JOURNAL OF FORENSIC SCIENCES

J Forensic Sci, January 2016, Vol. 61, No. S1
doi: 10.1111/1556-4029.12979
Available online at: onlinelibrary.wiley.com

PAPER

GENERAL

Robert J. McDown[1], *B.Sc.; Cihan Varol*[1], *Ph.D.; Leonardo Carvajal*[1], *M.Sc.; and Lei Chen*[1], *Ph.D.*

In-Depth Analysis of Computer Memory Acquisition Software for Forensic Purposes

ABSTRACT: The comparison studies on random access memory (RAM) acquisition tools are either limited in metrics or the selected tools were designed to be executed in older operating systems. Therefore, this study evaluates widely used seven shareware or freeware/open source RAM acquisition forensic tools that are compatible to work with the latest 64-bit Windows operating systems. These tools' user interface capabilities, platform limitations, reporting capabilities, total execution time, shared and proprietary DLLs, modified registry keys, and invoked files during processing were compared. We observed that Windows Memory Reader and Belkasoft's Live Ram Capturer leaves the least fingerprints in memory when loaded. On the other hand, ProDiscover and FTK Imager perform poor in memory usage, processing time, DLL usage, and not-wanted artifacts introduced to the system. While Belkasoft's Live Ram Capturer is the fastest to obtain an image of the memory, Pro Discover takes the longest time to do the same job.

KEYWORDS: forensic science, computer forensics, forensic tools, live forensics, memory acquisition, volatile data

One of the first steps in digital forensic investigation is making the exact copies of the storage devices and random access memory. With that way digital investigators create evidence duplicates to do their analysis on the acquired image and preserve the original data (1). The success of the digital investigation is dependent on the availability and quality of the data that is collected. The quality of the data is described as the degree to which information is fit for use in a particular application. Another description of the data quality is that the information is accessible, complete, timely, and free of defects (2). While conducting postmortem acquisition (dead analysis), the quality of the data can be retained in high level because of the necessary cautions the investigator may apply, such as using write blocking devices and using the best fit tools with no time limitations on the process. However, one other area the investigators have been focusing on is the volatile information collected from random access memory (RAM). The live memory acquisition conducted on a computer system introduces new artifacts to the system, in which the quality of the data cannot be retained in high level likewise the one from a dead analysis.

Digital forensic tools can be used from acquiring the exact copy of the original evidence to the analysis of the operating system, or even in the final stages of the reporting (3). In digital forensics, no matter which area is being focused on, there will be advantages and disadvantages of the available tools that can be used to solve a case. Determining which tool will not only provide accurate results, but satisfy other required criterions from the investigator, such as processing time of the tool, reporting capabilities, and left artifacts, is a dire need in digital forensics area. Previous analysis and tests of the forensics tools will help investigators to be familiar with them, and it will yield to correct selection of the tool according to their need. Therefore, this study is intended to provide an in-depth analysis and comparison of seven forensic software (tools) that collect digital evidence from physical memory (RAM) in 64-bit Windows 7 and 8 platforms, specifically these are AccessData's *Forensic Toolkit Imager,* Belkasoft's *Live Ram Capturer*, Technology Pathways' *Prodiscover*, Guidance Software's *Winen*, Architecture Technology Corporation's *Windows Memory Reader*, Mandiant's *Memoryze*, and Moonsols' *DumpIt*. This study also provides justification about the need for volatile memory forensics and why it is vital to collect the evidence without losing significant information from its content.

The rest of the study is structured as follows: first, volatile memory and importance of forensic analysis of the memory are discussed; second, volatile memory acquisition techniques and information about the software tested in this study are shared; third, related work in forensic tools evaluation is discussed. At the end, the study is finalized with the results of the comparison study followed by a discussion and conclusion section.

Volatile Memory and Forensic Importance

The volatile memory is defined as computer memory that requires power to maintain the stored information (4). The data that are stored on the memory will likely get lost when the machine is shut down or when the content is overwritten when using the computer. Compared to a regular storage analysis, it is hard to predict and generate meaningful and ready for use data because its constant change behavior. However, still the information located in the memory is forensically important as it can

[1]Department of Computer Science, Sam Houston State University, 1903 Ave I, Huntsville, TX 77341.

Received 29 Aug. 2014; and in revised form 6 April 2015; accepted 19 April 2015.

hold many types of artifacts that cannot be found in a regular hard drive analysis.

