Location Intelligence
Infrastructure Asset Management

Confirm®

Installing Confirm Connector System

v26.30a.AM

Information in this document is subject to change without notice and does not represent a commitment on the part of the vendor or its representatives. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, without the written permission of Confirm.

© 2026 Confirm. All rights reserved.

Products named herein may be trademarks of their respective manufacturers and are hereby recognized. Trademarked names are used editorially, to the benefit of the trademark owner, with no intent to infringe on the trademark.

Open Source Attribution Notice

The Confirm suite of products contain the following open source software:

- Feature Data Objects v 3.5.0, which is licensed under GNU Lesser General Public License, Version 2.1, February 1999 with the unRAR restriction. The license can be downloaded from: **http://fdo.osgeo.org/licenceAndGovernance.html**. The source code for this software is available from **http://fdo.osgeo.org/content/fdo-350-downloads**
- MrSID software (specifically the mrsid32.dll) is used under license and is Copyright © 1995-2002, LizardTech, Inc., 1008 Western Ave., Suite 200, Seattle, WA 98104. All rights reserved. MrSID is protected by U.S. Patent No. 5,710,835. Foreign patents are pending. Unauthorized use or duplication prohibited.

  Patented technology in the Software was developed in part through a project at the Los Alamos National Laboratory, funded by the U.S. Government and managed by the University of California. The U.S. Government has reserved rights in the technology, including a non-exclusive, nontransferable, irrevocable, paid-up license to practice or have practiced throughout the world, for or on behalf of the United States, inventions covered by the patent, and has other rights under 35 U.S.C. § 200-212 and applicable implementing regulations.

  For further information, contact Lizardtech.

- NodaTime, version number 1.3.10, which is licensed under the Apache license, version number 2.0. The license can be downloaded from http://www.apache.org/licenses/LICENSE-2.0 . The source code for this software is available from http://nodatime.org/.
- Chromium Embedded Framework, version 3, which is licensed under the New BSD License. The license can be downloaded from http://opensource.org/licenses/BSD-3-Clause. The source code for this software is available from http://code.google.com/p/chromiumembedded/downloads/list.
- Xilium.CefGlue, version 3, which is licensed under the MIT License (with portions licensed under the New BSD License). The licenses can be downloaded from http://opensource.org/licenses/MIT and http://opensource.org/licenses/BSD-3-Clause. The source code for this software is available from http://xilium.bitbucket.org/cefglue/.
- D3 Data Driven Documentation, version 3.4.1, which is licensed under the New BSD License. The license can be downloaded from from https://github.com/mbostock/d3/blob/master/LICENSE. The source code for this software is available from http://d3js.org/.
- OpenLayers, version 8.1, which is licensed under the BSD 2-Clause Licence. The license which can be downloaded from https://github.com/openlayers/openlayers/blob/master/LICENSE.md The source code for this software is available from https://github.com/openlayers/openlayers.
- Proj4js, version 1+, which is licensed under the Apache License, Version 2, January 2004. The license can be downloaded from http://www.apache.org/licenses/LICENSE-2.0.html. The source code for this software is available from http://trac.osgeo.org/proj4js/.
- requireJS, version 2.1.2, which is licensed under the MIT License or the New BSD License. The license can be downloaded from https://github.com/jrburke/requirejs/blob/master/LICENSE. The source code for this software is available from http://requirejs.org/.
- Apache Cordova, version 11.1.0, which is licensed under the Apache License, Version 2, January 2004. The license can be downloaded from http://www.apache.org/licenses/LICENSE-2.0.html. The source code for this software is available from http://phonegap.com/download/.
- Xilium.CefGlue, version 75.1, which is unlicensed. The source code for this software is available from https://gitlab.com/xiliumhq/chromiumembedded/cefglue.

- Chromium Embedded Framework, version 75.0, which is licensed according to the following criteria:

  Copyright (c) 2008-2014 Marshall A. Greenblatt. Portions Copyright (c) 2006-2009 Google Inc. All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  * Neither the name of Google Inc. nor the name Chromium Embedded Framework nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

  The source code for this software is available from http://opensource.spotify.com/cefbuilds/index.html#

October 02, 2026

Table of Contents

Specifications

Confirm Connector Installation 6

Specifications

The following sections outline all the Specifications that exist within the Confirm functionality.

