.NET Framework 4

**ASP.NET IIS Registration Tool (Aspnet\_regiis.exe)**

The ASP.NET IIS Registration tool (Aspnet\_regiis.exe) is used to register ASP.NET applications with Internet Information Services (IIS). This topic describes the options, syntax, and other information for using the tool.

Using the ASP.NET IIS Registration tool, you can perform tasks such as the following:

* Register or remove the .NET Framework ASP.NET installation with IIS.
* Create new ASP.NET application pools.
* Display the status of all installed versions of ASP.NET.

|  |
| --- |
| **Description: Important noteImportant** |
| The version of Aspnet\_regiis that is included with the .NET Framework version 4 has new features and capabilities, which are summarized in the [Options](http://msdn.microsoft.com/en-us/library/k6h9cz8h.aspx#Options) section. However, these features are available only when you install the .NET Framework 4 on Windows Vista, Windows Server 2008, or Windows 7. If you install the .NET Framework version 4 on any version of Windows XP or Windows Server 2003, the tool works the same as in version 3.5 and earlier of the .NET Framework. For details about how Aspnet\_regiis works on Windows XP or Windows Server 2003, see [ASP.NET IIS Registration Tool for the .NET Framework Version 3.5](http://go.microsoft.com/fwlink/?linkid=139901). |

The .NET Framework 4 can be installed side-by-side with previous versions of the .NET Framework on a single computer. If IIS was previously enabled on the computer, the setup process for the .NET Framework automatically registers ASP.NET 4 with IIS. However, if you install the .NET Framework 4 before you enable IIS, you must run the ASP.NET IIS Registration tool in order to register the .NET Framework with IIS and create application pools that use the .NET Framework 4.

|  |
| --- |
| **Description: Caution noteCaution** |
| For information about how to find the correct version of Aspnet\_regiis.exe, see [Finding the Correct Version of Aspnet\_regiis.exe](http://msdn.microsoft.com/en-us/library/k6h9cz8h.aspx#findingthecorrectversion) later in this document. |

aspnet\_regiis [options]

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifOptions

The **i**, **ir**, **iru** and **enable** options are typically the only options that you require in order to register ASP.NET applications with IIS.

