Aha, I have found a flaw in the logic to detect whether my program is running on 64-bit Windows

devblogs.microsoft.com/oldnewthing/20140904-00

September 4, 2014

Raymond Chen

Some time ago, I described how to detect programmatically whether you are running on 64-bit Windows, and one of the steps of the algorithm was "If you are a 64-bit program, then you are running on 64-bit Windows, because 32-bit Windows cannot run 64-bit programs." Every so often, somebody will claim that they found a flaw in this logic: "This algorithm may work today, but it assumes that the only version of Windows that can run 64-bit applications is 64-bit Windows. What if a future non-64-bit version of version of Windows runs 64-bit applications? Then your algorithm will incorrectly say that it is running on 64-bit Windows!" Yeah, but so what? Suppose you detect that the program is running on this hypothetical version of Windows that is not natively 64-bit but still runs 64-bit applications. What will your program do differently? How can you reason about the feature set and compatibility requirements of something that hasn't been invented yet? This is another case of *If you don't know what you're going to do with the answer to a question, then there's not much point in asking it*.

In this specific case, you should just continue about your normal business and let the emulation layer of the hypothetical future version of Windows do its job of giving you a 64-bit sky with 64-bit birds in the 64-bit trees.

Raymond Chen

Follow