# TTP Detail – T1574.013

## TTP Information

Name: KernelCallbackTable

Description: Adversaries may abuse the <code>KernelCallbackTable</code> of a process to hijack its execution flow in order to run their own payloads.(Citation: Lazarus APT January 2022)(Citation: FinFisher exposed ) The <code>KernelCallbackTable</code> can be found in the Process Environment Block (PEB) and is initialized to an array of graphic functions available to a GUI process once <code>user32.dll</code> is loaded.(Citation: Windows Process Injection KernelCallbackTable)  
  
An adversary may hijack the execution flow of a process using the <code>KernelCallbackTable</code> by replacing an original callback function with a malicious payload. Modifying callback functions can be achieved in various ways involving related behaviors such as [Reflective Code Loading](https://attack.mitre.org/techniques/T1620) or [Process Injection](https://attack.mitre.org/techniques/T1055) into another process.  
  
A pointer to the memory address of the <code>KernelCallbackTable</code> can be obtained by locating the PEB (ex: via a call to the <code>NtQueryInformationProcess()</code> [Native API](https://attack.mitre.org/techniques/T1106) function).(Citation: NtQueryInformationProcess) Once the pointer is located, the <code>KernelCallbackTable</code> can be duplicated, and a function in the table (e.g., <code>fnCOPYDATA</code>) set to the address of a malicious payload (ex: via <code>WriteProcessMemory()</code>). The PEB is then updated with the new address of the table. Once the tampered function is invoked, the malicious payload will be triggered.(Citation: Lazarus APT January 2022)  
  
The tampered function is typically invoked using a Windows message. After the process is hijacked and malicious code is executed, the <code>KernelCallbackTable</code> may also be restored to its original state by the rest of the malicious payload.(Citation: Lazarus APT January 2022) Use of the <code>KernelCallbackTable</code> to hijack execution flow may evade detection from security products since the execution can be masked under a legitimate process.

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Kill Chain Phases

**•** mitre-attack: persistence

**•** mitre-attack: privilege-escalation

**•** mitre-attack: defense-evasion

## Malware

* FinFisher

## APTs (Intrusion Sets)

* Lazarus Group