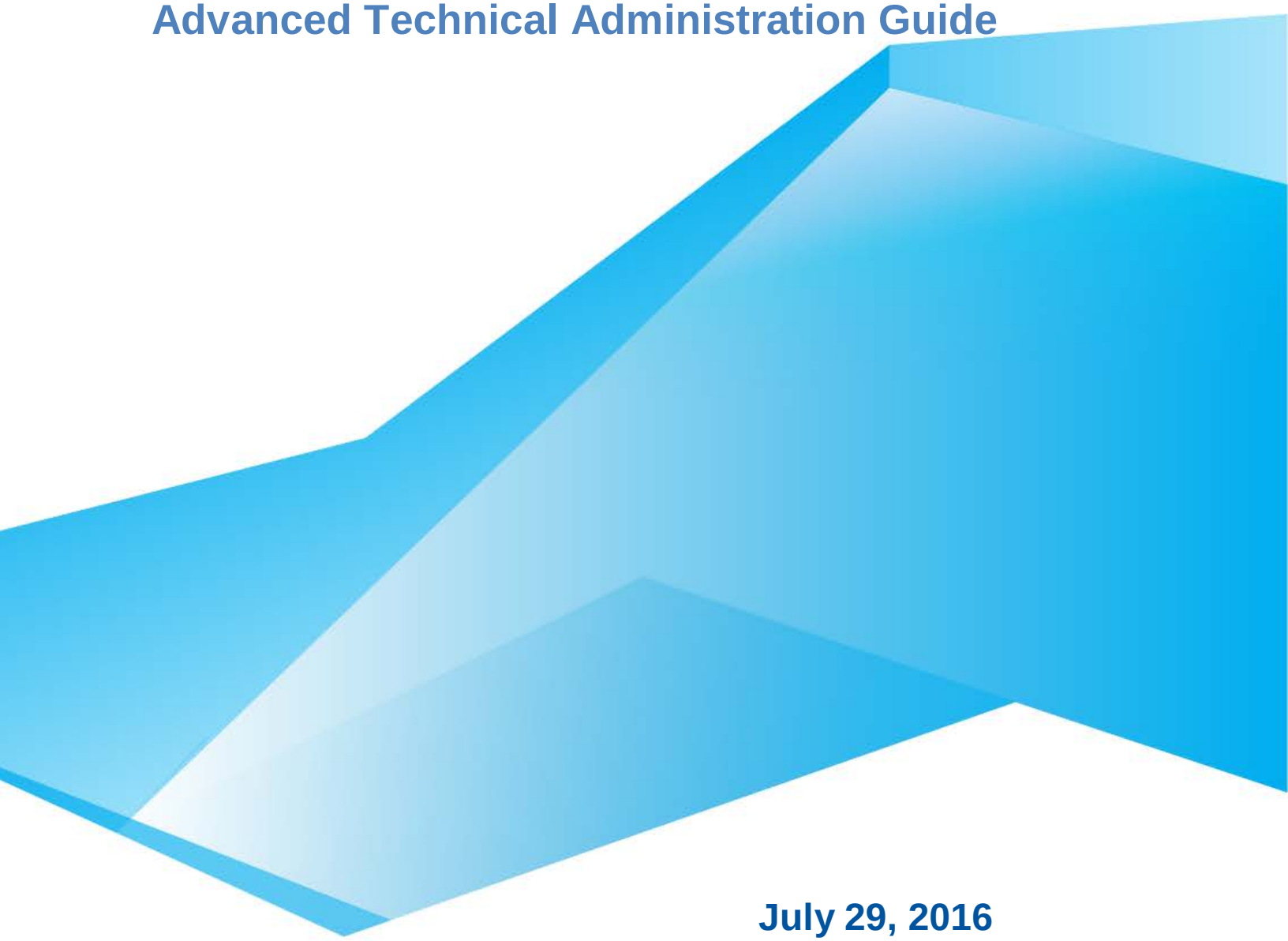
Deltek Know more. Do more.™

Deltek Vision® 7.6

Advanced Technical Administration Guide



July 29, 2016

While Deltek has attempted to verify that the information in this document is accurate and complete, some typographical or technical errors may exist. The recipient of this document is solely responsible for all decisions relating to or use of the information provided herein.

The information contained in this publication is effective as of the publication date below and is subject to change without notice.

This edition published July 2016.

Contents

Overview

This guide is a supplement to the *Deltek Vision Technical Installation Guide*. Topics covered in this supplement are for advanced deployments and may not be applicable to all installations of Vision.

Procedures included in this guide explain how to:

- Create a reverse proxy for SQL Reporting Services using IIS 7.0 Application Request Routing (ARR)
- Configure HTTP compression to improve application response time
- Configure the Secure Sockets Layer (SSL)
- Pre-deploy the Vision Smart Client to user workstations
- Configure integrated security
- Configure database session state
- Secure your product deployment

Consulting Is Available

This document provides additional insight into advanced technical administration topics for your Deltek application. These advanced topics are outside the bounds of the Deltek Customer Care Agreement and therefore are **not** covered by your Support Contract.

If you would like Deltek to provide additional insight into, or assistance with, the implementation of this material for your specific environment, experts in our Consulting Group are available to provide the specialized help that you need.

Adding Custom Notes to This Guide

If you would like to add custom notes to this guide that are specific to your company, Adobe® Reader® versions X and later provide this ability. If you do not already have Adobe Reader, you can download it free from Adobe.

To add a custom note using Adobe Reader:

1. On the Reader toolbar, click **Comment** at far right.
2. In the **Annotations** pane that displays, click  **Sticky Note**. The cursor changes to match the button.
3. Position the cursor at the location in the guide where you want the note to appear, and click. A note icon is inserted at the location and a text box pops up.
4. Enter your information in the text box.
5. Continue adding notes as needed.
6. Save the document.

Deltek recommends that you save the document to a slightly different filename so as to keep the original from being overwritten.

When reading the document, cursor over a note icon to see the information. Double-click a note icon to edit the information.

If You Need Assistance

If you need assistance installing, implementing, or using Vision, Deltek makes a wealth of information and expertise readily available to you.

Customer Services

For decades, Deltek has maintained close relationships with client firms, helping with their problems, listening to their needs, and getting to know their individual business environments. A full range of customer services has grown out of this close contact, including the following:

- Extensive self-support options through the Customer Care Connect Web portal.
- Phone and email support from Customer Care analysts
- Technical services
- Consulting services
- Custom programming
- Classroom, on-site, and Web-based training

Find out more about these and other services from the Customer Care Connect site.

Customer Care Connect Site

The Deltek Customer Care Connect site is a support Web portal for Deltek customers who purchase an Ongoing Support Plan (OSP).

The following are some of the many options you have at the Customer Care Connect site:

- Download the latest versions of your Deltek products
- Search Deltek's knowledge base
- Ask questions, exchange ideas, and share knowledge with other Deltek customers through the Deltek Connect Customer Forums
- Display or download product information, such as release notes, user guides, technical information, and white papers
- Submit a support case and check on its progress
- Transfer requested files to a Customer Care analyst
- Use Quick Chat to submit a question to a Customer Care analyst online
- Subscribe to Deltek communications about your Deltek products and services
- Receive alerts of new Deltek releases and hot fixes

If you need assistance using the Customer Care Connect site, the online help available on the site provides answers for most questions.

Access Customer Care Connect

To access the Customer Care Connect site:

1. Go to http://support.deltek.com.
2. Enter your Customer Care Connect **Username** and **Password**.
3. Click **Log In**.

If you do not have a username and password for the Customer Care Connect site, contact your firm's Vision Administrator.

If you forget your username or password, you can click the **Account Assistance** button on the login screen for help.

Additional Documentation

Release notes and other guides are available for this release. You can download these documents in two ways.

Deltek Software Manager

The Documents tab in Deltek Software Manager lists all of the documents associated with a release and lets you download the ones that you want.

To download documents:

1. On the Deltek Customer Care site, click the Product Downloads tab, then select **Launch Deltek Software Manager**.
2. When the Deltek Software Manager opens, highlight a release in the left pane.

   Do **not** enter a check next to the release name or click **Add to Download Queue**. If you do so, you will download the software as well as any documentation that you want.

3. Click the Documents tab to display a list of available documents for the release.
4. Select the documents that you want.
5. Click **View Download Queue** to see a list of documents that you selected.
6. Click **Download**.

Customer Care Site Enterprise Search

Use the search feature to find specific documents or to see a list of all documents associated with a release. Then open or download the ones that you want.

To download documents:

1. On the Deltek Customer Care site, click **Enterprise Search**.
2. Select **Release Documentation** as the **Source**.

3. Perform one of the following actions:
   - To see a list of all available documentation for a release, enter the product and release number (for example, **Vision 7.6**) in the search field.
   - To find a specific document, enter a description of the document (for example, **Vision 7.6 release notes**) in the search field.
4. Click on the document, then choose to open or save it.

Problem Displaying Online Help in Internet Explorer

If you use Internet Explorer and the help does not display correctly, you need to turn off Compatibility View for the browser. Click **Tools » Compatibility View settings**, and clear the **Display intranet sites in Compatibility View** check box. Then refresh the browser. You could also elect to use hosted help or the FQDN of your server in the URL (for example, server.domain) to bypass this issue. Make sure you remove deltek.com (if listed) from the **Websites you've added to Compatibility View** list.

Chapter 1: Creating a Reverse Proxy for SQL Reporting Using Application Request Routing (ARR)

Do I Need a Reverse Proxy?

Deltek Vision uses the Microsoft SQL Reporting Services WinForms report viewer control to render reports. This control requires a direct connection to the server running the SQL Reporting Services web service. Due to the nature of the Deltek Vision and SQL Reporting Services logical tier architectures and the available editions and licensing requirements of SQL Reporting Services, it is likely that the SQL Reporting Services web service will not be installed on the Vision web/application server in your deployment of Deltek Vision.

Typically, this is not a problem when Vision is deployed inside the intranet. However, when Vision is deployed where it is accessible directly via the Internet, the infrastructure requirements needed to support the configuration become complex because it is necessary to have multiple points of entry (one each for the Vision web server and SQL Reporting web service), multiple firewall configurations, and, potentially, multiple public DNS records with your Internet Service Provider (ISP). To complicate matters, if you have a two-tier deployment of Deltek Vision, this deployment may require that the server hosting your database is made accessible to the Internet, posing additional security risks.

A reverse proxy using Microsoft's Application Request Routing (ARR) extension for IIS allows the direct forwarding of requests through the Vision web server to the reporting services web service, with responses back to your Internet clients. This configuration resolves all of the issues described above.

The primary intent of a reverse proxy is to shield the SQL server from access via the Internet. Specifically, this is for two tier deployments where the SQL database and report server are on the same physical machine. Deltek does **not** recommend the use of the reverse proxy for a large number of users due to the potential performance impact that the reverse proxy component may have on the Vision web/application server.

Deltek supports the use of Application Request Routing 3.0.

Install Application Request Routing (ARR)

Follow the steps below to install ARR. These installation instructions are specific to version 3.0 of ARR.

Prerequisites

The following prerequisites must be met before installation:

- The Vision web/application server must be running one of the following:
  - Windows Server 2012 / IIS 8.0
  - Windows Server 2012 R2 / IIS 8.5
- Deltek Vision must be installed.
- The IIS configuration must include the IIS role service **Management Service**.

Important Information on the Use of Non-standard Ports

Before installing and configuring ARR, see Chapter "3: Configuring Secure Sockets Layer (SSL)," for information on how the reporting framework handles SSL requests and potential issues with the use of non-standard ports.

Download and Install ARR on Your Vision Web/Application Server

To download and install Application Request Routing on your Vision web/application server:

1. Go to the following URL to install ARR 3.0 via the Microsoft Web Platform installer: http://www.iis.net/downloads/microsoft/application-request-routing
2. Click **Install this Extension**.
3. On the Microsoft Web Platform Installer page, click **Install Now**.
4. On the File Download dialog box, click **Run** to run the ARRv3_0.exe file.
5. When the Web Platform Installer launches, choose to install Application Request Routing 3.0.

   The Web Platform Installer will ensure that all prerequisites required for the installation are also downloaded and installed.
6. Accept the license agreements.
7. When the Web Platform Installer has finished downloading and installing all components, click **Finish** and then click **Exit** on the Web Platform Installer main page.

Configure Application Request Routing (ARR)

To configure Application Request Routing:

1. Open Windows Explorer and create two folders under <drive>:\Program Files\Deltek\Vision\Web\, named:
   - Reports
   - ReportServer

   For example, enter: c:\Program Files\Deltek\Vision\Web\Reports.
2. Create a new Application Pool called **DeltekVisionReportingProxy**:
   a. In IIS Manager, expand the server name.
   b. Right-click **Application Pools**, and select **Add Application Pool**.
   c. Enter the name, and click **OK** to create the Application Pool.
3. Modify the Application Pool settings:
   a. Right-click the **DeltekVisionReportingProxy** Application Pool, and select **Advanced Settings**.
   b. Set **Enable 32 bit applications** to **false**.
   c. Configure the **Identity** to be the same account as your DeltekVisionAppPool. By default, this is the local DeltekVision Windows account.
   d. Set **Idle Time-out** to **0** (the default is 20).

e. Scroll down to see more Advanced Settings.

f. Set **Regular Time Interval (minutes)** to **0** (the default is 1740).

g. Set **Specific Times** to **00:15:00** (the default is 00:00:00).

4. Create IIS Applications to act as the proxy for the Reports (SQL RS Report Manager) and ReportServer (SQL RS web service):

   a. In IIS Manager, expand **Sites**.

   b. Right-click **Default Web Site**, and select **Add Application**.

   c. In the **Alias** field, enter **Reports**, configure it to use the DeltekVisionReportingProxy, and enter (or browse to) the physical path that you created in step 1.

   d. Click **OK** to create the Reports application.

5. Set up the ReportServer Application:

   a. Right-click **Default Web Site**, and select **Add Application**.

   b. In the **Alias** field, enter **ReportServer**, configure it to use the DeltekVisionReportingProxy, and enter (or browse to) the physical path you created in step 1.

   c. Click **OK** to create the ReportServer Application.

6. Add **Rewrite Rules** for each reporting application.

   a. Under **Default Web Site**, click **Reports Application**.

   b. Double-click **URL Rewrite**.

   If you do not see the URL Rewrite module, it's possible that Internet Services Manager was open when ARR was installed. Close and re-open Internet Services Manager.

   c. Click **Actions » Add Rules**.

   d. Select **Reverse Proxy**, and click **OK**.

   e. Click **OK** when you see the prompt: **Are you sure you want to enable proxy functionality?**

   f. On the Add Reverse Proxy Rules dialog box, enter the name of your SQL Reporting Services server in the **Inbound Rules** text box.

   If your Vision server is configured for SSL, Deltek recommends that you select **Enable SSL Offloading** (this is the default). With SSL Offloading enabled, you do not need an SSL certificate installed on the SQL Reporting Services server. The SSL certificate on the web/application server ensures that reporting functionality is encrypted between client and server.

   However, if you want to maintain SSL certificates on both the web server and the reporting server, clear the **Enable SSL Offloading** option**,** and make sure to use the **https://** prefix when configuring the report server URL in steps 6i and 9 below.

   Refer to Chapter 3, "Configuring Secure Sockets Layer (SSL)," for information on how the Vision reporting framework handles SSL requests.

   g. Click **OK** to create the reverse proxy rule.

h. Select the rule that was created and click the **Edit** link on the right, under **Inbound Rules**.

i. By default, the rewrite rule only includes the base URL for the server name entered. Edit the URL under **Rewrite URL** to have the correct Reporting Services application. The correct URL is:

http://<reportserver>/Reports/{R:1}

Make sure that there is a slash between Reports and {R:1}.

