Analysis Report for: b.txt

Overall Functionality

The VBA macro code embedded within a Microsoft Word document (.doc file) constitutes a malicious script. Upon opening the document, the `AutoOpen` subroutine automatically executes, downloading a malicious executable (`evil.exe`) from a remote server (`http://10.8.3.119:8080/evil.exe`) and then immediately executing it. This behavior is typical of a downloader or dropper malware.

Function Summaries

* **`URLDownloadToFileA`:** This function, declared from the `urlmon` library, downloads a file from a specified URL to a local file path.

* **Parameters:**
* `pCaller`: A long integer (presumably a handle or context, but its exact purpose within this code is unclear, likely 0 for no specific caller).
* `szURL`: A string containing the URL of the file to download (in this case, `http://10.8.3.119:8080/evil.exe`).
* `szFileName`: A string specifying the local file path where the downloaded file will be saved (`C:\Windows\system32\spool\drivers\color\evil.exe`).
* `dwReserved`: A long integer reserved for future use (set to 0).
* `lpfnCB`: A long integer (likely a callback function pointer, but here set to 0).

* **Return Value:** A long integer representing the success or failure of the download operation (likely an HRESULT error code). Error checking is absent.

* **`WinExec`:** This function, declared from the `kernel32` library, executes a specified command line.

* **Parameters:**
* `lpCmdLine`: A string containing the command line to execute (in this case, the full path to the downloaded `evil.exe`).
* `uCmdShow`: A long integer specifying how the executed program's window should be displayed (`SHOW_HIDE` which is not defined explicitly but implies hiding the window.).

* **Return Value:** A long integer representing the success or failure of the command execution, but there is no error handling.

Control Flow

The `AutoOpen()` subroutine is the only significant function. Its control flow is straightforward and linear:

1. **Download:** It calls `URLDownloadToFileA` to download `evil.exe` from the remote server to a system directory. The lack of error handling means that any download errors will be silently ignored.
2. **Execution:** It then immediately calls `WinExec` to execute the downloaded `evil.exe` with a hidden window, providing no opportunity for user interaction.

Data Structures

No complex data structures are used; the code only manipulates strings and long integers. The strings represent URLs and file paths, while the long integers serve as parameters and return values for the API calls.

Malware Family Suggestion

Based on its functionality, the code strongly suggests a **downloader/dropper** type of malware. The primary role is to fetch a malicious payload from a remote location and execute it. The choice of the `system32\spool\drivers\color` directory suggests an attempt to hide the downloaded file. Further analysis of `evil.exe` would be necessary to determine the precise nature of the payload (e.g., ransomware, backdoor, etc.). The use of `AutoOpen` indicates that this malware is designed to be self-propagating, infecting documents and executing upon opening. The use of the URLDownloadToFileA function is a clear indication of the malware aiming to download further malicious code and execute it upon opening the affected document.