



ForMaLity: Automated FORnsic MAlware Analysis using VolatiLITY

Parag H. Rughani, Ph. D.

Assistant Professor,

IFS, Gujarat Forensic Sciences University,

Gandhinagar (Guj) India

Abstract: Forensic analysis of volatile memory plays a crucial role in cyber crime investigation. It has been observed that when available, RAM dump helps forensic investigators in retrieving many useful information related to a crime. There are variety of tools available for RAM analysis including Volatility, which currently dominates open source RAM forensic tools. It has been experienced that many times forensic investigators do not think possibilities of having a malware in the RAM dump. And, if it is there, still they are not very expert Malware Analysts, so it becomes difficult for them to analyze possible malware in a RAM dump. Availability of tools like Volatility lets forensic investigators identify and correlate various components to conclude whether the crime was carried out using any malware or not. However, use of volatility requires knowledge of command line tool and dynamic as well as static malware analysis. This work is done to assist forensic investigators in detecting and analyzing possible malware from a RAM dump. The work is based on volatility framework and outcome is a single step automated tool which analyzes RAM dump and possible malware residing in that. Final report generated using this tool gives accurate details about possibilities of use of malware in committing a crime.

Keywords: Malware Forensics, Live Memory Forensics, Volatile Memory Analysis, Malware Analysis, Forensic RAM Analysis, Volatility, Automated RAM Forensics

1. INTRODUCTION

“Performing memory forensics has the potential to contribute significantly to any forensic investigation”, stated Kristine Amari in his article[1]. It indeed has become one of the crucial steps in forensic investigation of cyber crimes. Compared to dead box analysis, live forensics is very useful and powerful especially, when it is possible to get dump of RAM from the crime scene. Apart from retrieving and analyzing known artifacts like open ports and connections, running process, loaded dlls, clipboard and session details, passwords and much more, also another important and needed step in live forensics is to check possibilities of having a malware in acquired RAM dump. As this analysis can help in correlating other aspects to confirm whether help of any malware was taken in committing crime or not.

As most of the computers these days are connected to some network or at least to Internet, one cannot ignore possibility of a malware running in the system. Details of such malware running in the system may help in knowing about stolen information, unauthorized access, malicious activities and much more. Liming Cai, et. al. mentioned this importance by stating “We must access the computer system's physical memory to find more important information, such as the intruder's IP address, information about the running malicious program's, processes, worms, trojans and so on” in their paper[2].

Considering importance of live forensics, it is very important for forensic investigators to have expertise in understanding how malware work and how they can be identified from the live memory image or RAM dump. It may not be economically and technically feasible for the incident response team or forensics laboratory to have forensic experts with necessary malware analysis knowledge. Though, there are many commercial and

existing live forensic tools available, but none of them feature automated analysis.

The work mentioned in this paper is based on Volatility Framework[3] and it can be very handy for forensic investigators, especially in detecting malware in live memory image by single command. The work is done by keeping current limitation of Volatility framework in mind. The proposed solution called ForMaLity – automated FORnsic MAlware analysis using volatiLITY, is a user friendly and accurate solution to overcome above problems.

Following sections discuss more on existing solutions and proposed solution with its implementation.

2. RELATED WORK

Malware analysts are working from many years in automating analysis process to ease their work. Except forensic aspects as discussed in this paper, there are many open source sandboxes like cuckoo[4] which provide automation in malware analysis process. Manuel Egele, et. al. discussed various sandboxes and automated malware analysis tools in their survey[5]. While, Michael Bailey, et. al. proposed automated classification and analysis of Internet Malware in their work[6].

But, when we talk about RAM forensics, very little work is done in the automation of existing open source tools, and especially when we talk about automating Volatility Framework, there is reasonably very less work done. Tomer Teller, et. al. proposed a solution based on cuckoo, Volatility and IDA[7] in their paper at Blackhat [8], but it heavily depends on Cuckoo. While Logen, Höfken and Schuba provided a GUI solution as an extension to Volatility in their paper[9], though work proposed by them performs few basic tasks automatically, it still does not perform automatic steps

to find out malware. Another well known tool eVOLVe developed by James Habben[10], is a web based tool. Though eVOLVe provides a graphical user interface to the execution, it is still not completely implemented to carry out Malware Analysis. Further, eVOLVe asks user to pass profile of the image at the time of execution, which indicates that in most of the cases user will need to run Volatility separately to get the profile first and then one can pass it to eVOLVe. While, ForMaLity, proposed in next section overcomes above problems as it detects profile automatically as a part of automating malware process.

