This page is specific to

Microsoft Visual Studio 2010/.NET Framework 4

.NET Framework 4 - ASP.NET

**Encrypting and Decrypting Configuration Sections**

You can use the ASP.NET IIS Registration Tool (Aspnet\_regiis.exe) to encrypt or decrypt sections of a Web configuration file. ASP.NET will automatically decrypt encrypted configuration elements when the Web.config file is processed.

|  |
| --- |
| **Description: NoteNote** |
| The Aspnet\_regiis.exe tool is located in the *%windows%*\Microsoft.NET\Framework\*versionNumber* folder. |

You can also use the protected configuration classes in the [System.Configuration](http://msdn.microsoft.com/en-us/library/system.configuration.aspx) namespace to encrypt and decrypt sections of a Web configuration file, sections of a configuration file for an executable (.exe), or sections in the machine-level and application-level configuration files. For more information, see the [ProtectSection](http://msdn.microsoft.com/en-us/library/system.configuration.sectioninformation.protectsection.aspx) method of the [SectionInformation](http://msdn.microsoft.com/en-us/library/system.configuration.sectioninformation.aspx) class. For information on referencing a section of a Web.config file, see the [WebConfigurationManager](http://msdn.microsoft.com/en-us/library/system.web.configuration.webconfigurationmanager.aspx) class. For information on referencing configuration sections of files other than the Web.config file, see the [ConfigurationManager](http://msdn.microsoft.com/en-us/library/system.configuration.configurationmanager.aspx) class.

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifEncrypting a Web Configuration Section

To encrypt configuration file contents, use the Aspnet\_regiis.exe tool with the **–pe** option and the name of the configuration element to be encrypted.

Use the **–app** option to identify the application for which the Web.config file will be encrypted and the **-site** option to identify which Web site the application is a part of. The Web site is identified using the site number from the Internet Information Services (IIS) metabase. You can retrieve the site number from the INSTANCE\_META\_PATH server variable in the [ServerVariables](http://msdn.microsoft.com/en-us/library/system.web.httprequest.servervariables.aspx) collection. For example, when IIS is installed, a Web site named "Default Web Site" is created as site 1. In pages served from that site, the INSTANCE\_META\_PATH server variable returns "/LM/W3SVC/1". If you do not specify a **-site** option, site 1 is used.

Use the **–prov** option to identify the name of the [ProtectedConfigurationProvider](http://msdn.microsoft.com/en-us/library/system.configuration.protectedconfigurationprovider.aspx) that will perform the encryption and decryption. If you do not specify a provider using the **-prov** option, the provider configured as the defaultProvider is used.

|  |
| --- |
| **Description: NoteNote** |
| If you are using an [RsaProtectedConfigurationProvider](http://msdn.microsoft.com/en-us/library/system.configuration.rsaprotectedconfigurationprovider.aspx) instance that specifies a custom key container, you must create the key container before running the Aspnet\_regiis.exe tool. For more information, see [Importing and Exporting Protected Configuration RSA Key Containers](http://msdn.microsoft.com/en-us/library/yxw286t2.aspx). |

The following command encrypts the **connectionStrings** element in the Web.config file for the application SampleApplication. Because no **-site** option is included, the application is assumed to be from Web site 1 (most commonly Default Web Site in IIS). The encryption is performed using the [RsaProtectedConfigurationProvider](http://msdn.microsoft.com/en-us/library/system.configuration.rsaprotectedconfigurationprovider.aspx) specified in the machine configuration.

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl29_ctl00_ctl06_code');" \o "Copy Code)

aspnet\_regiis -pe "connectionStrings" -app "/SampleApplication" -prov "RsaProtectedConfigurationProvider"

When a page or other ASP.NET resource in the application is requested, ASP.NET calls the provider for the protected configuration section to decrypt the information for use by ASP.NET and your application code.

|  |
| --- |
| **Description: NoteNote** |
| To decrypt and encrypt a section of the Web.config file, the ASP.NET process must have permission to read the appropriate encryption key information. For more information, see [Importing and Exporting Protected Configuration RSA Key Containers](http://msdn.microsoft.com/en-us/library/yxw286t2.aspx). |

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifDecrypting a Web Configuration Section

To decrypt encrypted configuration file contents, you use the Aspnet\_regiis.exe tool with the **-pd** switch and the name of the configuration element to be decrypted. Use the **–app** and **-site** switches to identify the application for which the Web.config file will be decrypted. You do not need to specify the **–prov** switch to identify the name of the [ProtectedConfigurationProvider](http://msdn.microsoft.com/en-us/library/system.configuration.protectedconfigurationprovider.aspx), because that information is read from the **configProtectionProvider** attribute of the protected configuration section.

The following command decrypts the **connectionStrings** element in the Web.config file for the ASP.NET application SampleApplication:

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl30_ctl00_ctl01_code');" \o "Copy Code)

aspnet\_regiis -pd "connectionStrings" -app "/SampleApplication"