7. Repeat steps 6a through 6g for the ReportServer virtual directory.
   a. Under **Default Web Site**, click **ReportServer Application**.
   b. Double-click **URL Rewrite**.
   c. Click **Actions** » **Add Rules**.
   d. Select **Reverse Proxy**, and click **OK**.
   e. When prompted about enabling proxy functionality, click **OK**.
   f. On the Reverse Proxy Rules dialog box, enter the name of your SQL Reporting Services server in the **Inbound Rules** text box.
   g. Click **OK** to create the reverse proxy rule.
8. Select the rule that was created and click the **Edit** link on the right under **Inbound Rules**. By default, the rewrite rule only includes the base URL for the server name entered.
9. Edit the URL under **Rewrite URL** to have the correct Reporting Services application. The correct URL will be:

   http://<reportserver>/ReportServer/{R:1}

Test the Proxy Server

To test the proxy server:

1. Open Internet Explorer.
2. Browse to the following URLs. If ARR has been configured properly, your request will be proxied to the SQL Reporting Services server.
   - http://<VisionWebServer>/Reports

     where <VisionWebServer> is the Fully Qualified Domain Name of the web/application server.
   - http://<VisionWebServer>/ReportServer

     where <VisionWebServer> is the Fully Qualified Domain Name of the web/application server.

Configure Vision to Use the Reverse Proxy

To modify WebLink to use the Reverse Proxy:

1. To open WebLink, click **Start » All Programs » Deltek Vision** if on the web/application server or use http://<VisionWebServer>/Vision/Weblink.htm.
2. Enter the password to access WebLink.
3. Click the Report Server tab, and modify the Server URL to be the URL to access the new ReportServer virtual directory that you created on the Vision Web Server.
4. Typically, the Server URL is in the form http://<ReportServer>/ReportServer. Change this to http://<VisionWebServer>/ReportServer.

   No additional changes are necessary to the WebLink configuration. Be sure to change all databases that will use the reverse proxy.

   The Vision web server should be secured with an SSL certificate. This means that you do not need to secure the SQL Reporting Services server with an SSL certificate. Therefore, the URL entered above should start with **http://**, not **https://**.

   The proxy server forwards HTTPS traffic to HTTP via SSL Offloading. The Vision reporting framework automatically changes the URL prefix of the report server URL to use **https://** if Vision is accessed via HTTPS.
5. To test the Report Server configuration, click **Test » Report Server Configuration**.
6. After the configuration tests successfully, save your changes.

Troubleshooting

If you need help, contact the Deltek Global Services consulting group, VisionConsulting@deltek.com. The consulting group will provide an estimate of the cost for the help that you need.

Application Request Routing (ARR) Documentation

For additional documentation, go to http://www.iis.net/extensions/ApplicationRequestRouting.

Chapter 2: Configuring HTTP Compression

Configuring HTTP Compression for Vision can greatly reduce the size of HTTP (hypertext transfer protocol) requests and responses between the client and web server, which improves application response time. HTTP Compression is an available functionality built into Internet Information Services (IIS). By default, however, HTTP Compression is not enabled. This section explains how to install and configure HTTP Compression.

Three Configuration Methods for HTTP Compression

You can configure HTTP Compression using one of three methods. This document focuses on the first of the three methods. However, if you want to use the other methods, you can use the modified entries and settings from applicationhost.config, described at the end of this section.

- Use the appcmd IIS command line administrative utility. You must run this utility via an elevated command prompt such as **Run as Administrator**.
- Modify the applicationhost.config file directly. Deltek does **not** recommend that you modify the applicationhost.config file directly unless you are familiar with XML formatting. Be sure to make a backup of applicationhost.config before you make any changes.
- Use the Configuration Editor via the Internet Information Services administrative utility.

Install HTTP Compression IIS Role Services

To install HTTP Compression IIS Role Services:

1. Launch the Server Manager.
2. Click **Roles**.
3. Under Web Server (IIS), locate **Role Services** and check to see that the Static and Dynamic Content Compression role services have been installed.
4. If not, select **Add Role Services** and install both role services.

Alternative Procedure

Alternatively, you can install these role services using the Windows Package Manager (pkgmgr) from an administrative command prompt (for example, **Run as Administrator**):

start /w pkgmgr /iu:IIS-WebServerRole;IIS-Performance;IIS-HttpCompressionStatic;IIS-HttpCompressionDynamic

Configure HTTP Compression

To configure HTTP Compression:

1. Select one of the following actions:
   - If you want to enable compression at the server level, ensure that both static and dynamic compression are enabled via an elevated command prompt:

     C:\Windows\System32\Inetsrv\Appcmd.exe set config -section:urlCompression -doStaticCompression:true -doDynamicCompression:true

- If you want to enable compression for a particular web site, use the following command and replace **"Site Name"** with the name of the web site:

  C:\Windows\System32\Inetsrv\Appcmd.exe set config "Site Name" -section:urlCompression -doStaticCompression:true -doDynamicCompression:true

2. Set the static and dynamic compression levels via an elevated command prompt:

   C:\Windows\System32\Inetsrv\Appcmd.exe set config -section:httpCompression -[name='gzip'].staticCompressionLevel:9 -[name='gzip'].dynamicCompressionLevel:4

   The default dynamic compression level is zero.

Dynamic compression can significantly impact CPU resources. Refer to the following blog post for information and recommendations on setting compression levels. The command above uses recommendations from this blog post:

http://weblogs.asp.net/owscott/archive/2009/02/22/iis-7-compression-good-bad-how-much.aspx

3. Configure the content types that you want to compress. The default configuration compresses most static and dynamic content types used by the application.

   However, you must configure specific content types to compress the ClickOnce content types.

   C:\Windows\system32\inetsrv\appcmd set config /section:httpCompression /+dynamicTypes.[mimeType='application/octet-stream',enabled='true'] /commit:apphost

   C:\Windows\system32\inetsrv\appcmd set config /section:httpCompression /+dynamicTypes.[mimeType='application/x-ms-application',enabled='true'] /commit:apphost

   C:\Windows\system32\inetsrv\appcmd set config /section:httpCompression /+dynamicTypes.[mimeType='application/x-ms-manifest',enabled='true'] /commit:apphost

ClickOnce content types are considered dynamic. If you add them under the <statictypes> section, ClickOnce files are not compressed.

See the following Microsoft support article for additional guidance on setting content types:

http://support.microsoft.com/kb/969062

Additional Settings that May Impact HTTP Compression

You should test to ensure that HTTP Compression is working as expected before modifying these settings. Follow the instructions in the next section to determine if these settings are necessary in your environment.

The following additional settings may impact the functionality of HTTP Compression:

C:\Windows\system32\inetsrv\appcmd.exe set config -section:system.webServer/serverRuntime /frequentHitThreshold:1 /commit:apphost

C:\Windows\system32\inetsrv\appcmd.exe set config -section:system.webServer/serverRuntime /frequentHitTimePeriod:00:01:00 /commit:apphost

The default values are **2** and **00:00:10**, respectively.

For more information, see http://www.iis.net/ConfigReference/system.webServer/serverRuntime.

Test the HTTP Compression Configuration

Fiddler HTTP Debugging Proxy (http://www.fiddlertool.com) is a good tool for determining whether or not HTTP Compression is working as expected.

HTTP Compression Sections/Settings in applicationhost.config

The configuration of HTTP Compression described above modifies the three primary sections in applicationhost.config shown below. The specific settings that you modify are displayed in red:

- <urlCompression doStaticCompression="true" doDynamicCompression="true" />
- <httpCompression directory="%SystemDrive%\inetpub\temp\IIS Temporary Compressed Files">

  <scheme name="gzip" dll="%Windir%\system32\inetsrv\gzip.dll" staticCompressionLevel="9" dynamicCompressionLevel="4" />

  <staticTypes>

  <add mimeType="text/*" enabled="true" />

  <add mimeType="message/*" enabled="true" />

  <add mimeType="application/x-javascript" enabled="true" />

  <add mimeType="application/atom+xml" enabled="true" />

  <add mimeType="application/xaml+xml" enabled="true" />

  <add mimeType="*/*" enabled="false" />

  </staticTypes>

  <dynamicTypes>

  <add mimeType="text/*" enabled="true" />

  <add mimeType="message/*" enabled="true" />

  <add mimeType="application/x-javascript" enabled="true" />

```
            <add mimeType="application/octet-stream" enabled="true" />
            <add mimeType="application/x-ms-application" enabled="true" />
            <add mimeType="application/x-ms-manifest" enabled="true" />
            <add mimeType="*/*" enabled="false" />
        </dynamicTypes>
    </httpCompression>
```

- `<serverRuntime frequentHitThreshold="1" frequentHitTimePeriod="00:01:00" />`

Chapter 3: Configuring Secure Sockets Layer (SSL)

Important Information on SSL Configurations

Read this section to better understand:

- How the Vision reporting framework handles SSL requests.
- How the use of non-standard ports impacts functionality.
- What configurations are and are not possible using non-standard ports.

How the Reporting Framework Handles SSL

Each request to run a report in Vision includes several calls to the report server web service URL. Some of these calls are server-side (made from the Vision web/application server) and some of these calls are client-side (made from the Vision application on the user's workstation).

When Vision is configured for SSL, the SQL Reporting Services server must also be configured for SSL, or a reverse proxy must be configured to offload the SSL requests to HTTP before forwarding them to the report server. In either case, the default behavior is that server-side calls are always made using HTTP only, which requires that the report server have an HTTP binding configured.

An alternative approach is to use HTTPS for server-side calls. This approach works only if SSL is configured for use by both Vision and Reporting Services. Select the **Use HTTPS for Reporting Services server-side calls** option on the Report Server tab in WebLink to use HTTPS for server-side calls.

Client-side calls are always made using the protocol prefix used to access Vision (HTTP -> HTTP and HTTPS-> HTTPS). When you use SSL, the communication between the client and the server is always encrypted using SSL, whether or not a reverse proxy is used. If a reverse proxy is used and configured with SSL Offloading enabled, it will receive the HTTPS request and forward it to the report server over HTTP and will receive the response from the report server in HTTP and forward it back to the client over HTTPS. If a reverse proxy is not used, the SQL Reporting Services server will need to have an HTTPS binding in addition to the HTTP binding, or the **Use HTTPS for Reporting Services server-side calls** option must been selected to use HTTPS for server-side calls.

Non-Standard SSL Ports

While it is possible to use non-standard SSL ports with Vision, the default reporting framework behavior is that all server-side calls to the report server URL are made using HTTP. For this reason, if you are using a non-standard SSL port for your SQL Reporting Services URL (for example, http://<ReportServer>:4443/reportserver) in WebLink, you need to use a reverse proxy, such as ARR, and enable SSL Offloading. Alternatively, if you select the **Use HTTPS for Reporting Services server-side calls** option, you do not need to configure an HTTP binding for SSRS nor use a reverse proxy.

For more information about ARR, see Chapter 1, "Creating a Reverse Proxy for SQL Reporting Services Using Application Request Routing (ARR)."

In addition, you need to configure an HTTP port on SSRS with the same port value (for example, HTTPS web server port 4443 and SSRS HTTP port 4443). This will ensure that client requests to

the ARR reporting virtual directories work properly and that server-side calls from the web server to the report server also work properly.

Similar changes are required if you use a hardware- or software-based reverse proxy solution other than ARR. ARR is the only reverse proxy solution tested by Deltek.

You can successfully use a non-standard SSL port for your Vision URL, but to use non-standard ports with SSRS, you must do one of the following:

- Use a reverse proxy.
- Select the **Use HTTPS for Reporting Services server-side calls** option.
- Reconfigure your system to use standard HTTP/ HTTPS ports 80/443.

In a two- or three-tier Vision deployment where SSRS is on a different server than Vision, a configuration without a reverse proxy and SSL Offloading enabled would require that the same non-standard port be enabled on SSRS for both SSL and non-SSL bindings under the default reporting framework behavior. This is not possible due to the resulting port conflict. To resolve this issue, select the **Use HTTPS for Reporting Services server-side calls** option.

Likewise, using the default reporting framework behavior, you cannot configure a single server installation with non-standard ports for both Vision and SSRS, with or without ARR, because the same port would be required for both HTTP and HTTPS, resulting in a port conflict. However, you can have a single server installation of Vision and SSRS using standard HTTP/ HTTPS ports 443/80, with or without ARR. To resolve this issue, select the **Use HTTPS for Reporting Services server-side calls** option.

Secure the Vision Web Server

To configure Vision for use with SSL, you must either:

- Obtain an SSL certificate from an online certificate authority such as Verisign, Thawte, or Comodo, *or*
- Have access to a domain or stand-alone certificate authority on your network.

Request a Server Certificate

To complete the certificate request process:

1. Log on to the web server.
2. From Administrative Tools, open Internet Information Services Manager.
3. From the navigation pane at left, select your server navigation menu.
4. Double-click **Server Certificates** to display the Server Certificates window.
5. In the Actions pane, select one of the following options:
   - **Import** — If you already have a certificate for your server, select this action to import that certificate.
   - **Create Certificate Request** — Select this action to launch a wizard that guides you in creating a text file to submit to your Certificate Authority (CA) to obtain the actual SSL certificate for your web server.
   - **Complete Certificate Request** — If you used **Create Certificate Request** to request a certificate, select this action to complete your request and install your certificate.

- **Create Domain Certificate** — If you have a Certificate Authority on your domain, select this action to request your certificate.
- **Create Self-Signed Certificate** — Select this action to test SSL functionality or troubleshoot SSL certificate issues.

After you obtain and import your SSL Certificate, you create an SSL binding for your web server.

6. Expand **Sites**, and select your web site.
7. In the Actions pane, click **Bindings** to display the Site Bindings dialog box.
8. Click **Add**. The Add Site Binding dialog box displays.
9. From the **Type** drop-down list, select **https**. The **Port** value automatically changes to **443**.
10. From the **IP address** drop-down list, select your IP address or use the default setting **All Unassigned**.
11. From the **SSL Certificate** drop-down list, select your certificate.
12. Click **OK**.

Test the SSL Certificate and Binding

To test your new SSL certificate and binding, access your web site using **https://** as the URL prefix and make sure that everything is working correctly.

Secure SQL Server Reporting Services

The Reporting Services Configuration Manager does not directly support requesting and importing the SSL certificate, as IIS does. To request and import the SSL certificate on your Reporting Services server, you must use the Certificates MMC (Microsoft Management Console) snap-in, described below.

The SSL architecture of Vision is such that if you are using SSL for Vision, you **must** use SSL for Reporting Services. The only exception is in reverse proxy configurations, as described in “Important Information on SSL Configurations” above.

- You cannot run SSL for Vision without an SSL binding configured for Reporting Services.
- You cannot run Vision without SSL and still use SSL for Reporting Services.
- The Reporting Services web service URL in WebLink must reference the Fully Qualified Domain Name (FQDN) of the report server. This is specified in the SSL certificate. If the report server previously referenced a local netbios name, you must change it to the FQDN. The FQDN name must be in the format:

  http(s)://vision.companyname.com/reportserver

To secure SQL Server Reporting Services for Vision:

> If SQL Reporting Services and IIS are being used on the same server and you have already configured an SSL certificate for IIS, you do not need to use the Certificate MMC imported in steps 1 through 10 below. Start with Step 11.