|  |  |
| --- | --- |
| **Option** | **Description** |
| **-c** | Legacy option that does not apply when you run applications under ASP.NET version 4. For information about how this option works with previous versions of ASP.NET, see [ASP.NET IIS Registration Tool for the .NET Framework Version 3.5](http://go.microsoft.com/fwlink/?linkid=139901). |
| **-disable** | On IIS 6.0 and on IIS 7.0 running in Classic mode, disables the ASP.NET ISAPI extension. The command does not apply to IIS 7.0 in Integrated mode.  This option can be combined only with the **-i**, **-ir**, or **-r** options. It is not available with IIS versions earlier than IIS 6.0. |
| **-e** | Removes the client scripts for ASP.NET from the aspnet\_client subdirectory of each IIS site directory.  Only the client scripts for the ASP.NET version that is associated with Aspnet\_regiis.exe are removed. |
| **-ea** | Removes the client scripts for all versions of ASP.NET from the aspnet\_client subdirectory of each IIS site directory. |
| **-enable** | On IIS 6.0 and on IIS 7.0 in Classic mode, option enables the ASP.NET ISAPI extension. The command does not apply to IIS 7.0 in Integrated mode.  This option can be combined only with the **-i**, **-ir**, or **-r** options. It is not available with IIS versions earlier than IIS 6.0 or with ASP.NET versions earlier than ASP.NET version 2.0. |
| **-ga** *user* | Grants the specified user or group access to the IIS metabase and other directories that are used by ASP.NET. You typically use this option when you are creating a custom service account. For more information, see [How to: Create a Service Account for an ASP.NET 2.0 Application](http://go.microsoft.com/fwlink/?LinkId=189143).  This option is not available with ASP.NET versions earlier than ASP.NET 2.0.  For installations on Windows Vista, Windows Server 2008, or Windows 7, this option enables you to set ACLs for groups.  Description: Note**Note**  The version of Aspnet\_regiis.exe that is provided in ASP.NET versions 3.5 and earlier fails when specifying a local user or group. |
| **-i** | Installs ASP.NET 4 and updates existing applications to use the ASP.NET 4 version of the application pool. It updates both the IIS Classic mode and the IIS Integrated mode handler and script mappings in the IIS metabase.  This option creates two application pools, ASP.NET v4.0 and ASP.NET v4.0 Classic. The DefaultAppPool application pool and the Classic .NET AppPool application pools are set to the .NET Framework 4 version of the CLR.  The 32-bit version of Aspnet\_regiis.exe adds only 32-bit scriptmaps to the Applicationhost.config file. The 64-bit version of Aspnet\_regiis.exe adds 32-bit and 64-bit scriptmaps to the Applicationhost.config file. This is done because 64-bit versions of Windows Vista, Windows Server 2008, or Windows 7 installations that have IIS 7.0 enabled will also have Windows on Windows 64-bit (WOW64) support available.  Description: Note**Note**  In the **Windows Features** dialog box of the operating system, the option for ASP.NET integration is not selected. (This option is in the **Internet Information Services** section.) The **Windows Features** dialog box displays only the status of ASP.NET 2.0 integration. It does not display information about version 4. |
| **-ir** | Installs and registers ASP.NET 4. This option is the same as the **-i** option except that this option does not change the CLR version that is associated with any existing application pools. |
| **-iru** | If ASP.NET is not currently registered with IIS, performs the tasks described for **-i**.  If a previous version of ASP.NET is already registered with IIS, this option performs the tasks described for **-ir**. |
| **-k** *path* | This option is not supported under Windows Vista, Windows Server 2008, or Windows 7.  For information about how this option works when you run under Windows XP or Windows Server 2003, see [ASP.NET IIS Registration Tool for the .NET Framework Version 3.5](http://go.microsoft.com/fwlink/?LinkID=139901). |
| **-kn** *path* | This option is not supported under Windows Vista, Windows Server 2008, or Windows 7.  For information about how this option works when you run under Windows XP or Windows Server 2003, see [ASP.NET IIS Registration Tool for the .NET Framework Version 3.5](http://go.microsoft.com/fwlink/?LinkID=139901). |
| **-lk** | Lists the path and version of all IIS metabase keys where ASP.NET is mapped. This option is not supported on Windows Vista and later versions.  The keys that inherit ASP.NET scriptmaps from a parent key are not displayed. |
| **-lv** | Lists the status and installation path of all versions of ASP.NET that are installed on the computer. |
| **-norestart** | This option is not supported under Windows Vista, Windows Server 2008, or Windows 7.  For information about how this option works when you run under Windows XP or Windows Server 2003, see [ASP.NET IIS Registration Tool for the .NET Framework Version 3.5](http://go.microsoft.com/fwlink/?LinkID=139901). |
| **-r** | Performs the tasks described for **-i** and in addition switches all application pools to use the .NET Framework 4 CLR. |
| **-s** *path* | Updates scriptmaps and application-pool assignments for the specified application and for all sub-applications.  For a non-root Web application (not W3SVC\ or W3SVC\ROOT), the application pool settings for the non-root Web application and all its sub-applications are copied to a new application pool named ASP.NET V4.0 *original application pool name*, and the non-root Web application is then moved to the new application pool. The new application pool has the same settings as the original application pool, except that the copy is mapped to the .NET Framework 4 CLR and has ASP.NET 4 settings.  If the new application pool ASP.NET V4.0 *original application pool name* already exists, the step of creating and cloning the application pool is skipped. Instead, Aspnet\_regiis.exe uses the existing application pool and sets it to use the .NET Framework 4 CLR, but makes no other changes. The tool then moves the specified application and its sub-applications to the corresponding application-pool copy. Existing application-pool settings are left unchanged.  If the specified application is the Web root (W3SVC\ or W3SVC\ROOT), Aspnet\_regiis.exe updates all application pools to use the .NET Framework 4 CLR and ASP.NET 4. |
| **-sn** *path* | Performs the tasks described for **-s**, except that **-sn** is not recursive. Updates scriptmaps and application-pool assignments for only the specific application. No sub-applications are changed.  If the specified application is the Web root (W3SVC\ or W3SVC\ROOT), Aspnet\_regiis.exe changes the application pools named DefaultAppPool, Classic .NET AppPool, ASP.NET V4.0, and ASP.NET V4.0 Classic" to use the .NET Framework 4 CLR and ASP.NET 4. |
| **-u** | For uninstallation, leaves the two 4 versions of the application pools, leaves any applications that were in the two version 4 application pools.  This option also removes the **managedRuntimeVersion** attribute of application pools that were mapped to the .NET Framework 4 CLR. This reverts the application pools to the IIS 7.0 default value for the runtime, which is 2.0 on Windows Vista, Windows Server 2008, or Windows 7.  The option removes all explicitly set version 4 handler and module mappings. It also removes all scriptmaps (32-bit and 64-bit), regardless of whether the 32-bit or 64-bit version of Aspnet\_regiis.exe is used. |
| **-ua** | For uninstallation, leaves the two 4 versions of the application pools, and leaves any applications that were in the two version 4 application pools.  This option removes all ASP.NET registrations (both versions 2.0 and 4) from the Applicationhost.config file. It also removes all scriptmaps (32-bit and 64-bit), regardless of whether the 32-bit or 64-bit version of Aspnet\_regiis.exe is used.  Existing application pools will still be associated with the .NET Framework 2.0, because that is the default value for the **managedRuntimeVersion** attribute. |
| **-?** | Displays the command syntax and options for the ASP.NET IIS Registration tool. |

