22 – Covert Malware Launching

Launchers and Resource Sections
Process Injection
   DLL Injection
   Hook Injection
   Detours
   APC injection
Launchers and Resource Section

- Malware may be hidden in a launcher's resource section.
- Resource manipulation functions may be used by launcher to extract it:
  - FindResource
  - LoadResource
  - SizeofResource
- Launchers may contain privilege escalation code.
Process Injection

- Malware may be injected into a running process using
  - VirtualAllocEx
  - WriteProcessMemory
- In DLL injection, the process may be forced to load a DLL (containing malware in `DLLMain`)
- In Direct Injection, code is inserted into the memory address space of a process directly.
DLL Injection

- Launcher must find the victim process. Usually by searching using
  - CreateToolhelp32Snapshot
  - Process32First
  - Process32Next
- Lancher than then use OpenProcess to get the process handle.
- Injection usually done by call to CreateRemoteThread.
- Launcher may also call VirtualAllocEx to allocate space for the library name in victim.
- WriteProcessMemory is called to store the library name in the victim.
- Then GetModuleHandle, GetProcAddress, and CreateRemoteThread opening a new thread in the victim process that loads the specified library.
Direct Injection

- Usually used to inject shellcode (from an exploit).
- Three telltale functions:
  - `VirtualAllocEx`
  - `WriteProcessMemory`
  - `CreateRemoteThread`
- Often contains two calls to `VirtualAllocEx` and `WriteProcessMemory`, one for code and one for data.
- Call to `CreateRemoteThread` contains injected code start address of and data address.
- Extra work may need to be done by the malware to get access to necessary functionality.
Process Replacement

• Process replacement is the extreme case of process injection in which the entire process is overwritten.

• Process must be created in a suspended state by calling CreateProcess with the CREATE_SUSPENDED dwCreationFlag.

• After creation, replace process (usually with ZwUnmapViewOfSection, VirtualAllocEx, and WriteProcessMemory).

• Finally start process running with SetThreadContext (for entry point) and ResumeThread.
Hook Injection (1)

- Two purposes:
  - Insure that malware will run on receipt of a specific message
  - Insure that a specific DLL is loaded into a victim process

- Hooks can be
  - Local (initiating messages are sent to an internal process)
  - Remote (messages are destined for another process)

- Keyloggers, for example, usually hook using `WH_KEYBOARD` or `WH_KEYBOARD_LL` procedure types.

- `SetWindowsHookEx` is the primary function used to perform remote hooking. The hook procedure must call `CallNextHookEx` to let other hooks run.
Hook Injection (2)

- Loading into all threads may degrade system performance, thus malware often targets particular threads.
  - Identify the \texttt{dwThreadId} of the process whose thread is to be hooked.
  - Malware must do its dirty work and call \texttt{CallNextHookEx}
  - Install the malware by calling \texttt{SetWindowsHookEx} with appropriate \texttt{idHook} (WH\_CBT or other unlikely messages are good candidates).

- To execute the behavior, send the required message to the hooked thread.

- Malware may call \texttt{LoadLibrary} and \texttt{UnhookWindowsHookEx} in \texttt{DllMain} to clean up.
Detours

- MS library supporting instrumenting (modifying) OS and applications.
- Functions used by malware:
  - Import table modification
  - Attaching DLLs to existing programs
  - Add function hooks to running processes.
- When DLL is added, PE structure is modified to have .detour section containing original PE header and new IAT.
Asynchronous Procedure Call (APC) Injection

- Each thread has a list of associated APCs, which can be invoked by remote threads.
- Remote thread invokes APC by calling `QueueUserApc` providing the APC, thread, and data.
- Thread-targeting code will often call `CreateToolhelp32Snapshot`, `Process32First` and `Process32Next` followed by calls to `Thread32First` and `Thread32Next`.
- Alternatively malware may use `ZwQuerySystemInformation`.
- APC injection from kernel will use `KeInitializeApc` and `KeInsertQueueApc`.
Next Time

• Read PMA Chapter 13
• Find out what translate.py does on REMnux and how it could be used to decode xor encoded data.
• Do as many of the labs as you can.