

Re: windows error codes

Source:

<http://www.tech-archive.net/Archive/WinXP/microsoft.public.windowsxp.general/2004-12/1118.html>

From: David Candy (.)

Date: 12/01/04

Date: Thu, 2 Dec 2004 08:41:03 +1100

<http://oca.microsoft.com/en/windiag.asp>
Driver Development Tools: Windows DDK

Bug Check 0x1E: KMODE_EXCEPTION_NOT_HANDLED

The KMODE_EXCEPTION_NOT_HANDLED bug check has a value of 0x0000001E. This indicates that a kernel-mode program generated an exception which the error handler did not catch.

Parameters

The following parameters are displayed on the blue screen.

ParameterDescription

- 1The exception code that was not handled
- 2The address at which the exception occurred
- 3Parameter 0 of the exception
- 4Parameter 1 of the exception

Cause

This is a very common bug check. To interpret it, you must identify which exception was generated.

Common exception codes include:

0x80000002: STATUS_DATATYPE_MISALIGNMENT

An unaligned data reference was encountered.

0x80000003: STATUS_BREAKPOINT

A breakpoint or ASSERT was encountered when no kernel debugger was attached to the system. ¶

0xC0000005: STATUS_ACCESS_VIOLATION

A memory access violation occurred. (Parameter 4 of the bug check is the address that the driver attempted to access.)

For a complete list of exception codes, see the ntstatus.h file located in the inc directory of the Windows DDK.

Resolving the Problem

If you are not equipped to debug this problem, you should use some basic troubleshooting techniques. If a driver is identified in the bug check message, disable the driver or check with the manufacturer for driver updates. Try changing video adapters. Check with your hardware vendor for any BIOS updates. Disable BIOS memory options such as caching or shadowing.

If you plan to debug this problem, you may find it difficult to obtain a stack trace. Parameter 2 (the exception address) should pinpoint the driver or function that caused this problem.

If exception code 0x80000003 occurs, this indicates that a hard-coded breakpoint or assertion was hit, but the system was started with the /NODEBUG switch. This problem should rarely occur. If it occurs repeatedly, make sure a kernel debugger is connected and the system is started with the /DEBUG switch.

If exception code 0x80000002 occurs, the trap frame will supply additional information.

If the specific cause of the exception is unknown, the following should be considered:

Hardware incompatibility. First, make sure that any new hardware installed is listed on the Microsoft Windows Hardware Compatibility List (HCL).

Faulty device driver or system service. In addition, a faulty device driver or system service might be responsible for this error. Hardware issues, such as BIOS incompatibilities, memory conflicts, and IRQ conflicts can also generate this error.

If a driver is listed by name within the bug check message, disable or remove that driver. Disable or remove any drivers or services that were recently added. If the error occurs during the startup sequence and the system partition is formatted with NTFS file system, you might be able to use Safe Mode to rename or delete the faulty driver. If the driver is used as part of the system startup process in Safe Mode, you need to start the computer by using the Recovery Console to access the file.

If the problem is associated with Win32k.sys, the source of the error might be a third-party remote control program. If such software is installed, the service can be removed by starting the system using the Recovery Console and deleting the offending system service file.

Check the System Log in Event Viewer for additional error messages that might help pinpoint the device or driver that is causing bug check 0x1E. Disabling memory caching of the BIOS might also resolve the error.

You should also run hardware diagnostics, especially the memory scanner, supplied by the system manufacturer. For details on these procedures, see the owner's manual for your computer.

The error that generates this message can occur after the first restart during Windows Setup, or after Setup is finished. A possible cause of the error is a system BIOS incompatibility. BIOS problems can be resolved by upgrading the system BIOS version.

To get a stack trace if the normal stack tracing procedures fail

Use the KB (Display Stack Backtrace) debugger command to display parameters in the stack trace. Look for the call to NT!PspUnhandledExceptionInSystemThread. (If this function is not listed, see the note below.)