Compared to a regular hard disk analysis, memory analysis is not fully structured. Investigators follow certain rules and steps while taking consideration on the operating system that is in question and the associated file structure when they are conducting investigation on a hard drive. However, it is nearly impossible to predict what you can find in volatile memory and the location of it (5). Based on the discussed reasons, analysts need to be more careful when collecting and analyzing evidence from the memory.

Memory forensics provide forensically information that cannot be acquired from a traditional dead analysis conducted on a hard disk. For instance, acquiring a user password is not always feasible from the hard disk unless the user deliberately stores that information into the hard drive. However, in reality, when the user types their password or when data are encrypted, the passwords are loaded into and stored in the memory which can help the investigator to recover those (6). For example, a former University of Wisconsin–Madison student was charged with possessing child pornography after identifying the encryption keys, to decrypted Mitra's hard drive, in the memory (7).

Another information that will not get exposed with dead analysis is the running processes in the memory. All running, hidden or even maybe previously ran processes can be acquired from the memory. These processes not only help to identify the software that is running or was executed previously, but also will help in the identification of intruders. Specifically, nowadays, memory is one of the favorite locations for the viruses, Trojans and worms. The attackers use memory to store malware information to be more effective and not being easily exposed to the user. That is why, traditional forensic analysis of the hard disk will not expose the code or allow the investigator to understand how the attack is being executed or how to mitigate it. Moreover, open documents and running processes information from the memory can be extremely important to locate and track the infection, if malware is propagated to the resident files.

Besides running the processes in the memory, another valuable information that can be gathered from the volatile memory is the network information, such as the network connections and open ports. For example, physical memory contents were used in the case between Columbia Pictures Industry and Bunnell for copyright violation (8). Specifically, according to the complaint, the defendants were operating a Web site called TorrentSpy.com which provided links to Bit Torrent copyrighted files. The court ordered the defendants to provide RAM contents stored on Web servers due to the relevant data it could contain about records of users' activities (8). The network and open port information is also vital especially if the attack manipulates the data that is being supplied by network traffic analyzer tools. It is a more challenging job to hide the attack from the memory data instead of a network analyzer tool.

As the examples indicate, the importance of volatile memory and its content cannot be oversight. Therefore, this research focuses on evaluating the tools which acquire the content of the physical memory to find digital evidence that will not be found in a common hard drive analysis. All the memory acquisition tools have different characteristics, advantages, and drawbacks. Therefore, regardless which tool is used to obtain a RAM image, changes to the original evidence should be minimal in order to reduce the risk of losing evidence.

Volatile Memory Acquisition

Hardware- and software-based acquisitions can be used to obtain the content of the RAM. The main advantage of the hardware-based technique is that the device can take the image of the RAM without introducing additional software to the target system which will eventually minimize the changes on the data that will be retrieved. The main deficiency of this technique is the high cost of equipment. Besides the cost factor, the equipment installation is needed before the attack occurs in order to allow the investigator to collect the image of the RAM. On the other hand, software-based solutions introduce new items to the computer system and to the memory when they are executed. This has the potential to overwrite on forensically important information in the RAM. However, software-based acquisitions are a cost-effective solution, usually free, and readily available (9). These are the reasons leading digital investigation communities are employing software-based solutions as opposed to the hardware-based ones.

Based on a survey, we conducted with the practitioners in digital forensics, we chose the most commonly used shareware or freeware/open source tools that are used to acquire RAM images in 64-bit Windows-based machines and evaluated them by their effects on the system. These include the following: *Forensic Toolkit Imager* version 3.1.4.6, *Live Ram Capturer* version 1.0, *ProDiscover* version 8.2.0.5, *WinEn* version 7.9.0.111, *Windows Memory Reader* version 1.0.0, *Memoryze* version 3.0, and *DumpIt* version 1.3.2. *Forensic Toolkit Imager (FTK Imager)* is a freeware forensic tool developed by AccessData (10) that has been supporting digital investigators to perform a complete computer forensic examination. It offers forensic tools for acquiring forensic images of both physical and logical memory, reading forensic images, decrypting data, and reporting of digital evidence. *Live Ram Capturer* is a freeware tool developed by Belkasoft (11). Digital investigators use this tool across both 32- and 64-bit Windows operating systems to acquire the contents of physical memory. *ProDiscover* is a shareware forensic tool developed by Tech Pathways. Digital investigators acquire the contents of physical memory, logical memory, as well as system BIOS using this commercial tool. *ProDiscover* also generates reports to document the digital evidence results (12). *WinEn*, or otherwise known as Encase Forensic Imager, is a commercial imaging tool developed by Guidance Software (13). This tool allows for physical and logical imaging, image browsing and analysis, and encryption capabilities. They also include their proprietary EnScript scripting language used for investigative automation. *Windows Memory Reader* is a verbose command line tool developed by Architecture Technology Corporation (14). This tool allows digital investigators to acquire live physical memory, memory-mapped device data and allows for zero padding for memory gaps. *Windows Memory Reader* allows the investigator to create a hash of the image with supported types of MD5, SHA-1, SHA-256, and SHA-512 (14). This tool has built-in functionality to allow for forensic automation with existing tools the investigator may currently using. *Memoryze* is a freeware forensic tool that has been developed by Mandiant (15). This tool allows the collection and analysis of memory images. Analysis can be performed on a live system or on the acquired image. *Memoryze* contains batch files to collect data such as image of the physical memory, driver, processes, network ports, and other forensically important information that can be gathered from live analysis (15). *DumpIt* is a freeware command line tool that was developed by MoonSols (16). This tool

