Why does a corrupted binary sometimes result in “Program too big to fit in memory”?

devblogs.microsoft.com/oldnewthing/20060130-00

January 30, 2006

Raymond Chen

If you take a program and corrupt the header, or just take a large-ish file that isn’t a program at all and give it a “.exe” extension, then try to run it (Warning: Save your work first!), you will typically get the error “Program too big to fit in memory”. Why such a confusing error message? Why doesn’t it say “Corrupted program”?

Because the program isn’t actually corrupted. Sort of.

A Win32 executable file begins with a so-called “MZ” header, followed by a so-called “PE” header. If the “PE” header cannot be found, then the loader attempts to load the program as a Win16 executable file, which consists of an “MZ” header followed by an “NE” header.

If neither a “PE” nor an “NE” header can be found after the “MZ” header, then the loader attempts to load the program as an MS-DOS relocatable executable. If not even an “MZ” header can be found, then the loader attempt to load the program as an MS-DOS non-relocatable executable (aka “COM format” since this is the format of CP/M .COM files).

In pictures:

MZ	PE	Win32
	NE	Win16
	else	MS-DOS relocatable
else		MS-DOS non-relocatable

Observe that no matter what path you take through the chart, you will always end up at something. There is no exit path that says “Corrupted program”.

But where does “Program too big to fit in memory” come from?

If the program header is corrupted, then various fields in the header such as those which specify the amount of memory required by the program will typically be nonsensical values. The loader sees an MS-DOS relocatable program that requires 800KB of conventional memory, and that's where "Out of memory" comes from.

An MS-DOS non-relocatable program contains no such information about memory requirements. The rule for loading non-relocatable programs is simply to load the program into a single 64KB chunk of memory and set it on its way. Therefore, a program with no "MZ" header but which is larger than 64KB in size won't fit in the single 64KB chunk and consequently results in an "Out of memory" error.

And since people are certain to ask:

- "MZ" = the legendary Mark Zbikowski.
- "NE" = "New Executable", back when Windows was "new".
- "PE" = "Portable Executable", because one of Windows NT's claims to fame was its portability to architectures other than the x86.
- "LE" = "Linear Executable", used by OS/2 and by Windows 95 device drivers.

Raymond Chen

Follow