What is this extra thread in my process?
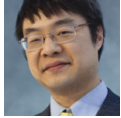
devblogs.microsoft.com/oldnewthing/20140212-00

Raymond Chen

A customer liaison asked:

> After applying Service Pack 2 to Windows Server 2003, my customer found that a simple MFC application (just using the template, no customization) has two threads when it is supposed to have only one. After five minutes, one of the threads exits. This doesn't happen on Windows Server 2003 RTM or Windows Server 2003 Service Pack 1.
>
> Here is a stack trace of the extra thread:
>
> ```
> 0:001> kP
> ntdll!KiFastSystemCallRet
> ntdll!NtWaitForMultipleObjects+0xc
> ntdll!EtwpWaitForMultipleObjectsEx+0xf7
> ntdll!EtwpEventPump+0x27f
> kernel32!BaseThreadStart+0x34
> ```
>
> The parameters to `EtwpWaitForMultipleObjectsEx` seem to be consistent with `WaitForMultipleObjectsEx`. Assuming that's the case, the parameters are
>
> | | |
> |---|---|
> | `nCount` | `= 0` |
> | `lpHandles` | `= 00x004f4470` |
> | `bWaitAll` | `= 0` |
> | `dwMilliseconds` | `= 300000` = 5 minutes |
> | `bAlertable` | `= TRUE` |
>
> Can you explain what the purpose of this thread is? Did this behavior change as a result of the update? It is important for the customer to know the purpose of this thread.

We asked, "Why does the customer need to know the purpose of this thread?"

We never heard back from the customer liaison. I guess it wasn't that important after all.

In case you cared: From the names of the functions, it looks like this is the ETW event pump thread.

Raymond Chen

Follow