Journal of Forensic Research and Criminal Investigation
JFRCI, 3(2): 57-63
www.scitcentral.com

Review Article: Open Access

Impact of Tools on the Acquisition of RAM Memory

Marcos Fuentes Martínez*

*Guardia Civil, Spain.

Received November 11, 2020; Revised December 09, 2020; Accepted January 01, 2021

ABSTRACT

When responding to a security incident in a system, a number of basic principles must be followed regarding the collection of evidences of the system. The capture of these evidences has to be done according to its order of volatility. In this sense, RAM memory constitutes the most important element to capture, given its extreme volatility. RAM memory must be acquired and analyzed because the data it holds, which may belong to the system itself or to any other device connected to it, can survive a certain amount of time in it. Since RAM is a constantly changing element, it must be stranded out that any action carried on the system under analysis will modify the contents of the RAM. In this article a comparative and an objective analysis has been carried out, showing the impact that the execution of some tools for the capture of RAM has on the system. This comparative study details both the private shared workspaces, for each of the processes executed by each of the tools used.

Keywords: DFIR, Digital Forensics, Incident response, RAM Memory, Windows, Impact of tools

INTRODUCTION

When acting on a security incident of any kind, document RFC 3227 (1), which sets out basic guidelines for action, must be taken into account. Among many other interesting aspects, as the first person who can intervene a System, are the guiding principles during evidence collection, which says that evidence should be collected from the most volatile to the least volatile, specifying this in point 2.1, on the order of volatility.

When intervening in a living system for subsequent analysis, the first technical action to be carried out is to dump the RAM. Due to the extreme volatility of RAM, its acquisition is a fundamental phase in the evidence collection.

It should be remembered that, in any case, RAM can and should be captured and analyzed because, sometimes, studying the non-volatile data will not be enough.

When studying a forensic image of a RAM, one plays with a certain advantage in the analysis. In the RAM there may bed at a that correspond to other data stored on the hard disk, or other types of data may be found, called an onymous data, which are not stored on the hard disk. Therefore, action must be taken quickly, altering, as little as possible, the RAM you want to capture.

The more time that passes, as well as the more activity that has taken place in the system, the less options exist to be able to recover useful information from the RAM memory because, this one, is constantly changing. Therefore, when using tools to capture RAM memory in alive system, it is altered and, therefore, the image of RAM memory that is being acquired is also altered. Because RAM memory does not freeze. Data can survive in RAM for a certain time.

For example, an image file that has been opened on a system from an external device could be recovered, with them eta data properties, through thumb nails or, even, fragmented; a carving can be made on the forensic image in the RAM, or, more effectively, on its memory pages; a timeline of this element can be carried out; hashes and/or credentials that are in use can be obtained from, for example, encrypted content in the System; Windows Registry hives, Event Logs, etc., can be exported; processes in execution, historical processes, hidden processes or network connections can be seen, just to give some examples. Everything will depend on the time elapsed since the action of the incident and the actions carried out in the System after wards. Very interesting and valuable data can be extracted from the RAM, which could consist of a type of information that Is key to the satisfactory resolution of the case. Depending on the type of case, it could even be solved with the analysis of the RAM memory.

Without a doubt, each scenario must be valued because, each one of them, will present its peculiarities and characteristics.

Corresponding author: Marcos Fuentes Martínez, Guardia Civil, Spain, E-mail: n4rr34n6@protonmail.com

Citation: Martínez MF. (2021) Impact of Tools on the Acquisition of RAM Memory. J Forensic Res Criminal Investig, 3(2): 57-63.

For example, it is not the same case, nor does it require the same study, the analysis of an email header as a case where child pornography or terrorism content is found.

But, before starting to analyse the RAM, it must be captured. For this reason, it is vital to choose the right tool to use to capture as much useful information as possible.

Many articles have spoken, on many occasions, of the existing tools available to carry out such an action, explaining its basic operation. But nothing has been said about the impact that the execution of these tools has on the RAM that we want to acquire.