allows for acquisitions of physical memory and stores the results as a raw file for later analysis.

As described earlier, each of the tools listed here have advantages and disadvantages over each other. Therefore, having an in-depth study of these tools, such as the amount of (not-wanted) artifacts introduced to the system and memory with usage of these tools, will provide vital information about those software and eventually will help the investigators choose the tool that satisfies their specific needs.

Background and Related Work

Digital forensic investigators know that learning to use a forensic software is usually not easily learn in short period of time (17). Learning how to use commercial or freeware/open source tools will depend on users' experience with computers and his or her explorer identity. For example, authors in (17) mention that a user with Linux experience will be able to learn open source tools faster than a user with experience only on Windows systems. To minimize this learning curve, digital investigators and other users should be trained and be familiar with different systems to select and use the forensic tools proficiently. Moreover, commercial and open source tools can differ in functionality and complexity, as well as in cost (18). Most of the market leading commercial tools cost thousands of dollars. On the other hand, freeware/open source tools are free.

A number of researchers in their work compared different forensic tools used to analyze the acquired data from systems. For instance, Manson et al. (17), made comparisons using three software forensic tools, namely Encase, FTK, and Sleuth Kit. In their work, FTK Imager was used to obtain the image of the compromised system. Then, these three tools were compared based on their capabilities when analyzing the obtained evidence. Although all of them were able to identify and reflect forensically important information, the timeframe of importing the image to FTK was slower compared to Sleuth Kit and Encase as the information gets indexed by FTK (17). According to the authors, the graphical user interface (GUI) used for FTK allow users to work in their analysis without the need of spending a lot of time in training. Encase has additional search features such as EnScript commands and string conditions which allow Encase to search data rapidly and efficiently (18). However, users need a considerable amount of training time to be effectively using the commands.

To evaluate the usability of commercial and open source digital forensic tools, authors in (18) conducted interviews and surveys with forensic experts to identify what digital forensic tools are preferred by digital investigators and users. According to the majority of the participants, open source tools are used when commercial tools do not do certain functions better. About 31% of the interviewed experts recognized that they are not familiar with open source tools, 29% said that their organizations have bought licenses for commercial tools, and 20% said that open source tools do not provide user support (18). For the use of graphical user interface or command line tool type question, most of the users indicated their preference on the tools that provide user interface. In addition, almost 50% of the interviewees said that they do not know programming languages (18). On the other hand, experienced programmers prefer to use of tools that incorporates programming and scripting languages. To the reporting and documentation question, experts indicated that they prefer to bookmark comments in the program itself, others prefer writing down the findings on a notepad, and the rest of them type their findings in a word document (18).

Most of the digital forensics tools comparison studies are focused on the environments that can analyze the collected evidence from digital devices. However, there are also a few studies focused on memory acquisition as well. For instance, Thing et al. (19) created an automated system to perform a live memory forensic analysis for mobile phones. The authors created memgrab, a memory acquisition tool to dump the processes running on the memory. The tool performs process trace system call for tracing the processes by controlling its execution and accessing to its address space (19). Schuster reflected the importance of memory acquisition process and their effects on the memory contents. He analyzed the pool allocation mechanism of the Microsoft Windows operating system (20). The author found that allocations from the nonpaged pool are reused based on their size and a last-in first-out schedule (20).