The first parameter to NT!PspUnhandledExceptionInSystemThread is a pointer to a structure, which contains pointers to an except statement:

```
typedef struct _EXCEPTION_POINTERS {
    PEXCEPTION_RECORD ExceptionRecord;
    PCONTEXT ContextRecord;
} EXCEPTION_POINTERS, *PEXCEPTION_POINTERS;
```

```
ULONG PspUnhandledExceptionInSystemThread(
    IN PEXCEPTION_POINTERS ExceptionPointers
)
```

Use the DD (Display Memory) command on that address to display the necessary data.

The first retrieved value is an exception record and the second is a context record. Use the .exr (Display Exception Record) command and the .cxr (Display Context Record) command with these two values as their arguments, respectively.

After the .cxr command executes, use the KB command to display a stack trace that is based on the context record information. This stack trace indicates the calling stack where the unhandled exception occurred.

Note This procedure assumes that you can locate NT!PspUnhandledExceptionInSystemThread. However, in some cases (such as an access violation crash) you will not be able to do this. In that case, look for ntoskrnl!KiDispatchException. The third parameter passed to this function is a trap frame address. Use the .trap (Display Trap Frame) command with this address to set the Register Context to the proper value. You can then perform stack traces and issue other commands.

Here is an example of bug check 0x1E on an x86 processor:

```
kd> .bugcheck get the bugcheck data
Bugcheck code 0000001e
Arguments c0000005 8013cd0a 00000000 0362cfff
```

microsoft.public.windowsxp.general: Re: windows error codes

kd> kb start with a stack trace

```
FramePtr RetAddr Param1 Param2 Param3 Function Name
8013ed5c 801263ba 00000000 00000000 fe40cb00 NT!_DbgBreakPoint
8013eccc 8013313c 0000001e c0000005 8013cd0a NT!_KeBugCheckEx+0x194
fe40cad0 8013318e fe40caf8 801359ff fe40cb00 NT!PspUnhandledExceptionInSystemThread+0x18
fe40cad8 801359ff fe40cb00 00000000 fe40cb00 NT!PspSystemThreadStartup+0x4a
fe40cf7c 8013cb8e fe43a44c ff6ce388 00000000 NT!_except_handler3+0x47
00000000 00000000 00000000 00000000 00000000 NT!KiThreadStartup+0xe
```

kd> dd fe40caf8 L2 dump EXCEPTION_POINTERS structure
0xFE40CAF8 fe40cd88 fe40cbc4 ..@...@.

kd> .exr fe40cd88 first DWORD is the exception record

Exception Record @ FE40CD88:

```
ExceptionCode: c0000005
ExceptionFlags: 00000000
Chained Record: 00000000
ExceptionAddress: 8013cd0a
NumberParameters: 00000002
Parameter[0]: 00000000
Parameter[1]: 0362cfff
```

kd> .cxr fe40cbc4 second DWORD is the context record

```
CtxFlags: 00010017
eax=00087000 ebx=00000000 ecx=03ff0000 edx=ff63d000 esi=0362cfff edi=036b3fff
eip=8013cd0a esp=fe40ce50 ebp=fe40cef8 iopl=0 nv dn ei pl nz ac po cy
vip=0 vif=0
cs=0008 ss=0010 ds=0023 es=0023 fs=0030 gs=0000 efl=00010617
0x8013cd0a f3a4 rep movsb
```

kd> kb kb gives stack for context record

ChildEBP RetAddr Args to Child

```
fe40ce54 80402e09 ff6c4000 ff63d000 03ff0000 NT!_RtlMoveMemory@12+0x3e
fe40ce68 80403c18 ffbc0c28 ff6ce008 ff6c4000 HAL!_HalpCopyBufferMap@20+0x49
fe40ce9c fe43b1e4 ff6cef90 ffbc0c28 ff6ce009 HAL!_IoFlushAdapterBuffers@24+0x148
fe40ceb8 fe4385b4 ff6ce388 6cd00800 ffbc0c28 QIC117!_kdi_FlushDMABuffers@20+0x28
fe40cef8 fe439894 ff6cd008 ffb6c820 fe40cf4c QIC117!_cq_d_CmdReadWrite@8+0x26e
fe40cf18 fe437d92 ff6cd008 ffb6c820 ff6e4e50 QIC117!_cq_d_DispatchFRB@8+0x210
fe40cf30 fe43a4f5 ff6cd008 ffb6c820 00000000 QIC117!_cq_d_ProcessFRB@8+0x134
fe40cf4c 80133184 ff6ce388 00000000 00000000 QIC117!_kdi_ThreadRun@4+0xa9
fe40cf7c 8013cb8e fe43a44c ff6ce388 00000000 NT!_PspSystemThreadStartup@8+0x40
```

