Obfuscated Files or Information: Dynamic API Resolution, Sub-technique T1027.007 - Enterprise

Archived: 2026-04-05 13:54:47 UTC

Adversaries may obfuscate then dynamically resolve API functions called by their malware in order to conceal malicious functionalities and impair defensive analysis. Malware commonly uses various Native API functions provided by the OS to perform various tasks such as those involving processes, files, and other system artifacts.

API functions called by malware may leave static artifacts such as strings in payload files. Defensive analysts may also uncover which functions a binary file may execute via an import address table (IAT) or other structures that help dynamically link calling code to the shared modules that provide functions.[1][2]

To avoid static or other defensive analysis, adversaries may use dynamic API resolution to conceal malware characteristics and functionalities. Similar to Software Packing, dynamic API resolution may change file signatures and obfuscate malicious API function calls until they are resolved and invoked during runtime.

Various methods may be used to obfuscate malware calls to API functions. For example, hashes of function names are commonly stored in malware in lieu of literal strings. Malware can use these hashes (or other identifiers) to manually reproduce the linking and loading process using functions such as `GetProcAddress()` and `LoadLibrary()`. These hashes/identifiers can also be further obfuscated using encryption or other string manipulation tricks (requiring various forms of Deobfuscate/Decode Files or Information during execution).[3][4][1]

Source: https://attack.mitre.org/techniques/T1027/007