Sans Institute InfoSec Reading Room also published detailed report about the techniques and tools for recovering and analyzing data from volatile memory (9). Although the work itself is very valuable, the comparison metrics used on the tools are very limited and the document only provides general information about the tools that can mostly work on Linux- or older Windows-based systems. On the other hand, the closest study to our work was conducted by Sutherland et al. (5). The authors described the importance of memory acquisition tools and compared them in detail, such as memory usage, registry keys, and dynamic link libraries (DLLs) as well. However, the tools that were tested were particularly designed to work on 32-bit Windows XP machines; thus, except ProDiscover either most of them are not compatible to work post 64-bit Windows XP machines or some of them are not widely used anymore. Therefore, our study, here, is hoping to enhance this work by increasing the comparison metrics and also reflect the most recent tools that can work on newer Windows operating systems.

Analysis and Results

In this investigation, a metrics was created for evaluation of the tools. Specifically, user interface capabilities, platform limitations, reporting capabilities, total execution time, occupied memory, shared and proprietary DLLs invoked, modified registry keys, and invoked files during the processing were evaluated by the use of the discussed memory acquisition tools. This selected metrics can be used by digital investigators or other professionals when they are required to acquire a physical memory image. Before getting into the details of the analysis and results, information on the survey and the test environment is shared.

Survey

As mentioned earlier, a survey was conducted among the practitioners of digital investigations to learn their preference of the live memory acquisition software. This survey (IRB #22703) was sent to total of 59 computer forensics laboratories or commercial companies specializes in Computer Forensics in the USA and we have received 41 responses serving nation and worldwide. In the survey, we specifically asked for the investigator to provide their order of preference/usage among memory acquisition tools on 64-bit Windows operating system. According to the results, we received from the participants, FTK Imager, ProDiscover, and Memoryze are widely employed and

preferred products by digital investigation organizations as shown in Fig. 1.

Test Environment

As, in a real-life scenario, the discussed memory acquisition software will be installed to a local machine for obtaining the content of the memory, it is desirable to conduct the testing on a physical device only. However, virtual environment also gives us the flexibility of testing each of these tools in a fully controlled environment. In detail, we were able to test the named software in a limited hardware characteristics, which can be faced in a regular investigation, compared to our physical device's specification. With this way, we hope to enlighten the community about performance comparisons of these memory acquisition tools with different hardware characteristics.

First, the discussed seven forensic tools were tested to acquire RAM content from a virtual machine running Microsoft Windows 7 Professional 64-bit operating system. The hardware characteristics of this virtual machine were as follows: 2 GB of physical memory (RAM), 2 processor cores at 3.10 GHz, and a 50 GB virtual hard disk. Besides the virtual machine, we also conducted the same case study in a physical device that is also running Microsoft Windows 7 Professional 64-bit operating system. The computer had 4 GB of physical memory, 2 processor cores at 3 GHz, and a 512 GB of hard disk. Both the virtual machine and physical devices were only executing the default Windows 7 processes to have common baseline when conducting the analysis. Moreover, both platforms were not connected to the Internet to prevent any possible computer state changes in the memory which can be caused by Internet services.

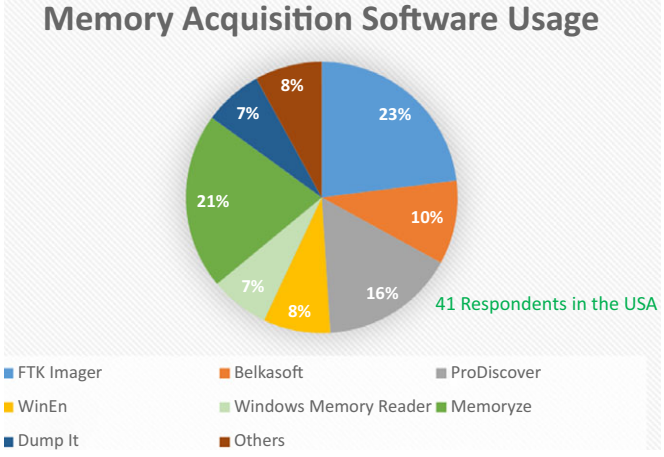

FIG. 1—*Memory acquisition software usage.*

User Interface

Hibshi et al. (18) mentioned that most users prefer a GUI interface environment and avoid the use of the command line tools on multipurpose forensic software. However, still command line tools are used in certain circumstances where there is no way to conduct a task with a software that has GUI.

As shown in Table 1, out of the selected tools, FTK Imager, Belkasoft Live Ram Capturer, and ProDiscover have a graphical user interface. The GUIs have a menu bar which is available at any time. For example, FTK Imager's "File" menu has an option called "Capture Memory" to obtain the RAM memory content. Belkasoft brings up a dialog box to specify the location to save the results and has a button called "Capture" that is used to start the memory acquisition process. ProDiscover uses its "Action" menu to capture the physical memory. On the other hand, digital investigators will need to know the specific commands to execute WinEn, Windows Memory Reader, Memoryze, or DumpIt, to acquire the physical memory content. For example, the command *memorydd.bat –output c:\results* is used for Memoryze and the command *wmr.exe –D –H SHA-256 C:\results* can used with Windows Memory Reader.

