jul. 2007.

[14] T. Hoppe, J. Dittmann. S. Kiltz, "A New Forensic Model and ITS Application To The Collection, Extraction And Long Term OF Screen Content OFF A Memory Dump," in *16th International Conference on Digital Signal Processing (DSP)*, Aegean island of Santorini, Greece, 2009.

[15] F. Olajide. N. Savage, "On the extraction of forensically relevant information from physical memory," in *World Congress on Internet Security (WORLDCIS-2011), Technically Co-Sponsored by IEEE UK/RI Computer Chapter*, London, 2011.