How to grant permissions to a custom assembly that is referenced in a report in Reporting Services

Article ID: 842419 - [View products that this article applies to.](http://support.microsoft.com/kb/842419/en-us" \l "appliesto)

[Expand all](javascript:void(0);) | [Collapse all](javascript:void(0);)

**Collapse image**[On This Page](javascript:void(0);)

**Collapse image**[SUMMARY](javascript:void(0);)

*This article describes a problem that occurs when you access a custom assembly in your report that does not have sufficient permissions on the resources in your environment. The article also provides steps to grant permissions to the custom assemblies that are used in the reports. To do this, follow these steps:*

1. *Assert permissions in your custom assembly.*
2. *Make the custom assembly available to Report Designer and to Report Server.*
3. *Reference the custom assembly in your report.*
4. *Modify the code access security settings for your custom assembly.*

[[http://support.microsoft.com/library/images/support/en-us/uparrow.gif](http://support.microsoft.com/kb/842419/en-us#top)Back to the top](http://support.microsoft.com/kb/842419/en-us" \l "top) | [Give Feedback](http://support.microsoft.com/kb/842419/en-us" \l "survey)

**Collapse image**[INTRODUCTION](javascript:void(0);)

Reports that access custom assemblies may not render successfully if the custom assemblies do not have sufficient permissions. This article describes how to modify the policy configuration files to grant additional permissions to a custom assembly that is accessed by a report.

[[http://support.microsoft.com/library/images/support/en-us/uparrow.gif](http://support.microsoft.com/kb/842419/en-us#top)Back to the top](http://support.microsoft.com/kb/842419/en-us" \l "top) | [Give Feedback](http://support.microsoft.com/kb/842419/en-us" \l "survey)

**Collapse image**[MORE INFORMATION](javascript:void(0);)

After you create a custom assembly by using the Microsoft .NET Framework, and then you access the custom assembly in your report and preview the report in Report Designer, the report may not render successfully. However, if you run the same report in DebugLocal mode in Report Designer or if you view the report by using Report Manager, the following error may be rendered in the report instead of the output of the referenced custom assembly:

#Error

Additionally, you may receive an error message that is similar to the following in Report Designer when you run the report in DebugLocal mode:

Request for the permission of type System.Security.Permissions.FileIOPermission, mscorlib, Version=1.0.5000.0, Culture=neutral, PublicKeyToken=b77a5c561934e089 failed.

This problem may occur when the custom assemblies that are referenced in your report do not have sufficient permissions.   
  
**Note** If you preview the report in Report Designer, the report runs all the report expressions by using the FullTrust permission set. The report does not use the security policy settings.  
  
To grant the required permissions to a custom assembly, follow these steps:

1. Assert permissions in your custom assembly.
2. Make the custom assembly available to Report Designer and to Report Server.
3. Reference the custom assembly in your report.
4. Modify the code access security settings for your custom assembly.

Assert permissions in your custom assembly

To acquire the required permissions to run the code in your custom assembly, you must assert the required permissions in your custom assembly code.   
  
**Note** The examples provided in this section use C# code.  
  
For example, if you access data in Microsoft SQL Server tables by using the "System.Data.SQLClient" namespace in your custom assembly, you must add the following code to your custom assembly before you use the "System.Data.SQLClient" namespace:

System.Data.SqlClient.SqlClientPermission pSql = new SqlClientPermission(System.Security.Permissions.PermissionState.Unrestricted);

pSql.Assert();

If you access a file in your custom assembly, you must add the following code to your custom assembly before you access the file:

FileIOPermission filePerm = new FileIOPermission(FileIOPermissionAccess.Read, "C:\TestFile");

filePerm.Assert();

Similarly, if you access a Web service in your custom assembly, you must add code that is similar to the following to assert the Web permissions:

System.Text.RegularExpressions.Regex urlRegEx = new System.Text.RegularExpressions.Regex(@"http://www\.webcontinuum\.net/.\*");

System.Net.WebPermission p = new System.Net.WebPermission(NetworkAccess.Connect,urlRegEx);

p.Assert();

For more information about asserting permissions in your custom assembly, visit the following MSDN Web site:

<http://msdn2.microsoft.com/en-us/library/aa179495(SQL.80).aspx>

Make the custom assembly available to Report Designer and to Report Server

To make your custom assembly available to Report Designer and to Report Server, you must copy the custom assembly to the following folders:

* *Installation Drive*:\Program Files\Microsoft SQL Server\80\Tools\Report Designer
* *Installation Drive*:\Program Files\Microsoft SQL Server\MSSQL\Reporting Services\ReportServer\bin

Reference the custom assembly in your report

To reference a custom assembly in your report, follow these steps:

1. Open the report project in Microsoft Visual Studio .NET 2003.
2. Open the report that must reference the custom assembly.
3. On the **Report** menu, click **Report Properties**.
4. In the **Report Properties** dialog box, click the **References** tab.
5. Under **References**, click the ellipsis button that is next to the **Assembly name** column header.
6. In the **Add References** dialog box, click **Browse**.
7. In the **Select Component** dialog box, locate and click your custom assembly, and then click **Open**.
8. In the **Add References** dialog box, click **OK**.
9. In the **Report Properties** dialog box, click **OK**.

