You can’t leave a critical section you never entered

devblogs.microsoft.com/oldnewthing/20090619-00

June 19, 2009

Raymond Chen

If you call `LeaveCriticalSection` on a critical section you never entered, the behavior is undefined.

Critical sections are an extremly high-traffic code path. Intense work goes into making them as fast as possible. Customers like to ask questions like “Exactly how many cycles does it take to enter a critical section? We’re not going to use them if they’re too slow.” They don’t say what they are going to do if critical sections are too slow for them, but the threat is real (even if the justification is bogus). I’ve seen programmers respond to code feedback of the form “You need a critical section here” with “No, I’m not going to bother. I’m afraid it’ll slow down my code.”

Given that critical sections are so heavily used, the algorithm gets tweaked regularly in order to improve performance. Sometimes the tweaks are minor; other times, the algorithm gets a major overhaul. Of course, the hope is that by changing only the insides, nobody will notice. On the other hand, people who relied on undefined behavior (like exiting a critical section they never entered and hoping that something meaningful would happen) are going to see changes in behavior.

I don’t know the precise reasons why the internals of critical sections changed, but I suspect it had to do with mitigating the effect of lock convoy.

Raymond Chen

Follow