Adventures in application compatibility: The cost of forgetting to specify a calling convention

devblogs.microsoft.com/oldnewthing/20210902-00

September 2, 2021

Raymond Chen

We saw last time that the Windows header files sometimes look at world through `__stdcall`-colored glasses, and that causes problems when the header file fails to specify an explicit calling convention.

The developers of one particular component made the mistake of omitting an explicit calling convention for one of their callback function pointer types, but it didn’t cause any immediate problems. Consumers who compiled with `__cdecl` as the default calling convention passed a `__cdecl` function pointer, but things happened to work out okay.

However, people reported that after installing a sevicing update, some programs that used that component started crashing. The reason is that the servicing update altered the code generation, and now the misplaced stack pointer started causing problems.

What we have here is a confluence of multiple mistakes. The feature team authored their header file incorrectly, failing to specify an explicit calling convention. This led to customers consuming the header file incorrectly, and passing callback function pointers that used the `__cdecl` calling convention instead of `__stdcall`.

Now the application compatibility adventure begins.

In addition to fixing the header files to be explicit about the calling convention (to prevent the problem from spreading), the component has to be modified so that it can be used with *either* calling convention.

```
declspec(naked) declspec(noinline)
void CALLBACK
WrapCallbackWithESPFix(WIDGETFILTERPROC filter, int a, int b)
{
    __asm
    {
        mov     edi, edi                ; hotpatch stub
        push    ebp                     ; establish stack frame
        mov     ebp, esp

        push    b
        push    a
#if _CONTROL_FLOW_GUARD
        mov     ecx, filter             ; call target
        call    [__guard_check_icall_fptr]
        call    ecx
#else
        call    filter                  ; make the call
#endif

        ; restore esp if the callee mismanaged it due to wrong calling convention
        mov     esp, ebp
        pop     ebp
        ret     12
    }
}
```

It so happens that this workaround didn’t hang around indefinitely. The component in question has a very small audience, and in particular, only one of the clients was encountering this problem. That customer made a fix for their program and deployed it via their update channel. The workaround was removed a little less than a year later.

Bonus reading: Throwing garbage on the sidewalk: The sad history of the rundll32 program.

Raymond Chen

Follow