Lab 1: Analysis of Sample1

Jonathan Harijanto

January 16, 2017

CS 373: Defense Against the Dark Arts - Winter 2017

Abstract

The purpose of this blog is to describe the observation from an unknown malware called `evil.exe`. This blog will cover the testing methodology for the malware and the knowledge that was gained from completing this lab.

I. BLOG

A. What I looked at:

In this first lab, I had to investigate an unknown malware called `evil.exe`. The only available information for me was the current version of the file and the size of it. I was able to retrieve these data from Right Click - Properties (Windows Operating System). Therefore, I need additional tools to study the malware's behavior further. The list of tools that I used are FakeNet, ProcessMonitor, ProcessExplorer and Flypaper.

B. How I looked at it:

To study the behavior of the malware accurately, I had to run all the tools before I executed (double-click) the malware. First, I ran both FakeNet and Flypaper to stop the real network connection and simulate a 'fake' ones. I did this because I wanted `evil.exe` to assume that a network connection still exists and try to contact the remote host. When this happen, I could observe its network activity with the host. Next, I ran ProcessMonitor to watch the malware's activity towards the file system, registry, and process in real-time. Finally, I executed ProcessExplorer to look up the malware process details when it is running.

The way I analyzed the malware was I played with each tool individually and noted all the information provided in a text editor. The first tool I focused on was ProcessMonitor because it gave me an overall idea what the malware was doing. After that, I closed ProcessMonitor and moved on to ProcessExplorer. Similarly, I continued with FakeNet because I needed to look at the network traffic. Lastly, I applied some manual testing using Windows search tools and command prompt's command.

C. What I found:

When I studied the malware using Process Monitor, I found out that the first thing it did was call different registry functions in the Windows registry. I noticed that `evil.exe` called `RegOpenKey` function to open a registry key, `RegQueryValue` to retrieve data in a registry key and `RegEnumValue` to enumerate the value of a registry key in both HKEY_LOCAL_MACHINE (HKLM) and HKEY_CURRENT_USER (HKCU). It is interesting to know that most `RegQueryValue` results are NAME NOT FOUND instead of SUCCESS.

Furthermore, I noticed that the malware started to create multiple files with DLL extensions (.dll) in the system32 directory. One of the examples is imm32.dll. This `evil.exe` also read multiple files and created some file mapping objects in the system32 directory too.

Further down the list in Process Monitor, I observed that the malware started to call different registry functions which are `RegDeleteValue` to remove a registry key and `RegEnumKey` to retrieve a sub-key from the registry key. However, these function calls only occurred in HKEY_LOCAL_MACHINE (HKLM). The last two things that the malware did were launched a new command prompt process and made a new directory called `ntldrs`. Inside this directory, the malware created four different files: `svchest.exe, funbots.bat, lsinter.gif, system.yf`. Then, it used the open command prompt to hide these files, using attrib command, except `svchest.exe`.

Moving on to a different tool called FakeNet. With this program, I discovered that the malware made several HTTP requests using the GET method. The first request was to a host named hisunpharm.com, where `evil.exe` asked for an application file called `pao.exe`. Next, it inquired an HTML file, and a text file from a host called timeless888.com. Furthermore, all these files were downloaded in gzip format.

The last tool that I used was Process Explorer. I saw that `evil.exe` consumed 75 MB of memory for itself (Private Byte). Also, I found various strings dumped in the Properties - Scan section. Some of these strings were related to command prompt instruction (`attrib`), and some of them are related to specific directories in Windows. In addition, some strings printed as the Registry function like `RegOpenKey`. Last but not least, I discovered some random words like *FsEjGsZJFs* and random phrases like *WHAT A FFFING DAY*.

After I had finished using all the tools, I decided to perform some manual tests. From this tests, I learned that, first, the malware hid the `ntldrs` directory from Windows Search. Second, the access

to `C:\Users\Admin\AppData\Local\`, where `pao.exe` is found, was blocked by the malware. Furthermore, the three files: Isinter.gif, funbots.bat, system.yf in `ntldrs` directory were set as hidden, unless I executed the `svchest.exe` (an executable file created by `evil.exe`). Finally, I tried to use the `attrib -s -h -r /s /d` command on the malware folder and nothing showed up.

II. Conclusion

From this lab experiment, I learned that `evil.exe` was able to exploit the Windows security by adding, retrieving, and deleting various registry keys. It also able to attack the system32 directory by adding and deleting several DLL extension files without permission. Furthermore, it could create its own network communication that able to download unknown files automatically. Thus, there is a high chance that `evil.exe` is a Trojan.