Why is there no supported way to get the command line of another process?

devblogs.microsoft.com/oldnewthing/20090223-00

February 23, 2009

Raymond Chen

Commenter Francisco Moraes wonders whether there is a supported way of getting the command line of another process. Although there are certainly unsupported ways of doing it or ways that work with the assistance of a debugger, there's nothing that is supported for programmatic access to another process's command line, at least nothing provided by the kernel. (The WMI folks have come up with Win32_Process.CommandLine. I have no idea how they get that. You'll have to ask them yourself.)

That there isn't is a consequence of the principle of not keeping track of information which you don't need. The kernel has no need to obtain the command line of another process. It takes the command line passed to the `CreateProcess` function and copies it into the address space of the process being launched, in a location where the `GetCommandLine` function can retrieve it. Once the process can access its own command line, the kernel's responsibilities are done.

Since the command line is copied into the process's address space, the process might even write to the memory that holds the command line and modify it. If that happens, then the original command line is lost forever; the only known copy got overwritten.

Raymond Chen

Follow