**Configuration Options**

|  |  |
| --- | --- |
| **Option** | **Description** |
| **-config+** | Enables remote access to ASP.NET configuration. |
| **-config-** | Disables remote access to ASP.NET configuration. |
| **-pa** *container account* | Grants permission for the specified user or group *account* to access the specified key container. This option can take the following modifiers:   * **-pku** Substitutes a user-specified container for the default machine container. * **-csp** *provider* Specifies the container provider to use. * **-full** Specifies that full access should be added (by default, access is read-only). |
| **-pc** *container* | Creates an RSA public/private key pair in the specified container. This option can take the following modifiers:   * **-size** *keySize* Specifies the key size. The default is 2048 bytes. * **-pku** Substitutes a user-specified container for the default key container. * **-exp** Specifies that private keys must be able to be exported. * **-csp** *provider* Specifies the container provider to use. |
| **-pd** *section* | Decrypts the configuration section. This argument can take the following parameters:   * **-app** *virtualPath* Specifies that decryption should occur at the level of the included path. * **-location** *subPath* Specifies the subdirectory to decrypt. * **-pkm** Specifies that the Machine.config file should be decrypted instead of the Web.config file. |
| **-pdf** *section webApplicationDirectory* | Decrypts the specified configuration section of the Web.config file in the specified physical (not virtual) directory. |
| **-pe** *section* | Encrypts the specified configuration section. This option can take the following modifiers:   * **-prov** *provider* Specifies the encryption provider to use. * **-app** *virtualPath* Specifies that encryption should occur at the level of the included path. * **-location** *subPath* Specifies the subdirectory to encrypt. * **-pkm** Specifies that the Machine.config file should be encrypted instead of the Web.config file. |
| **-pef** *section webApplicationDirectory* | Encrypts the specified configuration section of the Web.config file in the specified physical (not virtual) directory. |
| **-pi** *container file* | Imports an RSA public/private key pair to the specified *container* from the specified XML *file*. This option can take the following modifiers:   * **-pku** Substitutes a user-specified container for the default machine container. * **-exp** Specifies that private keys can be exported. * **-csp** *provider* Specifies the container provider to use. |
| **-pr** *container account* | Removes permission for the specified user or group *account* to access the specified key container. This option can take the following modifiers:   * **-pku** Substitutes a user-specified container for the default machine container. * **-csp** *provider* Specifies the container provider to use. |
| **-px** *container file* | Exports an RSA public/private key pair from the specified container to the specified XML file. This option can take the following modifiers:   * **-pku** Substitutes a user-specified container for the default machine container. * **-csp** *provider* Specifies the container provider to use. |
| **-pz** *container* | Deletes the specified key container. This option can take the following modifier:   * **-pku** Substitutes a user-specified container for the default machine container. |

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifRemarks

When multiple versions of ASP.NET are installed on a computer, the versions of ASP.NET are said to be running *side-by-side*. In this scenario, you must configure IIS to indicate which version of the ASP.NET ISAPI (aspnet\_isapi.dll) should process a page in an ASP.NET application. In IIS 7.0, in both Classic and Integrated mode, the CLR version set for the application pool determines which version of ASP.NET and which associated version of the CLR are used to run an application. In IIS 6.0, a scriptmap is used to associate an application with a specific ASP.NET ISAPI version, and the ISAPI version in turn determines which version of the CLR is used to run the application.