METHODS

Since the aim of acquiring RAM in a system is to collect as much useful information as possible, and since the integrityof that memory must be maintained in the best possible way, this study has been carried out, to show the impact of some of the most frequently used tools on the acquisition of RAM.

The data exposed shows the resource consumption of the memory itself, in its private space and in its shared space, for each of the processes executed by each tool, and the RAM acquisition time. Both factors, running processes and time, are key elements.

Two tests were carried out for this study, using two versions of Windows 10. The first system consists of a Windows 10, in its compilation number 17763.253, with an allocated RAM of 4,096 MB. This System has been downloaded from the official Microsoftsite (2) virtualized under Virtual Box (3) . The second system consists of a Windows 10, in its compilation number 17763.292, being a physical system, without virtualization, that has 15,306 MB of RAM memory.

To monitor the processes running on the various RAM acquisition tools, the 'Process Explorer' (4 5 6) tool has been chosen, in its version 16.22, which is available on Microsoft's official website (7).

The values that have been taken into account as a reference are those related to the private workspace, which consists of the memory dedicated to that monitored process, and which is not shared with other processes, as well as the workspace that is shared with other processes. This size is measured in kilobytes.

The reference values that have been taken into account for the execution times are those relating to the time marks corresponding to the creation and modification of the forensic image of the RAM, because the file is created at the same time as the dump of the RAM begins and is last modified when the last data is recorded, this is, the last bit.

In order not to lose any detail during the acquisition of the dumps, it has been decided to record the whole process on video, using the 'Record that' function of the 'Game Bar', which incorporates Windows 10 system (8).

Regarding the tool s tested, it has been decided to use some that are free of charge and more widely used, such as those listed below:

Belkasoft Live RAM Capturer (9-10).

DumpIT, in its version 3.0.20190124.1 (11).

FTK Imager Lite, in its version 3.1.1 (12).

Magnet RAM Capture, in its version 1.1.2 (13).

Memorize, in its version 3.0 (14).

Winpmem, in its version 3.2 (15).

Belkasoft Live RAM Capturer

During the acquisition of the RAM memory with this tool, two processes has been executed: The 'Ram Capture64.exe' process, as the parent process, and a child process 'con host.exe'. The 'conhost.exe' process is responsible for opening instances for each Windows console. That is, for each Windows console that is opened, a process 'conhost.exe' will appear.

The 'Ram Capture64.exe' process has presented a range of consumption values, in its private space, from 1,872-1,988, as minimum and maximum values. In its shared memory it has oscillated between 1,1476-11,672.

The process 'conhost.exe' has presented a range of consumption values, in its private space, of 7,260-7,344, as minimum and maximum values. In its shared memory it has oscillated between 16,768-16,816.

The time it took this tool to acquire the complete memory of the system was 3.150000003 minutes, as it can be seen from the time stamps relating to the creation and modification of the memory image.

DumpIt

This tool can be executed in two different ways: directly from the executable itself, or from a cmd console, where some parameters can be configured. Depending on how the tool is executed, some values can be found or others.

If this tool is executed from the executable itself, two processes can be found: 'DumpIT.exe', as the parent process, and a child process 'conhost.exe'.

The process 'DumpIT.exe' has presented a range of consumption values, in its private space, from 1,644-1,988, as minimum and maximum values. In its shared memory it has oscillated between 8,980-9,028.

The process 'conhost.exe' has presented a range of consumption values, in its private space, of 7,104-7,296, as

minimum and maximum values. In its shared memory it has oscillated between 17,132-17,200.

The time it took this tool to acquire the complete RAM memory of the system, with this type of execution, was 7.716666658 minutes, as it can be seen in the time stamps relating to the creation and modification of the memory image.

However, if this tool is executed from the cmd console, where some parameters can be configured, the parent process 'cmd.exe' can be seen, with its child process 'conhost.exe', and the parent process 'DumpIT.exe', with its child process 'conhost.exe'.

The 'cmd.exe' process has presented a range of consumption values, in its private space, of 6,108-11,484, as minimum and maximum values. In its shared memory it has oscillated between 14,460-16,776.

