OrgPublisher Executive and Silverlight Server Configuration

Table of Contents

Audience

The intended audience for this document is the OrgPublisher chart administrator, in conjunction with a server administrator, with permissions to turn Windows features on or off.

Intent

This document should be used to setup the web server that will be used to host published charts. Administrative rights may be required to perform most actions listed in this document. Consult your IT department to ensure all company policies are followed when configuring this server.

System Requirements

- A Microsoft® IIS (ISAPI enabled) web server running Windows® 2003, Windows® 2008 or Windows® 2012

- 2GB RAM minimum; more RAM may be required for larger charts

- Microsoft .NET Framework. Minimum version 4.0

  Note: *Silverlight (NPAPI) is not support in Google Chrome 45+

Windows Features and Roles requirements

Use the Control Panel\Server Manager to ensure the following Windows features and Roles are enabled:

- Roles
  - Web Server (IIS)
    - o Role Services
      - Application Development Features
        - .NET Extensibility
        - ISAPI Extensions
        - ISAPI Filters

- Features
  - .NET Framework 4.0 Features
    - o .NET Framework 4.0
    - o WCF Activation
      - HTTP Activation
      - Non-HTTP Activation

Your version of the .NET Framework may appear slightly different. Version 4.0 (or higher) of the .NET Framework is required.

Silverlight Security Restrictions for Cross-Domain and Cross-Scheme Network Access

Microsoft restricts cross-domain and cross-scheme access for security purposes.

This can affect the OrgPublisher Silverlight/cross browser charting feature if person photos are hosted using a different domain or scheme than is used for hosting the chart itself.

For example, a Silverlight chart may be hosted with an HTTP connection, but the chart's **Base photo file path or URL** (Photo Wizard) uses HTTPS to download the photos from a different server.

This combination of HTTP and HTTPS is called **cross-scheming**. The combination of accessing different servers is called **cross-domain**.
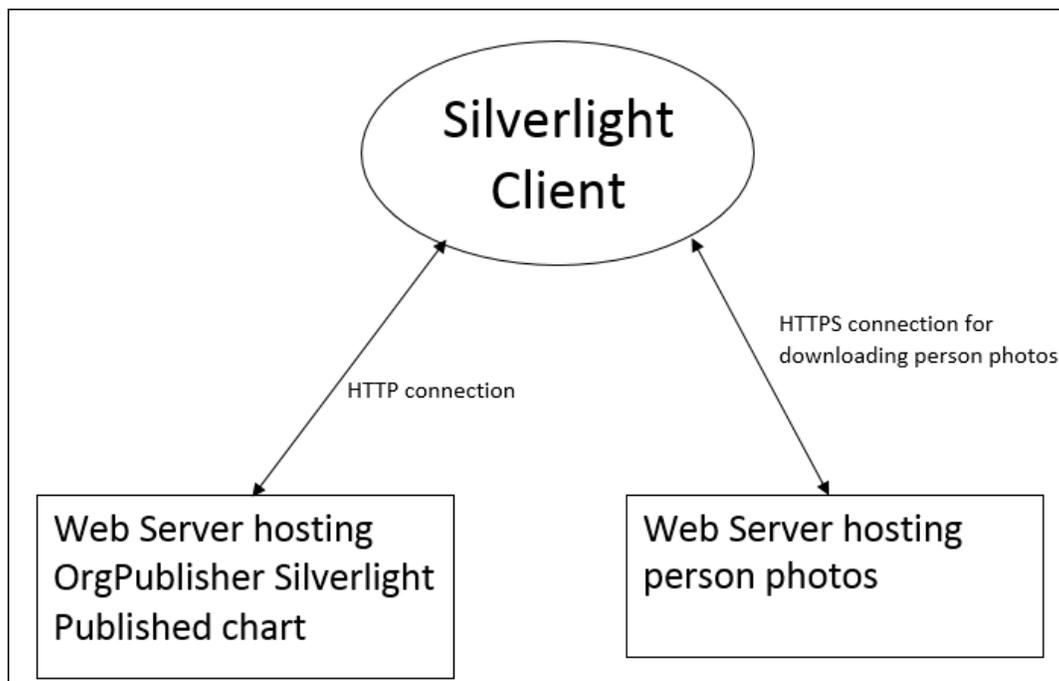

Figure 1.

