Wrong display of function Names by WinDbg

Posted by jd - 13 May 2010 - 20:26

I'm debugging an application for which I have no source. The app is crashing in a dll. In IDA pro I see:

call ds:__imp_strncpy which is the culprit of the crash

however in WinDBG I see:

call dword ptr

That's not what I want. I want WinDBG to show me exactly what IDA does. Can this be done?

==========

Re: Wrong display of function Names by WinDbg

Posted by Robert Kuster - 28 May 2010 - 16:43

JD, welcome.

Whenever you see a large offset like this mydll!wrongfuncanme+0x33360 it is always a sign of missing symbols.

Now strncpy is part of the CRT library and thus resides in MSVCR80.DLL or one of its siblings (usually there comes a new CRT version with every Visual Studio release). Because MSVCR80.DLL is a Microsoft DLL you can actually get its public PDB symbols from the Microsoft server. More precisely WinDbg will fetch the correct symbols automatically for you from the server, if you set it up correctly. You might read WinDbg. From A to Z! - "Symbols in WinDbg" at slide 24. Or simply check out the .symfix+ command...

If you aren't able to get the right PDB files any serious debugger will read at least the export symbols (functions) of your modules. Example: If you open MSVCR80.DLL in Dependency Walker you see all its exported functions. This is the same information that is read and used by IDA or by WinDbg if they fail to get the PDBs. For example in WinDbg:

```
0:000> ld MSVCR80
*** ERROR: Symbol file could not be found.  Defaulted to export symbols for
C:WINDOWSWinSxSx86_Microsoft.VC80.CRT_1fc8b3b9a1e18e3b_8.0.50727.4053_x-ww_e6967989M
SVCR80.dll
```

Bottom line: WinDbg should show you and will show you the same symbol information as IDA does.

I hope this helps,
Robert

==========