3. PROPOSED TOOL

As the demand of malware analysis residing in a RAM dump is increasing, an automated open source solution was expected from the forensics fraternity. The work is done to create such tool for assistance of Digital Forensic Investigators, who are assumed not to be expert malware analysts but are needed to have some mechanism by which they can easily identify presence of any malware in the RAM dump. The knowledge of such malware can help them in correlating various components and eventually can lead to solution of the crime.

The proposed tool called ForMaLity is an extension work to Volatility Framework and is based on python. It is made user friendly to ease extraction and analysis of malware from RAM dump. The biggest advantage of this tool is, user does not need to remember commands, their syntaxes or even when to use which command. This is very handy for those who do not prefer to work on command line utilities because they avoid remembering commands. The tool requires only one argument, full path of dump, for execution. Once execution is started, the tool does not require any input from the user.

The execution begins with determination of image information, which gives details about Operating System (including service packs, if there are any) and architecture (e. g. x86). This details indicate the dump is taken from a machine having which architecture and operating system. Volatility calls it a profile, which is mandatory to execute other volatility commands. As, this is one of the basic requirements of volatility, which user has to save and paste in each command, this not only makes the command lengthy, but it also becomes cumbersome process to paste same profile in each and every time, someone executes a command. Proposed tool ForMaLity overcomes this process as user neither needs to remember this profile for future steps nor he needs to worry how it is identified.

After getting the profile next step is to start malware analysis. For this purpose, ForMaLity first checks open connections at the time RAM was captured. Details of open connection is very useful in malware analysis as most of the

malware (including ransomware) are network based and work as botnet. These malware mostly need to connect to their command and control center either for next order or to send specific information like stolen CC details or passwords or files.

Details gathered in previous step by proposed tool contain list of processes which are communicating to IP addresses with the ports used for the communication. If any open connection is found then we can easily retrieve process ID of the process which initiated that communication. There are possibilities that these process may not be legitimate processes and may be performing malicious activities. It is very important here to understand that this may become difficult for a regular forensic investigator to check and analyze each running process, which also communicates with some IP, to confirm whether it is a malware or not. Again, since the steps are automated the burden from forensic investigator's shoulders have been released.

There is possibility of having multiple instances / processes of same executable file communicating to same IP address. The tool takes care of this aspect, by removing duplicate IP addresses and keeping only one instance required for further process.

In the next step, the tool automatically dumps the executable file which initiated the communication. This suspicious executable file was running when the RAM was captured, hence, it was loaded into the RAM. Dump of the process will allow the tool to check whether it is a malware or a legitimate file. Apart from need of dump in the automated process, it is also useful for future investigation, especially in behavioral analysis, which is not included in the tool. Having, a dump file will provide an opportunity to the forensic investigator to analyze it in detail or to send it to some malware research center to get more details about it.

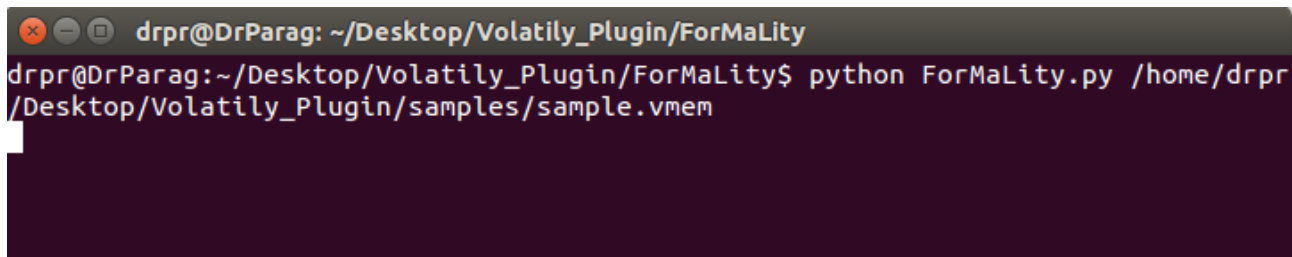
At last, the tool checks dumped executable files against VirusTotal database[11]. The tool automatically sends executable file to the VirusTotal using existing public API provided by VirusTotal. Since, VirusTotal contains a rich set of rules / signatures of malware, possibility of getting false positive is very low or almost negligible. Once sample is uploaded to Virus Total, ForMaLity, automatically requests for the report and as soon as the report is retrieved, it will display it and will also store it in file for future use.