Send feedback on this topic. / Built on Thursday, February 13, 2003

Knowledge Base

How to Troubleshoot a STOP 0x0000001E KMODE_EXCEPTION_NOT_HANDLED Error Message PSS ID Number: 275678

Article Last Modified on 10/14/2002

The information in this article applies to:

Microsoft Windows NT Server 4.0 Terminal Server Edition
Microsoft Windows 2000 Server
Microsoft Windows 2000 Advanced Server
Microsoft Windows 2000 Professional
Microsoft Windows NT Workstation 4.0
Microsoft Windows NT Server 4.0
Microsoft Windows NT Server, Enterprise Edition 4.0

This article was previously published under Q275678

SUMMARY

This article describes how to troubleshoot a "STOP 0x0000001E KMODE_EXCEPTION_NOT_HANDLED" error message.

MORE INFORMATION

A "STOP 0x0000001E KMODE_EXCEPTION_NOT_HANDLED" error message is a common type of STOP error message you may receive on a Windows NT-based computer. A "STOP 0x0000001E KMODE_EXCEPTION_NOT_HANDLED" error message indicates that an error condition was detected by the kernel and Windows NT was unable to continue running because of this error condition. The types of problems that can cause a "STOP 0x0000001E KMODE_EXCEPTION_NOT_HANDLED" error message are very similar to the problems that cause a "STOP 0x0000000A" error message, such as bad pointers, invalid addresses and other types of access violations. The STOP 0x0000001E bug check identifies an error that occurred in a section of code where no error handling routines exist. Note that most exceptions are generated directly in the section of code that is running.

The top four lines of a STOP 0x0000001E generally appear as:

```
STOP: 0x0000001E (0xAAAAAAAA,0xBBBBBBBB,0xCCCCCCCC,0xDDDDDDDD)
```

```
KMODE_EXCEPTION_NOT_HANDLED
```

```
AAAAAAAA from BBBBBBBB (CCCCCCCC,DDDDDDDD)
```

```
Address BBBBBBBB has base at XXXXXXXX – MODULE1.SYS Address CCCCCCCC has base at  
YYYYYYYY – MODULE2.SYS
```

The four hexadecimal parameters after the STOP code (0xAAAAAAAA, 0xBBBBBBBB, 0xCCCCCCCC, 0xDDDDDDDD) have the following meanings:

0xAAAAAAAA is a code that identifies the exception that was not handled.

0xBBBBBBBB is the address at which the exception occurred.

0xCCCCCCCC is the first parameter of the exception, and sometimes this is another address in code.

0xDDDDDDDD is the second parameter of the exception, which can vary in meaning.

Interpreting the Parameters

Usually the exception address identifies the driver or function that caused the problem. Always note this address and the date of the driver or image that contains this address.

The first parameter is a Windows NT error code, and all error codes are defined in the Ntstatus.h file (which can also be found in the Windows NT 4.0 SDK). The first parameter tells you the type of error.

The second parameter is also important because it tells you in what code module the error occurred. This can frequently point to an individual driver or piece of hardware that is at fault, which will generally be listed on the third line of the STOP screen.

The last two parameters vary depending upon the exception that has occurred. Typically, you can find a description of the parameters included with the name of the error code in the Ntstatus.h file.