The server hosting the photos must be configured to allow this cross-domain and cross-scheme behavior in Silverlight. The configuration involves placing a **ClientAccessPolicy.xml** file on the root IIS folder (for instance, **\inetpub\wwwroot**), and configuring this file to meet your required level of security. A copy of the **ClientAccessPolicy.xml** file is included in the install folder of the OrgPublisher desktop install.

The links below, to the Microsoft MSDN Library, take you to articles that explain how to place and configure this file.

Network Security Access Restrictions in Silverlight
URL Access Restrictions in Silverlight

EXAMPLE ClientAccessPolicy.xml file

This file allows any domain and either HTTP or HTTPS. Each user should adjust these settings according to their needs.

```xml
<?xml version="1.0" encoding="utf-8"?>
<access-policy>
 <cross-domain-access>
  <policy>
   <allow-from http-methods="*">
    <domain uri="*" />
    <domain uri="http://*" />
    <domain uri="https://*" />
   </allow-from>

   <grant-to>
    <resource path="/" include-subpaths="true"/>
   </grant-to>
  </policy>
 </cross-domain-access>
</access-policy>
```

Publishing Constraints

1. Do not publish other types of charts (ECharts, PluginX, etc.) in the same folder as you publish your Silverlight or OrgPublisher Executive for iPad charts.
2. Silverlight charts published to the same folder must have identical security settings (Unsecure, Standard, Active Directory, etc.).
3. Only one (1) OrgPublisher Executive for iPad chart can be published to a given publish location. Do not publish multiple OrgPublisher Executive for iPad charts to the same folder.

Internet Information Services (IIS) Configuration

The default root folder of IIS is **C:\inetpub\wwwroot**. You can create a subfolder in the root folder, or choose a different location to publish your chart.

The OrgPublisher publishing wizard prompts you for the folder path to publish your chart. Make note of the publish location selected in the publishing wizard.

1. Once the chart is published, use Internet Information Services (IIS) Manager to add a virtual directory and map it to the physical path/folder you selected during the publish process.

   The alias you specify will be part of the URL for this chart.

2. Create two app pools and name them similar to the name of the published chart; one for 64-bit applications and one for 32-bit applications.

3. Use the App Pool Advanced Settings dialog for each app pool to change the following:

   a. In the Process Model section, set the Identity to **ApplicationPoolIdentity**
   b. Set the .NET Framework version to **4.0**

4. Use the App Pool Advanced Settings dialog for the 32-bit application app pool to change the following:

   a. Set **Enable 32-bit Applications** to **True**

5. Set up authentication for the virtual folder in IIS:

   a. Select the virtual folder
   b. In the IIS section of the Features View (center pane), double-click the **Authentication** item
   c. Select the appropriate security model

      i. **Anonymous** for unsecure or standard security
      ii. In the dialog that opens, select **Application pool identity**
      iii. Windows for Active Directory (only for Silverlight, <u>not</u> OPE)
      iv. Ensure only one model is enabled

6. In the virtual folder, convert the **Service** folder to an Application by right-clicking the folder and selecting **Convert to Application**:

   a. Set the previously configured 64-bit app pool for the **Service** folder

7. Also in the virtual folder, convert the **Data** folder to application by right-clicking on the folder and selecting **Convert to Application**:

   a. Set the previously configured 32-bit app pool for the **Data** folder

8. Enable the ISAPI handler for the Silverlight Data folder:

   a. In the virtual folder, click **Data**
   b. In the IIS sections of the Features View (center pane), double-click **Handler Mappings**

      i. Click **Add Script Map** in the Actions pane on the right
      ii. In the **Request Path** enter the **<chart name>.dll**

iii. For the **Executable**, click the button to navigate to the folder where you published the chart and select the DLL matching the name of the published chart

iv. For the **Name**, enter the name of the published chart

v. Click OK to close the window

c. In the list of handler mappings:

i. Right-click the mapping you just created

ii. Click **Edit Feature Permissions**

iii. Select the **Execute** check box

iv. Click OK to close the window

9. Set folder permissions for the virtual folder:

a. For the 64-bit app pool, navigate to the **top level** folder in your publish-to location and set permissions, noted in Step 10

b. For the 32-bit app pool, navigate to the **Data** subfolder in your publish-to location and set permissions, noted in Step 10

10. Set security permissions for the virtual folder:

a. Right-click on the virtual folder and click **Edit permissions**

b. Open the **Security** tab and add the app pool to the list of objects, giving the app pool full permissions to the folder. For an app pool named MyChartAppPool, the object name would be

IIS AppPool\MyChartAppPool

Troubleshooting

For help troubleshooting problems viewing a chart, please contact customer support.