The tool takes care of all necessary things by making sure end user gets quick and accurate solution without knowing how to use powerful tool like Volatility.

4. IMPLEMENTATION

Step by step implementation with relevant screen-shots is discussed below:

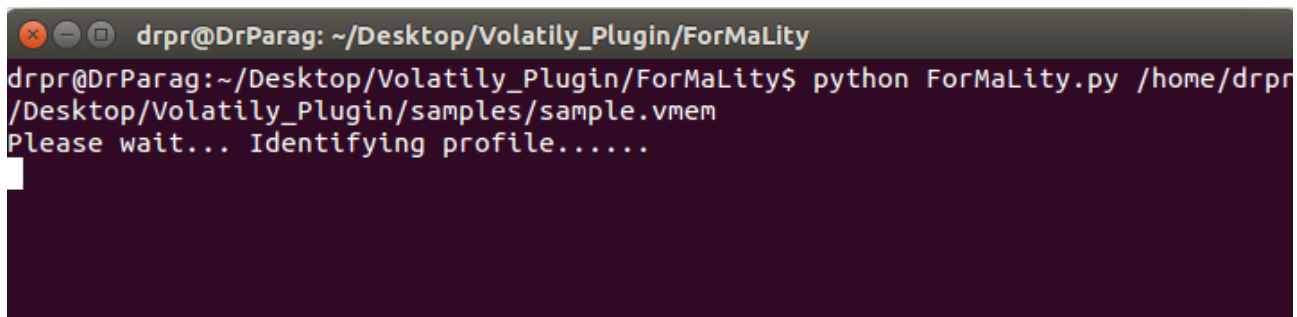
Step 1: Executing ForMaLity from command prompt, by providing sample image path.



```
drpr@DrParag: ~/Desktop/Volatily_Plugin/ForMaLity
drpr@DrParag:~/Desktop/Volatily_Plugin/ForMaLity$ python ForMaLity.py /home/drpr/Desktop/Volatily_Plugin/samples/sample.vmem
```

Figure 1. Executing ForMaLity

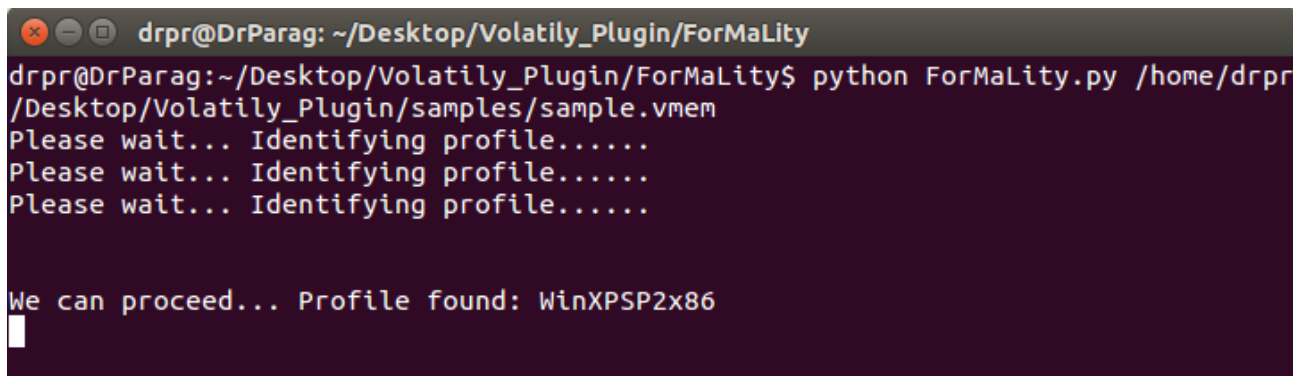
Step 2: Tool tries to identify profile, it may take longer time depending on sample.



```
drpr@DrParag: ~/Desktop/Volatily_Plugin/ForMaLity
drpr@DrParag:~/Desktop/Volatily_Plugin/ForMaLity$ python ForMaLity.py /home/drpr/Desktop/Volatily_Plugin/samples/sample.vmem
Please wait... Identifying profile.....
```