In this section

Confirm Connector Installation 6

Confirm Connector Installation

Introduction

The purpose of this document is to detail the installation instructions for the Confirm Connector.

The Confirm Connector provides a synchronous method of integrating systems to perform specific operations.

After reading these notes, you should be able to perform the following tasks:

- Install and test the Confirm Connector on an Application Server.
- Apply upgrades.

INSTALLATION SERVER

The purpose of the Installation server is to act as a central repository for all of your Confirm installation files, enabling you to install and upgrade your software from this server.

Confirm is supplied as a ZIP file. The documentation will assume this has been extracted to a suitable folder on your Installation Server. The extracted directories will be referred to as the 'Confirm Media' within all documents.

Confirm Media Directory Structure

Note: Directories shown only display those that apply to the Confirm Connector product; your Confirm Media will contain more directories than are listed here.

Confirm Media - Software folder	Description
Confirm Connector	Installation files for the Confirm Connector.
Prerequisites	Root directory for prerequisites
Framework.Net 3.5	Microsoft .NET Framework 3.5 redistributable
Framework.Net 4.0	Microsoft .NET Framework 4.0 redistributable
Framework.Net 4.6	Microsoft .NET Framework 4.6
Framework.Net 4.8	Microsoft .NET Framework 4.8
MapXtreme	MapXtreme redistributable
Visual C++ 2013 Redistributable	Visual C++ 2013 redistributable
Visual C++ 2015 Redistributable	Visual C++ 2015 redistributable

INSTALLING CONFIRM CONNECTOR

This section describes how to install, configure and upgrade the Confirm Connector on your chosen Application Server.

Confirm Connector Requirements

The following items must be installed or configured before attempting to install or upgrade the Confirm Connector: -

- Microsoft .NET Framework 3.5 SP1
- Microsoft .NET Framework 4.8
- Visual C++ 2013 Redistributable and Visual C++ 2015 Redistributable
- MapXtreme 9.1 redistributable

  Note: There is a new version of the MapXtreme Prerequisite in Confirm V19.0 and so this prerequisite software needs to be upgraded as well, when upgrading Confirm.

- Oracle or SQL Server Database client drivers (not supplied on Confirm Media)
- Confirm Configuration file, (Shared profile) see section 'Installation' Step 3 and section 'Connector Shared Profile (configuration) file' for more details
- Local Security Policy – Replace a process level token, see section 'Local Security Policy' for details
- Desktop Heap limits for background processes should be increased from the default values. See section 'Increase Desktop Heap' for details.
- If the Connector is being deployed as a Web Service, IIS needs to be installed, and for IIS 7 or above, the following IIS roles should also be installed:

Common HTTP Feature	Static Content
	Default Document
	Directory Browsing
	HTTP Errors
	WCF HTTP Activation (From Server 2012 onwards)
Application Development	ASP.NET
	.NET Extensibility
	ASP
	CGI
	ISAPI Extensions
	ISAPI Filters
	Server Side Includes
Health and Diagnostics	HTTP Logging
	Logging Tools
	Request Monitor
	Tracing
	Custom Logging
	ODBC Logging
Security	Request Filtering
Performance	Static Content
Management Tools	IIS Management Console
	IIS Management Scripts and Tools
	Management Service
FTP Server	FTP Service

