BiBi Wiper: A Malware Analysis Amidst the Israel-Hamas-ISIS Conflict

Archived: 2026-04-05 22:56:47 UTC

The wiper collects information about the date and time of the system.

The wiper execution produces system information related to system paths, processor cores, threads, rounds, and stats.

While executing the wiper, it renames the files to .BiBi extension and disrupts the file system.

The following string appears that it could be indicative of spyware or a keylogger.

After gathering information about the disk drives on the system, the malware uses the 'GetDriveTypeA' API function to determine the type of each drive, such as removable, CD-ROM, network, etc.

The malware uses the 'GetNativeSystemInfo' API function to determine the processor architecture of the system and whether it is a 32-bit or 64-bit processor.
Then, the wiper takes the number of processors of the system, moves the value to the eax register, and displays the value in the command prompt in the initial execution.

The malware loads the rstrtmgr.dll DLL file using the LoadLibraryA API function. If the DLL loads successfully, the GetProcAddress function will get the DLL's RmStartSession address.

The screenshot below demonstrates how the operation in Figure 15 is executed dynamically:

Wiper execution via the command line and specifying a specific path:

The malware uses the 'Sleep' API function to delay thread execution. This function is often employed for time-based evasion by adding delays in the code.

cmd.exe /c wmic shadowcopy delete

cmd.exe /c bcdedit /set {default} bootstatuspolicy ignoreallfailures

cmd.exe /c bcdedit /set {default} recoveryenabled no

The following commands executed can be viewed from the created cmd processes.

After the execution of the commands, the wiper malware begins the corruption action in the file system. The malware begins by utilizing the 'CreateFileW' API function to open files. It sets the 'dwCreationDisposition' parameter to the value of '3', which corresponds to 'OPEN_EXISTING.' This action allows the malware to determine whether the file exists in the file system or not.

The malware uses the ‘WriteFile’ API function to rewrite data to files in the file system. It handles the files using the ‘hFile’ handle. If the write operation fails, the malware takes action at the ‘loc_140017C5E’ memory location. The data written in the ‘lpBuffer’ pointer is what the malware writes when it handles a file.

The Wiper malware uses the ‘FindFirstFileExA’ API function to search for files, directories, and sub-directories. The starting point for the search is the path that the threat actor specified during the execution of the Wiper, or it defaults to the ‘C:\Users’ path inside the ‘lpFileName’ pointer.

Then, the malware continues the search progress with the ‘FindNextFileW’ function.

During the corruption of the files in the file system, the wiper changes their extension to ‘.BiBi1’.

Yara Rule

Detection