1. Click **Start » Run**.
2. In the **Open** field on the Run dialog box, enter **mmc** and click **OK**. The MMC console launches.
3. Click **File » Add/Remove Snap-in**. The Add or Remove Snap-ins dialog box displays.
4. Select **Certificates**, and click **Add**.
5. Select **Computer account**, and click **Next**.
6. Select **Local Computer**.
7. Click **Finish**, and then click **OK**. You should now see the certificate store.
8. Next you need to request a new certificate or import an existing certificate.
9. Right-click the Personal folder, and point to **All Tasks**.
10. Select one of the following actions:
    - If you have a domain Certificate Authority (CA), click **Request New Certificate**.
    - If you need to request a certificate from a stand-alone CA or an online CA, click **Advanced Operations » Create Custom Request**.
11. After you have your SSL certificate, import it using the following steps:
    a. Right-click the Personal folder.
    b. Click **All Tasks » Import** to launch the Certificate Import Wizard.
    c. Browse to the location of your SSL certificate and complete the import process.

    At this point the certificate is registered with the server. The next step is to register the certificate with SQL Reporting Services.
12. Click **Start » All Programs » Microsoft SQL Server » Configuration Tools** to open the Reporting Services Configuration Manager on the Report Server.

    The next steps are to create the SSL bindings for the Web Service URL and the Report Manager URL.
13. Under **Connect**, click **Web Service URL**. The Web Service URL window displays.
14. Click **Advanced**. The Advanced Multiple Web Site Configuration dialog box displays.
15. Under **Multiple SSL Identities for the Report Server Web Service**, click **Add**. The Add a Report Server SSL Binding dialog box displays.
16. Select a specific **IP Address** (if appropriate).
17. Click the drop-down list for the **Certificate** option. The certificate you imported in the previous steps should display. Select the certificate.
18. Click **OK** to add this URL to the system.

> If all communication to the report server will be done via SSL, you should also remove the HTTP binding from the configuration.

19. Repeat these steps for the Report Manager URL.
20. On the Web Server, launch WebLink, log in, and select the database.
21. On the ReportServer page, verify that the URL contains a reference to the Fully Qualified Domain Name (FQDN) of the report server. This is specified in the SLL certificate. If the report server previously referenced a local netbios name, you must change it to the FQDN. The FQDN name must be in the format:

    http(s)://vision.companyname.com/reportserver

Test the SSL Configuration

Test Vision using SSL URLs to ensure that the product is functioning correctly. To do this, trace a Vision SSL session using Fiddler (http://www.fiddlertool.com) or another HTTP tracing tool.

Chapter 4: Pre-Deploying Deltek Vision Smart Client to User Workstations

The ClickOnce deployment technology is used for delivering Windows-based applications to the user. The Deltek Vision Smart Client application uses this technology to check for new updates on the application/web server each time that the application is launched, and automatically installs them into the local user's profile (%USERPROFILE%\Local Settings\Apps\2.0\...).

To reduce the size of the initial client-side download when a user launches Vision, you can pre-deploy the Smart Client files to user workstations. This "Hybrid Deployment Model" installs the application by first looking in a specific folder on the workstation and, if no file is found there, downloading the file from the application/web server.

When you use the Hybrid Deployment Model (HDM), ClickOnce delivers about 15 files (enough to display the login page). After that, HDM takes over to deliver core application assemblies, software updates, language-specific satellite assemblies, and custom items.

ClickOnce Deployment Features

- Applications are installed per-user, not per-computer.
- Administrator privileges are not required.
- Applications do not have to be installed through Add/Remove Programs.
- Nothing is registered to the GAC (Global Assembly Cache).
- No ActiveX objects, plug-ins, or Java applets are used.
- ClickOnce Cache Location:
  - "c:\Users\USERPROFILE_Name\AppData\Local\Apps\2.0"

Files to be Deployed

You must repeat the process below each time that you upgrade your Vision application/web servers to a new release.

The files that must be pre-deployed to the user workstations are located on the application/web server in the **\Program Files\Deltek\Vision\WebClient** folder (where Vision 7.x is installed):

- DeploymentManifest.xml
- One or more zip files (as listed in the DeploymentManifest.xml)

You must copy all of the zip files, plus the DeploymentManifest.xml, to the workstation.

The date and time on the time stamp for each zip file must match the date and time shown in the DeploymentManifest.xml.

Deploy Files to a Workstation

Use the procedure below to deploy files to your workstation. Your workstation must be running Windows 7 or higher.

To deploy files:

1. Locate the \ProgramData\ directory, and create the Deltek directory.

   By default, the \ProgramData folder on Windows 7 and higher is hidden. You may need to select the **Show Hidden Files** option in Windows Explorer.

2. Copy the DeploymentManifest.xml file and **all** zip files from the application/web server into the Deltek directory.

   You must repeat this process each time that you upgrade your Vision application/web servers to a new release.

Chapter 5: Integrated Security Configuration for Vision

Vision includes an option for Windows Integrated Authentication, which allows users to log in one time for both Windows and the Vision application. You configure the use of Windows Integrated Security for each user's Vision account by using the Windows Domain network login as the username for that user. This allows the user to be logged in automatically to the Vision application as long as they are logged in to the domain. If the user is not properly logged in to the domain, the user is prompted for network credentials before they can log in to Vision. For example, non-domain workstation users, as well as users connecting to the network via an Internet connection, will receive a domain authentication challenge before they are logged in to Vision.

The use of Integrated Security in IIS **requires** a CAL (Client Access License) for each user who will access that web server. This is a Microsoft, not Deltek, licensing requirement.

Required Configuration Changes

To configure Windows Integrated Authentication, several changes are required at the domain level and in IIS, in addition to configuring your domain user accounts in Vision:

- You must configure a domain user account as the IIS Application Pool identity for the DeltekVisionAppPool in IIS. The domain account does not require domain administrative rights. By default, the Vision installation creates a local Windows account, "DeltekVision," to serve this function. However, a domain account is required to support trusted domains as well as the default IIS Windows Integrated Security configuration of using Kernel Mode Authentication.
- The domain account used for the Application Pool Identity needs the following rights on the Vision web/application server:
  - The account must be a member of the following local groups:
    - Administrators group
    - IIS_IUSERS group
  - The account requires the following local security policy rights:
    - Allow log on locally
    - Log on as a service
    - Log on as a batch job
- You must change the Vision IIS Application (virtual directory) from using Anonymous Access to Windows Integrated Security.
- If you do not wish to use the default of Kernel Mode Authentication, a Service Principal Name (SPN) must be created for the domain user account that is the Application Pool Identity. You must have domain administrative rights to create the SPN. See "Configure a Service Principal Name" for more information.

Configure the Application Pool Identity

To configure the Application Pool Identity to be a domain account:

1. Click **Server Manager » Configuration » Local Users and Groups » Groups** and add the domain user to the local Administrators and IIS_IUSRS group.
2. In Administrative tools, click **Security Settings » Local Policies » User Rights Assignment** to grant the domain user the necessary rights.
3. Click **Administrative Tools » Internet Information Services » Application Pools** and change the application pool identity.
4. Right-click **DeltekVisionAppPool**, and click **Advanced Settings**.
5. In the **Process Model » Identity** field, click the ellipses (**…**). The Application Pool Identity dialog box displays.
6. Select **Custom Account**.
7. Click **Set**. The Set Credentials dialog box displays.
8. In the **Username** field, enter the domain and user name in the following format: **Domain\Username**. Click **OK**.
9. Launch Vision on the web/application server to ensure that the application launches correctly. If not, review the application event logs to look for a problem.

Configure Vision IIS to Use Windows Integrated Authentication

Do **not** make any modifications to the security settings for the VisionClient IIS Application. The application represents the ClickOnce deployment and must continue using Anonymous Access.

To configure Vision IIS to use Windows Integrated Authentication:

1. From within Internet Information Services, expand the web site where the Vision application is installed.
2. Select the Vision application.
3. Double-click the **Authentication** icon under IIS.
4. Select **Anonymous Authentication**, and click **Disable** on the Actions pane.
5. Select **Windows Authentication**, and click **Enable** on the Actions pane.
6. With Windows Authentication still selected, click **Advanced Settings**. Ensure that the **Enable Kernel-mode authentication** option is selected, and click **Cancel**.

   The default configuration is to have **Enable Kernel-mode authentication** selected. If you clear **Enable Kernel-mode authentication**, you must create a Service Principal Name, which is documented later in this section. The default setting is acceptable for application authentication; however, if you wish to use Windows authentication for your database connection, you must complete the “Configure Windows Integrated Authentication for Internet Users (and Non-Domain Workstations)” procedure.
7. Launch the Vision application. On the login page you should see the **Windows Authentication** option.

   The **Windows Authentication** option only displays if you have multiple databases configured in WebLink. If there is only one database, the user is automatically logged into Vision, and this screen does not display.

Configure Vision for Windows Integrated Authentication

After the servers are configured to support Windows Integrated Authentication, you must configure Vision application domain users with their domain logins.

WebLink has options on the System Settings Tab that may alleviate performance issues when using Windows Integrated Authentication. Details for these configuration settings are documented in the WebLink help. Test changes to these settings thoroughly before you implement the changes in a production environment.

To configure a domain login for Vision:

1. Launch the Vision application, and log in as a user with the appropriate security rights.
2. Click **Configuration » Security » Users**, and create a new user.
3. Enter the domain username for the user you want to create (for example, the login ID used to log in to the Windows domain).
4. Complete the additional information required for this user.
5. Select the **Windows Authentication** option.
6. From the **Domain** drop-down list, select the domain for this user. If the domain is not listed, you can manually enter the netbios name of the domain.
7. Save your changes.

   When the user launches Vision, the login screen displays with the **User ID** and password blank and the **Windows Authentication** check box cleared unless the WebLink option **Automatically check Windows Authentication check box in WebLink** is selected.

   - After a login with **Windows Authentication** selected, the option is remembered for subsequent logins.
   - If there is only one database defined in WebLink and the application is configured for Windows Integrated Authentication, the user is automatically logged in to the application on all subsequent logins after the initial login.

Configure Windows Integrated Authentication for Internet Users (and Non-Domain Workstations)

A different authentication process applies to domain users who are configured for Windows Integrated Authentication but are accessing the application from a non-domain workstation or via the Internet.

To configure Integrated Authentication for Internet users:

1. Launch the Vision application. The Internet Explorer security prompt displays. This is expected because the user is not authenticated to the domain and IIS is configured for Windows Integrated Authentication so that only authenticated users can access without a challenge.

   Select the **Remember my credentials** option if you want to save your credentials for both the browser and the WinForms application. In the future, you will not be prompted for credentials.

2. Enter the domain credentials, and click **OK**. The Windows Login Credentials dialog box displays.

3. Enter values in the **Username**, **Password**, and **Domain** fields. This is necessary because the client side WinForms application is not able to use the previous credentials requested by, and processed by, Internet Explorer.

Configure Windows Authentication for the Vision Database Connection

The first step to use Windows Integrated Authentication for the Vision database connection is to grant the domain user account running the IIS Application Pool Identity the appropriate rights to the Vision database (and the Report Server and Session State databases, as needed).

To establish rights for SQL Server:

1. Identify the domain user account that is being used as the Application Pool Identity in IIS. See step 3 in the “Configure the Application Pool Identity” section**.**
2. In SQL Server Enterprise Manager, create a SQL login for this domain user account.
3. Click **User Mapping** and grant db_owner rights to the Vision database (and the Report Server and Session State databases).
4. Modify WebLink to use Windows Integrated Authentication for the various database connections. Complete steps 5 through 10 to enable these settings.
5. Launch WebLink, and enter the WebLink password when prompted.
6. Click **OK**. The WebLink screen displays.
7. From the **Current Database** drop-down list, select the database to which you want to connect.
8. Select the **Windows Authentication** option to use the domain Application Pool Identity user account to connect to the database.
9. If necessary, you can also enable Windows Authentication for the Report Server database connection. In this situation, the account requiring access may differ from the one used for the IIS Application Pool Identity. The account that will be used to make this connection is shown on this page as the **Windows Username** under the Report Server URL. If this is a different account than the IIS Application Pool Identity, you must grant db_owner rights to the Report Server databases and then select the Windows Integrated option for the Report Server database authentication.

   Optionally, if you are using SQL Server Session state, you can also enable Windows Authentication for that connection. This will use the IIS Application Pool Identity to make the database connection.

   WebLink has options on the System Settings Tab that may alleviate performance issues when using integrated authentication. See the WebLink help for information about these configuration settings. Changes to these settings should be thoroughly tested before implementing them in a production environment.
10. On each of the tabs in WebLink, select the test button to test the connection.

Configure a Service Principal Name

To disable Kernel Mode Authentication, you must create a Service Principal Name (SPN) for the domain user account that is the Application Pool Identity. The creation of the SPN requires domain administrative rights.

IIS Kernel Mode Authentication

When you use Windows Integrated Authentication, the default configuration of IIS is to use Kernel Mode Authentication. If you must disable Kernel Mode Authentication, follow the steps in this section to establish a Service Principal Name (SPN) for the Application Pool Identity.

In a default configuration of IIS, Kernel Mode Authentication is enabled.

To see if Kernel Mode Authentication is enabled:

1. Using an Administrator account, log on to the Vision web/application server domain.
2. To open Internet Information Services, click **Start » All Programs » Administrative Tools » Internet Information Services (IIS) Manager**.
3. Expand the server name, expand **Sites**, and select **Default Web Site** (or the site where Vision is installed).
4. Select the Vision virtual directory, and double-click **Authentication** in the Features view.
5. Select **Windows Authentication** and verify that the status is **Enabled.** (Anonymous Access should be **Disabled**.) If it is not, select **Enable** from the **Actions** menu.
6. With **Windows Authentication** still selected, click **Advanced settings** on the **Action** menu. The Advanced Settings dialog box displays.
7. If the **Kernel Mode Authentication** check box is selected, Kernel Mode Authentication is enabled.

Kernel Mode Authentication Implementation

The default configuration works for the Vision Windows Integrated Authentication application and database connections.

To disable Kernel Mode Authentication, clear the **Enable Kernel Mode Authentication** selection under the Advanced Settings of the Windows Authentication feature for the Vision virtual directory. Disabling Kernel Mode Authentication requires that a Service Principal Name be established for the Application Pool Identity.

Service Principal Names

Under the default configuration with Kernel Mode Authentication enabled, it is not necessary to create a Service Principal Name for the Application Pool Identity. The default SPNs created are sufficient.

If you do create an SPN for the Application Pool Identity, there will be a duplicate SPN issue that prevents Windows Integrated Security from authenticating anyone to the web site.

When Kernel Mode Authentication is disabled, complete the following steps to create a Service Principal Name for the Application Pool Identity of the DeltekVisionAppPool.

To create the Service Principal Name:

1. Log on to the server with domain administrative rights.
2. Run the following commands:
   - setspn –A http/<name of server> ApplicationPoolIdentity (Domain\Username)
   - setspn –A http/<fully qualified name of server> ApplicationPoolIdentity (Domain\Username)

Or, if appropriate, use the DNS name of the server:

- setspn –A http/<DNS name of server> ApplicationPoolIdentity (Domain\Username)

Refer to the following related Microsoft Knowledge Base article if you need additional details:

http://support.microsoft.com/?id=871179

Configure Authentication Persistence

When you use Windows Integrated Authentication in IIS, every request made by the client is authenticated, by default, using one of two Windows authentication providers: Negotiate or NTLM. This repeated authentication causes extra round trips between the client and server for each request and can impact performance, especially on latent connections. However, if you use Authentication Persistence, the server authenticates only the initial request from the client and does not perform authentication on subsequent requests on the same connection, thus improving performance.

