

## Get Public Key

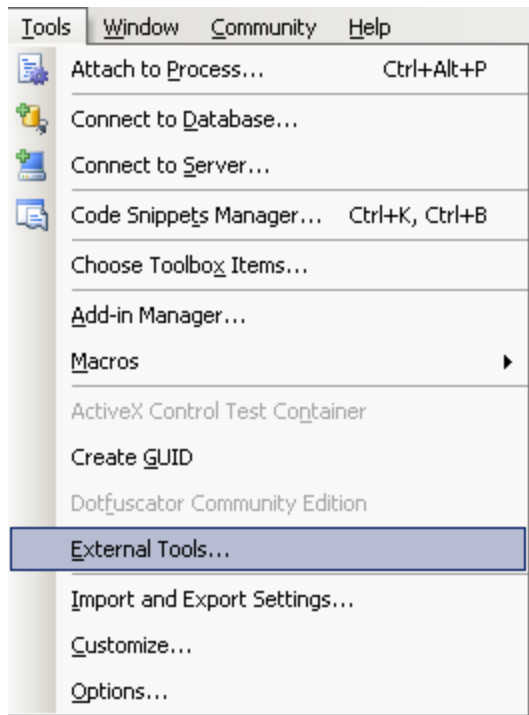
VS.NET 2005 makes it quite easy to sign .NET assemblies with a strong name key public key/private key pair directly from the properties of the project. But after the assembly is compiled, nothing in the default VS.NET 2005 IDE can obtain the public key that was embedded within the assembly. So, the VS.NET `sn.exe` command-line tool is often used to manually peek into the assembly for the public key using the following command:

```
sn -Tp [pathToAssembly]\[assemblyFileName]
```

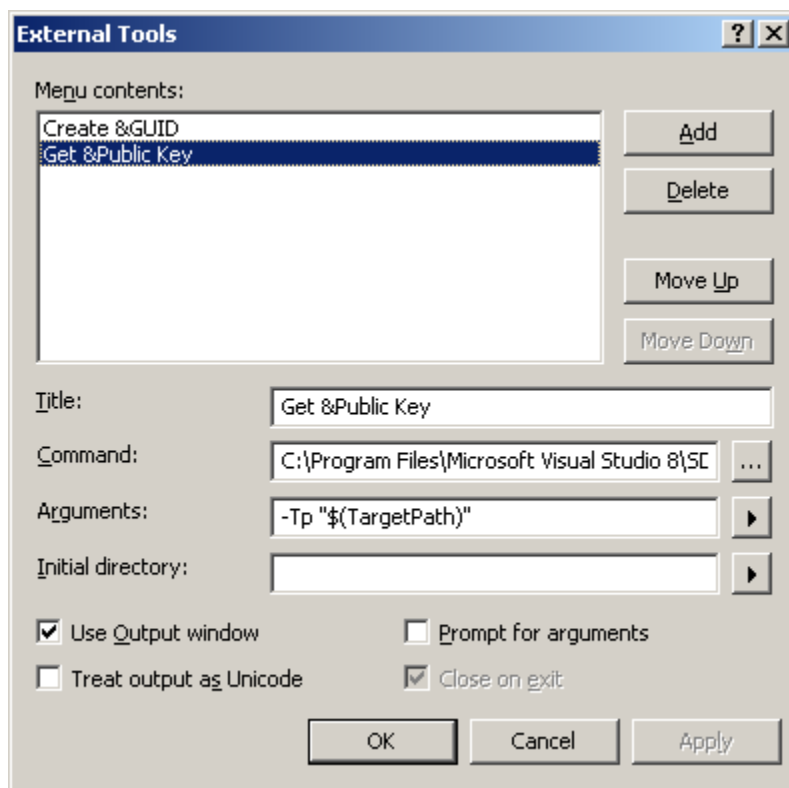
While this is not difficult, it is terribly inefficient. Not only does it require the developer to open a command window, but also remember the command, parameters, the case of those parameters, locate or type the full path to the assembly, and execute the command. What's worse, VS.NET knows exactly where the assembly was created; it just does not provide an interface to peek into the assembly to see the public key token. It is also possible to drag the DLL into the GAC to see the public key token but not the public key blob; .NET Reflector can also show an assembly's public key token.