FTP Extensibility
IIS Hostable Web Core

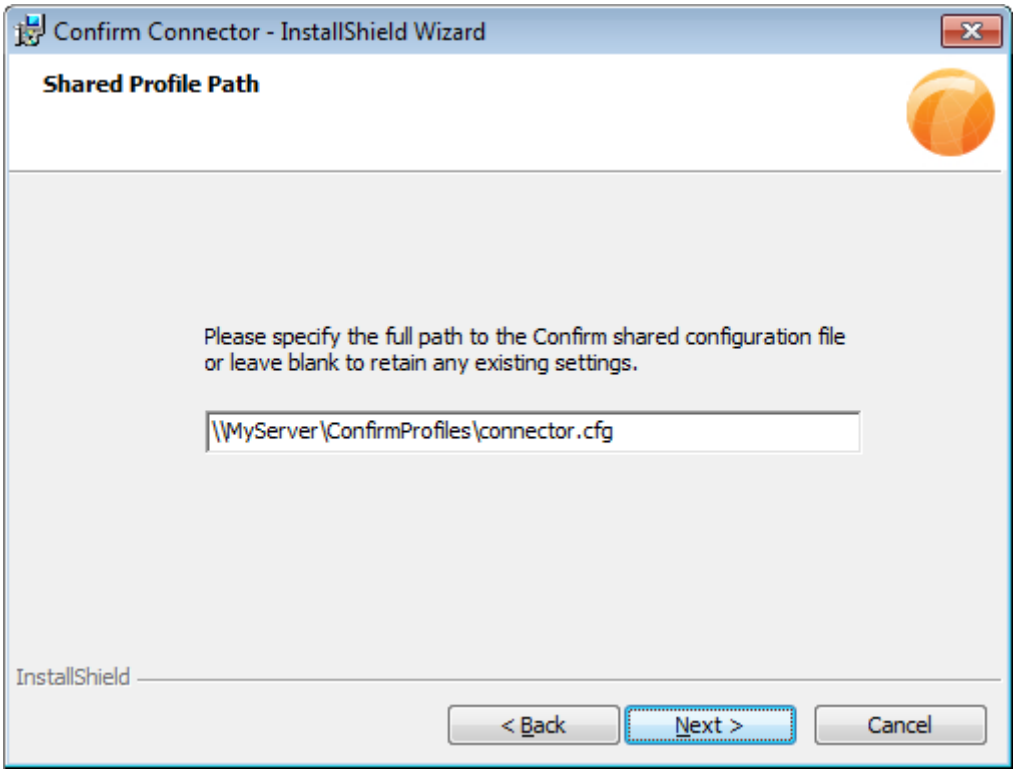
INSTALLATION

The Connector can be used as either a Web Service, or via COM for use with legacy systems.

If you only require the COM interface, install the product using the supplied ‘Install COM Connector only.bat’ batch file.

If you require the Web Service, run the setup.exe. Both installs will launch the same install wizard as documented in the following steps.

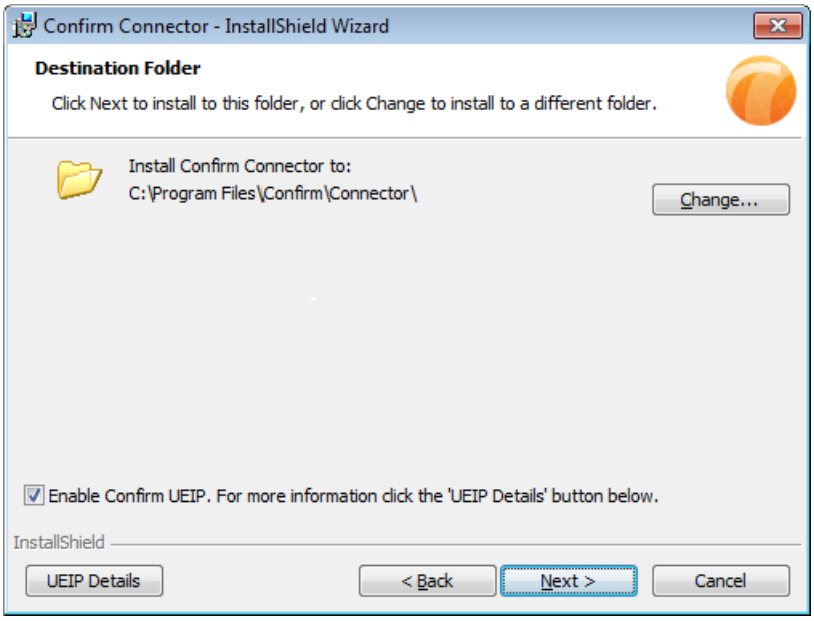
- Launch the installation by running the setup.exe or batch file as previous mentioned.These are located in the Connector directory in the Confirm Media.
- You will see the Confirm Connector Installer welcome screen displaying the version being installed.
- Click ‘Next’ to continue.