If there were no parameters to the error code, these will be 0x00000000.

microsoft.public.windowsxp.general: Re: windows error codes

For example, in the following STOP error message, an access violation (0xC0000005) occurred in module Srv.sys, which is the kernel mode server service:

STOP: 0x0000001E (0xC0000005, 0xFCA733B9, 0x00000000, 0x00000000)

KMODE_EXCEPTION_NOT_HANDLED 0xC0000005 from 0xFCA733B9 (0x0, 0x0)Address FCA733B9 has base at FCA70000 – SRV.SYS

Note that no parameters went with this error code.

How to Troubleshoot a "STOP 0x0000001E KMODE_EXCEPTION_NOT_HANDLED" Error Message

Try to replace the driver that is identified in the STOP error message, either with a known good copy from your installation media, or with an updated version from the manufacturer.

Disable the driver that is identified in the STOP error message or any newly installed drivers.

Verify that any new hardware or software is properly installed. Disconnect the new hardware or replace it to see if this resolves the issue.

If you have a video driver that was not supplied with the operating system, try switching to the standard VGA driver or a driver that is compatible with the operating system.

View the following Microsoft Hardware Compatibility List Web site to verify that all your hardware and drivers are compatible with the operating system:

<http://www.microsoft.com/hcl>

Run any system diagnostics that are supplied by your computer manufacturer, especially a RAM check. If this is a new installation of the hardware or software, contact the manufacturer for any requires updates for drivers or firmware.

Disable all filter drivers, such as remote control software, antivirus programs, backup programs, and so on.

Keywords: kberrmsg kbhowto kbtshoot KB275678

Technology: kbNTTermServ400 kbNTTermServSearch kbwin2000AdvServ kbwin2000AdvServSearch kbwin2000Pro kbwin2000ProSearch kbwin2000Search kbwin2000Serv kbwin2000ServSearch kbWinAdvServSearch kbWinNT400search kbWinNTS400 kbWinNTS400search kbWinNTsearch kbWinNTSEnt400 kbWinNTSEntSearch kbWinNTSsearch kbWinNTW400 kbWinNTW400search kbWinNTWsearch

Send feedback to Microsoft

© 2003 Microsoft Corporation. All rights reserved.

--

<http://www.uscricket.com>

"motts" <motts@discussions.microsoft.com> wrote in message news:5EFC5E1B-5517-426D-8EAD-383FF4ECB

> It's a different stop message everytime (e.g. 0x0000001e, or 0x0000004d these
> may not be exactly right because i am at work right now). I can get the
> specific numbers when i get home this evening. But that is really what drives
> me crazy because the meeages are always different.

>

> "Will Denny" wrote:

>

>> Hi

>>

>> See if this helps:

>>

>> <http://groups.google.com/groups?q=ipnathlp&hl=en&btnG=Google+Search>

>>

>> Which particular Stop Code(s) were you getting?

>>

>> --

>>

>> Will Denny

>> MS-MVP - Windows Shell/User

>> Please reply to the News Groups

Re: windows error codes

microsoft.public.windowsxp.general: Re: windows error codes

```
>>
>>
>> "motts" <motts@discussions.microsoft.com> wrote in message
>> news:65546067-5ECC-47AA-9A73-35B6FEE34EE1@microsoft.com...
>> >I have had xp since april of 2004 and i have had to reinstall it three
>> >times
>> > because of stop messages and eventually not being able to boot into
>> > windows
>> > at all. The latest reinstall was about a month and a half ago and i am
>> > starting to get stop messages again. In looking at previous questions
>> > posted
>> > here someone had the same problem and it was suggested that it was being
>> > caused by norton a/v. I uninstalled that and i was still getting stop
>> > messages. In searching the web i saw that some of these problems were
>> > being
>> > caused by a creative sound card, so i uninstalled that and so far so good
>> > for
>> > the stop messages. But in event viewer i am getting an error code called
>> > "ipnathlp". Can anyone tell me what this means? I have an u s robotics
>> > 8054
>> > router connected to my computer. If this code is a serious problem how do
>> > i
>> > fix it?
>>
>>
>>
```