**Analyzing stuxnet affected memory dump using Volatility**

Stuxnet could be the first advanced malware. It is thought that it was developed by the United States and Israel to attack Iran's nuclear facilities. It attacked Windows systems using a zero-day exploit and It was focused on SCADA (Supervisory control and Data Acquisition systems) in order to  affect critical infrastructures

Stuxnet automatically executes itself and drops files onto the system by exploiting a vulnerability in various Windows versions (CVE-2010-2568) that allows malicious code to run when a specially crafted shortcut icon is displayed.

Stuxnet shares similarities with an [Autorun](https://www.f-secure.com/v-descs/worm_w32_autorun.shtml) worm, as it usually arrives via an infected USB thumb drive or other removable media, and once on an infected computer, will save copies of itself on other removable media for propagation to new victim machines.

An attacker can subvert this operation with a specially crafted .LNK file, which is pointed to a specially crafted Control Panel module (in reality, the malware). When the system attempts to resolve the shortcut file's icon, the vulnerability is triggered and the Control Panel module is automatically executed. The user does not need to click on the icon in order for the malware to be executed.

**Infection**

On execution, the malware drops the following files onto the system:

2 files ( mrxcls.sys and mrxnet.sys) - Dropped in C:\Windows\System32\Drivers folder

C:\Windows\inf\oem7a.PNF - An encrypted DLL file, the trojan-dropper's main component

C:\Windows\inf\mdmcpq3.PNF - An encrypted data file

C:\Windows\inf\mdmeric3.PNF

C:\Windows\inf\oem6c.PNF

An alert user may recognize the presence of a Stuxnet infection if the following items are present (which is helpful if the infected machine has no antivirus product installed):

The 2 dropped files, mrxcls.sys and mrxnet.sys, are found in C:\Windows\System32\Drivers folder

The registry keys associated with the 2 dropped drivers are visible:

HKLM\System\CurrentControlSet\Services\Services\MRxNet

HKLM\System\CurrentControlSet\Services\Services\MRxCls

**Execution**

The encrypted DLL file contained in the dropped oem7a.PNF file is injected into a process, using the following name structure:

[normaldll].ASLR.[random] - e.g., Kernel32.dll.aslr.21af34

The injection is performed by the mrxcls.sys file, which is responsible for attaching and copying the DLL into the target process. The rest of the injection routine is carried out by 2 additional components embedded in the mrxcls.sys file, which are also loaded into the same process space.

Mrxcls.sys also injects code to these processes:

services.exe

svchost.exe

lsass.exe

**Submit a report detailing the following aspects:-**

1. **STUXNET description**
2. **What is ASLR ?**
3. **Analysis of STUXNET**
   1. **Find the profile and when the image was dumped**
   2. **Identify the process that is affected? (multiple instances of the process would be running on the machine)**
   3. **Find the process id for the process affected by the stuxnet?**
   4. **Analyze the sockets**

**( The malicious process will not bound to any port )**

* 1. **Analyze the DLL of the suspicious process ( The malicious process has fewer DLLs )**
  2. **Find hidden DLLs using ldrmodule .**

**Challenge :-**

Company X has contacted you to perform forensics work on a recent incident that occurred. One of their employees had received an email from a fellow co-worker that pointed to a PDF file. Upon opening, the employee did not seem to notice anything, however recently they have had unusual activity in their bank account. Company X was able to obtain a memory image of the employee’s virtual machine upon suspected infection. Company X wishes you to analyze the virtual memory and report on any suspected activities found. Questions can be found below to help in the formal report for the investigation.

**Questions:**

List the processes that were running on the victim’s machine. Which process was most likely responsible for the initial exploit?

List the sockets that were open on the victim’s machine during infection. Are there any suspicious processes that have sockets open?

List any suspicious URLs that may be in the suspected process’s memory.

Are there any other processes that contain URLs that may point to banking troubles? If so, what are these processes and what are the URLs?

Were there any files that were able to be extracted from the initial process? How were these files extracted?

If there was a file extracted from the initial process, what techniques did it use to perform the exploit?

List suspicious files that were loaded by any processes on the victim’s machine. From this information, what was a possible payload of the initial exploit be that would be affecting the victim’s bank account?

If any suspicious files can be extracted from an injected process, do any anti-virus products pick up the suspicious executable? What is the general result from anti-virus products?

Are there any related registry entries associated with the payload?

What technique was used in the initial exploit to inject code in to the other processes?