Source of Extra Round Trips

The default Windows authentication provider is Negotiate, which causes the client and server to "negotiate" an authentication method that both can support. On a typical Active Directory network, the default authentication method is Kerberos. On non-domain, or more specifically, internet-based connections, the default authentication method is NTLM.

If you view the connections for a single user in IIS logs, you see something like this:

NTLM:

2016-01-05 16:24:51 10.5.12.16 POST /Vision/MethodCall.aspx - 80 - 10.5.4.5 - - 401 2 5 484

2016-01-05 16:24:51 10.5.12.16 POST /Vision/MethodCall.aspx - 80 - 10.5.4.5 - - 401 1 2148074254 15

2016-01-05 16:24:51 10.5.12.16 POST /Vision/MethodCall.aspx - 80 DOMAIN\user 10.5.4.5 - - 200 0 0 501

2016-01-05 16:24:51 10.5.12.16 POST /Vision/MethodCall.aspx - 80 - 10.5.4.5 - - 401 2 5 15

2016-01-05 16:24:51 10.5.12.16 POST /Vision/MethodCall.aspx - 80 - 10.5.4.5 - - 401 1 2148074254 0

2016-01-05 16:24:51 10.5.12.16 POST /Vision/MethodCall.aspx - 80 DOMAIN\user 10.5.4.5 - - 200 0 0 15

2016-01-05 16:24:51 10.5.12.16 POST /Vision/MethodCall.aspx - 80 - 10.5.4.5 - - 401 2 5 0

2016-01-05 16:24:51 10.5.12.16 POST /Vision/MethodCall.aspx - 80 - 10.5.4.5 - - 401 1 2148074254 0

2016-01-05 16:24:51 10.5.12.16 POST /Vision/MethodCall.aspx - 80 DOMAIN\user 10.5.4.5 - - 200 0 0 15

2016-01-05 16:24:54 10.5.12.16 POST /Vision/MethodCall.aspx - 80 - 10.5.4.5 - - 401 2 5 15

2016-01-05 16:24:54 10.5.12.16 POST /Vision/MethodCall.aspx - 80 - 10.5.4.5 - - 401 1 2148074254 0

2016-01-05 16:24:54 10.5.12.16 POST /Vision/MethodCall.aspx - 80 DOMAIN\user 10.5.4.5 - - 200 0 0 140

Kerberos:

2016-01-05 17:37:46 10.5.12.16 POST /Vision/MethodCall.aspx - 80 - 10.5.4.5 - - 401 2 5 0

2016-01-05 17:37:47 10.5.12.16 POST /Vision/MethodCall.aspx - 80 DOMAIN\user 10.5.4.5 - - 200 0 0 93

2016-01-05 17:37:47 10.5.12.16 POST /Vision/MethodCall.aspx - 80 - 10.5.4.5 - - 401 2 5 0

2016-01-05 17:37:47 10.5.12.16 POST /Vision/MethodCall.aspx - 80 DOMAIN\user 10.5.4.5 - - 200 0 0 15

2016-01-05 17:37:47 10.5.12.16 POST /Vision/MethodCall.aspx - 80 - 10.5.4.5 - - 401 2 5 0

2016-01-05 17:37:47 10.5.12.16 POST /Vision/MethodCall.aspx - 80 DOMAIN\user 10.5.4.5 - - 200 0 0 15

2016-01-05 17:37:50 10.5.12.16 POST /Vision/MethodCall.aspx - 80 - 10.5.4.5 - - 401 2 5 0

2016-01-05 17:37:50 10.5.12.16 POST /Vision/MethodCall.aspx - 80 DOMAIN\user 10.5.4.5 - - 200 0 0 144

For NTLM, note that there are two 401 (unauthorized) HTTP status codes (401.2 and 401.1) for each HTTP 200 (success) status code. Each of these 401/401/200 codes represents a single request from the client.

With Kerberos, there is only one 401 HTTP status code for each 200 status code.

Each 401 represents an extra round trip between the client and server, which can decrease overall performance, especially on latent connections, such as from remote offices over a slower WAN link or via the Internet.

Configuration Options for Authentication Persistence

If performance is a concern, you can enable Authentication Persistence via the Advanced Server Connection Settings on the System Settings tab of the Weblink utility.

Select one of the following configuration options.

Option	Description
Share a single HTTP connection for all HTTP requests and establish authentication persistence when using Windows Integrated Authentication (supports Kerberos only)	Select this option if you are sure that all connections are authenticating via Kerberos (for example, if you have an Active Directory domain to which all users authenticate and you do not have Vision open to the Internet). Be sure that **Negotiate** is at the top of the list in the Providers link under the Windows Authentication configuration for the Vision application in IIS. You can further validate if Kerberos is in use by reviewing the IIS logs and comparing them to the examples above.
Share a single HTTP connection for all HTTP requests when using Windows Authentication (supports Kerberos and NTLM) *requires additional configuration – see help documentation for more details*	Select this option to support both Kerberos and NTLM authentication if Vision is open to the Internet. You must also change the IIS configuration to add the **authPersistNonNTLM** setting, described in this Microsoft KB article: https://support.microsoft.com/en-us/kb/954873.
None Selected (default)	The default behavior of IIS does not change.

If the client is not going through a proxy to access the web server, select the related **Disable automatic proxy detection on HTTP requests (can improve performance over high latency connections)** check box to disable automatic proxy detection. This approach only has an impact on highly latent connections (100ms or higher).

Use IIS Logs to Confirm Authentication Persistence

You can review IIS logs using Excel to see if authentication persistence is configured properly.

Fiddler is a great tool for debugging HTTP issues. However, to determine if authentication persistence is enabled and working, use IIS logs. Examining IIS logs lets you see what is occurring from the perspective of the server rather than the client.

To review IIS logs to confirm authentication persistence:

1. Copy the log to your desktop or to another working location.
2. Open the log in Notepad.
3. Remove all of the header information or the log will not parse properly:
   a. At the top of the log, note the first four rows:

      #Software: Microsoft Internet Information Services 8.0

      #Version: 1.0

      #Date: 2016-01-05 16:24:50

      #Fields:

   b. After #Fields, find the word **date** followed by all of the other column headers.
   c. In Notepad, place your cursor just before the **d** in **date**, press ENTER, and then delete the first four rows.
   d. Search through the log for additional instances of this header information, because IIS restarts will cause the header information to repeat in the log. Delete any additional instances that you find.
4. Open the log with Excel:
   a. Browse to the location where you edited the log file.
   b. On the Open File dialog box, select **All Files** from the **File Type** drop-down list. By default, you will only see Excel files.
   c. When the Text Import Wizard starts, select the **Delimited** option and click **Next**.
   d. Select **Space** as the delimiter and click **Finish**.
5. When the file displays in Excel, locate the **c-ip** column and filter on unique client IP addresses.
6. Examine the requests.

   If persistence is configured properly, you will see a few 401s at the beginning of the user's session, but the majority of requests will display as 200s.

   It is normal to see other 401s. As long as they are not repetitive, persistence is configured properly.

Chapter 6: Configuring Database Session State for Vision

Session state information is typically stored in memory on the web server in the IIS Application Pool process serving the application (w3wp.exe). Database session state is normally not a consideration unless you will be load balancing multiple front-end Vision web/application servers and you would like to isolate your user's session information from a failure or error on one web server where their session information may be lost.

Use the WebLink utility to configure Vision to store session state information in a database. Be aware that Deltek has written our own session state model and does not rely on ASP.NET session state.

Create the Session State Database (Optional)

If you want to store session state information in the Vision database, WebLink automatically configures the database table where session state information is stored. However, if you want to store the database table in a database other than your Vision database, you must create a separate database and login for this purpose.

Configure Vision for Database Session State

Before you make this change, make sure that you are making it during a maintenance window when no one is logged into Vision. Changing the session state invalidates all active user sessions.

To configure Vision for database session state, complete the following steps on the web/application Server:

1. Launch and log in to the Vision WebLink utility.

   The shortcut for WebLink is under the Deltek Vision program group on the Start menu.

2. Click the System Settings tab.
3. In the drop-down field, change **Store Session State in Memory** to **Store Session State in SQL Server**. You will receive a message that switching session state modes will force a restart of ASP.NET and all users will lose their sessions.
4. Click **Yes**.
5. In the **SQL Server** field, enter the name of the database server where the session state database exists.
6. In the **Database Name** field, enter the name of the session state database.
7. If you plan to use Windows Authentication for the database connection, select the **Windows Authentication** option. This disables the **SQL Username** and **SQL Password** fields.

   See Chapter 5, "Integrated Security Configuration for Vision," for details on Windows authentication.

8. In the **SQL username** field, enter the SQL Login ID with rights to that database.

9. In the **SQL password** field, enter the password for the SQL Login ID that you entered in the previous step.
10. Click **Test Database Connection** to validate the connection information that you entered.
11. Click **Apply** to save your changes.

    A message displays that prompts you to configure this database to store session state information.
12. Click **Yes** to create a table in the database called **FW_SessionState**. A message displays to tell you that WebLink has successfully configured the server (database) to store the Vision session state.

To verify that the session state is being stored correctly:

1. Access your database server via a query utility.
2. Log in to Vision.
3. Run the following query in Query Analyzer to verify that a row has been added to the table. There will be one row for every user logged in.

```
Use <Session State Database>
Go
Select * from FW_SessionState
Go
```

If Vision is not closed out properly, user sessions can be orphaned. A process server job named “Delete Old Sessions” runs every night to remove orphaned user sessions.

Chapter 7: Securing Your Deltek Vision Deployment

The default installation of Deltek Vision creates a variety of user accounts on the Vision physical tiers (Database, Web/Application, Report, and Process Server). These user accounts include local Microsoft Windows user accounts and SQL Server Login IDs (for both Windows and Mixed Mode authentication).

This section guides you through making the necessary changes to secure your Vision deployment on the various physical tiers. Use it to change accounts so that they are unique to your firm and do not include any Deltek default user accounts or passwords.

Most of these changes require Administrative rights on your servers, so be sure to log in with the proper account. Do **not** log in using the DeltekVision local account because you will be deleting or disabling this account on all Vision servers.

If You Have Deployed Several Logical Tiers with the Same Windows Account

You may have deployed several logical tiers, all using the same Windows account and all located on the same physical server. For example, in a single-server installation, the DeltekVision local Windows account is used as all of the following:

- The Application Pool Identity
- The Reporting Services access account
- The Process Server service account
- A Windows SQL Login account

If the account is serving multiple roles, you may not need to delete or disable the accounts as many times as indicated in these instructions.

Web/Application Tier

The web/application tier installation creates a local Windows user account named **DeltekVision**. This account is also added to the Local Administrators group and the IIS_IUSRS group and is configured as the Application Pool Identity of the DeltekVisionAppPool.

To secure the web/application tier and customize the Application Pool Identity:

1. Change the Application Pool Identity.
2. Select one of the following actions:
   - If you are using a Windows domain, create a domain user account, or use an existing one. Then add this user to the Local Administrators group and IIS_IUSRS group on the web/application server.
   - If you are not using a Windows domain, create a new local Windows user account and add that user to the same Windows groups.
3. Log on to the domain on the Vision web/application server using an Administrator account.
4. Click **Start » All Programs » Administrative Tools » Internet Information Services (IIS) Manager** to open Internet Information Services.

5. Expand the Server name, and click **Application Pools**.
6. Select the **DeltekVisionAppPool**, and select **Advanced Settings** from the Action pane on the right-hand side.
7. Place your mouse pointer in the **Identity** field, and click the ellipses (…) button to set the identity.
8. Select the **Custom account** option, and click **Set**.
9. In the **User name** field on the Set Credentials dialog box, enter the Application Pool Identity in the form <Domain>\<Username>.
10. In the **Password** and **Confirm password** fields, enter the user's password.
11. Click **OK** three times to set the identity.

    After this process is completed, if you are using Windows Integrated Authentication for the SQL Server connection, you need to add the Domain user to the Local User (not Administrators) group on the SQL Server and grant this new Domain user dbo (database owner) rights to your Vision database(s).

See "Database Tier" for more information.

12. Change the Process Server service account. See "Process Server Tier" for more information. By default, the Process Server service is installed on every web/application server, as well as on any server installed as a Dedicated Process Server.
13. Click **Computer Management » Local Users and Groups » Users** and then delete or disable the local Deltek Vision Windows user account on the web/application server.

Database Tier

The Database tier installation creates a local Windows user account on the SQL Server named **DeltekVision,** as well as a SQL Server Login ID, also named **DeltekVision**.

SQL Server has two modes of authentication:

- Windows Integrated
- Mixed Mode

If you are unsure which mode of authentication you are using:

1. Click **Start » All Programs » Deltek Vision » WebLink** to launch the Vision WebLink utility on the web/application server.
2. If you have not set a password for WebLink, click **Change Password**, and enter a unique password. The **Change Password** button is visible only when accessing WebLink via localhost on the web/application server.
3. Log in to WebLink, and select your database from the **Current Database** drop-down list.
4. Review the information on the General tab to identify your method of SQL authentication.
   - If the **Windows Authentication** check box is selected, then you are using Windows Integrated Authentication.
   - If the **Windows Authentication** check box is cleared and a SQL username and password are filled in, then you are using SQL Server or Mixed Mode authentication.

Windows Integrated Authentication

If you are using Windows Integrated Authentication for the SQL Server connection, the local Windows user account is created on the database tier. You need to update the database tier with the new user account that you created for the Vision Application Pool Identity in the Web/Application Tier section.

See Chapter 5, “Integrated Security Configuration for Vision,” for detailed information on configuring Windows Integrated Security for web/application and database connections.

To update the database tier with the new user account:

1. Select one of the following options:
   - If you are using a Domain user account for the IIS Application Pool Identity, add this Domain user to the Local Users (not Administrators) group on the SQL Server.
   - If you are using a Local user account as the IIS Application Pool Identity, use the Computer Management utility from Administrative Tools to create a local user on the database server with the same username and password as you used on the web/application server.

   Administrative rights to your database server are not necessary for the domain or local user account described above.
2. Create a new Windows login in SQL Server for the Domain or Local user account that is being used for the IIS Application Pool Identity. Create the new SQL Login using SQL Server Management Studio from the Security - Logins folder.
3. Grant this new Windows login dbo (database owner) rights to your Vision database(s).
4. On the web/application server, launch the WebLink utility.

5. Log in to WebLink, and select your database from the **Current Database** drop-down list.
6. If it is not already selected, select the **Windows Authentication** check box.
7. To ensure that the database connection information was updated correctly, click the **Test » Database Connection** to validate the connection.
8. Use the **Computer Management** utility under Administrative Tools to delete or disable the local DeltekVision Windows user account on the Database server.
9. Use SQL Server Management Studio to delete or disable the DeltekVision Windows Login ID from within SQL Server. Deleting the Windows User Account does not remove it from SQL Server. However, disabling it disables the account in SQL Server.

Mixed Mode Authentication

If you are using Mixed Mode Authentication for the SQL Server database connection, you need to create a unique SQL Server login.

