ransomware_c2.exe

Likely Malicious

Name:	ransomware_c2.exe
Media type:	application/x-msdownload; format=pe64
SHA-256:	dc40d6f0cb069d6b89ba4742d4601fbb8186330168fb53111797932df5488972
Report ID:	2f92ee99-b211-410a-a44b-1d1358c3abe2
Submission Date:	6/20/2025, 12:12:16 PM UTC

peexe | anti-debug | overlay | packed | packer_detected

Analysis Overview

Threat Indicators, triggered during analysis

PE imports APIs used for code injection (thread execution hijacking)

Verdict

LIKELY MALICIOUS

Origin

Input File

Found API reference GetThreadContext@KERNEL32.dll	LIKELY MALICIOUS
Found API reference ResumeThread@KERNEL32.dll	LIKELY MALICIOUS
Found API reference SetThreadContext@KERNEL32.dll	LIKELY MALICIOUS
Found API reference SuspendThread@KERNEL32.dll	LIKELY MALICIOUS

MITRE Techniques

Tactic	Technique
Defense Evasion	Thread Execution Hijacking

Adversaries may inject malicious code into hijacked processes in order to evade process-based defenses as well as possibly elevate privileges. Thread Execution Hijacking is a method of executing arbitrary code in the address space of a separate live process.

Tactic	Technique
Persistence	Hijack Execution Flow

Adversaries may execute their own malicious payloads by hijacking the way operating systems run programs. Hijacking execution flow can be for the purposes of persistence, since this hijacked execution may reoccur over time. Adversaries may also use these mechanisms to elevate privileges or evade defenses, such as application control or other restrictions on execution.

Found indicators for heavy vm detection or system enumeration

Verdict

LIKELY MALICIOUS

Origin

Input File

The file is potentially trying to avoid defenses by detecting a sandbox environment LIKELY MALICIOUS

PE imports APIs possibly used for VM Detection

Verdict

SUSPICIOUS

Origin

Input File

Found API reference `GlobalMemoryStatusEx@KERNEL32.dll`

MITRE Techniques

Tactic	Technique
Defense Evasion	Dynamic API Resolution

Tactic	Technique
Discovery	System Information Discovery

An adversary may attempt to get detailed information about the operating system and hardware, including version, patches, hotfixes, service packs, and architecture. Adversaries may use the information from [System Information Discovery](https://attack.mitre.org/techniques/T1082) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions.

Tactic	Technique
Defense Evasion	Virtualization/Sandbox Evasion

Adversaries may employ various means to detect and avoid virtualization and analysis environments. This may include changing behaviors based on the results of checks for the presence of artifacts indicative of a virtual machine environment (VME) or sandbox. If the adversary detects a VME, they may alter their malware to disengage from the victim or conceal the core functions of the implant. They may also search for VME artifacts before dropping secondary or additional payloads. Adversaries may use the information learned from [Virtualization/Sandbox Evasion](https://attack.mitre.org/techniques/T1497) during automated discovery to shape follow-on behaviors.

Tactic	Technique
Defense Evasion	System Checks

Adversaries may employ various system checks to detect and avoid virtualization and analysis environments. This may include changing behaviors based on the results of checks for the presence of artifacts indicative of a virtual machine environment (VME) or sandbox. If the adversary detects a VME, they may alter their malware to disengage from the victim or conceal the core functions of the implant. They may also search for VME artifacts before dropping secondary or additional payloads. Adversaries may use the information learned from [Virtualization/Sandbox Evasion](https://attack.mitre.org/techniques/T1497) during automated discovery to shape follow-on behaviors.

PE imports APIs used for anti-debugging purposes

Verdict	Tags
SUSPICIOUS	anti-debug

Origin

Input File

Found API reference `OutputDebugStringA@KERNEL32.dll`

MITRE Techniques

Tactic	Technique
Defense Evasion	Debugger Evasion

PE imports APIs used to access the internet

Verdict

SUSPICIOUS

Origin

Input File

Found API reference `HttpAddRequestHeadersA@WININET.dll`

Found API reference `HttpOpenRequestA@WININET.dll`

Found API reference `HttpQueryInfoA@WININET.dll`

Found API reference `HttpSendRequestA@WININET.dll`

Found API reference `InternetCloseHandle@WININET.dll`

Found API reference `InternetConnectA@WININET.dll`

Found API reference `InternetOpenA@WININET.dll`

MITRE Techniques

Tactic	Technique
Command and Control	Application Layer Protocol

Adversaries may communicate using application layer protocols to avoid detection/network filtering by blending in with existing traffic. Commands to the remote system, and often the results of those commands, will be embedded within the protocol traffic between the client and server.

