This page is specific to

Microsoft Visual Studio 2010/.NET Framework 4

.NET Framework 4 - ASP.NET

**Specifying a Protected Configuration Provider**

You can encrypt and decrypt sections of a Web.config file using a [ProtectedConfigurationProvider](http://msdn.microsoft.com/en-us/library/system.configuration.protectedconfigurationprovider.aspx) class. The following list describes the protected configuration providers included in the .NET Framework:

* [DpapiProtectedConfigurationProvider](http://msdn.microsoft.com/en-us/library/system.configuration.dpapiprotectedconfigurationprovider.aspx). Uses the Windows Data Protection API (DPAPI) to encrypt and decrypt data.
* [RsaProtectedConfigurationProvider](http://msdn.microsoft.com/en-us/library/system.configuration.rsaprotectedconfigurationprovider.aspx). Uses the RSA encryption algorithm to encrypt and decrypt data.

Both providers offer strong encryption of data. However, if you are planning on using the same encrypted configuration file on multiple servers, such as a Web farm, only the [RsaProtectedConfigurationProvider](http://msdn.microsoft.com/en-us/library/system.configuration.rsaprotectedconfigurationprovider.aspx) enables you to export the encryption keys and import them on another server.

The .NET Framework also enables you to define your own protected configuration providers. For details, see [Implementing a Protected Configuration Provider](http://msdn.microsoft.com/en-us/library/wfc2t3az.aspx).

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifConfiguring a Protected Configuration Provider

You can specify your own protected configuration provider or you can use one of the providers built into ASP.NET. By default, the Machine.config file specifies the following protected configuration providers:

* An [RsaProtectedConfigurationProvider](http://msdn.microsoft.com/en-us/library/system.configuration.rsaprotectedconfigurationprovider.aspx) instance named RsaProtectedConfigurationProvider. This is the configured as the default provider.
* An [DpapiProtectedConfigurationProvider](http://msdn.microsoft.com/en-us/library/system.configuration.dpapiprotectedconfigurationprovider.aspx) instance named DataProtectionConfigurationProvider.

You specify a protected configuration provider in the **configProtectedData** section of a configuration file. If you want to specify your own provider with custom settings, you can declare a new provider instance using the **add** element of the **providers** element. You can identify your provider instance as the default provider using the **defaultProvider** attribute of the **configProtectedData** element.

The following example configures an [RsaProtectedConfigurationProvider](http://msdn.microsoft.com/en-us/library/system.configuration.rsaprotectedconfigurationprovider.aspx) instance with the name SampleProvider and sets it as the default provider.

[Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl33_ctl00_ctl03_code');" \o "Copy Code)

<configuration>

<configProtectedData defaultProvider="SampleProvider">

<providers>

<add name="SampleProvider"

type="System.Configuration.RsaProtectedConfigurationProvider,

System.Configuration, Version=2.0.0.0, Culture=neutral,

PublicKeyToken=b03f5f7f11d50a3a,

processorArchitecture=MSIL"

keyContainerName="SampleKeys"

useMachineContainer="true" />

</providers>

</configProtectedData>

</configuration>

You use the name of the protected configuration provider instance when encrypting configuration sections. For more information, see [Encrypting and Decrypting Configuration Sections](http://msdn.microsoft.com/en-us/library/zhhddkxy.aspx).

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifProtected Configuration Provider Options

Each protected configuration provider exposes options that you can set using attributes of the provider's declaration in the configuration file. All providers require the **type** and **description** attribute as well as the **keyName** for the provider instance. Beyond that, the options are unique to each provider type.

The following table describes the configuration options for the [RsaProtectedConfigurationProvider](http://msdn.microsoft.com/en-us/library/system.configuration.rsaprotectedconfigurationprovider.aspx).

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| **type** | The type of protected configuration provider. The following example shows a type definition for the [RsaProtectedConfigurationProvider](http://msdn.microsoft.com/en-us/library/system.configuration.rsaprotectedconfigurationprovider.aspx):  [Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl34_ctl00_ctl02_code');" \o "Copy Code)  type="System.Configuration.RsaProtectedConfigurationProvider,  System.Configuration, Version=2.0. 0.0,  Culture=neutral,  PublicKeyToken=b03f5f7f11d0a3a,  processorArchitecture=MSIL" |
| **description** | A description of the provider instance. |
| **keyContainerName** | The name of the RSA key container used to encrypt or decrypt the contents of the Web.config file.  Description: Note**Note**  The ASP.NET process must have read access to the specified RSA key container. You can grant access to an RSA key container with the **Aspnet\_regiis.exe** tool using the **-pa** switch. For more information, see [Importing and Exporting Protected Configuration RSA Key Containers](http://msdn.microsoft.com/en-us/library/yxw286t2.aspx). |
| **useMachineContainer** | **true** if the RSA key container is a machine-level key container; **false** if the RSA key container is a user-level key container. For more information, see [Encrypting Configuration Information Using Protected Configuration](http://msdn.microsoft.com/en-us/library/53tyfkaw.aspx). |
| **useOAEP** | **true** to use Optimal Asymmetric Encryption Padding (OAEP) when encrypting and decrypting; otherwise, **false**. For more information, see the [RSAOAEPKeyExchangeFormatter](http://msdn.microsoft.com/en-us/library/system.security.cryptography.rsaoaepkeyexchangeformatter.aspx) class. |
| **cspProviderName** | The name of the Windows cryptography API (crypto API) cryptographic service provider (CSP). For more information, see [ProviderName](http://msdn.microsoft.com/en-us/library/system.security.cryptography.cspparameters.providername.aspx). |

The following table describes the configuration options for the [DpapiProtectedConfigurationProvider](http://msdn.microsoft.com/en-us/library/system.configuration.dpapiprotectedconfigurationprovider.aspx).

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| **type** | The type of protected configuration provider. The following example shows a type definition for the [DpapiProtectedConfigurationProvider](http://msdn.microsoft.com/en-us/library/system.configuration.dpapiprotectedconfigurationprovider.aspx):  [Copy Code](javascript:CopyCode('ctl00_MTCS_main_ctl34_ctl00_ctl09_code');" \o "Copy Code)  type="System.Configuration.DpapiProtectedConfigurationProvider,  System.Configuration, Version=4.0.0.0,  Culture=neutral,  PublicKeyToken=b03f5f7f11d0a3a,  processorArchitecture=MSIL" |
| **description** | A description of the provider instance. |
| **keyEntropy** | An application-specific value to include with the encryption key to protect against other applications being able to decrypt encrypted information. For more information, refer to the **OptionalEntropy** parameter of the **CryptProtectData** method of the Windows data protection API (DPAPI). |
| **useMachineProtection** | **true** to use machine-specific protection; **false** to use user-account-specific protection. When **true**, any process running on the computer can unprotect data, and it is recommended that you restrict access to encrypted data using an Access Control List (ACL). For more information, see the CRYPTPROTECT\_LOCAL\_MACHINE value for the **dwFlags** parameter of the **CryptProtectData** method of the Windows data protection API (DPAPI). |

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