To create the SQL Server login:

1. If you haven't already done so, secure the **sa** account with a unique password using SQL Server Management Studio from the Security - Logins folder.
2. Create a unique SQL Server login ID and password using SQL Server Management Studio.
3. Grant the new login dbo (database owner) rights to your Vision database(s) and, if appropriate, the Reporting Services databases (ReportServer and ReportServerTempDB).
4. If you want to use a different account for report server database access, create a second SQL login in SQL Server Management Studio and manually update the Report Server tab in WebLink with the new connection information. Be sure to test the connection before saving your changes.
5. Log in to WebLink, and select your database from the **Current Database** drop-down list.
6. On the General tab, enter the new SQL Server login username and password.
7. To ensure that the database connection information was updated correctly, click the **Test » Database Connection** to validate the connection.
8. Update the Report Server tab with the new connection information for the Report Server databases.
9. Use SQL Server Management Studio to delete or disable the DeltekVision SQL login ID that the installation created on the Database server.

Report Tier

The Report tier installation creates a local Windows user account named DeltekVision on the Report Server (SQL Reporting Services server). This Windows user account is also granted System Administrator and Content Manager Rights in SQL Reporting Services.

When you created the database tier account, access rights were automatically given to the Report Server databases for the new user account.

To secure the Report tier and customize the Report tier account:

1. Select one of the following actions:
   - If you have a Windows Domain, create or have a Domain user account created, and use the Computer Management utility under Administrative Tools to add this user to the Local Administrators group on the Report server. Click **Computer Management** » **Local Users and Groups** » **Administrators**, and add the new Windows account.
   - If you are not using a Windows Domain, you need to create a new local Windows user account and add that user to the Local Administrators group.

   The next step is to use Report Manager to grant this new account the necessary rights in Reporting Services.
2. Use http://<report_server>/reports to open Report Manager, replacing <report_server> with the name of your Report Server.

   To access Report Manager, you must already have been granted rights to the Report Server. You may also need to launch Internet Explorer using the **Run as Administrator** option.
3. Click the **Site Settings** link in the upper right corner and add the new account to the Administrators role.
4. Delete the DeltekVision account from this role.
5. Click the Properties tab or the **Folder Settings** button (depending on your version of SQL Server).
6. Add your new account to the Content Manager Role.
7. Delete the DeltekVision account from that role.
8. Use the Computer Management utility from Administrative Tools to delete or disable the local DeltekVision Windows user account on the Report Server.

Process Server Tier

The Process Server tier installation creates a local Windows user account named DeltekVision on the Process Server.

By default, the Process Server service is installed on every web/application server, as well as on any server installed as a Dedicated Process Server. Therefore, you should perform the following steps on every web/application server as well as on every dedicated Process Server where the process server service will run.

In this procedure, you change the Process Server Service Account (Windows account on the Process Server tier).

To secure the Process Server tier and customize the Process Server service:

1. Select one of the following actions:
   - If you have a Windows Domain, create or have a Domain user account created. Use the Computer Management utility under Administrative Tools to add this user to the Local Administrators group on the Report server. Click **Computer Management** » **Local Users and Groups** » **Administrators**, and add the new Windows account.
   - If you are not using a Windows Domain, you need to create a new local Windows user account and add that user to the Local Administrators group.

2. To change the service to run your new account, click **Start » Control Panel » Administrative Tools » Services**, and locate the Deltek Vision Process Server service.
3. Update the **Log On As** column to reflect the new user information that you created above.
4. Use the Computer Management utility from Administrative Tools to delete or disable the local DeltekVision Windows user account on the Process server.

If You Have Multiple Servers

Your Vision installation should now be secured with customized username and password information unique to your firm on every tier. If you have multiple web/application, report, or process servers, make sure that you repeat the same steps on each physical server, using the same user account information that you used on the first server for that tier.

Chapter 8: Reporting Services Logging

Reporting Services include several different types of logging to help debug reporting services issues. You can enable two kinds of logging:

- Trace logging, which provides more detailed logging on errors or warnings seen in the Reporting Services logs
- HTTP logging, which helps identify issues with HTTP related issues for Report Manager or the Report Server web service

For more information, see the Microsoft library article: http://msdn.microsoft.com/en-us/library/ms157403.aspx.

Enable Reporting Services Trace Logging

To configure trace logging:

1. Stop the Reporting Services service.
2. Modify the ReportServer\bin\ReportServerService.exe.config file, as shown in the "Changes for the ReportServer\bin\ReportServerService.exe.config File" section.
3. Restart the Reporting Services service.

   The trace level rules are as follows for a particular component and are defined in the ReportServerService.exe.config file:

   - If there is a component-wide trace level defined in RSTrace/Components, then it takes precedence.
   - If the trace level is defined for **all**, it uses that level (for example, **all:3**).
   - If neither is defined, the default trace level DefaultTraceSwitch defined in system.diagnostics/switches is used.

For more information, see the Microsoft library article: http://msdn.microsoft.com/en-us/library/ms156500.aspx.

Components for Which You Can Enable Tracing

You can enable tracing for the following components:

- Library
- ConfigManager
- WebServer
- NtService
- Session
- BufferedResponse
- RunningRequests
- DbPolling
- Notification
- Provider

- Schedule
- Subscription
- Security
- ServiceController
- DbCleanup
- Cache
- Chunks
- ExtensionFactory
- RunningJobs
- Processing
- ReportRendering
- HtmlViewer
- DataExtension
- EmailExtension
- ImageRenderer
- ExcelRenderer
- PreviewServer
- ResourceUtilities
- ReportPreview
- UI
- Crypto
- SemanticModelGenerator
- SemanticQueryEngine
- AppDomainManager
- HttpRuntime

Changes for the ReportServer\bin\ReportServerService.exe.config File

Make the highlighted modifications to the ReportServer\bin\ReportServerService.exe.config file:

```
<configuration>
  <configSections>
    <section name="RStrace"
type="Microsoft.ReportingServices.Diagnostics.RSTraceSectionHandler,Microsoft.ReportingServi
ces.Diagnostics" />
  </configSections>
  <system.diagnostics>
    <switches>
      <add name="DefaultTraceSwitch" value="3" />
    </switches>
  </system.diagnostics>
  <RStrace>
```

```
<add name="FileName" value="ReportServerService_" />
<add name="FileSizeLimitMb" value="32" />
<add name="KeepFilesForDays" value="14" />
<add name="Prefix" value="tid, time" />
<add name="TraceListeners" value="debugwindow, file" />
<add name="TraceFileMode" value="unique" />
<add name="HttpTraceFileName" value="ReportServerService_HTTP_" />
<add name="HttpTraceSwitches" value="date,time,
clientip,username,serverip,serverport,host,method,uristem,uriquery,protocolstatus,bytesreceived,t
imetaken,protocolversion,useragent,cookiereceived,cookiesent,referrer" />
<add name="Components"
value="all:3,http:3,Library:4,EmailExtension:4,Subscription:4,Schedule:4,Notification:4,DbPolling:
4,NtService:4" />
</RStrace>
```

Errors in the Reporting Services Log File

Several lines from a reporting services log file showing errors are shown below:

session!ReportServer_0-1!e10!09/11/2011-13:14:51:: i INFO: LoadSnapshot: Item with session: pvon0l55nrycom3uczjnho45, reportPath: , **userName: KL\deltekadmin not found in the database**

library!ReportServer_0-1!e10!09/11/2011-13:14:51:: e ERROR: **Throwing Microsoft.ReportingServices.Diagnostics.Utilities.ExecutionNotFoundException: Execution 'pvon0l55nrycom3uczjnho45' cannot be found,** ;

Info: Microsoft.ReportingServices.Diagnostics.Utilities.ExecutionNotFoundException: Execution 'pvon0l55nrycom3uczjnho45' cannot be found

The sections (or “Components”) that can be traced are identified at the beginning of each log entry and appended with an exclamation point. For example, if you want verbose logging for the errors in the example above, enable library and session verbose logging as follows:

```
<add name="Components" value="all:3,Library:4,Session:4" />
```

Enable Reporting Services HTTP Logging

For the Microsoft library article on this topic, see the following link:

http://msdn.microsoft.com/en-us/library/bb630443.aspx

The Reporting Services service runs its own http.sys listener to accept standard HTTP/HTTPS requests on standard HTTP ports (80/443). Unlike Internet Information Services, HTTP logging is not enabled by default, but can be enabled following the steps in the MSDN article referenced above. This logging helps you troubleshoot HTTP and authentication related issues.

You can also use Fiddler to trace the HTTP requests from client to report server for help troubleshooting these kinds of issues. Obtain Fiddler at http://www.fiddler2.com.

Chapter 9: Deltek Vision Transaction Document Management

Vision Transaction Document Management (TDM) uses Microsoft SQL Server FILESTREAM technology to store and retrieve documents in a SQL Server database. Deltek has chosen to configure FILESTREAM functionality and to store these documents in a separate database rather than in your Vision transactional database. These documents include transaction-related supporting documents as well as Adobe InDesign templates.

This chapter will help you configure FILESTREAM and Vision TDM.

Your separate Vision and FILESTREAM (TDM) databases must be backed up on the same schedule so that they will be in sync if a restore is needed.

Prerequisites

Install or upgrade to the current version of Vision.

FILESTREAM Best Practices

The following article provides detailed information on FILESTREAM best practices:

http://msdn.microsoft.com/en-us/library/dd206979(v=sql.105).aspx

Some examples of FILESTREAM best practices noted in the MSDN article are:

- Disable short file names on FILESTREAM computer systems because they take significantly longer to create. To disable short file names, use the Windows fsutil utility.
- Regularly defragment FILESTREAM computer systems.
- Use 64-KB NTFS clusters. Compressed volumes must be set to 4-KB NTFS clusters.
- Disable indexing on FILESTREAM volumes and use the Windows fsutil utility to set disablelastaccess**.**
- Disable antivirus scanning of FILESTREAM volumes, if possible. If antivirus scanning is necessary, avoid setting policies that will automatically delete offending files.
- Ensure that your nightly backup routine (and any other backup processes) back up the FILESTREAM database at the same time as your Vision transaction database. Metadata for uploaded files is stored in the Vision database, while the actual uploaded file data is stored in the FILESTREAM database. If you restore one or both databases and the database backups are out of sync, data issues may occur. See Data Synchronization Issue later in this chapter.

The Vision Backup Utility application, on the Utilities menu, automatically backs up both the Vision transaction database and the FILESTREAM database.

Installation Overview

Use the steps in the following sections to configure Vision TDM with SQL Server FILESTREAM. The four primary steps are:

1. Identify the SQL Server to host the FILESTREAM database.
2. Enable FILESTREAM on SQL Server.
3. Identify the Physical Disk Location of the FILESTREAM Data.
4. Create the FILESTREAM database.

Identify the SQL Server to Host the FILESTREAM Database

In many Vision configurations, the SQL Server that hosts the Vision transaction database also hosts the FILESTREAM database. However, Vision TDM has been developed to allow the FILESTREAM database to exist on a separate SQL Server instance.

Enable FILESTREAM on SQL Server

You must enable FILESTREAM on the SQL Server instance intended to host the FILESTREAM database before you can create the database. Because FILESTREAM is not enabled by default, you must enable FILESTREAM during the SQL Server installation or after SQL Server is installed. Refer to the appropriate section for your installation.

Enable FILESTREAM During SQL Server Installation

To enable FILESTREAM during SQL Server installation:

1. Open the Database Engine Configuration.
2. Click the FILESTREAM tab and ensure that the following options are selected:
   - **Enable FILESTREAM for Transact-SQL access**
   - **Enable FILESTREAM for file I/O streaming access**
   - **Allow remote clients to have streaming access to FILESTREAM data**

   By default, the Windows share name created for FILESTREAM access will be the SQL Server instance name (default SQL instances are named MSSQLSERVER). Deltek recommends that you use the default selections.

3. Click **Next** to continue.

Enable FILESTREAM after SQL Server is installed

To enable FILESTREAM after installing SQL Server:

1. On the database server, open the SQL Server Configuration Manager.
2. Right-click your SQL Server service, and click **Properties** on the shortcut menu.
3. Click the FILESTREAM tab, ensure that all three options are selected, and click **OK**.

   By default, the Windows share name created for FILESTREAM access will be the SQL Server instance name (default SQL instances are named MSSQLSERVER). Deltek recommends that you use the default selections.

In addition, complete the following configuration settings in SQL Server properties:

1. Open SQL Server Management Studio.
2. Right-click the server, and click **Properties**.
3. Select the **Advanced** page.
4. Check to ensure that the **Running Values** are displaying that the **Filestream Access level** is set to **Full access enabled**.
5. Click **OK**, and restart the SQL Server service.

Identify the Physical Disk Location of the FILESTREAM Data

Before you create the FILESTREAM database, you must determine where the FILESTREAM data will be stored. By default, the FILESTREAM data is stored in a folder and sub-folders in your SQL Server Data folder. Depending on your SQL Server installation and disk configuration, it may be better to place this data on a separate physical disk, partition, RAID array, and so on.

See the “FILESTREAM Best Practices” section for more information.

Create the FILESTREAM Database

To create the FILESTREAM database:

1. Open the SQL Server Management Studio (SSMS).
2. Right-click the Databases folder, and click **New Database** on the shortcut menu.
3. Enter a name for the database.

   The name of the FILESTREAM database **must** be your Vision database name with **FILES** appended to it.

   For example, if your Vision database is **VisionDemo**, then your FILESTREAM database will be **VisionDemoFILES**.
4. Select the Filegroups page from the left menu.
5. Click **Add Filegroup** in the FILESTREAM section to add a new Filegroup.
6. Enter a name for the filegroup.

   The name of the Filegroup **must** be your Vision FILESTREAM database name with **_FS** appended to it.

   For example, if your Vision FILESTREAM database is **VisionDemoFILES**, then the name of the Filegroup will be **VisionDemoFILES_FS**. This name is necessary for the WebLink utility to correctly create the FILESTREAM database objects (tables, indexes, and so on).
7. Select the General page from the left menu.
8. Click **Add** to add the FILESTREAM data file.
9. Enter the following information for the FILESTREAM data file:
   - **File Type** — Select **Filestream Data**.
   - **Logical Name** — For consistency, give this the same name as the Filegroup (for example, **VisionDemoFILES_FS**).

- **Path** — Select the physical path to the FILESTREAM data.

See the “Identify the Physical Disk Location of the FILESTREAM Data” section for more information.

The other fields applicable to the SQL Server Data and Log files are not applicable to FILESTREAM data.

Configure Database Access for FILESTREAM Database

In addition to configuring SQL Server to support FILESTREAM, you must make changes to your Vision configuration to properly support Vision TDM.

IIS Application Pool Configuration

The FILESTREAM database connection is only supported using Windows Integrated Authentication. In Vision, the identity of the DeltekVisionAppPool is the account that will make this connection to the FILESTREAM-enabled TDM database. You will need to ensure that the Application Pool is running as a Windows account (Domain or Local), and that this account has db_owner rights to the FILESTREAM database.

Although the FILESTREAM database connection requires Windows Integrated Authentication, you do not need to make any changes to the database connection for your Vision transaction database to support FILESTREAM. For example, if you are using a SQL Login to access your Vision database, no changes are needed.

If you are using a Local account, the same Windows account name (with the same password) must exist on both the Vision web/application server and the FILESTREAM enabled SQL Database server for Vision to access the FILESTREAM Windows file share.

Configure and Validate the FILESTREAM Database with WebLink

After the FILESTREAM database has been created and the IIS Application Pool identity has been granted db_owner rights to the FILESTREAM database, use the WebLink Utility to create the FILESTREAM database schema.

To configure the database schema and verify the configuration:

1. Launch and log in to the WebLink utility on the Vision web/application server.
2. From the **Current Database** drop-down list, select the Vision database that will be used with Vision TDM.
3. Select the **Enable FILESTREAM** check box. You will receive a message asking if you would like to verify that the FILESTREAM database is configured correctly.

