Analysis Report for: 4EF020190955B8B13ACE8B8A13AEB39F.exe

Decoded using latin-1...

Overall Functionality

This code is a batch script (despite the `.c` extension, it's not C code; it's a Windows batch script) designed to aggressively reset Windows Update components. It stops Windows Update and BITS services, deletes various system folders and files related to Windows Update, and then re-registers numerous DLLs and OCXs. Finally, it modifies the security descriptors of the BITS and Windows Update services and sets their startup type to automatic. The script's intent appears to be to resolve Windows Update issues, but its methods are drastic and potentially damaging. The inclusion of a URL pointing to a website related to repairing Windows Update further suggests this intent, however, the aggressive nature of the script raises serious concerns.

Function Summaries (Note: There are no functions in the traditional C sense; these are commands within a batch script)

The script uses several Windows command-line utilities:

* `net stop`: Stops a specified Windows service. Parameters are the service names (e.g., `wuauserv`, `bits`, `cryptsvc`). No return value in the batch script context; success or failure is implied by the command's exit code.
* `rd`: Removes a directory. Parameters are the directory path, `/s` (remove subdirectories), and `/q` (quiet mode). No return value; success/failure is implied.
* `del`: Deletes a file or files. Parameters are the file paths. No return value; success/failure is implied.
* `cd`: Changes the current directory. Parameter is the new directory path. No return value.
* `regsvr32`: Registers a DLL or OCX file. Parameter is the file path and `/s` (silent mode). No return value; success/failure is implied.
* `sc.exe sdset`: Sets the security descriptor of a service. Parameters are the service name and the security descriptor string (a complex access control list). No return value; success/failure is implied.
* `sc config`: Configures a service. Parameters are the service name, `start= auto` (sets the startup type to automatic). No return value; success/failure is implied.
* `echo`: Displays text on the console. Parameter is the text to display. No return value.
* `pause`: Pauses the script execution, waiting for user input. No parameters, no return value.
* `cls`: Clears the console screen. No parameters, no return value.

Control Flow

The script's control flow is sequential. It executes commands one after another. There are no loops or complex conditional statements. Error handling is minimal; if a command fails, the script continues executing subsequent commands. This lack of error handling is a major weakness.

Data Structures

There are no explicitly defined data structures. The script uses environment variables like `%systemroot%`, `%windir%`, and `%ALLUSERSPROFILE%`. These variables hold system directory paths.

Malware Family Suggestion

While not strictly malware, this script exhibits characteristics that could be exploited by malicious actors. Its aggressive nature (deleting system folders and files, altering service security descriptors) makes it highly suspicious. A malicious variant could easily be created by:

* **Adding malicious code execution:** The script could be modified to download and execute malware after performing its "Windows Update reset" actions.
* **Data exfiltration:** Modified versions could steal sensitive data during the "cleanup" process.
* **Backdoor creation:** Altered service security descriptors could create a backdoor allowing remote access.

Therefore, although the script itself might not be a complete malware program, its functionality and the ease with which it could be weaponized strongly suggest it could be classified as a **potentially unwanted program (PUP)** or a tool used as part of a more complex malware infection. It carries significant risk and should never be run on a system unless its actions are fully understood and authorized. The lack of error checking and logging further increases the danger.