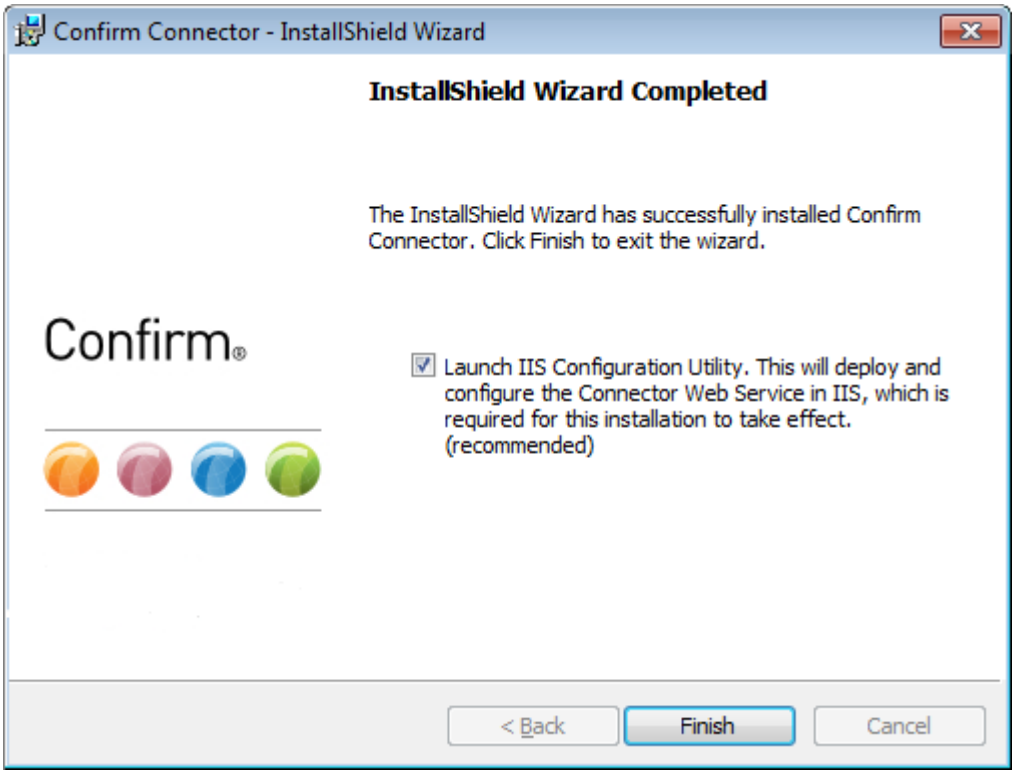
- On first installation you can specify where the configuration file is located which contains the database profiles. On upgrade this step is skipped and the existing location is used. The Connector uses a configuration file in the same way as the Confirm client. Enter the path to the configuration file (as shown in the above screenshot)"<path>\<filename>.cfg"'.

  Note: It is advised that the Confirm Connector uses a separate configuration file from the Confirm client. Make a copy of the Confirm client configuration file and place it into the location you are specifying.

- Click ‘Next’ to continue.



- If you are upgrading Confirm, the current installation directory will be ignored in favour of the default Destination Folder as the upgrade will remove the previous installation as part of the upgrade. Change the Destination Folder if required.
- If you have not previously enabled the Confirm User Experience Improvement Program(UEIP) on the target machine, you will see a checkbox allowing you to enable it now. Click the 'UEIP Details' button for full details of this program.
- Click 'Next' to continue.
- You are now ready to install the Connector
- Click 'Install' to begin the installation process.



- When installation is complete you will get the option to launch the IIS Configuration Utility, provided you are not running the COM only installation. It is recommended that you run this utility so that the

Web Service can be deployed and configured in IIS. The installation will not upgrade the Web Service automatically as this should be located outside of the Program Files installation folder.