   By default, WebLink prefills:

   - The name of the SQL Server being used for the Vision database
   - The name of the FILESTREAM database (required and grayed out)

   If you are hosting the FILESTREAM database on a different instance of SQL Server, click **No** and modify the name of the FILESTREAM SQL Server. After you make the necessary changes, click **Test Connection** on the WebLink menu to re-test and create the FILESTREAM database schema.

Your FILESTREAM database can exist on a different SQL Server instance. For more information, refer to the “Identify the SQL Server to Host the FILESTREAM Database” section.

4. Click **Yes** to validate the configuration. A message displays indicating that the FILESTREAM database is not configured for use and asking if you would like to configure it.

5. Click **Yes** to create the FILESTREAM database objects (tables, indexes, and so on). When the objects are created, a message displays saying that the configuration is complete.

Files Administration Utility in Vision

The Vision Files Administration utility allows you to search for and view files that were uploaded into Vision. This includes supporting documents that were uploaded for Vision transactions, as well as InDesign templates that were imported or created using the Vision Merge Templates application.

To view the documents that have been uploaded using the Files Administration utility:

1. Open the Vision application.
2. On the Vision Utilities menu, click **Files Administration**.
3. On the Files Administration dialog box, use the **Date Range** fields to select the start and end dates to define the date range to search. Vision defaults to use the past three days.
4. Click **Refresh Files List** to populate the Files grid. Records that match the start and end date criteria display.
5. To further refine your results set, complete one or more of the following actions:
   - Enter specific text that you want to find. Vision searches the **File Name** and **Description** fields to locate the matching text.
   - Open the lookup and select a **User ID**.
   - Use the drop-down list to select a Vision application. This drop-down list displays the applications that allow supporting documents.
6. Click **Refresh Files List** to activate the search. The Files grid lists all of the documents that match your criteria.
7. Click the **File Name** link to open the associated PDF.

If you receive an error message stating that the file is missing, the Vision and TDM databases are not in sync. Refer to the “Data Synchronization Issue” section for additional information.

Data Synchronization Issue

The Databases Out of Sync dialog box displays when the files in the Vision and FILESTREAM databases are not synchronized. This file mismatch can occur when there is a database backup or restore on one database, but not the other. In this situation, the **File Name** link cannot open the selected file. Click **OK** to return to the Files Administration utility. Then contact your system administrator for details.

Troubleshooting FILESTREAM

This section lists potential problems, causes, and solutions for issues with FILESTREAM.

Problem 1

You test the database connection for the first time in WebLink, and WebLink cannot create the FILESTREAM database objects.

Possible Cause	The FILESTREAM filegroup name is not in the required format: **[VisionFILESTREAMDBName]_FS**.
Solution	Reformat the FILESTREAM filegroup name.

Problem 2

You receive a “FILESTREAM data cannot be placed on empty filegroup” error.

Possible Cause	The FILESTREAM filegroup is not configured.
Solution	Configure the FILESTREAM filegroup, and confirm that the FILESTREAM filegroup name is in the required format: **[VisionFILESTREAMDBName]_FS**.

Problem 3

When a user attempts to upload a document, Vision displays a message that it has not been configured to upload supporting documents.

Possible Cause	The **Enable FILESTREAM** option is not selected in WebLink and/or FILESTREAM is not configured properly (WebLink is unable to connect to the FILESTREAM database or the FW_Files table was not created.)
Solution	Confirm that the **Enable FILESTREAM** option is selected in WebLink and that FILESTREAM is configured properly.

Problem 4

When you test the FILESTREAM configuration in WebLink, a message displays saying that the FILESTREAM database cannot be opened and that the login failed for the user account running the IIS Application Pool Identity.

Possible Cause	The FILESTREAM database has not been created or has not been created with the required naming format, or the Identity of the DeltekVisionAppPool in IIS has not been granted db_owner rights to the FILESTREAM database.
Solution	Confirm that the FILESTREAM database exists and has been properly named and that the IIS Application Pool Identity has the required database rights.

Using FILESTREAM with Other SQL Server Features

Refer to the following article for detailed information on using FILESTREAM with other SQL Server options:

http://msdn.microsoft.com/en-us/library/bb895334(v=sql.105).aspx

FILESTREAM and SQL 2012 Availability Groups

http://msdn.microsoft.com/en-us/library/hh510261.aspx

TDE (Transparent Data Encryption)

FILESTREAM can be used with TDE although the FILESTREAM data is not encrypted.

Log Shipping

Log shipping supports FILESTREAM. Both the primary and secondary servers must be running SQL Server 2012 or later, and have FILESTREAM enabled.

Database Mirroring

Database mirroring does not support FILESTREAM. A FILESTREAM filegroup cannot be created on the principal server. Database mirroring cannot be configured for a database that contains FILESTREAM filegroups.

Failover Clustering

For failover clustering, FILESTREAM filegroups must be put on a shared disk. FILESTREAM must be enabled on each node in the cluster that will host the FILESTREAM instance.

SQL Server Express

SQL Server Express supports FILESTREAM. The 4 GB database size limit does not include the FILESTREAM data container.

How to Use FILESTREAM in a Firewall-Protected Environment

To use FILESTREAM in a firewall-protected environment, both the client and server must be able to resolve DNS names to the server that contains the FILESTREAM files. FILESTREAM requires that the Windows file-sharing ports 139 and 445 be open.

The “client” in your Vision TDM deployment is the web/application server, so if your Vision deployment has a firewall between the web/application server and the FILESTREAM database server, then the ports referenced above must be open between the servers.

Queries to Join the Vision Transaction DB and FILES DB

The following query will obtain the file sizes in the FILES database by joining two [Vision] and [Vision]Files databases. This will work if the databases are on the same SQL Server.

```
SELECT DATALENGTH(a.FileData) as FileSize, b.FileName, b.ContentType FROM
[VisionDBName]FILES.dbo.FW_Files a inner join [VisionDBName].dbo.FW_Files b ON
a.FileID=b.FileID
```

The following query will obtain the file sizes in the FILES database by joining two [Vision] & [Vision]Files databases. This will work if the databases are on Linked SQL Servers.

```
SELECT DATALENGTH(a.FileData) as FileSize, b.FileName, b.ContentType FROM
[FILESTREAMDBServer].[VisionDBName]FILES.dbo.FW_Files a inner join
[VisionDBName].dbo.FW_Files b ON a.FileID=b.FileID
```

You must create the link between the servers first. See SQL Server Books Online for information on how to create Linked Servers:

http://msdn.microsoft.com/en-us/library/ms130214(v=sql.105).aspx

Chapter 10: Configure an Alternate Database for Vision Reporting

When you use Vision Reporting, Vision creates two different workloads on your SQL Server Database Engine: Transactional and Reporting. These workloads are necessary for proper functionality, but they also require a tremendous amount of resources from your SQL Server.

Vision provides a means to offload the reporting workload to a different SQL Server, which will help reduce the drain on your SQL Server resources. You can configure a copy of your Vision database for access, and configure the connection string information in WebLink on the Report Server tab in the Alternate Database for Reporting section.

The following reports use the alternate database:

- Dashboard reports
- Reports in the Reporting menu applications (minus Purchasing reports)

These reports use the alternate database whether they are previewed, directly printed, emailed, run via the process server, or processed in another way.

All other reports, including posting logs, billing (interactive and batch) reports, timesheet and expense reports, and so on, will continue to run their queries against the Vision transaction database.

Visualization reports do not use SQL Reporting Services and are not affected by this feature.

Alternate Database for Reporting

The primary benefit of using the Alternate Database for Reporting configuration in WebLink is that it can be used with any version or edition of SQL Server.

With SQL Express, the Alternate Database for Reporting must be located on the same SQL Express instance as the transaction database. This is a limitation of SQL Express because Reporting Services for SQL Express can only use Local databases.

When using the Alternate Database for Reporting configuration in WebLink, consider the following:

- Find an appropriate method to create a copy of the database on the second SQL Server (for example, transactional replication, log shipping, database backup/restore, or third party tools that support SQL Server snapshot backup).

  Deltek has not completed testing and does not provide support for the underlying database copy/synchronization methodology that you choose.

- Ensure that the database copy used for reporting is kept in sync with the transaction database. Not keeping the databases in sync will result in stale data for reporting purposes.
- Ensure that the SQL login used for authentication has read-only access to the database.

The following links are to Microsoft documentation that will help you choose an appropriate database replication/synchronization methodology:

- Transactional Replication:

  http://msdn.microsoft.com/en-us/library/ms151198.aspx
- Log Shipping:

  http://msdn.microsoft.com/en-us/library/ms187103.aspx
- Backup/Restore:

  http://msdn.microsoft.com/en-us/library/ms187048.aspx
- Third Party Tools that support Snapshot Backups:

  http://msdn.microsoft.com/en-us/library/ms189548(v=sql.105).aspx

SQL Server Database Mirroring is not supported for the Alternate Database for Reporting functionality as the mirrored database is not accessible for read-only queries. Also, Database Mirroring does not support SQL Server FILESTREAM, which is required for Vision Transaction Document Management (TDM).

Configure the Alternate Database for Reporting in WebLink

To configure an alternate database for reporting:

1. Identify and implement a methodology to create a copy of your Vision transaction database on a second SQL Server. For testing purposes, you can perform a backup/restore.
2. Identify and implement a methodology to ensure that the data is synchronized between the databases within a timeframe differential suitable to your business needs.
3. Create a login that has read-only rights to this database copy. This can be accomplished by granting db_datareader rights, rather than db_owner rights, for the SQL Server login that is used for the alternate database for reporting.
4. Launch the Vision WebLink Utility and select the Vision transaction database entry that you will configure for an alternate reporting database.
5. Click the Report Server tab.
6. Select the **Use Alternate Database for Reporting** option. When you click **OK**, Vision displays a message reminding you to make sure that the database specified is kept up to date with the Vision database and that the login has only read rights to the database.
7. Click **OK** to continue. Complete the following fields to enter the connection string information for the alternate database:

Field	Description
Server Name	Enter the name of the SQL Server hosting the alternate database.
Database name	Enter the name of the alternate database.

Field	Description
Windows Authentication	Select this check box if you are using Windows Authentication for the database connection. The identity of the DeltekVisionAppPool will need read-only rights to the alternate database.
Database Username/Password	If you are not using Windows Authentication, enter the SQL server login with read-only rights to the alternate database.

8. From the WebLink menu, click **Test » Alternate Database for Reporting** to validate the connection.

Troubleshooting

Identify the Connection String Used in a Report

After configuring the Alternate Database for Reporting or Availability Group options, you must validate that reports are running against the correct database.

Preview the Report

To acquire the connection string by previewing the report:

1. Click the construction hat icon  on the Reporting toolbar.

   If you don't see the icon, maximize the report.

2. From the **View Report Information** drop-down list, select **Report Data Source**.
3. Click the **View** button.

   You will be prompted to **Open** or **Save** the XML file.
4. Click **Open**, and the file will open using the application configured to open XML files. On most computers, that will be your default browser.
5. Review the ConnectString element for the following attributes:
   - **Data Source** — This is either the database server specified in the Alternate Database for Reporting configuration or, if you use Availability Groups, the Availability Group listener.
   - **Initial Catalog** — This is either the database name specified for the Alternate Database for Reporting configuration or, if you use Availability Groups, the Vision database name.
   - **ApplicationIntent=ReadOnly** — This only displays when you use Availability Groups and if the report was run against the Read Only reporting database.
   - **MultiSubnetFailover** – This only displays when you use Availability Groups.

Chapter 11: Configure Microsoft SQL Server Availability Groups

Vision supports the use of Microsoft SQL Server AlwaysOn Availability Groups. A SQL Server Availability Group provides an all-inclusive High Availability and Disaster Recovery solution for SQL Server databases. The specific features of Availability Groups directly supported by Vision are:

- **Readable Secondary Replicas/Read Only Routing** — This feature configures a readable replica of your transaction database that you can use to offload report queries from the primary transaction database.

  Only the following reports use Read Only Routing:

  - Dashboard reports
  - Reports that you run from the Reporting menu (except Purchasing reports)

  Reports that use the replica database will do so regardless of how they are processed: when they are previewed, directly printed, emailed, run via the process server, or generated in another way.
- **Multi-subnet Failover support** — This feature provides failover support when the Primary and Secondary replicas of the SQL Server Availability Groups are on different network subnets.

This chapter focuses specifically on the SQL Server Availability Group features supported by Vision. For information about other SQL Server Availability Group features, refer to the Microsoft documentation:

http://msdn.microsoft.com/en-us/library/ff877884.aspx

These AlwaysOn Architecture Guides provide valuable information about solution design:

http://blogs.msdn.com/b/sqlalwayson/archive/2012/07/03/alwayson-architecture-guides.aspx

Prerequisites

Refer to the MSDN documentation for prerequisites, restrictions, and recommendations for AlwaysOn Availability Groups:

http://msdn.microsoft.com/en-us/library/edbab896-42bb-4d17-8d75-e92ca11f7abb

Specific prerequisites for Vision are:

- Enterprise Edition of SQL Server 2012 or higher
- At least two SQL Server server nodes
- Standard Edition of Windows Server 2012 or later; the specific operating system feature that is required to support Availability Groups is Windows Server Failover Clustering.

  If you are using Transaction Document Management (which uses SQL Server FILESTREAM) and Windows Server 2012 Failover Clustering, you will need to obtain and install the following Microsoft hotfix:

  http://support.microsoft.com/kb/2835620
- A Windows file share on a server other than the SQL Servers to which the SQL service accounts have write access. This is required to configure Availability Groups.

Installation Overview

These are the primary installation steps, described in more detail in the sections that follow.

1. Use Server Manager to install the following features on **all** nodes:
   - Failover Clustering
   - Failover Clustering Tools (part of Remote Administration Tools)
   - .NET Framework 4.5.2
2. Create a file share on a different server that the SQL Server service account(s) will have full control rights to.

If you are using Analysis Cubes with Availability Groups, see the "Configure Analysis Cubes for Availability Groups" section for more information.

Create the Windows Server Failover Cluster (WSFC)

The WSFC will cluster applications and services. Your specific configuration depends on your intended use of SQL Server Availability Groups. For example, you may want to use a SQL Server Failover Cluster Instance (FCI) in addition to using Availability Groups. One of the primary differences in the cluster configuration of an FCI versus an Availability Group is the need to provide shared storage.

Refer to the Microsoft documentation for in-depth details about configuring Windows Server Failover Clustering for your intended use.

When you configure Windows Server Failover Cluster:

- A cluster virtual network is created. You will need an IP address and DNS name for the cluster and server names for the nodes that will be members of the cluster.
- No shared storage is needed, so you can clear the check box to add available storage.
- A static IP or DHCP can be used for the cluster network. Deltek recommends a static IP for production environments. The example below uses DHCP.
- A DNS name is created automatically, or you can create a DNS entry before configuring the cluster.
- You will need domain rights as the process creates a computer account in the domain for the cluster virtual network.
- Two virtual networks are created, one for the Windows Server Failover Cluster and one for the Availability Group listener. Use a unique name for the cluster which is not SQL-specific but will be easily identifiable as the Windows cluster.
- When Vision connects to the SQL Server, it will not be connecting to the WSFC name. Vision will connect to the Availability Group Listener (created later). The WSFC is enabled for the fail-over functionality of Availability Groups.