There are 32-bit and 64-bit versions of Aspnet\_regiis.exe. For Windows Vista and Windows Server 2008, you can run the 32-bit version of the tool on a 64-bit installation of IIS. This lets you run mixed 32-bit and 64-bit worker processes on the same 64-bit system. The following table lists special cases and limitations that apply when you run the 32-bit version of the tool on a 64-bit version of IIS.

|  |  |
| --- | --- |
| **IIS Version** | **Special cases for 32-bit versions of Aspnet\_regiis.exe** |
| 6.0 | You can run the 32-bit version of Aspnet\_regiis.exe under a 64-bit IIS 6.0 installation on Windows Server 2003 with the following restrictions:   * The 32-bit version runs only if IIS has been set to run in 32-bit mode. * The 64-bit version runs only if IIS has been set to run in 64-bit mode. |
| 7.0 | On Windows Vista, Windows Server 2008, or Windows 7, when you run the 32-bit version of Aspnet\_regiis.exe under IIS 7, the tool does not automatically set the **enable32BitAppOnWin64** attribute of the **applicationPools** element. On a default 64-bit operating system installation, with Windows on Windows 64 enabled, you can use 32-bit Aspnet\_regiis.exe to configure IIS 7, but you will not be able to run any 32-bit applications, unless you explicitly change the **enable32BitAppOnWin64** attribute to enable running the application in Windows on Windows 64. For more information about IIS 7.0 configuration options, see the documentation for the [applicationPools](http://go.microsoft.com/fwlink/?LinkID=135728) element in the Applicationhost.config file. |

The ASP.NET IIS Registration tool is commonly used with the **-s** or **-sn** option to remap an ASP.NET application to the version of the .NET Framework that it is associated with. Use the **-s** option to update an application at the specified root path and in all of its subdirectories. If you do not want applications in subdirectories to be updated, use the **-sn** option.

|  |
| --- |
| **Description: NoteNote** |
| The *path* parameter refers to the root path of the application, not the physical path. For example, W3SVC/1/ROOT/SampleApp1. |

You can also use the ASP.NET IIS Registration tool to view information about ASP.NET. To list the status and installation path of all of the versions of ASP.NET that are installed on your computer, use the **-lv** option.

The **-norestart** option prevents a restart of the World Wide Web Publishing Service after ASP.NET script maps are installed or updated. If you are running multiple Web sites in IIS and they are in various application pools, and if you do not want all of them to be restarted, use this option.

Any changes made using the **ASP.NET** tab on the Internet Information Service console application also cause all application pools to be recycled. If you want to change the ASP.NET version that an application uses without causing all application pools to be recycled, run the following two commands to change the version.

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl40_ctl00_ctl02_code');" \o "Copy Code)

aspnet\_regiis -s w3svc/<instance>/root -norestart

iisapp /a <app pool> /r

The ASP.NET IIS Registration tool can also be used to install and uninstall the linked version of ASP.NET. Use the **-i** option to install ASP.NET and to update the script maps of all existing ASP.NET applications. Use the **-ir** option to install ASP.NET without updating the script maps. To uninstall the ASP.NET version that is associated with the tool, use the **-u** option. If you want to uninstall all versions of ASP.NET from the computer, use the **-ua** option.

You can also use the ASP.NET IIS Registration tool to view information about ASP.NET. To list the status and installation path of all of the versions of ASP.NET that are installed on your computer, use the **-lv** option. If you want to see the paths to all of the IIS metabase keys where ASP.NET is mapped, use the **-lk** option.

You can use Aspnet\_regiis.exe to install and remove system client-side script, such as script for client-side validation. Use the **-c** option to install client script for the version of ASP.NET that is associated with the tool. (Script is installed in the Aspnet\_client subdirectory of each IIS site directory.) To remove the client-side script for just the ASP.NET version that is associated with the tool, use the **-e** option. To remove the client-side script for all installed versions of ASP.NET, use the **-ea** option.

