

Re: Windbg: Disable user mode debugging

Source:

<http://www.tech-archive.net/Archive/Development/microsoft.public.win32.programmer.kernel/2008-01/msg00122.htm>

- *From:* "Armin Zingler" <az.nospam@xxxxxxxxxxx>
 - *Date:* Thu, 10 Jan 2008 15:13:20 +0100
-

"David Craig" <drivers@xxxxxxxxxxx> schrieb

"Armin Zingler" <az.nospam@xxxxxxxxxxx> wrote in message
news:%23nXh6JzUIHA.3400@xxxxxxxxxxxxxxxxxxxxxxxxxxxx
> "David Craig" <drivers@xxxxxxxxxxx> schrieb
>> This group is not applicable because of the win32 in the title.
>> Win32 is a subsystem meaning 32-bit Windows.
>
> I don't have 64-bit Windows.
>
> So you don't have 64-bit Windows. The native system is NOT Windows.
> It is NT or some other magical derivative from Dave Cutler. It
> does not do Windows. It has no user interface components. Win32 is
> subsystem that does video, mouse, and keyboard, plus a few other
> things via kernel32.dll and win32k.sys (plus others). The kernel
> debugger is an extension to ntoskrnl and maybe the hal too. Device
> drivers, other than video, cannot call routines in win32k.sys. Device
> drivers use exports from ntoskrnl, hal, and other device
> drivers.

I have no doubts about this, however, I think this is hairsplitting
in view of the actual problem. I do know that the kernel does not have
a UI, but you were the one who wrote "32-bit Windows". I was only
talking about finding the right group. I could also say that
the whole product is "Windows" even if some parts will never be
seen as windows on the screen. Nevertheless, no need to elaborate
on this deeper, ok? :-)

>> The native system upon which
>> the win32 subsystem runs is the place where the kernel debugger
>> runs. You can have a Unix subsystem also run and a win16
>> subsystem until Vista.
>
> Sorry, I don't understand the distinction made. The Win32 groups
> have always been there for "native" Win32 programming. Maybe

Re: Windbg: Disable user mode debugging

> /this/ group is for "programming" only, but debugging is also a
> part of it. Again, there is no other "kernel" group.
>
The use of 'kernel' for both places is a bad usage of the word.

? I don't understand you. Neither did I choose the name for /this/ group, and as such, nor is it wrong to post to this group because it is a problem with using the kernel debug engine. Yes, /now/ I know that is in particular a VS problem – maybe caused by the kernel implementation when handling both debuggers at a time – but I wasn't aware of this when I started the thread. Sometimes this turns out afterwards.

Try windbg newsgroup

No, they would send me away just like you do. The problem is there even if Windbg does not run. But again, /now/ I know where the problem is.

and maybe the development.device.drivers newsgroup

I don't want to develop device drivers.

where you may find some assistance. The Visual Studio newsgroups are probably more closely related, but very few people do both types of debugging at the same time.

Yes, like I said, I guess most people won't know anything about kernel debugging at all there.

It just doesn't work well. If you have a device driver, just stub out all user requests with an error. Run your user mode app and make sure you see the correct parameters showing up in the call. After it appears to work well, then debug the device driver. Finally, go back and make sure the app handles the returned data correctly. Hard to both at the same time especially if you get into that mangled code stuff.

I don't know how this relates to the situation described. Again, the kernel debugger will be active all the time because it is "waiting" for a BSOD. In addition, it is a (user mode) development machine writing managed .Net applications. I do /not/ actively

Re: Windbg: Disable user mode debugging

Re: Windbg: Disable user mode debugging

do "mixed" debugging all the time. I actively only write managed applications. Only as soon as a BSOD will occur, maybe tomorrow or maybe in four weeks, I will turn to my other machine and use Windbg to analyze the BSOD problem.

>> The slowness is expected since you have two debuggers trying to
>> control the machine and getting in each other's way.
>
> Maybe, but why? One is for user mode, the other for kernel. That's
> why there must not be a conflict. The slowness is due to data
> exchange between the machines (verified by varying serial port
> speed). There is no reason for this massive transfer because
> debugging a user mode
> application has nothing to do with the other machine that is
> /only/ there for kernel debugging, so why transfer anything?
> It's a local matter and nothing else.
>
All breakpoints and other debugger support is done in the real kernel and not the Win32 subsystem. When you have two debuggers something has to decide who gets called.

Yes, but obviously this doesn't work well. Otherwise it wouldn't communicate over the serial connection for local-only issues. Again, "/debug=noumex" has been used.

>> I don't try and do both at the same time since I don't usually
>> write any application code, but there may be some way to do it.
>> Maybe the VS newsgroup or either of the windbg newsgroups can
>> help. I know the windbg group has people from Microsoft
>> answering questions.
>
> Windbg is just the UI, as you say. The problem is /only/ on the
> target machine where the internal kernel debugger and VS seem to
> have a
> conflict that mustn't be there. It has nothing to do with Windbg
> on the other machine.
>
The conflict cannot help but be there. The CPU only issues one interrupt. Kernel mode components (NT) have to handle those interrupts and decide if the system should blue screen, the process should be terminated, or a debugger should get a chance to handle it.

Ok, I will have to accept it.

Anyways, thanks for your participation! I consider this thread closed now. ;)

Re: Windbg: Disable user mode debugging

Armin

.