In their interview with experts, authors in (18) reveal that users need skills and knowledge to use forensics tools and also state that no investigators are able to conduct a digital investigation without taking training. In addition, 68% of the interviewers prefer intensive training and 24% prefer tools with GUI interfaces because those require less training (18). However, compared to a complete forensic tool, the acquisition of the physical memory is very straight forward and not difficult. As we reflected above, minimum training can be considered enough to obtain the physical memory content no matter whether the tool is running with a GUI or command line interface. It is clear that using the GUI is easier compared to the command line tools. However, the required commands for command line tools are kept in minimum to avoid any complexity from the user side. Therefore, minimum training should be sufficient enough to use any of these tools.

Operating System

Out of the selected tools, Belkasoft, ProDiscover, WinEn, Windows Memory Reader, Memoryze, and DumpIt are designed to operate on the Microsoft Windows platform, both in 32- and 64-bit operating systems, as shown in Table 1. On the other hand, FTK Imager works on multiple platforms.

Reporting Capabilities

The reporting capabilities are critical in a digital investigation. Therefore, when choosing a tool, users might prefer live

TABLE 1—*User interface and operating system compatibility.*

Tool	User Interface	32-bit XP	32-bit W7	64-bit W7	32-bit W8	64-bit W8	Linux
FTK Imager	GUI	✓	✓	✓	✓	✓	✓
Belkasoft	GUI	✓	✓	✓	✓	✓	×
ProDiscover	GUI	✓	✓	✓	✓	✓	×
WinEn	Command line	✓	✓	✓	✓	✓	×
Windows Memory Reader	Command line	✓	✓	✓	✓	✓	×
Memoryze	Command line	✓	✓	✓	✓	✓	×
DumpIt	Command line	✓	✓	✓	✓	✓	×

memory acquisition tools with built-in reporting features. However, reports might not necessarily have the information needed by the users. Out of the tested software, the largest reporting data are generated by ProDiscover. During the execution of ProDiscover forensic tool, it shows a report which includes information such as project number, project description, image files, disks, evidence of interest, clusters of interest, file signature mismatch, registry keys of interest, event log entries of interest, Internet activity information, search results, project notes, and physical page size that a digital investigator may need to document. Once the RAM image is acquired, additional information is shown, such as the acquisition start–finish times and dates, the time that the tool took to complete the process as well as the image file size. Memoryze does not create a report like ProDiscover. In contrast, Memoryze creates a folder called audit to store XML files about the batch results such as the issues found on the memory image. On the other hand, FTK Imager does not generate a report. FTK (AccessData's Forensic Toolkit) is needed to create a customize report from the memory image file (10). Rest of the tools, Belkasoft, Winen, Windows Memory Reader, and DumpIt do not generate a report from the memory acquisition. However, no matter which tool is selected, memory analysis software, such as Volatility (21) can be used to create reports from the memory image files.

Processing Time

As shown in Fig. 2, we can conclude that digital forensic tools that are based on command line interface are generally faster in obtaining the physical memory than the forensic tools with graphical user interfaces. For example, in physical device, the required time to acquire 4 GB of physical memory using DumpIt tool was 34 sec, Memoryze took 50 sec, Windows memory reader processed in 60 sec, and it was 47 sec for WinEn. In contrast, GUI tools such as Pro Discover took 114 sec and FTK Imager took 68 sec to obtain the 4 GB RAM contents. However, another GUI type of program, Belkasoft only took 33 sec to process the 4 GB of RAM. Although the numbers are less, we see similar ranking among the tools on the virtual machine setting as well (Fig. 2). There are a couple of reasons why Belkasoft process the task quickly: (i) executing less number of DLLs and modifying small number of registry keys and files, (ii) running in kernel mode so the processes have higher priority, while most of the other tools run in user mode resulting in slower output.

Occupied RAM

While the tools are being executed by the investigator, each of them will use a portion of the memory to conduct its task. With these left (not-wanted) artifacts on the memory, it is possible to lose other important clues from it. Thus, it may yield to an incomplete recovery of the memory contents. Therefore, the amount of Megabytes occupied in the memory is also vital when selecting the tool to be used in an investigation.

