Can handled exceptions be seen with WinDbg

Posted by Will Steele - 07 Jan 2010 - 00:02

I am looking at this post: Tess' ASP blog. In it, she writes a simple throw exception for demonstration purposes. With this in mind, can any exception (in this case, a handled exception) be caught in WinDbg with the SOS extension? Conversely, are only unhandled exceptions caught by WinDbg? I saw she used the sxe command and thought it might be similar.

==========

Re:Can handled exceptions be seen with WinDbg

Posted by Robert Kuster - 17 Jan 2010 - 18:07

Hi Will,

here I go again. Sorry for the little delay this time.
Can handled exceptions be seen with WinDbg
Short answer: Nope. A handled/dismissed exception isn't an exception anymore.

Long answer: It depends. The first thing one should know about exceptions is that on Windows (either the environment is Win32, MFC, .NET, or the kernel) exceptions are handled by the OS. To us exceptions are made available through language extensions, for example through the try & except constructs in C++ or C#. Thus it is the OS or more precisely its exception dispatcher that takes care and dispatches exceptions. From the perspective of the exception dispatcher there are two different situations to cover:A debugger is attached to the application in questionNo debugger is attached to the application
In case there is a debugger the exception dispatcher does the following:
Attempts to notify the process's debugger (say, WinDbg). This is known as a first-chance exception.
If WinDbg handles the exception (gH - go, exception handled) that's it. In this case the exception dispatcher won't notify anyone else about the exception because it has been dismissed.If WinDbg didn't handle the exception, the OS then tries to locate a frame-base exception filter (a C++ or C# catch statement, for instance).
Again, if the exception filter handles the exception that's it. In this case the exception dispatcher won't notify anyone else about the exception.If along the way nobody handled the exception, the exception dispatcher will notify the associated debugger again. This is known as a second-chance or last-chance exception.
Again, if the associated debugger handles the exception that's it. Your application will continue to run fine as if nothing has happened.
But if the associated debugger doesn't handle even the second chance exception (for instance, in WinDbg: gN - go not handled command) the process will terminate.The bottom line:
if a debugger is attached, this debugger will always be notified about any exception in the first place (first-chance exception)once an exception is handled there is no way to display it in a debugger thereafter; no debugger will be notified about handled exceptionsFinally please check WinDbg. From A to Z! once again.
Refer to slides 17-20 about the exception dispatcher and exceptions in general.Refer to slides 85-86 about debugging exceptions with WinDbg.Refer to slides 89-92 about event filters in WinDbg. In particular you should enable CLR exceptions and CLR notification exceptions for managed applications as otherwise WinDbg will ignore them altogether.

Said all that there is one more tool to be aware off called Application Verifier. Application Verifier is a runtime verification tool that monitors an application's interaction with the Windows OS. Among other things it profiles and tracks all (handled an unhandled) exceptions that occur within a process. Again, check out WinDbg. From A to Z! and refer to slides 97-102. While Application Verifier will display handled exceptions to you, note that it is just a tracking/logging tool of events that occurred in the past. By no means it will halt a debugger on handled exceptions since this isn't how exceptions were meant to work in the first place.

Last but not least: You might also try out the sos!DumpAllExceptions command.

I hope this helps, RK :)

==========

Re: Can handled exceptions be seen with WinDbg

Posted by ephraimdov - 17 Jun 2015 - 08:14

_____

More about....Exception HandlingException Handling

Dov

==========