function plus offset question Posted by Cliff Hupper - 30 Jul 2010 - 20:20

Hi, when I'm looking at a threads call stack what do the numbers after the function mean?

As indicated below as

libxx!classxx::Perform+0x4fe

and

libxx!classxx::Perform+0x60c

what are the +0x4fe and 0x60c?

Thanks.

2 Id: 330.370 Suspend: 1 Teb: 7ffdc000 Unfrozen

ChildEBP RetAddr

00deff14 7c90df5a ntdll!KiFastSystemCallRet

00deff18 7c8025db ntdll!ZwWaitForSingleObject+0xc

00deff7c 7c802542 kernel32!WaitForSingleObjectEx+0xa8

00deff90 0040da14 kernel32!WaitForSingleObject+0x12

00deffa8 0040db22 libxx!classxx::Perform+0x4fe

00deffb4 7c80b729 libxx!classxx::Perform+0x60c

00deffec 00000000 kernel32!BaseThreadStart+0x37

Re: function plus offset question Posted by huku - 02 Aug 2010 - 18:59

The 0x* numbers are the offsets within a function code where a 'call' instruction is issued.

ntdll!KiFastSystemCallRet ntdll!ZwWaitForSingleObject+0xc

This means that ZwWaitForSingleObject calls KiFastSystemCallRet via a 'call' instruction located 0xc bytes away from the start of ZwWaitForSingleObject's code.

Re: function plus offset question Posted by Cliff Hupper - 02 Aug 2010 - 22:41

And you're talking bytes as in the function + number of bytes of compiled code at runtime/debugging, not number of bytes of src correct?

Thanks.

PDF created: 05 Nov 2025, 11:50

Thinking debugging? Thing www.windbg.info! PDF created: 05 Nov 2025, 11:50 Re: function plus offset question Posted by huku - 02 Aug 2010 - 23:38 Yeap... exactly. Re: function plus offset question Posted by Cliff Hupper - 03 Aug 2010 - 00:04 Great! Thanks for the help. I have so much trouble sorting through these details in the normal help file / chm.

www.windbg.info - WinDbg.info

Thanks again!