The occupied RAM results differ slightly between the virtual machine environment and physical drive as shown in Fig. 3. Except Memoryze, the rest of the command line tools used less memory than the GUI version ones during our tests. Memoryze occupied the greatest amount of memory, due to its multithreading and implementation of its output. For instance, it opens a second terminal after the execution to display a colorful display which leads to the use of more memory. Throughout the imaging process, the memory usage of the tools remained fairly constant with a single small size peak, which may be attributed to the calculation of hash values for the created image file. Overall, we can conclude that the risk of losing important clues when using memory intensive GUI applications and Memoryze are greater than using command line applications.

Loaded DLLs

According to the results obtained in our analysis, we can conclude that digital forensic tools that are based on a command line interface or light weight GUI ones loaded the least amount of DLLs when obtaining the physical memory compared to the forensic tools with a graphically intensive user interfaces as shown in Fig. 4. Forensic tools that use a command line interface do not require the amount of supported files and DLLs as GUI tools with the same functionality. Belkasoft, although being a GUI tool, does not invoke as many files or DLLs as the other GUI tools because its only functionality is to acquire a forensic image. The other GUI tools offer a wide range of functionality and dynamic user interface; however, they invoke more DLLs and files than any of the others. If the tools are relied on DLLs, this can pose a problem when there are concerns that the operating system may have been compromised.

Modified Windows Registry Keys and Invoked Files

While forensic tools are being executed by investigators, each of them makes changes to the Windows registry keys. According

