Analysis Report for: gSWnOkdBMma.vbs

This code snippet is not C code; it's VBScript (VBS). C and VBScript are distinct programming languages. Attempting to analyze it as C would be fundamentally incorrect.

Overall Functionality

The VBScript code attempts to terminate a process and then delete two files. It targets a specific process identified by a seemingly hexadecimal filename ("75CDA3C0A569EA2C67F3D33927F21031.exe") and then deletes both that executable and a VBScript file ("gSWnOkdBMma.vbs"). The use of `WScript.Sleep 5000` introduces a 5-second delay between terminating the process and attempting to delete the executable. The `On Error Resume Next` statement is a particularly worrying sign, as it suppresses error handling, potentially masking failures and making debugging and analysis significantly harder.

Function Summaries

* **`GetObject("winmgmts:{impersonationLevel=impersonate}!\\.\root\cimv2")`**: This retrieves a WMI (Windows Management Instrumentation) object, providing access to system information, including running processes. It uses impersonation to potentially elevate privileges.

* **`CreateObject("Scripting.FileSystemObject")`**: Creates a FileSystemObject, which provides methods for interacting with the file system (creating, deleting, and manipulating files and folders).

* **`WMI.ExecQuery("SELECT * FROM Win32_Process WHERE Name = '75CDA3C0A569EA2C67F3D33927F21031.exe'")`**: This executes a WMI query to find all processes with the specified name.

* **`Process.Terminate`**: This terminates a process (within the loop).

* **`FSO.FileExists("path")`**: Checks if a file exists at the given path.

* **`FSO.DeleteFile("path")`**: Deletes a file at the given path.

Control Flow

1. **Error Handling:** `On Error Resume Next` is used, disabling standard error handling. This is extremely dangerous as errors could go unnoticed.

2. **WMI Process Termination:** The script queries for the target process. If found, it iterates through the results (although there's likely only one) and calls `Process.Terminate` on each process object.

3. **Delay:** A 5-second delay (`WScript.Sleep 5000`) is introduced before attempting to delete the file. This might be intended to ensure the process is completely closed before deleting the file.

4. **File Deletion:** The script attempts to delete the target executable. It then unconditionally deletes "gSWnOkdBMma.vbs".

5. **Object Cleanup:** The `FSO` and `WMI` objects are explicitly released.

Data Structures

The main data structures used are:

* **WMI object (`WMI`)**: Represents the connection to the WMI service.
* **FileSystemObject (`FSO`)**: An object for file system manipulation.
* **Process Collection (`Processes`)**: A collection of `Win32_Process` objects representing running processes that match the query.
* **Individual Process Object (`Process`)**: A single `Win32_Process` object representing one running process (inside the loop).

Malware Family Suggestion

Given the functionality of terminating a specific process and deleting associated files, this VBScript strongly suggests malicious intent. It exhibits characteristics consistent with several malware families, including:

* **Backdoor:** The script could be part of a larger backdoor installation process. The script might be deleting its own components.

* **Ransomware (component):** It is highly suspicious of malware which eliminates its own installation files after running.

* **Self-removing malware:** The script actively removes itself and a related executable, which is a common technique for malware to avoid detection.

The use of a seemingly random hexadecimal filename for the executable adds to the suspicion of malicious intent. The script's aggressive removal of files and the lack of error handling point towards a need for stealth and obfuscation, traits common in malicious software. Without more context,

it's impossible to assign it definitively to a specific malware family, but its behavior is highly indicative of malicious activity. It is crucial to treat this script as potentially harmful.