- Un-tick the box if you wish to run the utility later, or if you intend to manually configure IIS. Be sure if manually setting up the Web Service that you deploy the contents of the installed WebDeploy to a suitable location outside of the Program Files folder such as C:\inetpub\Confirm\Connector. For full details of deploying using the utility see section 'Deploying Confirm Connector as a Web Service using IIS Configuration Utility'.
- Click ‘Finish’ to exit the installer and optionally launch the IIS Configuration Utility.

You can always modify the Connector installation after the install steps have been completed by going to the Add / Remove Programs screen in the Windows Control Panel. From here the install wizard will let you select features of the install such as the Web Service or the COM based Connector.

Upgrading Confirm Connector

The Confirm Connector will be upgraded only if you are upgrading to new version of confirm connector, which is having new product version. To upgrade the Confirm Connector from an existing installation, follow the steps below: -

- Stop Connector Web Services that are running in IIS. Ensure there are no ConnectorConsole.exe processes running in the background on your server before proceeding to the next step.
- Run the installation process as per section 'Installation'.

Connector Configuration

Connector Shared Profile (configuration) file

A ‘Shared Profile’ or ‘Configuration’ file (.cfg) is used to connect to a database. The file is similar to the Confirm configuration file used to connect the Confirm client.

- Create a copy of your existing Confirm configuration file and save it to a location for your Connector to access.

The Connector knows where to look for this file by reading the registry value for ‘CFGFile’ string,which is shown in section 'Connector Configuration Settings' or if an override is being used as shown in section 'Deploying an additional Confirm Connector on the same IIS Server'.

Note: To create the configuration file, follow the steps outlined in the ‘Installing Confirm Main System’ CHA19255 document.

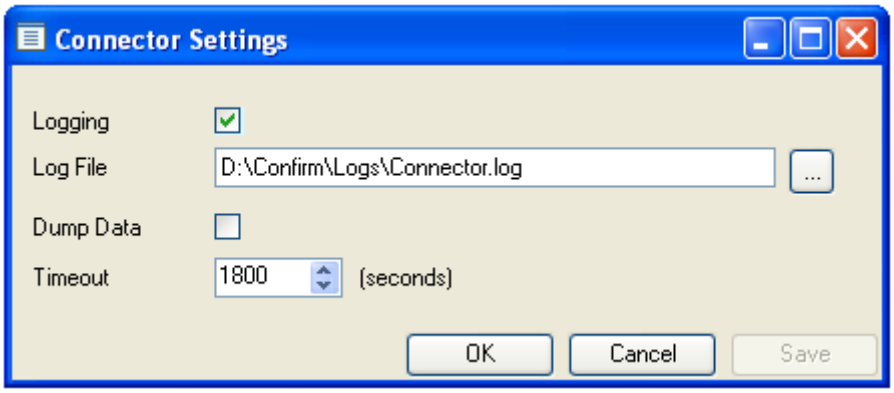
- You will need to manually append the following onto the DBParm= entry as shown below. This will prevent the ODBC connect dialogue from appearing if there are database connection problems.

  ,ConnectOption='SQL_DRIVER_CONNECT,SQL_DRIVER_NOPROMPT'

  Example: - DBParm=ConnectString='Driver=SQL Server;Server=localhost\instancename;Database=Confirm;UID=confirm;PWD=%',DelimitIdentifier='No',ConnectOption='SQL_DRIVER_CONNECT,SQL_DRIVER_NOPROMPT'

Connector Configuration Settings

The Confirm Connector contains logging settings can be set via the Confirm client using the “System Administration > System Configuration > Connector Settings” screen.



These are saved in the Confirm database and will always override any values specified in the Connector registry entries shown below. Note that the registry setting will be used up to the point of connecting to the database; this is to ensure that failures to connect to the database are logged.

The Connectors registry entries contain additional values which do not need to be changed. The registry values can be viewed from the following keys.