The process 'conhost.exe', dependent on the process 'cmd.exe', has presented a range of consumption values, in its private space, of 8,520-8,524, as minimum and maximum values. In its shared memory it has oscillated between 22,180-22,284.

The process 'DumpIT.exe' has presented a range of consumption values, in its private space, of 1,688-1,776, as minimum and maximum values. In its shared memory it has oscillated between 8,900-8,988.

The process 'conhost.exe' has presented a range of consumption values, in its private space, of 7,192-7,280, as minimum and maximum values. In its shared memory it has oscillated between 16,540-17,012.

The time it took this tool to acquire the complete RAM of the system, with this type of execution, was 4.8666662 minutes, as it can be seen in the time stamps relating to the creation and modification of the memory image.

FTK Imager Lite

During the acquisition of the RAM with this tool, a single process called 'FTK Imager.exe' was executed.

This process has presented a range of consumption values, in its private space, of 21,588-22,024, as minimum and maximum values. In its shared memory it has oscillated between 50,764-51,744.

The time it took this tool to acquire the complete RAM memory of the system was 3.5833333 minutes, as it can be seen from the time stamps relating to the creation and modification of the memory image.

Magnet RAM Capture

During the acquisition of the RAM with this tool, a single process called 'MagnetRAMCapture.exe' has been executed.

This process has presented a range of consumption values, in its private space, of 9,656-10,484, as minimum and maximum values. In its shared memory it has oscillated between 32,812-34,296.

The time it took this tool to acquire the full RAM of the system was 4.066666664 minutes, as it can be seen from the time stamps relating to the creation and modification of the memory image.

Memoryze

This tool is executed through the cmd console, so the following processes were presented during the acquisition: a parent process 'cmd.exe' with a child 'conhost.exe' process, and a parent process 'Memorize.exe' with a child 'conhost.exe' process. In addition to these processes, a 'netsh.exe' process is presented at the end of the acquisition, which is a command line utility, dependent on the 'Memoryze.exe' process.

The 'cmd.exe' process has presented a range of consumption values, in its private space, of 5,708-11,020, as minimum and maximum values. In its shared memory it has oscillated between 14,512-16,320.

The process 'conhost.exe', dependent on the process 'cmd.exe', has presented a range of consumption values, in its private space, of 7,604-7,688, as minimum and maximum values. In its shared memory it has oscillated between 19,904-20,032.

The 'Memoryze.exe' process has presented a range of consumption values, in its private space, of 3,616-3,644, as minimum and maximum values. In its shared memory it has oscillated between 11,860-11,960.

The process 'conhost.exe' has presented a range of consumption values, in its private space, of 7,276-8,180, as minimum and maximum values. In its shared memory it has oscillated between 16,804-17,924.

The process 'netsh.exe', has presented a memory consumption, in its private space, of 980. In its shared memory it has presented a value of 120.

The time that this tool took to acquire the complete RAM memory of the system, with this type of execution, was 6.583333332 minutes, as it can be seen in the time stamps relating to the creation and modification of the memory image.

Winpmem

This tool, which is executed via the cmd command line, will have a parent process 'cmd.exe', with two dependent processes: 'conhost.exe' and 'winpmem_3.2.exe'.

The 'cmd.exe' process has presented a range of consumption values, in its private space, of 2,828-5,920, as minimum and maximum values. In its shared memory it has oscillated between 4,944-4,992.

The process 'conhost.exe', dependent on the process 'cmd.exe', has presented a range of consumption values, in its private space, of 7,552-7,640, as minimum and maximum values. In its shared memory it has oscillated between 20,016-20,064.

The process 'winpmem_3.2.exe' has presented a range of consumption values, in its private space, from 1,840-3,824, as minimum and maximum values. In its shared memory it has oscillated between 6,720-8,948.

The time it took this tool to acquire the complete RAM memory of the system was 5.116666668 minutes, as it can be seen from the time stamps relating to the creation and modification of the memory image.