However, there is hope. VS.NET allows developers to setup external tools. The `sn.exe` command-line tool is external. So, with a little configuration of the IDE, a **Get Public Key** icon can be included directly on the toolbar. Then, just click the icon to get the public key token and public key blob from any compiled assembly that is in focus in the Solution Explorer. Here are the steps:

1. From the VS.NET IDE, choose the **External Tools** option from the **Tools** menu:



2. Click the **Add** button.

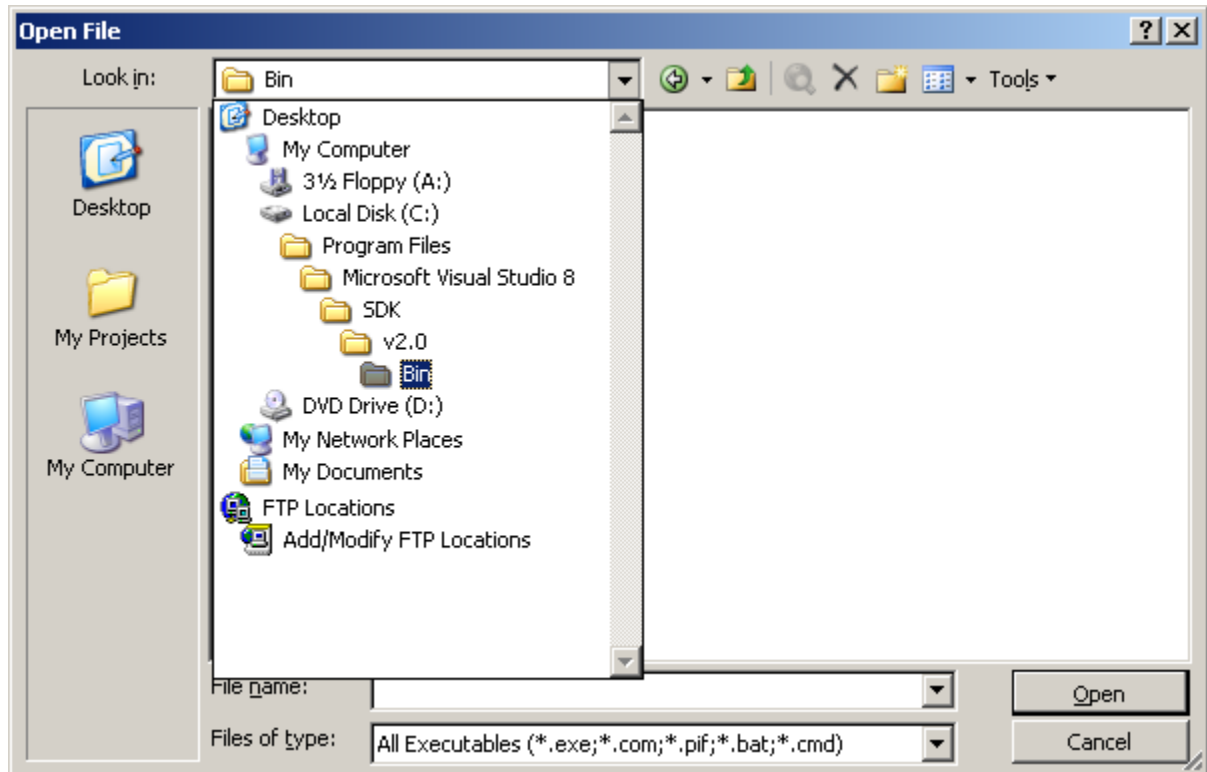


3. For the **Title**, enter something like **Get &Public Key** (the ampersand in front of the P will allow quick key access to the new menu option):

4. For the **Command**, click the ellipsis button to browse and select the `sn.exe` command.

By default, it can be found in `c:\Program File\Microsoft Visual Studio`

`8\SDK\v2.0\Bin` directory:



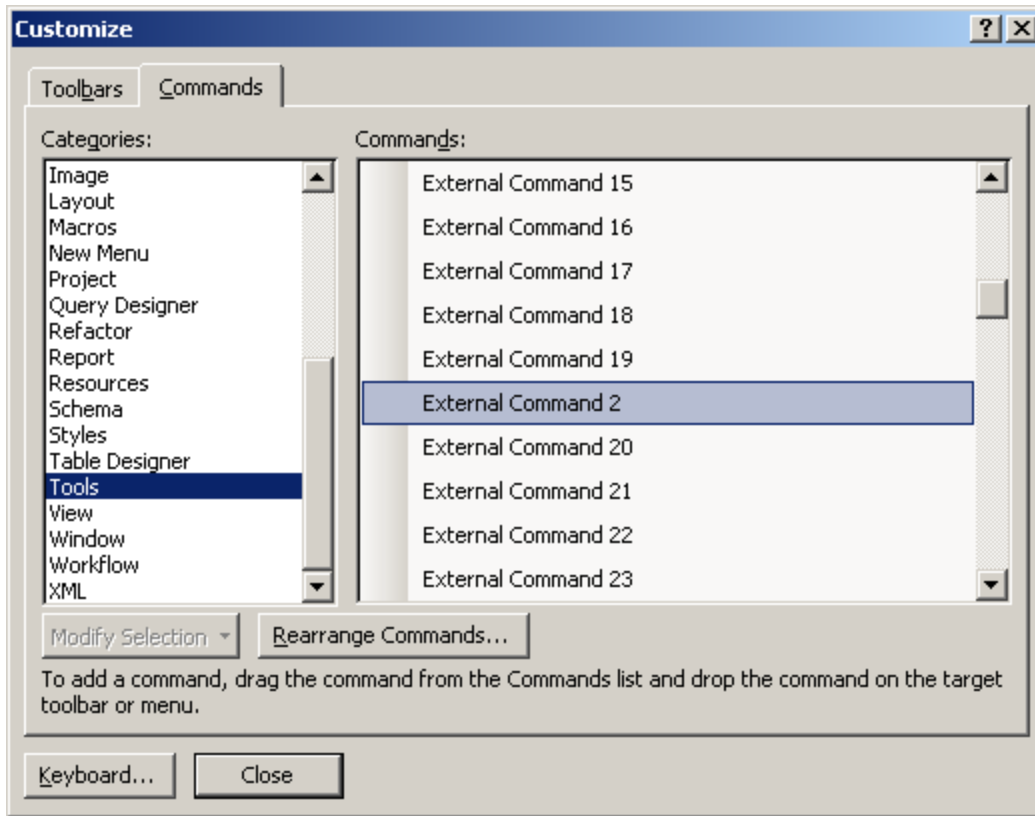
5. For the Arguments, enter `-Tp "$ (TargetPath) "`. The capital "T" provides the public key token for the assembly. Be sure not to use a lowercase "t" as that will provide the public key token for the file not the assembly. It looks similar, but it is not what WSS v3 or CAS will be looking for. The lowercase "p" will provide the public key blob which may be needed when working with CAS. The `"$ (TargetPath) "` (don't forget the quotes) will provide the location where VS.NET created the assembly including the full path, assembly name, and extension. All the variables that VS.NET knows in this context

can be viewed by clicking the arrow to the right of this textbox. Placing quotes around this variable will ensure that the command executes properly even if there are spaces in the path.

6. Select the checkbox to the left of the **Use Output window** option. This way the output will be displayed within the VS.NET IDE.
7. Click the **OK** button to add the command to the list of External Tools on the VS.NET **Tools** menu.