PE imports APIs used to enumerate local disc drives

Verdict

SUSPICIOUS

Origin

Input File

Found API reference `FindFirstVolumeW@KERNEL32.dll`

Found API reference `FindNextVolumeW@KERNEL32.dll`

Found API reference `FindVolumeClose@KERNEL32.dll`

Found API reference `GetVolumeInformationW@KERNEL32.dll`

MITRE Techniques

Tactic	Technique
Discovery	File and Directory Discovery

Adversaries may enumerate files and directories or may search in specific locations of a host or network share for certain information within a file system. Adversaries may use the information from [File and Directory Discovery](https://attack.mitre.org/techniques/T1083) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions.

PE section has an unusual entropy

Verdict

SUSPICIOUS

Origin

Input File

`.data` has an unusual entropy `0.396092832088` which may indicate packed data

MITRE Techniques

Tactic	Technique
Defense Evasion	Software Packing

Adversaries may perform software packing or virtual machine software protection to conceal their code. Software packing is a method of compressing or encrypting an executable. Packing an executable changes the file signature in an attempt to avoid signature-based detection. Most decompression techniques decompress the executable code in memory. Virtual machine software protection translates an executable's original code into a special format that only a special virtual machine can run. A virtual machine is then called to run this code.

Contains an overlay

Verdict	Tags
SUSPICIOUS	overlay

Origin

Input File

Input file has a `1866894` byte overlay at offset `1472512`

Executable may be carrying a suspicious packed payload

Verdict

SUSPICIOUS

Origin

Input File

A non-installer executable is not digitally signed and contains high-entropy (packed) data likely to be executed

Matched a relevant YARA rule

Verdict

SUSPICIOUS

Origin

Input File

Matched YARA rule pe_number_of_sections_uncommon with strength 0.5 (PE has an unusual number of sections (<2 or >10))