This tool allows the capture of memory using the network, which can be carried out using the Net catutility, but this would mean setting up another extraordinary process under the name 'nc.exe' that would have a memory consumption, in its private workspace, of about 624Kb.

RESULTS AND DISCUSSION

Any tool for RAM memory acquisition will always dump the entire memory of the system. All the files generated will have the same size unless they are compressed or splitted.

In the case of performing the acquisition process with the Winpmem tool, the resulting file will be larger than the system's memory because, in addition to this, it extracts and acquires other types of data. For this reason, the resulting file will be a '.zip' file, which is a container that cannot be directly analysed and which must be decompressed. Inside it, the image of the physical memory is found under the name of 'Physical Memory'.

It has been commented in some articles that, some tools, give problems with RAM sizes over 8 GB. This is not true. The main problem that exist s is that the memory profile is not identified correctly The forensic image profile of the RAM must be. correctly identified before proceeding with the analysis of the memory. All the memory images created with the tools shown in this study can be analysed with the appropriate tools, such as Volatility.

Objective data: Acquisition times

The objective data of the tests they have carried out are set out below. The first data to be presented will be the one relating to time. The time, established in seconds, that a tool takes to acquire the System's RAM memory.

As it can be seen **(Figure 1)**, in the tests run, the fastest tool has been Belkasoft Live RAM Capturer, while the slowest has been DumpIT, running from the command prompt, where a format type and output path were specified. However, the DumpIT tool, if executed directly, without using the Command Prompt, is not the slowest, leaving that position to Memoryze. The difference between the fastest and slowest tool is 274 seconds.

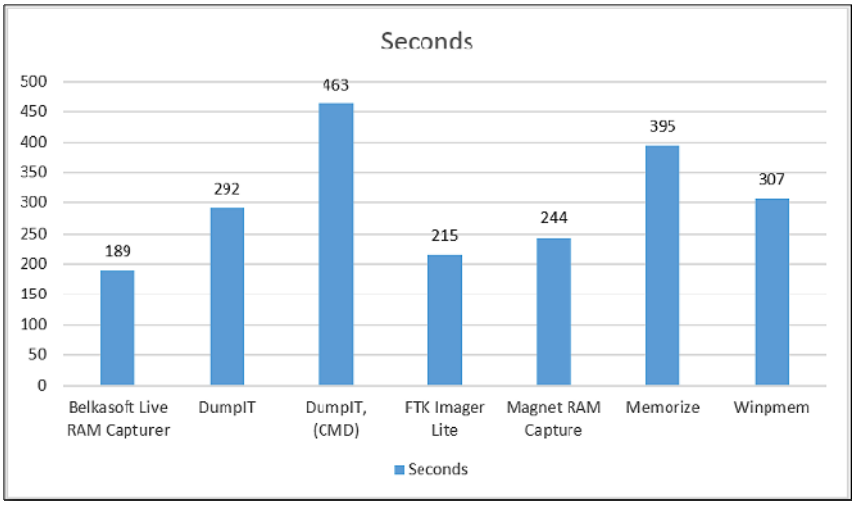

Figure 1. Time, in seconds, used by each of the tools used.

As mentioned at the beginning of this article, RAM memory is constantly changing. In other words, it contains highly volatile information. Therefore, the 274 second difference between the fastest and slowest tool is a very long time. With the course of this time, the possibilities of recovering elements of interest decrease. Elements that, with a proper intervention, could be found in the RAM memory.

Objective data: RAM memory consumption

Below are the objective data regarding RAM consumption, in its private workspace, for each of the tools tested. This consumption is measured in Kilobytes.

Because memory is constantly changing, processes will never have a single value. And they will not even present the same range of values in another similar execution.

In the tests carried out **(Figure 2)**, the tool that has used the fewest private resources has been DumpIT, with a minimum value of 1,644 Kilobytes, compared to the 21,588 Kilobytes used by FTK Imager Lite. Even at maximum values, the DumpIT tool consumes few er resources, with a maximum value of 1,988 Kilobytes (16), as opposed to the 22,024 Kilobytes maximum value of FTK Imager Lite. The difference between the two minimum values is 19,944 Kilobytes. A lot of information can be found in this space. Vital information that could be lost by not thinking about that consumption, in that size.

