If you can detect the difference between an emulator and the real thing, then the emulator has failed

devblogs.microsoft.com/oldnewthing/20100519-00

May 19, 2010

Raymond Chen

Recall that a corrupted program sometimes results in a "Program too big to fit in memory" error. In response, Dog complained that while that may have been a reasonable response back in the 1980's, in today's world, there's plenty of memory around for the MS-DOS emulator to add that extra check and return a better error code. Well yeah, but if you change the externally visible behavior, then you've failed as an emulator. The whole point of an emulator is to mimic another world, and any deviations from that other world can come back to bite you. MS-DOS is perhaps one of the strongest examples of requiring absolute unyielding backward compatibility. Hundreds if not thousands of programs scanned memory looking for specific byte sequences inside MS-DOS so it could patch them or hunted around inside MS-DOS's internal state variables so it could modify them. If you move one thing out of place, those programs stop working. MS-DOS contains chunks of "junk DNA", code fragments which do nothing but waste space, but which exist so that programs which go scanning through memory looking for specific byte sequences will find them. (This principle is not dead; there's even some junk DNA in Explorer.)

Given the extreme compatibility required for MS-DOS emulation, I'm not surprised that the original error behavior was preserved. There is certainly some program out there that stops working if attempting to execute a COM-style image larger than 64KB returns any error other than 8. (Besides, if you wanted it to return some other error code, you had precious few to choose from.)

Raymond Chen

Follow