The **Get Public Key** command can be used immediately from the **Tools** menu; just press **<Ctrl+T+p>**. Optionally, follow the following steps to put this new command onto a toolbar to make it really handy:

8. Right-click anywhere in the toolbar area and choose **Customize...** (the last option) from the content menu.
9. On the **Commands** tab, choose **Tools** from the **Categories** list:

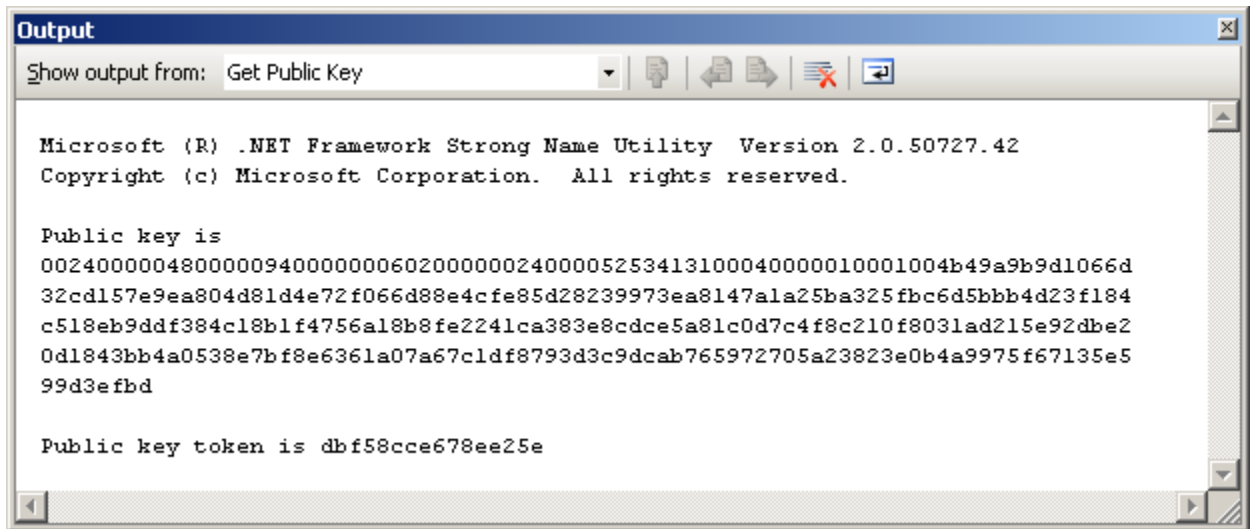


10. Scroll down in the **Commands** list until **External Command n** is visible (n is the number of commands down the External Tools list in the Tools menu the new **Get Public Key** is found; in the student VPC environment this will be **External Command 2**).
11. Select the **External Tools 2** option and drag it to a toolbar that will make it easy to choose when the public key is needed (note, placing this option onto a toolbar will show as External Command 2 until the window is closed; also, placing this option onto the menu containing the Help option would require additional setup not covered here.)



12. Click the **Close** button.

13. Be sure that the current project has focus in the Solution Explorer and that the project has been signed (AKA: strong named) and compiled. Then click the **Get Public Key** icon in the toolbar or **<Ctrl+T+p>** and an **Output** window similar to the following will be generated within the VS.NET IDE:



The screenshot shows the 'Output' window in the Visual Studio .NET IDE. The window title is 'Output' and it has a toolbar with icons for 'Show output from', 'Copy', 'Paste', 'Print', and 'Refresh'. The 'Show output from' dropdown is set to 'Get Public Key'. The output text is as follows:

```
Microsoft (R) .NET Framework Strong Name Utility Version 2.0.50727.42
Copyright (c) Microsoft Corporation. All rights reserved.

Public key is
00240000048000009400000006020000002400005253413100040000010001004b49a9b9d1066d
32cd157e9ea804d81d4e72f066d88e4cfe85d28239973ea8147a1a25ba325fbc6d5bbb4d23f184
c518eb9ddf384c18b1f4756a18b8fe2241ca383e8cdce5a81c0d7c4f8c210f8031ad215e92dbe2
0d1843bb4a0538e7bf8e6361a07a67c1df8793d3c9dcab765972705a23823e0b4a9975f67135e5
99d3efbd

Public key token is dbf58cce678ee25e
```