Maybe it could be thought, and believe, that these values, this comparison, are enough to determine whether to choose one tool or another. But you must also think about the rest of the processes that are executed by the system with each one of the tools, and about the work space shared with other processes. For this reason, the information corresponding to this data is also presented, where the values obtained with the total sum of the consumption of each of the tools are shown below.

As it can be seen **(Figure 3)**, the tool with the lowest total consumption, in the tests carried out, was DumpIT, with direct execution, without the use of the cmd console, with a value of 24,860 Kilobytes. On the other hand, the tool with the highest consumption is Memoryze, with a total value of 88,264 Kilobytes. A difference in total consumption of 53,404 Kilobytes can be seen. Certainly, a huge amount of information can be saved, found and/or lost, in that workspace, in that size.

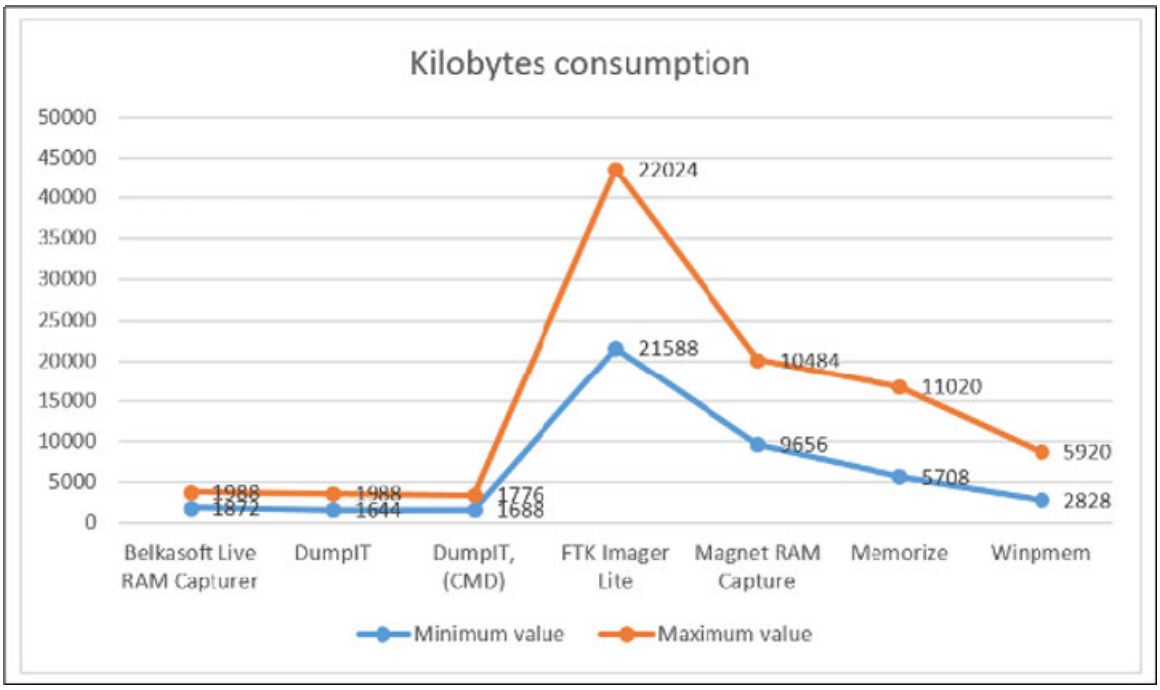

Figure 2. Consumption, measured in Kilobytes, for each of the tools used, in the private workspace.

