TP02: Memory Analysis with Volatility

1. Before we begin

In this **Practice**, we'll take a memory dump and analyse it using Volatility Framework.

2. What we'll learn

⊡ How to use Volatility plugins or commands to analyse memory dumps.
⊡ How to use online resources for malicious activities detection.

3. Volatility

Volatility[1] is an interesting tool for memory analysis and is available for Windows and Linux. Volatility is a command-line tool that allows us to quickly pull out useful information such as what processes were running on the device, network connections, and processes that contained injected code. You can even dump DLL's and processes for further analysis.

Volatility also supports the analysis of memory dumps from Linux devices and a wide range of plugins[2] have been designed by the forensic community.

For the actual practice we will use a memory dump extracted from a suspected windows machine: memdump.mem

4. Volatility Commands for memory forensics

In Volatility tool a variety of commands are available for processes exploration and analysis, we will show here the very important among them:

- **Display Volatility options:**

`volatility -h` : This command displays volatility available options.

- **Extract image information**

`imageinfo` : It displays the image memory profile which is used for memory analysis.

The profile corresponds to OS of memory dump for example: `--profile=Win7SP1x86`
Corresponds to a **Windows 7 SP1 in 32 bits operating system.**

[1] https://www.volatilityfoundation.org/
[2] https://github.com/volatilityfoundation/volatility/

- Display processes list:

`pslist` : It we allows to list the running processes in the memory dump.

During a memory analysis, we should identify legal[3] and illegal processes. From the results of this command applied on **memdump.mem** file we can see that :
`rad5163B.tmp.exe` and **`rfhyMVOQxfc.exe`** are suspected processes because they have suspected names. Whereas smss.exe, winlogon.exe and services.exe represent OS processes.

- Display processes tree:

`pstree` : This command allows to identify process parent or child processes, thus we can note suspected structures for example a Command line CMD is child process of Internet Explorer.

It is very important to get an idea of what process spawned another process. This makes it easier to spot suspicious process activity as you can see what process launched 'cmd.exe' or 'powershell.exe' for example and see if this looks like legitimate activity or not.

Using `pstree` is a great way to spot malicious processes masquerading as legitimate Windows processes. Windows processes will always run from set locations on disk and their parent process tends to be a set process. For example 'taskhostw' will always run from '%systemroot%\system32\taskhostw.exe' and its parent process will always be 'svchost.exe'. So if we spot 'taskhostw' running from any other location or with a different parent process then it is definitely something we want to take a closer look at.

Applying this command on suspected memory dump we can see that process with **`PID 1416`** is child process of process having **`PID 3544`** corresponding to **Winword** process.

- **Display hidden processes**

`psxview` : It allows to display hidden processes.

For a hidden process **`pslist`** and **`psscan`** columns are assigned to **`False`**.

- **Process DLL files**

`dlllist` **:** It extracts DLL files used by a given process.

We can test this command for suspected process with PID = 1808.

- **Register analysis**

[3] A detailed document about Windows legal or legitimate processes is available here : https://digital-forensics.sans.org/media/SANS_Poster_2018_Hunt_Evil_FINAL.pdf

`hivelist` : This command displays register information and subsequent files.

`hashdump` : Extracts pass words hashes of windows accounts.

- **Network connections analysis**

`netscan` : This command displays active network connections.

We can see that identified processes achieve several connections to the following IP address:

`172.16.169.164:4444`

- **Injected code detection**

`malfind` **:** It displays suspected processes which may contain injected code.

After execution of this command we can see header information of process that has PID = 1416 with head MZ.

- **Process dump**

`procdump` **:** like dumpfiles this command extracts a determined process with it's all DLL files and saves them into a specified directory.

Processes dumping is used to perform detailed analysis of malicious processes. For example we can use **Windows Defender** and **Malware Bytes** or any **anti-virus** software to scan the dumped DLL files and search for eventual malicious processes.

- **Mutual exclusion detection**

`mutantscan` : It displays Mutex (*Mutual Exclusion*) primitives (i.e., synchronization mechanisms)
Mutex can be used by malicious software to do not re-infect the same machine**.** This information allows to identify a particular malware, it represents an indicator of compromised.

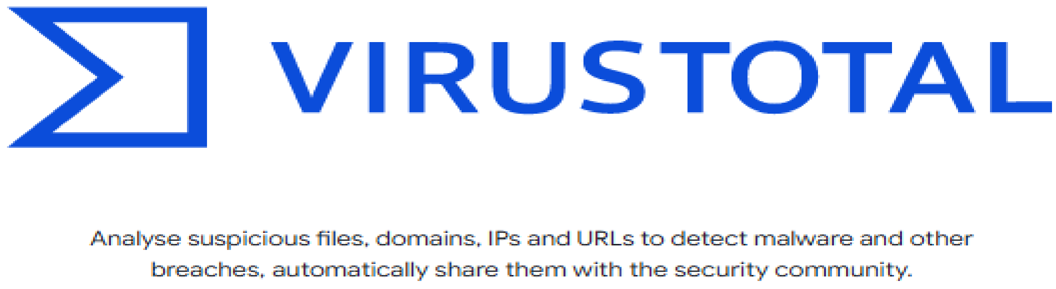
- **Services extraction**

`svcscan` : It lists all executed services on the compromised machine.

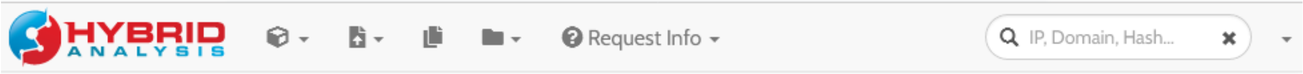
Malwares can use Windows services as a persistent way which they allow them to survive during reboot. It is interesting to analyze these elements to detect persistence mechanisms. **Emotet**[4] malware have used this technique to re-run.

5. Online resources for malicious activities detection

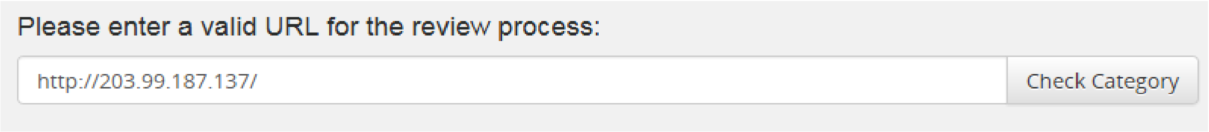
In a memory analysis task, online resources such **virustotal**[5], **payload security**[6] and **hybrid-analysis**[7] websites can be used to verify suspected processes.



Analyse suspicious files, domains, IPs and URLs to detect malware and other breaches, automatically share them with the security community.



To verify malicious URLs we can also use[8]:



Please enter a valid URL for the review process:

http://203.99.187.137/

[4] https://www.cisa.gov/news-events/alerts/2018/07/20/emotet-malware

[5] https://www.virustotal.com/gui/home/upload

[6] Payload Security's VxStream Sandbox is an automated malware analysis system for enterprises, governments, universities, SOCs and IR teams. At the core of **Hybrid Analysis**, a unique technology implementing in-depth memory analysis extracting more malicious processes and indicators.

[7] https://www.hybrid-analysis.com

[8] http://sitereview.symantec.com/#/