Matched YARA rule mersenne_twister_constants with strength 0.5

PE file contains many sections

Verdict

SUSPICIOUS

Origin

Input File

The PE executable contains 20 sections

PE has a thread-local-storage (TLS) callback

Verdict

SUSPICIOUS

Tags

anti-debug

Origin

Input File

TLS entrypoint at virtual address 0x4000ff20

TLS entrypoint at virtual address 0x1

TLS entrypoint at virtual address 0x4000ff00

TLS entrypoint at virtual address 0x4001e870

PE imports APIs used to manipulate/query other processes

Verdict

SUSPICIOUS

Origin

Input File

Found API reference OpenProcess@KERNEL32.dll

PE imports suspicious modules

Verdict

SUSPICIOUS

Origin

Input File

Imported module wininet.dll is marked as suspicious

PE section name contains interesting characters

Verdict

SUSPICIOUS

Origin

Input File

The name of the section /4 contains interesting characters

The name of the section /19 contains interesting characters

The name of the section /31 contains interesting characters

The name of the section /45 contains interesting characters

The name of the section /57 contains interesting characters

The name of the section /70 contains interesting characters

The name of the section /81 contains interesting characters

The name of the section /97 contains interesting characters

The name of the section /113 contains interesting characters

PE section size is empty

Verdict

SUSPICIOUS

Tags

packed

Origin

Input File

Section .bss indicates a raw size of 0

PE imports APIs used to hide other imports

Verdict

NO THREAT

Origin

Input File

Found API reference `GetProcAddress@KERNEL32.dll`

Found API reference `LoadLibraryA@KERNEL32.dll`

MITRE Techniques

Tactic	Technique
Defense Evasion	Dynamic API Resolution

PE has an uncommon section name

Verdict

NO THREAT

Origin

Input File

Section name `/4` is unusual

Section name `/19` is unusual

Section name `/31` is unusual

Section name `/45` is unusual

Section name `/57` is unusual

Section name `/70` is unusual

Section name `/81` is unusual

Section name `/97` is unusual

Section name `/113` is unusual

PE imports APIs to create or remove directories

Verdict

NO THREAT

Origin

Input File

Found API reference `RemoveDirectoryW@KERNEL32.dll`

PE imports APIs used to access or modify environment variables

Verdict

NO THREAT

Origin

Input File

Found API reference `getenv@api-ms-win-crt-environment-l1-1-0.dll`

PE imports APIs used to create/terminate threads

Verdict

NO THREAT

Origin

Input File

Found API reference `_beginthreadex@api-ms-win-crt-runtime-l1-1-0.dll`

PE imports APIs used to write data on files

Verdict

NO THREAT

Origin

Input File

Found API reference `SetEndOfFile@KERNEL32.dll`

Found API reference `fflush@api-ms-win-crt-stdio-l1-1-0.dll`

Found API reference `fputc@api-ms-win-crt-stdio-l1-1-0.dll`

Found API reference `fputs@api-ms-win-crt-stdio-l1-1-0.dll`

Found API reference `fwrite@api-ms-win-crt-stdio-l1-1-0.dll`

PE imports interesting APIs

Verdict

NO THREAT

Origin

Input File

Import `CreateHardLink@kernel32.dll` is marked as interesting

Import `DeleteFile@kernel32.dll` is marked as interesting

Import `GetCurrentProcess@kernel32.dll` is marked as interesting

Import `GetCurrentProcessId@kernel32.dll` is marked as interesting

Import `GetCurrentThread@kernel32.dll` is marked as interesting

Import `GetCurrentThreadId@kernel32.dll` is marked as interesting

Import `GetThreadContext@kernel32.dll` is marked as interesting

Import `GetThreadPriority@kernel32.dll` is marked as interesting

Import `GetThreadTimes@kernel32.dll` is marked as interesting

Import `MoveFile@kernel32.dll` is marked as interesting

Import `QueryPerformanceFrequency@kernel32.dll` is marked as interesting

Import `RaiseException@kernel32.dll` is marked as interesting

Import `SetProcessAffinityMask@kernel32.dll` is marked as interesting

Import `SuspendThread@kernel32.dll` is marked as interesting

Import `VirtualQuery@kernel32.dll` is marked as interesting

OSINT source detected benign resource(s)

Verdict

NO THREAT

Origin

Input File

OSINT provider `OPSWAT_REPUTATION` detected resource `a73f26a8d504043f785d7360e8febf2eeb8522ec873a0d4dd5d1d4bfd1e67d3d` as `NO_THREAT`

PE imports APIs used to create temporary files

Verdict

NO THREAT

Origin

Input File

Found API reference CreateFileW@KERNEL32.dll

Found API reference GetTempPathW@KERNEL32.dll

File Details

FileMagicDescription:	PE32+ executable (console) x86-64, for MS Windows
Size:	3.18 MB
Architecture:	64 Bits binary
SubsystemReadable:	IMAGE_SUBSYSTEM_WINDOWS_CUI
Date:	Wed Jun 18 14:59:58 2025
Packers (DiE):	Packer detected(HEUR)
IsDigitallySigned:	false
IsDotNet:	false
IsPacked:	false
Icon:	-

Hashes

MD5:	d45ba3248cbdc0952d2a691f6727a97f
SHA-1:	ae46fc37b831f60b3670ddfa8ba77dec31843872
SHA-256:	dc40d6f0cb069d6b89ba4742d4601fbb8186330168fb53111797932df5488972
SHA-512:	cf010962e5a233b30a4526852a66ba945ec6effdb4a0ea9cf3c19a18b601da1bf081ee952b23d191466fc18c544af7b65b954158dd24b36d043eb6929eebecf1
Imphash:	d4c48269e200e2a037f12fc8d0f08993
Fsiofuzzyhash:	6beb04d3f3b44fea02bd6cbe52ae004ad5faf19b91dfb8d1997f4004efe5dae2
Authentihash:	c125c0f3150dd980aa33fa8dad7c92bc3fe0b5bf21daa8571aef00fa5da36cd0
Ssdeep:	49152:AAUbeehGOZJOJ1auiXTW3nc1itb15hBDqqmj+foQ:AAOhGOZJAiXTEc1y15hBDqqmj+foQ

Visualization