Multi-subnet Clustering

If your WSFC (and SQL Server Availability Group Listener) are configured so that your WSFC nodes are on different network subnets, Multi-subnet Failover support is automatically turned on.

The **MultisubnetFailover=True** option is automatically added to the connection string.

Refer to the following MSDN documentation for more information:

- http://msdn.microsoft.com/en-us/library/hh213417(v=sql.110).aspx#SupportAgMultiSubnetFailover
- http://msdn.microsoft.com/en-us/library/hh213080.aspx

Install and Configure WSFC

Use the following procedures to install and configure Windows Server Failover Clustering.

Depending on your operating system, the steps in these procedures may vary slightly. The procedures in this section are based on Windows Server 2012 Standard Edition.

Install the Failover Clustering Feature

To install the failover clustering feature:

1. Open the Server Manager utility.
2. Access the **Local Server**, and scroll down to **Roles and Features**.
3. From the **Tasks** drop-down list, select **Add Roles and Features**.
4. In the Add Roles and Features wizard, click **Next** until you get to the Select Features page.
5. Select the **Failover Clustering** option.
6. If prompted, click **OK** to install any dependent features.
7. Complete the wizard to perform the installation.
8. If prompted, reboot the server.

Configure Failover Clustering using Failover Cluster Manager

To configure the failover clustering feature:

1. From Administrative Tools, open the Failover Cluster Manager.
2. Under **Management**, click **Create Cluster**. The Create Cluster Wizard will guide you through the process.
3. On the Select Servers page of the wizard, browse to or enter the names of the servers that will be part of the cluster.
4. On the Validation Warning page, select **Yes** to run the Cluster Validation tests, and click **Next**.

   This process creates a report that identifies any problems that need to be addressed before creating the cluster.

5. When the validation is completed, provide a name for the cluster in the **Cluster Name** field.
6. Click **Next** to create the cluster.

   If you are configuring a WSFC that does not require shared storage, you can clear the **Add all eligible storage to the cluster** option.

Install SQL Server on Each Node

After configuring the WSFC, you must perform the SQL Server installation on each node in the cluster.

Installation Requirements and Notes

- SQL Server 2012 or later Enterprise Edition is required for Availability Groups.
- If you are only configuring Availability Groups (not a Failover Cluster Instance), you must perform a New SQL Server stand-alone installation, not a New SQL Server failover cluster installation, on each node. If you choose a SQL Server Cluster installation, it may fail the pre-requisites of shared storage as this is not a requirement of Availability Groups. However, it may be a requirement for your specific configuration.
- Only the SQL Database Engine can use Availability Groups. Even though the Availability Groups use a WSFC, this is not a true cluster and will not provide fault tolerance for other SQL Server services (Analysis Services or Reporting Services) if installed.
- To be fault tolerant, Analysis Services must be part of an actual Failover Cluster Instance (FCI) where SQL is installed using a New SQL Server failover cluster installation.
- An FCI may be used together with an Availability Group to enhance the availability of an availability replica. However, to prevent potential race conditions in the WSFC cluster, automatic failover of the Availability Group is not supported to or from an availability replica that is hosted on an FCI.
- Reporting Services uses a Scale-out Deployment, which is not a cluster.
- You can use the same or different service accounts on each node, but all accounts must have rights to the file share as outlined in the Prerequisites section.
- FILESTREAM functionality can be used with Availability Groups and will require that FILESTREAM be enabled on all fail-over nodes.

  For more information, see http://msdn.microsoft.com/en-us/library/hh510261.aspx.

  You may also need the following Microsoft hotfix:
  http://support.microsoft.com/kb/2835620

- Although this is not a specific requirement, you should consider performing the exact same SQL Server installation on each node, including the same installation and data paths and instance name.
- For proper failover support, the failover nodes should have comparable hardware resources.
- Refer to the MSDN documentation for prerequisites, restrictions, and recommendations for AlwaysOn Availability Groups:

  https://msdn.microsoft.com/en-us/library/edbab896-42bb-4d17-8d75-e92ca11f7abb

After installing SQL Server on all nodes in the cluster, restore your Vision transaction database and configure the SQL Reporting Services databases (ReportServer and ReportServerTempDB) on the Primary node in the cluster.

Configure Database Login

For the Vision and Reporting Services databases that are part of an Availability Group to be immediately available in the event of an Availability Group failover, you must complete these steps on **all** failover nodes. The following rights need to be granted to the login used to access the databases (Vision and Reporting Services databases) that are part of the Availability Group:

Permission	Required for
Dbo = Database Owner rights to all databases in the Availability Group	All databases in the Availability Group
View Any Definition	Availability Groups
View Server State	Availability Groups

When you back up and restore a database to another server, the Login on Server A has a different SID (Security Identifier) than the same login on Server B. This issue is typically resolved using the sp_change_users_login stored procedure. However, since the database on the Secondary Replica will be in read-only mode, you cannot fix the login.

You can resolve this issue by using the sp_help_revlogin stored procedure, which can be found in the following Microsoft Support article:

http://support.microsoft.com/kb/918992

Once the procedure is created in the master database, execute it to get the list of SQL Logins with their associated SIDs and the CREATE statement to create the login on the Secondary Replicas. Here is an example.

```
CREATE LOGIN [DeltekVision] WITH PASSWORD =
0x0200E0E05D60876CCE39BD9209515FB63C5589D6C939F3AB56A6CE9DBFBF49A9410F66F098408
27135F800725E25A77714FDFA31FB6C18BCB46561217947C3749F0380A18AF5 HASHED, SID =
0x0124F12258D9BD49BE649C2D7A6DA838, DEFAULT_DATABASE = [master], CHECK_POLICY =
OFF, CHECK_EXPIRATION = OFF
```

Prior to configuring the Availability Groups, run the CREATE statement(s) created from executing the sp_help_revlogin on your server on all Secondary Replicas.

Create Availability Groups

You must create the Availability Group on each individual node.

To create an Availability Group:

1. Open the SQL Server Configuration Manager.
2. Right-click the SQL Server service, and click **Properties**.
3. On the AlwaysOn High Availability tab, select the check box to enable the Availability Groups for each node.

4. Select the databases to be included in the availability group (all of these databases will failover together if there is a failover). At a minimum, include the Vision and Reporting Services databases (ReportServer and ReportServerTempDB).

   The database must be in FULL recovery mode and a FULL database backup must have been taken of the database prior to starting the Availability Group wizard.

   For more information on performing backups and restores of databases in FULL recovery model, refer to the following MSDN documentation:

   https://msdn.microsoft.com/en-us/library/ms187048(v=sql.110).aspx

5. In SMS, start the Availability Group Wizard.
6. Expand **AlwaysOn High Availability**, right-click **Availability Groups**, and click **New Availability Group Wizard** on the shortcut menu.
7. Select a name for the Availability Group.

   This is not the virtual name of the Availability Group listener, but it can be the same. The name should be descriptive of the databases the Availability Group includes (for example, VisionAG) because multiple Availability Groups can exist on the same servers.
8. Select the databases that you want to include in the Availability Group. The wizard will tell you whether or not they meet the requirements (for example, FULL recovery and FULL backup taken).
9. Specify the Availability Group configuration (nodes, failover mode, and synchronization mode).

   Refer to the following settings to complete the fields on this form:

Field	Setting
Server Instance	Four allowed
Initial Role	Primary or Secondary
Automatic Failover	Can only configure two nodes
Synchronous Commit — Synchronous versus asynchronous commit identifies the relative amount of data loss in the event of a failure versus the performance of the synchronization.	Can configure a maximum of three.
Readable Secondary — Sets the access rights for the Server Instance (for read-only reporting and so on).	▪ **No** — Connections are not allowed to Secondary Replicas. ▪ **Yes** — Connections are allowed in read-only mode. ▪ **Read-intent only** — Connections using the **ApplicationIntent=ReadOnly** keyword are used for Read Only Routing.

Read-intent only is used for the Read Only Routing feature of Availability Groups. This is described later in this document, and it defines the specific functionality supported by Vision.

10. Click the Endpoints tab. The Endpoint for each instance is created automatically.

    Backup Preferences determine whether or not backups can be taken from the replicas other than the Primary (another feature of Availability Groups).

11. Click the Listener tab. You use this tab to create the Availability Group Listener.
12. From the **Network Mode** drop-down list, select the type of listener: **Static IP** or **DHCP**. The port will be the same as the one used by the SQL Server port (default is 1433).

If you are using multiple subnets, use the **Add** button to add the IP address for the listener on the second subnet. This option is available only if you are using Static IP addresses. Refer to the following MSDN documentation for more information:

http://msdn.microsoft.com/en-us/library/hh510226(v=sql.110).aspx

13. Click **Next**.
14. On the Data Synchronization Preference screen, enter the shared path or use the **Browse** button to select the location that all nodes/SQL service accounts can access. This determines how the databases are going to synchronize to the replicas.

Read Only Routing Configuration

Vision architecture changes have been made to support the Read Only Routing feature. This feature allows certain report queries to be run against the read-only copy of the Vision transaction database on a Secondary Replica of the Availability Group. The benefit of this feature is that it allows much of the Vision reporting workload to be offloaded from the database on the Primary replica to the Secondary Replica, which frees resources for the transaction workload.

- Read Only Routing requires that the connection string uses the **ApplicationIntent=ReadOnly** keyword. The Vision Reporting architecture has been specifically modified to allow for this change in the connection string when the database configuration specified in WebLink is configured to use Availability Groups.
- With this keyword and the proper configuration (queries below), the Availability Group will automatically route connections with this keyword to the read-only secondaries configured for **Read-intent only**.
- Even though you have configured the Availability Group for Read-intent only secondaries, manual queries are required to configure the Read Only Routing aspect of the configuration.

For more information about Read Only Routing, read the following MSDN article:

http://msdn.microsoft.com/en-us/library/hh710054(v=sql.110).aspx

Read Only Routing Queries

Two queries are required to modify the Availability Group configuration to support Read Only Routing:

- Configure the Read Only Routing URL
- Configure Read Only Routing Lists

Configure the Read Only Routing URL

Read Only Routing URLs are different from Availability Group Endpoints, which were automatically configured earlier. In the Availability Group configuration above, there are two nodes, each configured to allow Read-intent only connections when the node is in secondary mode. (A node in Secondary mode is promoted to Primary when a failover occurs.)

- The following query identifies existing Read only routing URLs:

```
select read_only_routing_url from sys.availability_replicas

Query Result (if present; otherwise the query will return NULL):
read_only_routing_url
tcp://CAMDEVSQL12AG1:1433
tcp://CAMDEVSQL12AG2:1433
```

- The following blog post identifies a script that can be run against each replica to calculate the read-only routing URL:

  http://blogs.msdn.com/b/mattn/archive/2012/04/25/calculating-read-only-routing-url-for-alwayson.aspx.

  Example output from the script is below:

```
Read-only-routing url script v.2012.1.24.1
This SQL Server instance version is [11.0.2100.60]
This SQL Server instance is a standard (not clustered) SQL Server
instance.
This SQL Server instance is enabled for AlwaysOn.
This SQL Server instance is NOT a Sql Azure instance.
This SQL Server instance DAC (dedicated admin) port is 1434
This SQL Server instance is listening to all IP addresses (default mode).
This SQL Server instance is listening on fixed tcp port(s) (it is not
configured for dynamic ports), this is a recommended configuration when
using read-only routing.
This SQL Server instance resides in domain 'dev.ads.deltek.com'
This SQL Server instance FQDN (Fully Qualified Domain Name) is
'CAMDEVSQL12AG1.dev.ads.deltek.com'
This SQL Server instance port is 1433
************************************************************************

The read_only_routing_url for this SQL Server instance is
'tcp://CAMDEVSQL12AG1.dev.ads.deltek.com:1433'
************************************************************************
```

- The following statements configure the Read only routing URL for each node:

```
ALTER AVAILABILITY GROUP [SQL12AG1]
MODIFY REPLICA ON N'CAMDEVSQL12AG1' WITH
(SECONDARY_ROLE(READ_ONLY_ROUTING_URL=N'tcp://CAMDEVSQL12AG1.dev.ads.deltek.com:1433'))
ALTER AVAILABILITY GROUP [SQL12AG1]
MODIFY REPLICA ON N'CAMDEVSQL12AG2' WITH
(SECONDARY_ROLE(READ_ONLY_ROUTING_URL=N'tcp://CAMDEVSQL12AG2.dev.ads.deltek.com:1433'))
```

  Where

  - `[SQL12AG1]` is the name of the Availability Group (not the listener).
  - `CAMDEVSQL12AG1` is node 1 and `CAMDEVSQL12AG2` is node 2.
  - `tcp://CAMDEVSQL12AG1.dev.ads.deltek.com:1433` is the Read only routing URL for node 1 and `tcp://CAMDEVSQL12AG1.dev.ads.deltek.com:1433` is the Read only routing URL for node 2.

Configure Read Only Routing Lists

Read Only Routing Lists provide a priority order for the routing of Read-intent only connections among nodes in the Availability Group configured for Read Only Routing.

- The following query identifies existing Read Only Routing Lists and shows that when CAMDEVSQL12AG1 is the Primary, the priority order will be the replica (CAMDEVSQL12AG2) and then itself if the replica is not available (and vice versa if CAMDEVSQL12AG2 is the Primary)

```
select g.name, r1.replica_server_name, l.routing_priority,
r2.replica_server_name, r2.read_only_routing_url
from sys.availability_read_only_routing_lists as l
join sys.availability_replicas as r1 on l.replica_id = r1.replica_id
join sys.availability_replicas as r2 on l.read_only_replica_id =
r2.replica_id
join sys.availability_groups as g on r1.group_id = g.group_id
```

Query Result:

name	replica_server_name	routing_priority	replica_server_name	read_only_routing_url
SQL12AG1	CAMDEVSQL12AG1	2	CAMDEVSQL12AG1	tcp://CAMDEVSQL12AG1:1433
SQL12AG1	CAMDEVSQL12AG2	1	CAMDEVSQL12AG1	tcp://CAMDEVSQL12AG1:1433
SQL12AG1	CAMDEVSQL12AG1	1	CAMDEVSQL12AG2	tcp://CAMDEVSQL12AG2:1433
SQL12AG1	CAMDEVSQL12AG2	2	CAMDEVSQL12AG2	tcp://CAMDEVSQL12AG2:1433

- The following statements configure the Read Only Routing Lists for this configuration:

```
ALTER AVAILABILITY GROUP [SQL12AG1]
MODIFY REPLICA ON N'CAMDEVSQL12AG1' WITH (PRIMARY_ROLE
(READ_ONLY_ROUTING_LIST = (N'CAMDEVSQL12AG2',N'CAMDEVSQL12AG1')))

ALTER AVAILABILITY GROUP [SQL12AG1]
MODIFY REPLICA ON N'CAMDEVSQL12AG2' WITH (PRIMARY_ROLE
(READ_ONLY_ROUTING_LIST = (N'CAMDEVSQL12AG1',N'CAMDEVSQL12AG2')))
```

Where:

- `[SQL12AG1]` is the name of the Availability Group (not the listener).
- `CAMDEVSQL12AG1` is node 1 and `CAMDEVSQL12AG2` is node 2.

Monitoring Availability Groups

The following tools are available for monitoring the status of an Availability Group:

- Availability Group Dashboard
- System and Dynamic Management Views (DMVs)
- System Monitor (PerfMon)
- Windows PowerShell

Availability Group Dashboard

To display the Availability Group Dashboard, complete the following the steps:

1. Launch SQL Server Management Studio, and connect to the Primary Replica.
2. Right-click the **Availability Group** folder, and click **Show Dashboard** on the shortcut menu.