Figure 2. Identifying Profile

Step 3: Tool automatically identifies profile and saves it for future use

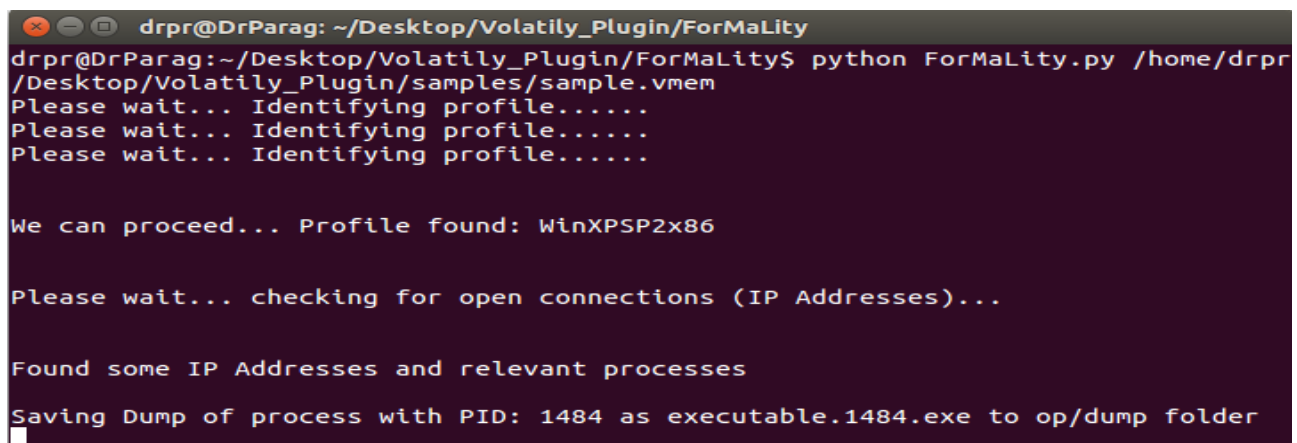


```
drpr@DrParag: ~/Desktop/Volatily_Plugin/ForMaLity
drpr@DrParag:~/Desktop/Volatily_Plugin/ForMaLity$ python ForMaLity.py /home/drpr/Desktop/Volatily_Plugin/samples/sample.vmem
Please wait... Identifying profile.....
Please wait... Identifying profile.....
Please wait... Identifying profile.....

We can proceed... Profile found: WinXPSP2x86
```

Figure 3. Found Profile

Step 4: Performs next step to get open connections and to dump process involved in such connections



```
drpr@DrParag: ~/Desktop/Volatily_Plugin/ForMaLity
drpr@DrParag:~/Desktop/Volatily_Plugin/ForMaLity$ python ForMaLity.py /home/drpr/Desktop/Volatily_Plugin/samples/sample.vmem
Please wait... Identifying profile.....
Please wait... Identifying profile.....
Please wait... Identifying profile.....

We can proceed... Profile found: WinXPSP2x86

Please wait... checking for open connections (IP Addresses)...

Found some IP Addresses and relevant processes
Saving Dump of process with PID: 1484 as executable.1484.exe to op/dump folder
```

Figure 4. Saving process dump

Step 5: After taking a dump of executable file, it automatically sends dump to VirusTotal for Malware Scan

```
drpr@DrParag: ~/Desktop/Volatility_Plugin/ForMaLity
drpr@DrParag:~/Desktop/Volatility_Plugin/ForMaLity$ python ForMaLity.py /home/drpr/
/Desktop/Volatility_Plugin/samples/sample.vmem
Please wait... Identifying profile.....
Please wait... Identifying profile.....
Please wait... Identifying profile.....

We can proceed... Profile found: WinXPSP2x86

Please wait... checking for open connections (IP Addresses)...

Found some IP Addresses and relevant processes

Saving Dump of process with PID: 1484 as executable.1484.exe to op/dump folder
Please wait... Scanning (executable.1484.exe) for Malware...

sending file .....executable.1484.exe to VirusTotal
```