Modify the code access security settings for your custom assembly

To grant the required permissions to the custom assembly, you must modify the code access security settings in the policy configuration files that correspond to Report Designer and to Report Server. To do this, follow these steps:

1. Create the permission set element and the code group element for your custom assembly that specifies the required permissions. To do this, follow these steps:
   1. Click **Start**, point to **Settings**, and then click **Control Panel**.
   2. In Control Panel, double-click **Administrative Tools**.
   3. In the Administrative Tools window, double-click **Microsoft .NET Framework 1.1 Configuration**.
   4. In the left pane of the .NET Configuration 1.1 window, expand **Runtime Security Policy**.
   5. Under **Runtime Security Policy**, expand **Machine**.
   6. Under **Machine**, right-click **Permission Sets**, and then click **New**.
   7. On the **Identify the new Permission Set** page, click **Create a new permission set**.
   8. Type **TestPermissions** in the **Name** box, and then click **Next**.
   9. On the **Assign Individual Permissions to Permission Set** page, click **Security** in the **Available Permissions** list, and then click **Add**.
   10. In the **Permission Settings** dialog box, click **Grant assemblies the following security permissions**, and then click to select the following check boxes:
       * **Enable assembly execution**
       * **Assert any permission that has been granted**

**Note** You can also turn on the other listed permission settings based on your requirements.

* 1. Click **OK**.
  2. On the **Assign Individual Permissions to Permission Set** page, add any additional permissions from the**Available Permissions** list to the **Assigned Permissions** list.
  3. Click **Finish**.
  4. In the .NET Configuration 1.1 window, expand **Code Groups** under **Machine**.
  5. Under **Code Groups**, right-click **All\_Code**, and then click **New**.
  6. On the **Identify the new Code Group** page, click **Create a new code group**, type **TestCodeGroup** in the **Name**box, and then click **Next**.
  7. On the **Choose a condition type** page, click **URL** in the **Choose the condition type for this code group** list.
  8. Type the absolute path of your custom assembly (*Installation Drive*:\Program Files\Microsoft SQL Server\80\Tools\Report Designer\*Custom assembly name*, for example) in the Report Designer folder in the **URL**box, and then click **Next**.
  9. On the **Assign a Permission Set to the Code Group** page, click **Use existing permission set**, and then click**TestPermissions** in the list.
  10. Click **Next**, and then click **Finish**.

The permission set and the code group that correspond to your custom assembly are created according to your specifications in the Security.config file on your computer.

1. Add the permission set and the code group that correspond to your custom assembly in the security policy configuration files for your report server. To do this, follow these steps:
   1. Open the Security.config file, the Rspreviewpolicy.config file, and the Rssrvpolicy.config file.  
        
      **Note** The following list provides the default locations of the Security.config file, the Rspreviewpolicy.config file, and the Rssrvpolicy.config file.
      * By default, the Security.config file is located in the *Installation Drive*:\WINDOWS\Microsoft.NET\Framework\v1.1.4322\config\security.config folder.
      * By default, the Rspreviewpolicy.config is located in the *Installation Drive*:\Program Files\Microsoft SQL Server\80\Tools\Report Designer\rspreviewpolicy.config folder.
      * By default, the Rssrvpolicy.config folder is located in the *Installation Drive*:\Program Files\Microsoft SQL Server\MSSQL\Reporting Services\ReportServer\rssrvpolicy.config folder.
   2. Copy the **PermissionSet** element that corresponds to the TestPermissions permission set from the Security.config file.
   3. Paste the **PermissionSet** element that corresponds to the TestPermissions permission set in the Rspreviewpolicy.config file and the Rssrvpolicy.config file as the last sub-element of the **NamedPermissionSets**element.
   4. Copy the **CodeGroup** element that corresponds to the TestCodeGroup code group from the Security.config file.
   5. Paste the **CodeGroup** element that corresponds to the TestCodeGroup code group in the Rspreviewpolicy.config file and in the Rssrvpolicy.config file after the first **CodeGroup** element.
   6. Remove the **PermissionSet** element that corresponds to the TestPermissions permission set and the **CodeGroup**element that corresponds to the TestCodeGroup code group from the Security.config file.
   7. Save the Security.config file, the Rspreviewpolicy.config file, and the Rssrvpolicy.config file.

After you access the custom assembly in your report and you run the report in DebugLocal mode, the report renders successfully.

[[http://support.microsoft.com/library/images/support/en-us/uparrow.gif](http://support.microsoft.com/kb/842419/en-us#top)Back to the top](http://support.microsoft.com/kb/842419/en-us#top) | [Give Feedback](http://support.microsoft.com/kb/842419/en-us#survey)

**Collapse image**[REFERENCES](javascript:void(0);)

For more information about code access security in Reporting Services, visit the following MSDN Web site:

<http://msdn2.microsoft.com/en-us/library/aa902638(SQL.80).aspx>

For more information about the using custom assemblies with reports in Reporting Services, visit the following MSDN Web site:

<http://msdn2.microsoft.com/en-us/library/aa179513(SQL.80).aspx>

For more information about debugging the custom assemblies in Reporting Services, visit the following MSDN Web site:

<http://msdn2.microsoft.com/en-us/library/aa179526(SQL.80).aspx>

[[http://support.microsoft.com/library/images/support/en-us/uparrow.gif](http://support.microsoft.com/kb/842419/en-us#top)Back to the top](http://support.microsoft.com/kb/842419/en-us#top) | [Give Feedback](http://support.microsoft.com/kb/842419/en-us#survey)

**Collapse image**[Properties](javascript:void(0);)

Article ID: 842419 - Last Review: April 3, 2007 - Revision: 2.2

**APPLIES TO**

* Microsoft SQL Server 2000 Reporting Services

|  |  |
| --- | --- |
| **Keywords:** | kbpolicy kbpermissions kbsettings kbsecurity kbdeployment kbreport kberrmsg kbinfo KB842419 |