The unintentionally-expanding scope of the SEM_NOGPFAULTERRORBOX flag

devblogs.microsoft.com/oldnewthing/20230227-00

February 27, 2023

Raymond Chen

In the old days, the fancy graphics cards were 640 × 480, computer memory was measured in megabytes, and there was no Windows Error Reporting. Back in those days, the `SEM_NOGPFAULTERRORBOX` flag was the last check before we displayed the “box of doom”:

```
;-----------------------------------------------------------------------;
; Display_Box_of_Doom -- Display the Unrecoverable Application Error
;            box that everyone seems to dislike so much.
;
; ...
;
;-----------------------------------------------------------------------;

cProc   Display_Box_of_Doom,<PUBLIC,NEAR>

    parmW   action
    parmD   lpText
cBegin
    SetKernelDS
    push    es
    mov es,curTDB                   ; did app disable exception
    test    es:[TDB_ErrMode],02h    ;  message box?
    pop es
    jnz nf_dont_ask
```

I included a snippet of the code just to show off the opening sentence of the comment block.

This logic was ported to 32-bit Windows as a corresponding final check before displaying the fatal error dialog box.

Around the Windows 2000 time frame, a little bit of code was added between the “final check” and the error dialog box in order to check whether the process belongs to a job object that has disabled the error dialog box.

And that created an environment where the “final check” was no longer right next to the thing the check was protecting.

The code to launch the just-in-time debugger was inserted right before the crash dialog, which means that it came *after* the check for `SEM_NOGPFAULTERRORBOX`: If you disable the crash dialog, you also disable just-in-time debugging.

And when Windows Error Reporting got added to the system, they also added the error reporting code right before the error dialog, which means that if you disable the crash dialog, you also disable Windows Error Reporting.

The test for `SEM_NOGPFAULTERRORBOX` moved further and further away from the thing it was protecting. Whenever people wanted to stick their "one last minute thing before showing the crash dialog" code, it got stuck in that slot, and yet another feature ended up being accidentally encompassed by that flag.

Related reading: The gradual erosion of the `SEM_NOOPENFILEERRORBOX` error mode.