(32bit) “HKEY_LOCAL_MACHINE\Software\Southbank Systems\Confirm.AM.Connector”

(64bit) “HKEY_LOCAL_MACHINE\Software\Wow6432Node\Southbanksystems\Confirm.AM.Connector”
An explanation of all the settings are shown below:

Registry Setting	Stored in database	Description
CFGFile	No	Location of the database configuration file.
EXEDirectory	No	The location of the Confirm Connector.
InstallPath	No	This value is set by the installation of the Connector and is set to the location where the MSI installer was run from.
DumpData	Yes	When enabled, additional debugging information will be added into the log file as set via the Log File setting and additional IN and OUT files will be created for each request/response. This can create a large number of files very quickly, so it is advised to only use this when troubleshooting problems. To enable DumpData set the value to ‘Yes’. Both this value and the database value are used independently and when the DumpData is enabled in the registry, the registry log file directory is used. The IN and OUT files are created in subfolders by year, month and day, e.g. …\2013\01\21\...IN.log.
LogFile	Yes	This value is used for the Confirm Connector log file. The default path will point to the Connector install location.
LoggingOn	Yes	This value will cause the Confirm Connector to write all errors to the log file. Default value is set to ‘Yes’.
TimeOut	Yes	This value is used to produce a Time Out error when the Connectors processing doesn’t complete within the time allocated. This value is defaulted to ‘1800000’ milliseconds(30 minutes). If this value is not set then a default of 30minutes is used

To enable Active Directory authentication for Confirm Connector, set the registry values to the below mentioned path

(32bit) “HKEY_LOCAL_MACHINE\SOFTWARE\Confirm”

(64bit) “HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Confirm”

Registry Setting	Description
AuthProviderType	• Confirm - To Use Confirm Authentication. • ActiveDirectory - To Use ActiveDirectory.

Note: When DumpData is turned on the Connector will record every transaction and the log file will grow exponentially, it is advised not to have this turned on during normal operation.

Note: Changing the Local Registry settings will not affect what is stored in the database unless the database setting is blank.

Mixed Mode of Authentication

Mixed mode of authentication allows both Active directory and Confirm users to be authenticated simultaneously. Before this change either Active directory or Confirm authentication was possible but not both.

Mixed mode can be enabled from Confirm Configuration Service.

Configuring Mixed Mode Authentication

A configuration file is required to enable Mixed mode. Please follow steps below:

- Open a text editor and create a new file.
- Copy and paste the following text in the file:

  {"AuthProviderType":"Mixed"}

- Save and name the file as "Filename.json".
- To enable Mixed Mode authentication for Confirm Connector, set the registry values in Configuration Service to the below mentioned path

(32bit)“HKEY_LOCAL_MACHINE\SOFTWARE\Confirm\ConfigurationService”

(64bit)“HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Confirm\ConfigurationService”

Registry Setting	Description
AdConfig	Stores Active Directory configurations.Path of AD json file.

Local Security Policy (Not required for Windows 2008 or above)

In order for the Connector to utilise the .asp requests, either as a standalone asp page or a web service you will need to modify a security policy on the server. Perform the following steps to enable the policy.

- Open the Local Security Setting screen ‘Start > Programs > Administrative Tools >Local Security Policy’.
- Navigate to ‘Security Settings > Local Policies > User Rights Assignment’ and select‘Replace a process level token’.
- Add ‘Service’ to this policy and ensure that both Local Policy Setting and Effective Policy Settings are enabled.

Note: If the Effective Policy Setting is not enabled and is greyed out you may need to reboot for the policy to become Effective.

SQL Server 2005/2008/2012

An additional step is required for the Connector to work with SQL Server 2005, 2008 and 2012.This needs to be performed on the database server.

Run the following SQL whilst logged in to the ‘Master’ database: -

- Grant view server state to Confirm
- Grant view server state to shared

This grants access to the syslockinfo table in the Master database. Without this done, the Connector will generate an error.

Note: You will need to run this command for each data set in the database replacing Confirm with the username in the database (i.e. highways, property etc.).

Increase Desktop Heap

The Connector uses a number of background processes names ‘ConnectorConsole.exe’. These are launched by the Connector Web Service to process requests. Windows limits the amount of Desktop Heap these processes have access to, which can cause problems in high load situations where multiple copies of the ‘ConnectorConsole.exe’ processes are run in parallel. To ensure stable running of the Connector under load please make a change to the following registry setting on your server.

