Analysis Report for: E55F5636F54DDC7178839A8BC8218A0A.exe

This code is written in VBScript, not C. The presence of `Dim`, `Set`, `CreateObject`, `WScript`, and `FileSystemObject` clearly indicates this. Analyzing it as C code would be incorrect.

****Overall Functionality**

This VBScript code monitors a specific process (Process ID 652) on the system. If the process is not running, it launches a specific executable (`MpDefenderCoreService.exe` located in a seemingly unusual directory `C:\Windows.old\Users\All Users\Documentos`) and then deletes itself. This behavior is highly suspicious and indicative of malware.

****Function Summaries**

The code doesn't define functions in the traditional sense. Instead, it uses built-in VBScript objects and methods:

* **`CreateObject("WScript.Shell")`**: Creates a WScript.Shell object, providing access to system-level functionalities, including executing programs (`Exec` method). No parameters, returns a Shell object.

* **`CreateObject("Scripting.FileSystemObject")`**: Creates a FileSystemObject, enabling file system manipulation such as deleting files (`DeleteFile` method). No parameters, returns a FileSystemObject.

* **`GetObject("winmgmts:\\.\root\cimv2")`**: Retrieves a connection to the WMI (Windows Management Instrumentation) service, allowing access to system information, including running processes. No parameters, returns a WMI object.

* **`objWMIService.ExecQuery(sQuery)`**: Executes a WMI query against the connected WMI service. `sQuery` (string) is the WMI query. Returns a collection of objects matching the query.

* **`WS.Exec(mainFilePath)`**: Executes the specified executable file (`mainFilePath`). The parameter is the path to the executable. The return value isn't directly used here but represents the process ID of the launched program.

* **`FSO.DeleteFile WScript.ScriptFullName`**: Deletes the current VBScript file itself using the `FileSystemObject`. The parameter is the full path to the script file (obtained via `WScript.ScriptFullName`).

* **`WScript.Sleep 5000`**: Pauses the script execution for 5000 milliseconds (5 seconds). No parameters.

****Control Flow**

The script's main logic resides within a `Do While True` loop. Inside the loop:

1. It initializes `isExists` to `false`.
2. It queries the WMI for running processes.
3. It iterates through the process list. If it finds a process with the matching ID ("652"), it sets `isExists` to `true` and exits the loop.
4. If the loop completes with `isExists` still `false` (the target process is not running), it launches `MpDefenderCoreService.exe`, deletes itself using `FSO.DeleteFile`, and exits the `Do While` loop.
5. If the target process is running, it waits for 5 seconds (`WScript.Sleep 5000`) and then reiterates the process.

****Data Structures**

The primary data structure is the collection of `Win32_Process` objects returned by the WMI query. This is essentially a list of running processes on the system. Other variables are simple strings and booleans.

****Malware Family Suggestion**

The behavior strongly suggests a **self-replicating malware**, potentially a **rootkit** or **backdoor**. The code's intent is to ensure a specific executable remains running. The deletion of the script itself after launching the executable makes it harder to detect and remove. The suspicious location of the executable (`C:\Windows.old`) and the use of a seemingly benign name (`MpDefenderCoreService.exe`, mimicking a legitimate Microsoft component) are classic obfuscation techniques employed by malware. The persistent monitoring of the process suggests the malicious program may be performing harmful activities in the background. The `On Error Resume Next` line is also highly suspicious, as it hides any potential errors. Without further analysis of the `MpDefenderCoreService.exe` itself, it is impossible to fully determine its functionality, but based solely on the VBScript, it is a highly probable malware variant.