Figure 5. Sends file to VirusTotal for Malware Scan

Step 6: Once the file is sent, the tool will automatically fetch, save and display report from VirusTotal.

```
drpr@DrParag: ~/Desktop/Volatility_Plugin/ForMaLity

Please wait... checking for open connections (IP Addresses)...

Found some IP Addresses and relevant processes

Saving Dump of process with PID: 1484 as executable.1484.exe to op/dump folder
Please wait... Scanning (executable.1484.exe) for Malware...

sending file .....executable.1484.exe to VirusTotal
sent file.....

Please wait... Waiting for report....

executable.1484.exe is a Malware because.....

6 positive scans found from total 55 scans...

Malware scan complete, please check op/vtReport.txt...
drpr@DrParag:~/Desktop/Volatility_Plugin/ForMaLity$
```

Figure 6. Retrieves, Saves and Displays report

Step 7: Report is saved to vtReport.txt file for future use as shown below.

```
vtReport.txt (~/Desktop/Volatility_Plugin/ForMaLity/op) - gedit

1 VirusTotal Scan... Report
2 #####
3 executable.1484.exe is a Malware.....
4
5 because 6 positive scans found from total 55 scans...
6
7 For more details check dump folder or refer
8 https://virustotal.com/en/file/48db195007e5ae9fc1246506564af154927e9f3fbfca0b4054552804027abbf2/analysis/
9
10 #####
11
12
13
```

Figure 7. Content of saved report

5. CONCLUSION

Outcome of this work will provide useful and crucial assistance to forensic investigators in analyzing live memory dumps for use of possible malware. Tool was tested with 5 various samples and gave accurate results for all the samples. Automation, accuracy and user friendliness of the tool will help in speeding up the live forensics. This tool may also reduce cost of training forensic investigators for malware analysis.

6. FUTURE SCOPE

Though, the tool meets all the current requirements for automatic forensic malware analysis, there is still scope of enhancement. The tool can be further extended to include more aspects of malware analysis. The tool currently supports only existing profiles, so a provision can be made available in the tool to accommodate new profiles. Last but not the least, a GUI to this automated process can make it possible to get results in couple of clicks.

REFERENCES

- [1] Kristine Amari, "Techniques and Tools for Recovering and Analyzing Data from Volatile Memory ", in SANS Institute InfoSec Reading Room, 2009
- [2] Liming Cai, Jing Sha ,Wei Qian, "Study on Forensic Analysis of Physical Memory" in the proceedings of 2nd International Symposium on Computer, Communication, Control and Automation (3CA 2013), 2013, p.no. 221-224
- [3] The Volatility Foundation - <http://www.volatilityfoundation.org/>
- [4] Cuckoo Sandbox - A malware analysis system, <https://www.cuckoosandbox.org>
- [5] Manuel Egele, Theodoor Scholte, Engin Kirda And Christopher Kruegel, "A Survey on Automated Dynamic Malware Analysis Techniques and Tools" in the ACM Computing Surveys Volume 44 Issue 2, 2012, Article No. 6 p.no. 1–49.
- [6] Michael Bailey, Jon Oberheide, Jon Andersen, Z. Morley Mao, Farnam Jahanian, Jose Nazario, "Automated Classification and Analysis of Internet Malware" in Recent Advances in Intrusion Detection, Volume 4637 of the series Lecture Notes in Computer Science, 2007. p.no. 178-197
- [7] IDA - Multi-processor disassembler and debugger, <https://www.hex-rays.com/products/ida/>
- [8] Tomer Teller, Adi Hayon, "Enhancing Automated Malware Analysis Machines with Memory Analysis" , Blackhat Arsenal – 2014, p.no. 1-5
- [9] Steffen Logen, Hans Höfken, Marko Schuba, "Simplifying RAM Forensics - A GUI and Extensions for the Volatility Framework", in the Seventh International Conference on Availability, Reliability and Security (ARES), 2012, p.no. 620 - 624
- [10] eVOLve by JamesHabben, <https://github.com/JamesHabben/evolve>
- [11] VirusTotal - facilitates the quick detection of viruses, worms, trojans, and all kinds of malware, <https://virustotal.com/>