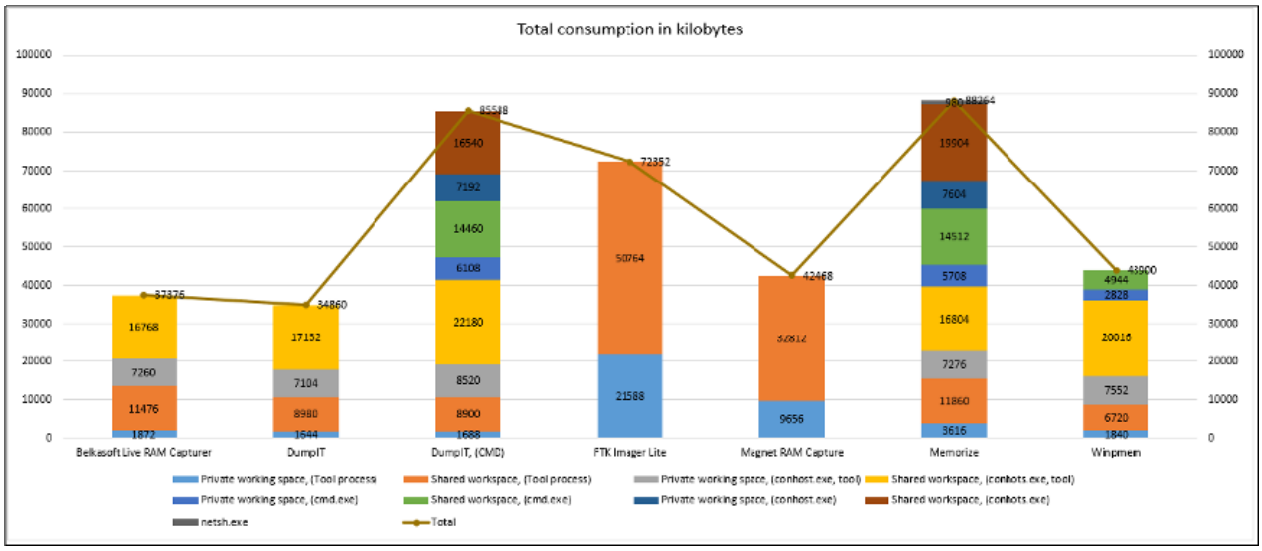

Figure 3. Total RAM memory consumption, measured in Kilobytes, for each of the tools used.

CONCLUSION

In this article it has been presented only some small tests that have been carried out with some of the free RAM acquisition tools and that are considered to be of more extended use. Other similar tests could be carried out with other tools.To name a few examples, this same study could be carried out,comparing as mallutility, such as MDD (17-18), with the OS Forensics suite (19 20 21).

Since RAM is constantly changing, no tool will have a single value, either in terms of resource consumption or time. It will not even display the same range of values in two different executions. It is not possible to obtain two identical RAM dumps. It all depends on the case. Everything depends on the system. Everything depends on what is being executed at that moment.

In my humble opinion, I believe that this study is an excellent way to compare the way in which the different tools works, without making subjective assessments, full of interests or opinions, since it is a question of presenting objective data in a real environment.

Each user can use the tool with which is most comfortable, regardless of which one it is, without taking into consideration what has been seen in this article, or, it can be taken into consideration that, since memory presents very volatile, constantly changing information,one must choose carefully what is going to be executed and how it is going to be executed.

Each user can evaluate only one factor in the use of the tools or can take into account everything that needs to be evaluated: The memory consumption of each of the tools, both in their private and in the shared workspaces, the time that each tool invests in carrying out its function, or the fact that there are tools that provide a final report with information on the memory profile that has been worked on.

The final objective of this work is to show that the tool to be used must be well chosen and that the impact that this tool has on the RAM of the system must be calculated. A memory that is being acquired to carry out a later study on it. A study that contains key information for the resolution of a case. Information that will be lost if the appropriate tool is not used properly.

