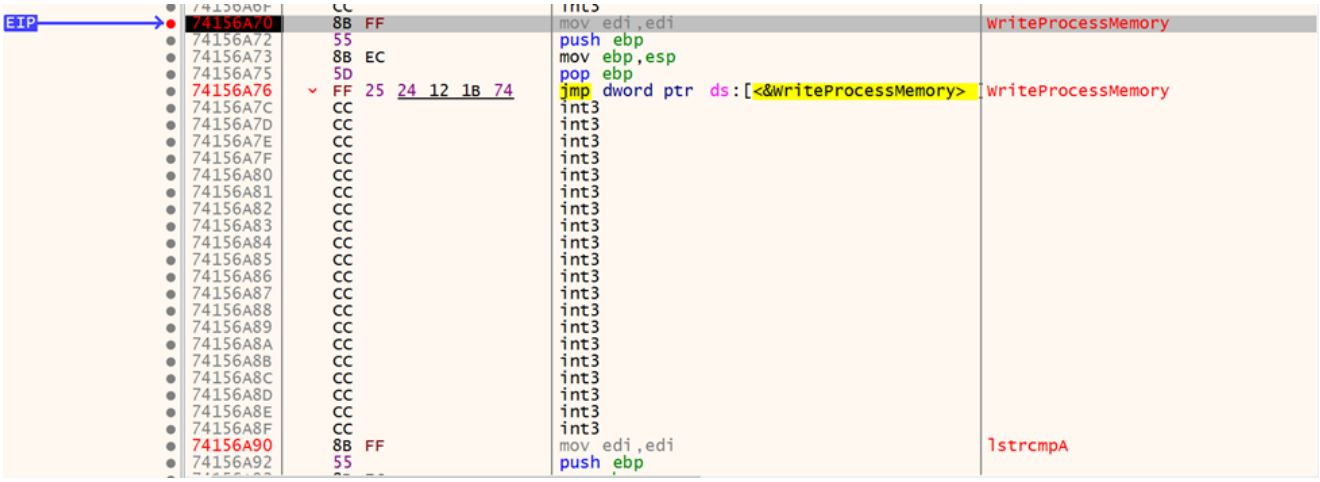
Researched by: Ilan Duhin

medium.com/@llandu/vawtrak-malware-824818c1837

Ilan Duhin March 19, 2023



Ilan Duhin

Mar 19

.

4 min read

“VawTrak” Malware

Executive Summary:

“Vawtrak” is a banking Trojan –malware that attempts to steal credentials from banks.

The Banker gains access to bank accounts via custom key logging, utilizing the access of a wide range of login credentials, such as passwords stored in browsers, FTP client private keys, or information stored within remote desktop settings.

To communicate, the Banker utilizes SOCKS connection and exfiltrates information such as screenshots and video captures.

Technical Analysis:

Unpacking Process:

From analyzing the malware in IDA, I see suspicious API such **CreateToolhelp32Snapshoot** call that retrieves running processes, I guess that the malware will use it to get snapshot of them until it finds a legitimate process to inject his malicious code.

using CreateToolhelp32Snapshot
In addition, I have checked on online sandboxes such as Any.run & Hybrid Analysis to see additional info about the injection and I find that it tries to inject into the **explorer.exe process.**

Process Tree from Any.run
After the conclusions, I choose to put my BP on **WriteProcessMemory** because the malware try to inject her code into other process so this call is perfect to use.

After placing a breakpoint on **WriteProcessMemory**, in order to catch the injection of the malware, and checking the functionality of the API calls within the code in MSDN, the parameter to dump the MZ header is clearly shown.

The parameter required to dump the packed information is the third parameter, according to MSDN. The parameter is “lpBuffer” — A pointer to the buffer that contains data to be written in the address space of the specified process or in other words **“holds our unpacking file data”.**

Memory Map permissions
The dumped memory file:
After cleaning the junk code:
the malware immediately writes itself into the autoruns paths, in order to have a foothold on the host upon startup or restart.

After running, “Vawtrak” creates a child process with same name as the original running process. 30 seconds into the run, the original malware process terminates itself, and removes itself from the original running path and copies itself into APPDATA\LOCAL\TEMP, **in order to elevate privileges** (because of existence Writing privileges at this path).

After establishing itself, the malware, through the injected process drops additional PE files, which contain **DLL and an executable**.

Drop the dll from pe-sieve after I dumped the implemented files into the folder to get more information. IDA also verify us that it is a dll file.

the DLL contains is creating a snapshot and list of all the currently running processes (as we mentioned earlier), this is usually done by reconnaissance malwares in order to target specific artifacts within the host.

When reconnaissance is complete, the malware extracts its C2 server from a seed that the malware file contains — hard coded. It will perform certificate validation in order to check if the server is still available, if not, the malware goes to sleep for a random amount of time.

In addition to certificate validation, the Banker checks if any reconnaissance information has been retrieved. If not, the malware does not initiate communication methods.

Encrypted data start with the C2 server:
DNS query to C2 server
Following all these steps, “VawTrak” will attempt to spread through the network utilizing SMB — a legitimate Windows file-sharing protocol.

After completing all activities within the host and attempting to preform lateral movement, the malware wipes itself off the host and terminates its process.

Tries to do lateral movement to another computers via SMB protocol.
looking for RDP sessions:
reg add HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\WDigest \UseLogonCredential / — auth of http protocol, stored in plaintext user credentials.

· **By default the key isn’t shows in registry.**