- Run RegEdit as an Administrator and browse to the following registry location:HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\SubSystem
- Select the ‘Windows’ key within this location and edit its value.
- Find within the value the string ‘SharedSection=’ which is followed by 3 numbers such as ‘SharedSection=1024,20480,768’.
- Edit the third value (in the above example this will be 768) to be 4096, so our example will look like ‘SharedSection=1024,20480,4096’. Do not change any other values in this string, even if they differ from those in this example.
- Save your changes by clicking ‘OK’. The server will now need to be restarted for the changes to take effect.

DEPLOYING THE CONNECTOR

The Confirm Connector is deployed as a Web Service.

Deploying Confirm Connector as a Web Service using IIS Configuration Utility Confirm

Confirm Connector IIS Configuration Utility is supplied as part of the Confirm Connector Install.It can be run directly at the end of the installation wizard, or, can be run at any time from the shortcut located in the Start menu:

Start Menu -> Confirm -> Confirm Connector IIS Configuration Utility.

The Confirm Connector IIS Configuration Utility can be used to deploy the Connector in two ways, Single or Multiple web sites.

Single Web Site

This is the default, and recommended deployment option for new installations of Confirm Connector. A single Application Pool and Virtual Directory will be created, giving access to all Connector Web based interfaces from a common base URL. The different interfaces within the Connector can then be accesses using the relevant endpoint.

- ConfirmConnector.asmx – for SOAP based requests
- ConnectorAspNet.aspx – for Mobile HTTP requests

Multiple Web Site

When upgrading an existing installation on the Connector, you may be using two separate Virtual Directories in IIS as the ConnectorWebService and ConnectorAspNet sites existed in separate directories. These have now been merged into a single directory, however, to avoid the need to change the URL's used to access these sites, multiple Virtual Directories can be used to point to the same single physical directory. This in effect creates two Virtual Directories which have different URL's but identical functionality. The configuration is provides for backwards compatibility only.

Recommendations

The TestConnector.aspx page used previously to test the Connector installation no longer exists in a separate directory and does not require its own Virtual Directory. It can be found alongside the other Connector pages.

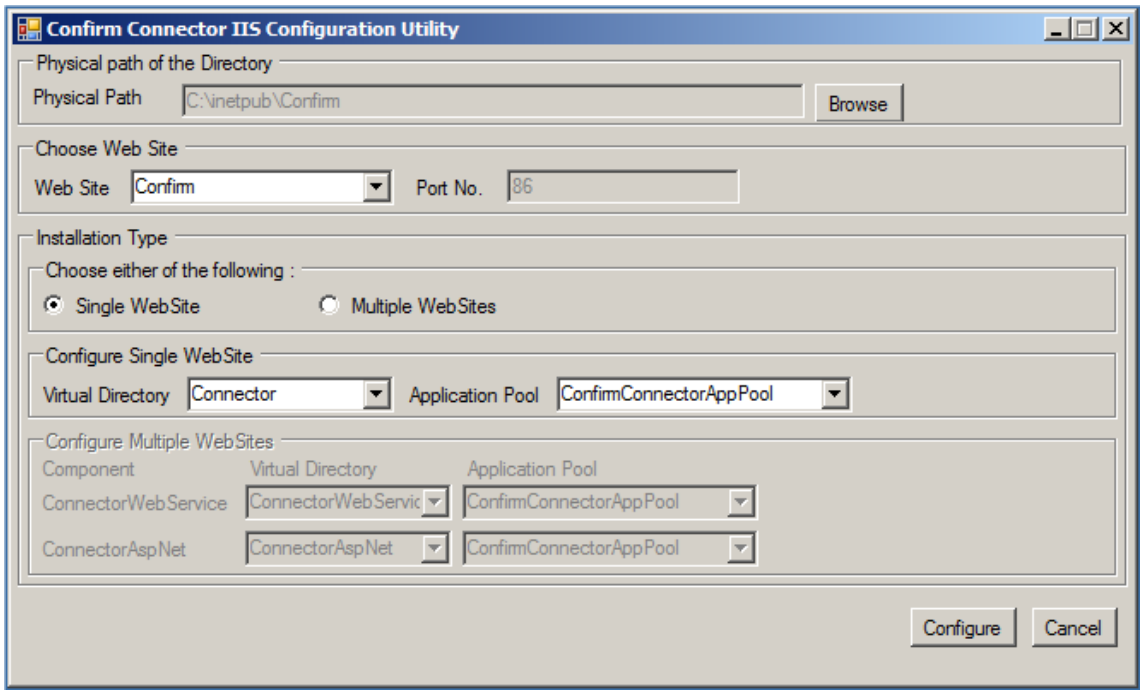
On Windows 2008 or above servers the Confirm Connector IIS Configuration Utility should be granted Admin rights when requested to enable correct access to IIS.