When you are using impersonation and if the impersonation account does not have Read access to the IIS metabase, a [COMException](http://msdn.microsoft.com/en-us/library/system.runtime.interopservices.comexception.aspx) exception with the message "Access is denied" is thrown when you try to access [DirectoryEntry](http://msdn.microsoft.com/en-us/library/system.directoryservices.directoryentry.aspx). This can occur when the Code subdirectory of the application exists in the application root directory. This error can also occur when an application is mapped to a UNC share and the UNC account does not have Read access to the metabase. In both cases, running the ASP.NET IIS Registration tool with the **-ga** *user* option for the impersonation or UNC account fixes the issue.

For more information about scriptmaps and application root paths, see the [IIS documentation](http://go.microsoft.com/fwlink/?LinkId=47856) on the Microsoft Web site.

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifExamples

The following command installs the ASP.NET version that is associated with the version of the ASP.NET IIS Registration tool that you are using, and updates existing ASP.NET applications. On IIS 6.0, the command updates the scriptmaps of existing ASP.NET applications that are currently mapped to an earlier version of ASP.NET. On IIS 7.0, the command updates both classic mode and integrated mode handlers in the IIS metabase. It also switches the DefaultAppPool and the Classic .NET AppPool applications pools to use the .NET Framework 4 CLR.

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl41_ctl00_ctl00_code');" \o "Copy Code)

aspnet\_regiis -i

The following command installs the ASP.NET version that is associated with the tool without updating existing applications. On IIS 7.0, the command does not update the version of the .NET Framework associated with the DefaultAppPool and Classic .NET AppPool application pools.

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl41_ctl00_ctl01_code');" \o "Copy Code)

aspnet\_regiis -ir

The following command displays the status and installation path of all versions of ASP.NET that are installed on the computer.

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl41_ctl00_ctl02_code');" \o "Copy Code)

aspnet\_regiis -lv

The following command updates a Web application and all of its sub-applications to use the ASP.NET version that is associated with the tool. On IIS 6.0, the scriptmaps are updated. On IIS 7.0, the application is mapped to a new application pool.

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl41_ctl00_ctl03_code');" \o "Copy Code)

aspnet\_regiis -s W3SVC/1/ROOT/SampleApp1

The following command performs the same functions as the **-s** command, except that it does not affect applications in subdirectories.

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl41_ctl00_ctl04_code');" \o "Copy Code)

aspnet\_regiis -sn W3SVC/1/ROOT/SampleApp1

The following command grants the specified user or group accounts access to the IIS metabase and the other directories used by ASP.NET.

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl41_ctl00_ctl05_code');" \o "Copy Code)

aspnet\_regiis -ga MyUserGroup W3SVC/1/ROOT/SampleApp1

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifFinding the Correct Version of Aspnet\_regiis.exe

Aspnet\_regiis.exe is installed in the Microsoft.NET Framework directory. If the computer is running multiple .NET Framework versions side-by-side, multiple versions of the tool might be installed. The following table lists the locations where the tool is installed for different versions of the .NET Framework.

|  |  |
| --- | --- |
| **Version of .NET Framework** | **Location of Aspnet\_regiis.exe file** |
| .NET Framework version 1 | %windir%\.NET\Framework\v1.0.3705 |
| .NET Framework version 1.1 | %windir%\Microsoft.NET\Framework\v1.1.4322 |
| .NET Framework version 2.0, version 3.0, and version 3.5 (32-bit systems) | %windir%\Microsoft.NET\Framework\v2.0.50727 |
| .NET Framework version 2.0, version 3.0, and version 3.5 (64-bit systems) | %windir%\Microsoft.NET\Framework64\v2.0.50727 |
| .NET Framework version 4 (32-bit systems) | %windir%\Microsoft.NET\Framework\v4.0.30319 |
| .NET Framework version 4 (64-bit systems) | %windir%\Microsoft.NET\Framework64\v4.0.30319 |

Directories for the .NET Framework versions 3.0 and 3.5 do not have a copy of the tool. For more information, see the entry [How to set an IIS Application or AppPool to use ASP.NET 3.5 rather than 2.0](http://go.microsoft.com/fwlink/?LinkId=186691) on Scott Hanselman's blog.