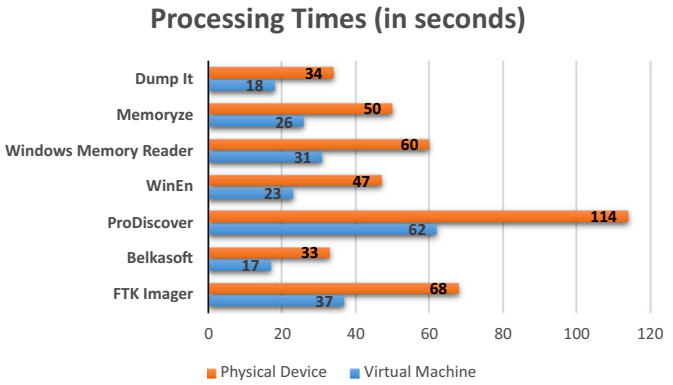

FIG. 2—*Processing times (in seconds).*

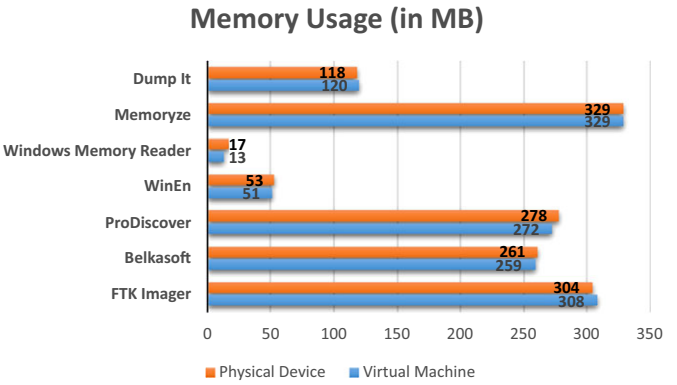

FIG. 3—*Memory usage (in MB).*

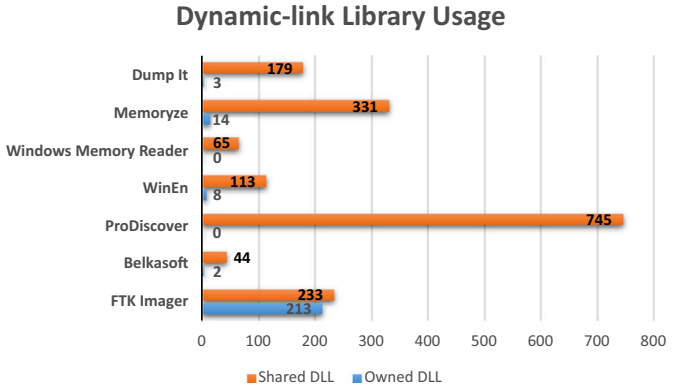

FIG. 4—*Dynamic-link library usage (virtual machine and physical device).*

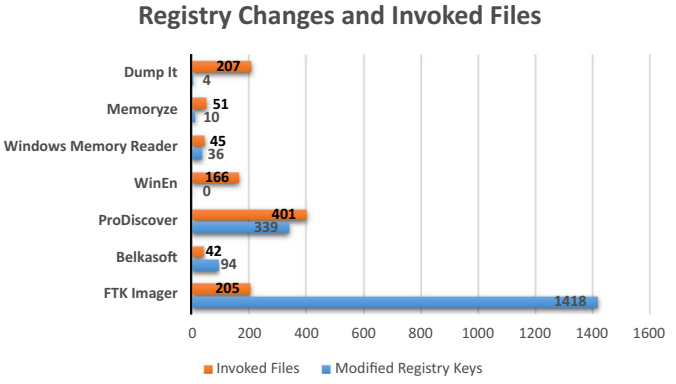

FIG. 5—*Registry changes and invoked files (virtual machine and physical device).*

to our test, GUI forensic tools changed more registry keys than the command line-based tools as shown in Fig. 5. Moreover, while the tools are being executed by the investigator, each of them will also invoke resident files located in the hard drive. With these invoked files on the system, the investigators not only adding artifacts to the system but also leading to possible loss of other important clues from it. Therefore, the number of invoked files is also vital when selecting the tool to be used for the investigation. As reflected in Fig. 5, Belkasoft and the command line tools invoke the fewer number of files in the computer system than the GUI-type applications.

Discussion and Conclusion

The importance of acquiring volatile data is critical because it can contain information not available during a dead forensic analysis. Most digital investigators and users may be concerned about the risk of losing the clue due to the artifacts or fingerprints left by forensic tools. This study presented the risk assessment of forensics tools when acquiring data from physical memory with a side-by-side comparison.

Although GUI tools are easier to learn than the command line tools, as we have discussed earlier the command line-based tools can be fully operated with a single line command written in command line prompt. Therefore, employing a user interface may not be a tool selection criterion for an investigator who has some primitive digital investigation skills. However, Belkasoft scored in the range of the command line tools although lacking in any functionality other than memory image acquisition. The command line tool, Windows Memory Reader, had a better score and offers much more functionality. The investigator should work with ProDiscover if its generated report is crucial for the investigation. Processing time should also be one of the main distinguishing attribute when selecting the tool. Although one can argue that the time difference among the tools are not distant, we need to keep in mind that these results are obtained from 2 and 4 GB of physical memories. While the current standards are pushing for 16–32 GB, the processing times that we show in the Fig. 2 will at least linearly increase. Also, although the size of the hard drive is not playing an important role in the processing time, the read–write speeds of the hard drive will have positive or negative influence on it. If the main goal of the investigation is to contain or trace an attack, then even a couple of seconds can be important to obtain as much as information from the system. Therefore, an investigator should adjust his selection of the tool based on the crime presence.

All seven forensic tools analyzed in this study left fingerprints. The portion of RAM used when the tool is being executed differs from each tool. Approximately, FTK Imager and Pro Discover left 10 times more artifacts compared to Belkasoft and Windows Memory Reader, 8 times more than WinEn, and 5 times more than DumpIt and Memoryze. These artifacts can overwrite forensically important content on the RAM, which will negatively affect the investigation. Except Belkasoft, the number of DLLs loaded from command line type tools is less than the number of DLLs loaded from the GUI-type ones. The same conclusions can be made for the modified Windows registry keys and invoked resident files located at the hard drive when the tools are being executed.

As mentioned earlier, Internet connection was not present when conducting the experiments. There are a couple of main reasons why investigators turn off Internet connection when obtaining RAM contents these are as follows: (i) threat of losing valuable data from RAM because of background activities of other Internet-demanded applications; (ii) some Trojans allow the attacker to view the computer screen in real time. At a minimum, the computer's Internet connection must be disabled so that information is not sent to the attacker. Moreover, for our investigation, it was vital to make a complete and correct comparison of the acquired images while the computer was always in the same state. Besides our Internet-less work, to test the completeness of the tools, later we have also enabled the Internet connection and opened a number of Web browsers, Web applications, and VoIP applications, such as Skype while obtaining the content of the RAM. After collecting the memory images with using the mentioned seven tools, the files were imported to Volatility (21), PoolTools (22), and Winhex (23) to check the contents of the memory images. From all the memory images, we were able to gather the same information about network connections, list of running processes, user names, such as the ones used for Facebook and Skype, loaded DLLs, and open files for a process. Thus, from a digital investigation point of view, the obtained information from all these tools can provide the same information.

