Reverse Engineering Beta

Where can I find the 64-bit version of shell32.dll on Windows?

3

I'm trying to debug Control Panel and I'd like to disassemble `shell32.dll`. Because control panel is a 64-bit executable, it loads the 64-bit version of the dll (contrary to the name). When I view the disassembled code in debug mode, I can confirm that it is indeed 64-bit. Ida claims that it's located at `C:\WINDOWS\system32\shell32.dll`; however this dll is entirely 32-bit. I also checked `C:\WINDOWS\SysWOW64\shell32.dll`, but it's also 32-bit.

Can someone explain what's going on here?

Thanks!

asked Oct 29 '15 at 22:52

George Rampleson
18 1 5

2 Answers

3

What you are seeing is the result of WoW file-system redirection. The effect occurs when a 32-bit executable requests a copy of a file in the Windows directory. Since a 64-bit result would probably make no sense to a 32-bit executable, you get the 32-bit copy instead.

If you use Explorer to copy the file from the system32 directory, and then examine the result, you will find that it is a 64-bit executable.

answered Oct 30 '15 at 14:52

peter ferrie
4,176 1 11 30

3

As @peter-ferrie said, 32-bit processes will use `C:\WINDOWS\SysWOW64\shell32.dll` instead of `C:\WINDOWS\system32\shell32.dll` if you specify `C:\WINDOWS\system32\shell32.dll` .

To force a 32-bit process to use the actual 64-bit version, you can specify the following file path: `C:\WINDOWS\Sysnative\shell32.dll`

This saves you the trouble of having to use Explorer to make a copy of the 64-bit DLL.

answered Nov 2 '15 at 19:35

Jason Geffner
19.3k 1 25 59

oops i was actually wondering how i can force my 32 bit process to load a 64 bit dll then reread the whole qa to come to the conclusion that you mean use sysnative to force 32 bit ida to load a 64 bit dll is that right – blabb Nov 2 '15 at 21:24

Yes, that's right. – Jason Geffner Nov 2 '15 at 21:48

+1 for the `Sysnative` folder – rev Nov 2 '15 at 23:54