What could be happening in Safe Mode to make my heap corruption bug go away?

devblogs.microsoft.com/oldnewthing/20161130-00

November 30, 2016

Raymond Chen

A customer had a program that encountered heap corruption bugs, and they found that the bugs didn't occur when the system was running Safe Mode. What is so special about Safe Mode that makes the bug go away, and how can we get that Safe Mode-like behavior all the time?

While we're at it, let's make the entire plane out of the black box.

In Safe Mode, the system loads only essential device drivers, and in particular, the video driver specifically tailored for your video card is not used. Instead, the system uses a plain vanilla video driver with no hardware acceleration or any other fancy features.

The plain vanilla video driver can affect how applications behave. Since there is no hardware acceleration, the program may be sent into alternate code paths which employ software emulation. It also changes the video DLLs loaded into the process, and that will affect the address space layout as well as alter the process's heap usage. Both of these things may perturb the memory map enough so that the buggy behavior manifests itself differently.

For example, suppose you had a use-after-free bug that accidentally zeroed a byte of memory that had already been freed back into the heap. The change in address space layout means that the heap may move to a different location in memory, causing pointers to have slightly different values, and maybe the result is that in Safe Mode, the pointer has a value of 0x00123456, so that clearing the high-order byte to zero has no effect. Or maybe the change in memory allocation pattern caused by the switch to the plain video driver means that the byte that got accidentally zeroed out hadn't yet been reused by another heap allocation, so writing to it has no perceptible effect (because nobody was using it).

The heap is a chaotic system, since it is highly sensitive to the exact pattern of memory allocation and deallocation (which can be nondeterministic due to multi-threading), so it doesn't take much at all to make the consequences of a heap corruption bug vary wildly.

Raymond Chen

Follow