Based on the results we have shown, users or digital investigators may choose a tool based on the number of fingerprints or artifacts that each tool introduces to the system. Therefore, employing a command line application or simple GUI application, like Belkasoft, as the volatile data acquisition tool in an investigation is crucial to retain the most data from the memory. Unless it is more important to track the attacker than to preserve the digital information, the Internet connection should be turned

off when collecting the volatile memory. After collecting the memory image, the data can be analyzed with a variety of memory analysis tools, such as the ones mentioned above.

As a future work, this study can be expandable to include other memory image acquisition tools used in different platforms, such as *LiMe* for Linux and *Mac Memory Reader* for Mac OS X, and also other Windows operating system-based tools, such as *windd*. Moreover, this study can be expandable to evaluate the mass remote memory acquisition tools.

References

1. Noureldin SH, Hashem S, Abdalla S. Computer forensics guidance model with cases study. Proceedings of the 2011 Third International Conference on Multimedia Information Networking and Security (MINES '11); November 4–6; Shangai, China. Washington, DC: IEEE Computer Society, 2011;564–71.
2. Varol C, Bayrak C. Estimation of quality of service in spelling correction using Kullback–Leibler divergence. Exp Syst Appl 2011;38(5):6307–12.
3. Austin RD. Digital forensics on the cheap: teaching forensics using open source tools. Proceedings of the 4th Annual Conference on Information Security Curriculum Development Conference; September 28–29; Kennesaw, GA. New York, NY: ACM, 2007.
4. Schatz B. BodySnatcher: towards reliable volatile memory acquisition by software. Digit Investig 2007;4:126–34.
5. Sutherland I, Evan J, Tryfonas T, Blyth A. Acquiring volatile operating system data tools and techniques. ACM SIGOPS Oper Syst Rev 2008;42(3):65–73.
6. Vidas TM. The acquisition and analysis of random access memory. J Digit Forensic Pract 2006;1(4):315–23.
7. Blanchard A. Former UW student faces jail time of child porn possession; http://badgerherald.com/news/2011/01/17/former-uw-student-fa/#.U95qcvldW18 (accessed December 4, 2014).
8. Columbia Pictures v. Bunnell; http://law.lexisnexis.com/litigation-news/articles/article.aspx?groupid=eQSqfLggRQQ=&article=G3e/MZEnBrM= (accessed December 4, 2014).
9. Amari K. Techniques and tools for recovering and analyzing data from volatile memory, SANS Institute, 2009; http://www.sans.org/reading-room/whitepapers/forensics/techniques-tools-recovering-analyzing-data-volatile-memory-33049 (accessed September 3, 2014).
10. AccessData Software Forensic Toolkit (FTK), http://www.accessdata.com (accessed August 6, 2014).
11. Live Ram Capturer, http://forensic.belkasoft.com/en/ec (accessed August 6, 2014).
12. ProDiscover Forensics. http://www.techpathways.com/ (accessed August 6, 2014).
13. Winen, https://www.guidancesoftware.com/ (accessed August 6, 2014).
14. Windows Memory Reader, http://www.atcorp.com/ (accessed August 6, 2014).
15. Software Downloads Memoryze, http://www.mandiant.com/resources/download/memoryze (accessed August 6, 2014).
16. DumpIt, http://www.moonsols.com/ (accessed August 6, 2014).
17. Manson D, Carlin A, Ramos S, Gyger A, Kaufman M, Treichelt J. Is the open way a better way? Digital forensics using open source tools. Proceedings of the 40th Hawaii International Conference on Systems Sciences (HICSS '07); January 3–6; Waikoloa, HI. Washington, DC: IEEE Computer Society, 2007;266b.
18. Hibshi H, Vidas T, Cranor LF. Usability of forensics tools: a user study. Proceedings of the 2011 Sixth International Conference on IT Security Incident Management and IT Forensics; May 10–12; Stuttgart, Germany. Piscataway, NJ: IEEE Explore, 2011;81–91.
19. Thing VLL, Kian-Yong N, Chang EC. Live memory forensics of mobile phones. Digit Investig 2010;7:74–82.
20. Schuster A. The impact of Microsoft Windows pool allocation strategies on memory forensics. Digit Investig 2008;5:58–64.
21. Volatility, http://code.google.com/p/volatility/ (accessed December 13, 2014).
22. PoolTools, http://computer.forensikblog.de/en/2007/11/pooltools-version-130.html (accessed December 14, 2014).
23. Winhex, http://www.x-ways.net/winhex/ (accessed December 14, 2014).

Additional information and reprint requests:
Cihan Varol, Ph.D.
Computer Science Department
Sam Houston State University
AB1 216F
Huntsville, TX 77341
E-mail: cvarol@gmail.com