REFERENCES

1. Brezinski, D., & amp; Killalea, T.(2002). Guidelines for Evidence Collection and Archiving. Available online at: https://www.ietf.org/rfc/rfc3227.txt
2. Get a Windows & nbsp;10 development environment. (2020). Available online at: https://developer.microsoft.com/en-us/windows/downloads/virtual-machines
3. Download VirtualBox. (n.d.). Available online at: https://www.virtualbox.org/wiki/Downloads
4. Russinovich, M. (2011). Mysteries of Memory Management Revealed,with Mark Russinovich (Part 1of 2). Available online at: https://channel9.msdn.com/Events/TechEd/NorthAmerica/2011/WCL405
5. Russinovich, M. (2020). Process Explorer (Version 16.22) [Computer software]. Available online at: https://docs.microsoft.com/en-us/sysinternals/downloads/process-explorer
6. Russinovich, M. (2020). Windows Sysinternals. Available online at: https://docs.microsoft.com/en-

us/sysinternals/

7. Background recording settings in Captures on Windows 10. (2020). Available online at: https://support.microsoft.com/en-gb/help/4027144/windows-10-background-recording-settings-in-captures
8. Belkasoft. (2020).Capture Live RAM Contents with Free Tool from Belkasoft! Available online at: https://belkasoft.com/ram-capturer
9. Suiche, M. (2019). Your favorite Memory Tool kit is back... FOR FREE! Available online at: https://blog.comae.io/your-favorite-memory-toolkit-is-back-f97072d33d5c
10. Suiche, M. (2020). DumpIt (Version 3.0.20190124.1) [Computer software]. Available online at: https://my.comae.com/
11. Access Data. (2020). FTK Imager Lite (Version 3.1.1) [Computer software]. Available online at: https://accessdata.com/product-download/ftk-imager-lite-version-3-1-1
12. Magnet Forensics. (2020). MAGNET RAM Capture (Version 1.1.2) [Computer software]. Available online at: https://www.magnetforensics.com/resources/magnet-ram-capture/
13. Fuentes M (2019). First steps with Volatility. Available online at: https://unminioncurioso.blogspot.com/2019/03/dfir-first-steps-with-volatility.html
14. Fuentes M (2020). OP Tanjawi: Forensic Techniques on Fire - Forensic Analysis to VirtualBox. Available online at: https://unminioncurioso.blogspot.com/2020/03/dfir-op-tanjawi-forensic-techniques-on.html
15. Fire Eye (2020). Memoryze (Version3.0) Computer software. Available online at: https://www.fireeye.com/services/freeware/memoryze.html
16. Cohen, M. (2020).WinPmem (Version 3.2) Computer software. Available online at: http://www.rekall-forensic.com/
17. Cohen,M.(2020).WinPmem. https://rekall.readthedocs.io/en/ghpages/Tools/pmem.html
18. Stotts, B. (2016, February 11). Mdd. Retrieved September 03, 2020, from https://sourceforge.net/projects/mdd/
19. PassMark® Software Pty Ltd. (2020). Pass Mark OS Forensics - Digital Investigation. Retrieved, from https://www.osforensics.com/osforensics.html
20. Mcleanbyron. (2018). Memory Management (Memory Management) - Win32 apps. Available online at: https://docs.microsoft.com/en-us/windows/win32/memory/memory-management
21. Markruss. (2017,). Windows Internals Book - Windows Sysinternals. Retrieved September 03, 2020, from https://docs.microsoft.com/en-us/sysinternals/learn/windows-internals
22. Microsoft Corporation. (2010). Memory Sizing Guidance for Windows 7. Retrieved September 03, 2020, from https://support.microsoft.com/en-us/help/2160852/ram-virtual-memory-pagefile-and-memory-management-in-windows
23. Welcome to Virtual Box.org! (2020). Retrieved September 03, 2020, from https://www.virtualbox.org/
24. "This is a copy edited, publisher-produced PDF of an article published in the International Journal of Cyber Forensics and Advanced Threat Investigations (CFATI): [2020]. Some rights reserved. The definitive publisher-authenticated version
25. Fuentes Martínez, M. (2020). Impact Of Tools On The Acquisition Of RAM Memory. International Journal of Cyber Forensics and Advanced Threat Investigations is available online at [https://conceptechint.net/index.php/CFATI/article/view/12]."