

RE: Enumerating modules inside a process in a 64 bit system

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework/2007-09/msg00374.html>

- *From:* wawang@xxxxxxxxxxxxxxxxxxxxxx ("Walter Wang [MSFT]")
 - *Date:* Thu, 27 Sep 2007 10:00:36 GMT
-

Hi Ramon,

Welcome to MSDN Managed Newsgroup!

Internally .NET's Process.Modules is using function EnumProcessModules from PSAPI.dll. This function has a known issue that it cannot work across 32/64 bit process boundary. Therefore enumerating another 64-bit process from 32-bit process or vice versa doesn't work correctly.

For an .NET process, by default it's compiled as "Any CPU", the generated IL code will JIT (Just-in-Time) compiled as 32-bit or 64-bit native code if you run it on a 32-bit or 64-bit OS accordingly.

The Office system now only has 32-bit version.

Therefore, when you run your "Any CPU"-compiled assembly on x64 system to enumerate WinWord.exe, the result is not complete.

We fixed this issue by adding a new function called EnumProcessModulesEx to PSAPI.dll (<http://msdn2.microsoft.com/en-us/library/ms682633.aspx>), but we currently cannot use it in this case:

- * it only works on Windows Vista or Windows Server 2008
- * currently .NET 2.0 Framework don't have a service pack or hotfix to make Process.Modules use this new API

As a workaround, if your application currently only needs to enumerate modules of WinWord, since WinWord is always 32-bit, I suggest you compile your .NET application specifically as 32-bit process instead of AnyCPU. This way it could correctly enumerate process of WinWord. Here's the steps on how to do this:

- * Select your project in Visual Studio 2005 Project Explorer, open menu "Build/Configuration Manager", this will bring up the "Configuration Manager" dialog
- * In the "Active solution platform" combobox, select "<New...>", this will bring up the "New Solution Platform" dialog

RE: Enumerating modules inside a process in a 64 bit system

- * In the "Type or select the new platform" combobox, select "x86", click button "OK".
- * Click button "Close" to close "Configuration Manager" dialog
- * Now you can build your assembly as 32-bit.

I hope this helps. Please feel free to let me know if there's anything else I can help.

Sincerely,
Walter Wang (wawang@xxxxxxxxxxxxxxxxxxxxxx, remove 'online.')

Microsoft Online Community Support

=====
For MSDN subscribers whose posts are left unanswered, please check this document: <http://blogs.msdn.com/msdnets/pages/postingAlias.aspx>

Get notification to my posts through email? Please refer to <http://msdn.microsoft.com/subscriptions/managednewsgroups/default.aspx#notifications>. If you are using Outlook Express/Windows Mail, please make sure you clear the check box "Tools/Options/Read: Get 300 headers at a time" to see your reply promptly.

Note: The MSDN Managed Newsgroup support offering is for non-urgent issues where an initial response from the community or a Microsoft Support Engineer within 1 business day is acceptable. Please note that each follow up response may take approximately 2 business days as the support professional working with you may need further investigation to reach the most efficient resolution. The offering is not appropriate for situations that require urgent, real-time or phone-based interactions or complex project analysis and dump analysis issues. Issues of this nature are best handled working with a dedicated Microsoft Support Engineer by contacting Microsoft Customer Support Services (CSS) at <http://msdn.microsoft.com/subscriptions/support/default.aspx>.

=====
This posting is provided "AS IS" with no warranties, and confers no rights.

.