System and Dynamic Management Views (DMVs)

The following MSDN article provides a variety of System Views and DMVs that can be used to monitor the health and status of the WSFC and Availability Groups:

http://msdn.microsoft.com/en-us/library/ff878305.aspx

System Monitor (PerfMon)

A variety of System Monitor counters can be used to monitor the performance of Availability Groups. Refer to the following for more information on the available counters and how to use them:

http://technet.microsoft.com/en-us/library/ff877954.aspx

Windows PowerShell

The following links are a four-part MSDN series on using PowerShell to monitor Availability Groups:

- http://blogs.msdn.com/b/sqlalwayson/archive/2012/02/13/monitoring-alwayson-health-with-powershell-part-1.aspx
- http://blogs.msdn.com/b/sqlalwayson/archive/2012/02/13/monitoring-alwayson-health-with-powershell-part-2.aspx
- http://blogs.msdn.com/b/sqlalwayson/archive/2012/02/13/monitoring-alwayson-health-with-powershell-part-3.aspx
- http://blogs.msdn.com/b/sqlalwayson/archive/2012/02/15/the-always-on-health-model-part-4.aspx

Flexible Failover Policy

The Failover Policy controls the Failover feature of Availability Groups. For more Information on this feature, read the following:

http://msdn.microsoft.com/en-us/library/hh710061(v=sql.110).aspx

Failover Condition Level and Health Check Timeout

Transact-SQL Value	Level	Automatic Failover Initiated On…
1	One	Server down. The SQL Server service stops because of a failover or restart.
2	Two	Server unresponsive. Any condition of lower value is satisfied, the SQL Server service is connected to the cluster and the health check timeout threshold is exceeded, or the current primary replica is in a failed state. This is the default level.
3	Three	Critical server error. Any condition of lower value is satisfied or an internal critical server error occurs.
4	Four	Moderate server error. Any condition of lower value is satisfied or a moderate server error occurs.
5	Five	Any qualified failure conditions. Any condition of lower value is satisfied or a qualifying failure condition occurs.

Failover condition is determined by WSFC executing sp_server_diagnositcs at regular intervals.

The following query identifies the existing Failover Policy:

```
select name,failure_condition_level,health_check_timeout from
sys.availability_groups
```

Query Result:

name	failure_condition_level	health_check_timeout
SQL12AG1	3	30000

The following statements configure the Failover Policy for this configuration:

```
ALTER AVAILABILITY GROUP AG1 SET (FAILURE_CONDITION_LEVEL = 1); //default is 3
ALTER AVAILABILITY GROUP AG1 SET (HEALTH_CHECK_TIMEOUT = 60000); //default is
30000
```

Configure Vision and Reporting Services to Use Availability Group Listener

For the final steps to configure Vision and Reporting Services to correctly use the Read Only Routing feature in Availability Group configuration, you must configure the following to use the Availability Listener:

- Report Server Configuration Tool to configure Reporting Services

- WebLink utility for your Vision database, Report Server database, and FILESTREAM database

Configure Vision for Availability Groups

To configure Vision to use the Availability Group configured in the preceding sections:

1. Launch the WebLink utility, and select or add the Vision transaction database that is part of the Availability Group.
2. Enter the SQL Server login ID that has been granted the necessary SQL Server rights to the Availability Group database. Refer to the "Configure Database Login" section of this chapter for further information.

   If you are adding a new WebLink entry, when you click out of the password field for the SQL Login ID, WebLink:

   - Issues a query to identify the Availability Group that the Vision database is part of.
   - Displays the **Use Availability Groups** check box.
   - Populates the name of the **Availability Group** (this is not the listener name).

   If you see the **Use Availability Groups** check box, but it is not enabled, check to see if the **Use Alternate Database for Reporting** check box is selected on the Report Server tab. You cannot use both the Availability Groups and Alternate Database for Reporting features because they provide the same functionality but are designed for different versions of SQL Server.

   Although you are modifying the server connection information for the Vision database, the Vision FILESTREAM database (if applicable), and the Reporting Services database to use the Availability Group listener, this action only ensures that these databases can still connect to the new Primary node in the event of a failover.

   The only features that use the Vision database on the Secondary (Read Only) Replica are the database queries for the specific reports described earlier in this document. This includes reports on the Reporting Applications menu (minus Purchasing reports) and Dashboard reports.

3. To use Availability Groups, complete the following actions:
   a. Select the **Use Availability Groups** check box.
   b. Change the **SQL Server** name to be the Availability Group listener name.
   c. If you are using FILESTREAM and the FILESTREAM database is part of the Availability Group (which it should be if the database is on the same SQL Server as Vision), confirm that the **FILESTREAM SQL Server** is using the Availability Group Listener name.
   d. On the Report Server tab, confirm that the Server Name specified in the Report Server Database Access is using the Availability Group listener name.

Configure Reporting Services to Use the Availability Group Listener

If Reporting Services have not yet been configured, follow the steps in the *Deltek Vision Technical Installation Guide* to configure Reporting Services. When you enter the Database Server name to use for the Report Server databases, use the Availability Group Listener.

If Reporting Services is already configured to use the Primary Node server name:

1. Open the Reporting Services Configuration Manager.
2. Select the Database menu.
3. Click the **Change Database** button.
4. Select the **Choose an existing report server database** option, and click **Next**.
5. On the **Connect to the Database Server** screen, change the **Server Name** to be the Availability Group Listener, and click **Next**.
6. Select the existing ReportServer database from the drop-down list, and click **Next** to complete the re-configuration.

Configure Analysis Cubes for Availability Groups

If you are deploying Vision Analysis Cubes in a configuration that includes Availability Groups, you must first configure the Analysis Cubes to use the Primary Node server name (**not** the Availability Group listener) in your Availability Groups configuration. This requires that you install both Analysis Services and Integration Services on the Primary Node and any Failover Nodes.

For more information, see the *Deltek Vision 7.6 Installation and Configuration Guide for Performance Management (Analysis Cubes and Performance Dashboards)*.

Manually Re-Configure Analysis Cubes upon Availability Group Failover

Vision Analysis Cubes are not Availability Group aware, which means that in the event of an Availability Group failover, the data update that occurs via the SQL Agent refresh job will also fail. This happens because the Vision transaction database is now on a Secondary (read-only) Node or is unavailable. If the reason for the failover is database-specific and the server is still operational, you can follow the steps below to reconfigure the Analysis Cubes to connect to the transaction database on the new Primary Node of the Availability Group. If the server is not operational after the failover of the Availability Group, you will need to build the cubes from scratch on the new Primary Node.

Analysis Cubes remain available when an Availability Group failover occurs.

To manually reconfigure Analysis Cubes when an Availability Group failover occurs:

1. The Secondary Node hosts the Vision Data Warehouse (DW) and Analysis Cubes databases. On the Secondary Node, create a linked server that points to the new Primary Node of the Availability Group:
   a. Open SQL Server Management Studio. Expand **Server Objects » Linked Servers » New Linked Server**.

b. Enter the server name (the new Primary Node, not the Availability Group listener) and choose **SQL Server** as the **Server Type**.

c. On the Security page, change the radio button for **Be made using the login's current security context**. Click **OK**.

2. Modify the LoadCFGTables and LoadUDFData stored procedures in the DW database to use the fully qualified path to the new Primary server:

   a. Expand (+) the <VisionDB>DW database.

   b. Expand (+) Programmability » Stored Procedures.

   c. Locate the LoadCFGTables stored procedure.

      i. Right-click **LoadCFGTables** and choose **Modify**.

      ii. Locate each instance of the <[VisionDBName]>.dbo.<table> with <LinkedServerName>.<[VisionDBName]>.dbo.<table>. There will be two entries in the script.

      For example, [VisionDemo].dbo.FW_CFGSystem would change to CAMDEVSQL12AG2.[VisionDemo].dbo.FW_CFGSystem, where VisionDemo is the name of the Vision transaction database and CAMDEVSQL12AG2 is the Linked Server configured in Step 1 above (the now Primary Node server name of the Availability Group after the failover).

      iii. Execute the modified script to update the stored procedure.

   d. Repeat for the LoadUDFData stored procedure. Nine entries must be modified.

3. Modify the VisionETL_Config.dtsconfig to point to the new Primary server:

   a. Open Windows Explorer and browse to <drive>:\Program Files \Deltek\Vision\Analysis\ETL_2K8\Jobs\<VisionDBName>_en-US

   b. Edit the VisionETL_Config.dtsconfig file with Notepad.

   c. Modify the Data Source of the Vision database to be the new Primary Node of the Availability Group (the server name, not the Availability Group listener).

```
<?xml version="1.0"?>
  <DTSConfiguration>
   <DTSConfigurationHeading>
    <DTSConfigurationFileInfo GeneratedBy="ADSDELTEKCOM\sonnyrai"
GeneratedDate="7/27/2007 4:29:36 PM" />
   </DTSConfigurationHeading>
   <Configuration ConfiguredType="Property"
Path="\Package.Connections[Vision].Properties[ConnectionString]"
ValueType="String">
    <ConfiguredValue>Data Source=CAMDEVSQL12AG2;Initial
Catalog=VisionDemo;Provider=SQLNCLI11;Integrated Security=SSPI;Auto
Translate=False;</ConfiguredValue>
   </Configuration>
   <Configuration ConfiguredType="Property"
Path="\Package.Connections[VisionDW].Properties[ConnectionString]"
ValueType="String">
```

<ConfiguredValue>Data Source=CAMDEVSQL12AG1;Initial Catalog=VisionDemoDW;Provider=SQLNCLI11;Integrated Security=SSPI;Auto Translate=False;</ConfiguredValue>

</Configuration>

<Configuration ConfiguredType="Property" Path="\Package.Connections[VisionCubes].Properties[ConnectionString]" ValueType="String">

<ConfiguredValue>Data Source=CAMDEVSQL12AG1;Initial Catalog=Deltek Vision Analysis - VisionDemo;Provider=MSOLAP.5;Integrated Security=SSPI;Impersonation Level=Impersonate;</ConfiguredValue>

</Configuration>

</DTSConfiguration>

4. Run the SQL Agent DW/Cube Refresh job to ensure that the job completes successfully.

Troubleshooting

Issue

If the **Use Availability Groups** check box does not display for your Vision database when selected in WebLink, execute the following query to see if it returns anything:

```
SELECT e.name, s.database_name
FROM sys.availability_groups_cluster AS e
 INNER JOIN sys.availability_databases_cluster AS s
 ON e.group_id = s.group_id
```

The query should return a result set showing the name of the Availability Group and each database included in the Availability Group.

Issue

When you use Availability Groups, a system health check query is run to determine the health of the Availability Group. If the result of this query returns 0 or 1, the system will fall back to running all reports against the Primary Replica and the Read Only Routing will effectively be disabled. The health check query is shown below:

select synchronization_health from sys.dm_hadr_availability_group_states

Synchronization Health Values

Value	Description
0	Not healthy. None of the availability replicas have a healthy **synchronization_health** (2 = HEALTHY).
1	Partially healthy. The synchronization health of some, but not all, availability replicas is healthy.
2	Healthy. The synchronization health of every availability replica is healthy.

The following error displays if the SQL login used for the Vision database does not have View Definition or View Server State permissions.

FrameworkException:

The user does not have permission to perform this action.

Call Stack:

{\b Query: }

select synchronization_health from sys.dm_hadr_availability_group_states

See the “Configure Database Login” section.

Solution:

Grant View Definition and View Server State permissions to the SQL Login.

Identify the Connection String Used by the Application or Process Server

To validate that the **MultiSubnetFailover=True** keyword is being added to the connection string for your Availability Group configuration, add the following setting to the web.config file under the <ApplicationSettings> tag:

<add key="LogConnectString" value="Y"/>

When this option is set to **Y**:

- The application connection string (created at login) is logged in the ConnectionString.txt file named in the application Logs directory.
- The Process Server connection string is logged in the ProcessServerConnectString.txt file in the same location.

Review these logs to ensure that the **MultiSubnetFailover=True** keyword is added to the connection string. After you have validated that the correct connection string is being issued, change the value of the LogConnectionString setting to **N**.

Identify the Connection String Used in a Report

After configuring the **Alternate Database for Reporting** or **Availability Group** options, you must validate that reports are running against the correct database. Preview the report to check the connection string.

To review the connection string by previewing the report:

1. Display any report.
2. Click the construction hat icon  on the Reporting toolbar.

   If you don’t see the icon, maximize the report.

3. From the **View Report Information** drop-down list, select **Report Data Source**.
4. Click the **View** button.

   You will be prompted to **Open** or **Save** the XML file.

5. Click **Open** to open the file using the application configured to open XML files (usually the default browser).
6. Review the ConnectString element for the following attributes:
   - **Data Source** — This is either the database server specified in the Alternate Database for Reporting configuration or, if you use Availability Groups, the Availability Group listener.
   - **Initial Catalog** — This is either the database name specified for the Alternate Database for Reporting configuration or, if you use Availability Groups, the Vision database name.
   - **ApplicationIntent=ReadOnly** — This only displays when you use Availability Groups and if the report was run against the Read Only reporting database.
   - **MultiSubnetFailover** – This only displays when you use Availability Groups.

Chapter 12: Configure a Shared Location for Databases.enc

If your Vision deployment includes multiple web/application servers, or even just a dedicated process server, the following steps will eliminate the need to synchronize changes made to databases.enc across your servers.

To configure a shared path for databases.enc:

1. Ensure that the databases.enc file is synchronized across all servers.
2. Identify a server that can host the file share. This can be any server as long as it is located in the same data center as your Vision deployment.
3. Create a Windows file share on that server (for example, \\server\share).
4. Grant the service account(s) running the IIS Application Pool Identity and the Process Server service a minimum of modify rights to the share you created.
5. Modify the Vision web.config file (..\Vision\Web\web.config) on all web/application and process servers:
    a. Under <appSettings>, locate the DatabasesEncDirectory entry and uncomment it out. (It will be commented out by default.)
    b. For the value of this setting, enter the share path, not including the databases.enc file name. It should look similar to this, where \\server\share\ is the actual UNC path to your file share:

    <add key="DatabasesEncDirectory" value="\\server\share\" />
6. Copy the databases.enc file to the share.
7. Rename the databases.enc file on all web/application and process servers to **databases.old**.
8. Restart IIS and the Process Server service on all applicable servers and run tests to ensure that Vision and WebLink can be accessed on all web/application servers and that the Process Server service is processing jobs correctly.
9. Make sure to check the Application Event Logs on all servers for any errors or warnings.

Alternative to a Shared Databases.enc File (Old Method)

As an alternative to having a shared path for your databases.enc file, complete the following steps to synchronize changes made to databases.enc across your servers:

1. Launch the WebLink application on one web/application server and configure your connection and application settings. The settings are saved to a local encrypted databases.enc file on the server.
2. Copy the databases.enc file from your web/application server to all other web/application and process servers.
3. Restart the Deltek Vision Process Server Windows service and IIS on each machine.
4. Repeat this process whenever any update is made in WebLink.

Deltek is the leading global provider of enterprise software and information solutions for government contractors, professional services firms and other project- and people-based businesses. For decades, we have delivered actionable insight that empowers our customers to unlock their business potential. 20,000 organizations and millions of users in over 80 countries around the world rely on Deltek to research and identify opportunities, win new business, recruit and develop talent, optimize resources, streamline operations and deliver more profitable projects. Deltek – Know more. Do more.®

deltek.com