On completion of the Install process, always run the Confirm Connector IIS Configuration Utility straight away to ensure the deployed version of Connector is correctly upgraded.

When backwards compatibility of URL's is not needed, always use the Single Web Site Mode as mentioned in section 'Single Web Site' as this minimises future Administration overhead.

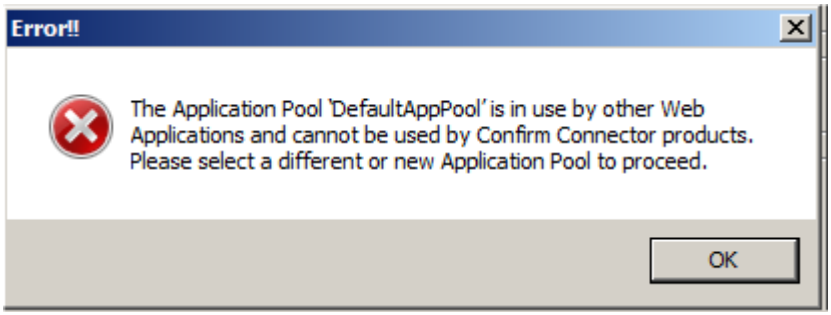
The Application Pool used by Confirm Connector should not be used by any other application.Future upgrades may require changes to the Application Pool that could be incompatible

How to use the Confirm Connector IIS Configuration Utility

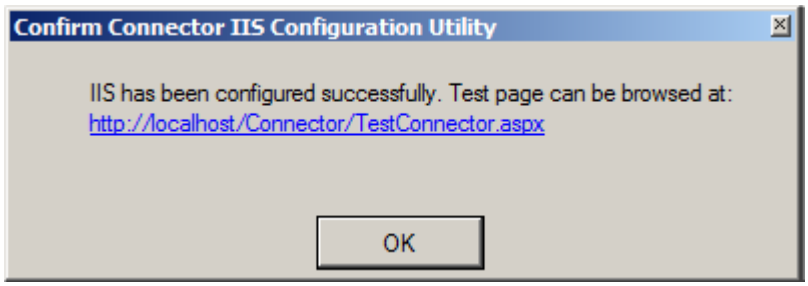


Steps to deploy Confirm Connector in Single web Site:

- By default, the physical path of the Connector deployment directory is set to:'C:\Inetpub\Confirm'. In case you want to change it, you can do that using 'Browse' button.
- Select the Web Site from the list of existing web sites on IIS, under which you want to create virtual directory OR type in the name of new web site in editable box, enter the port no. for that new website, and the same will be created afresh for you.
- Select Virtual Directory: You can either select from already existing Virtual directories or you can create a new Virtual Directory by just typing in the name in the Virtual Directory box above.
- Select Application Pool: You can either select from already existing Application Pools or you can create a new Application Pool by just providing the name in the editable Application Pool box above.
- Click 'Configure'.
- If the selected 'Application Pool' already has some websites apart from the value selected in 'Virtual Directory' drop-down, then you will be prompted with following error message:



- On clicking 'OK', you can either reselect application pool or can create a new Application Pool by just typing in the name of the new application pool in the Application Pool drop down box.
- Configuring IIS may take some time depending on the selections made. When it is completed, a success message will be displayed as below:

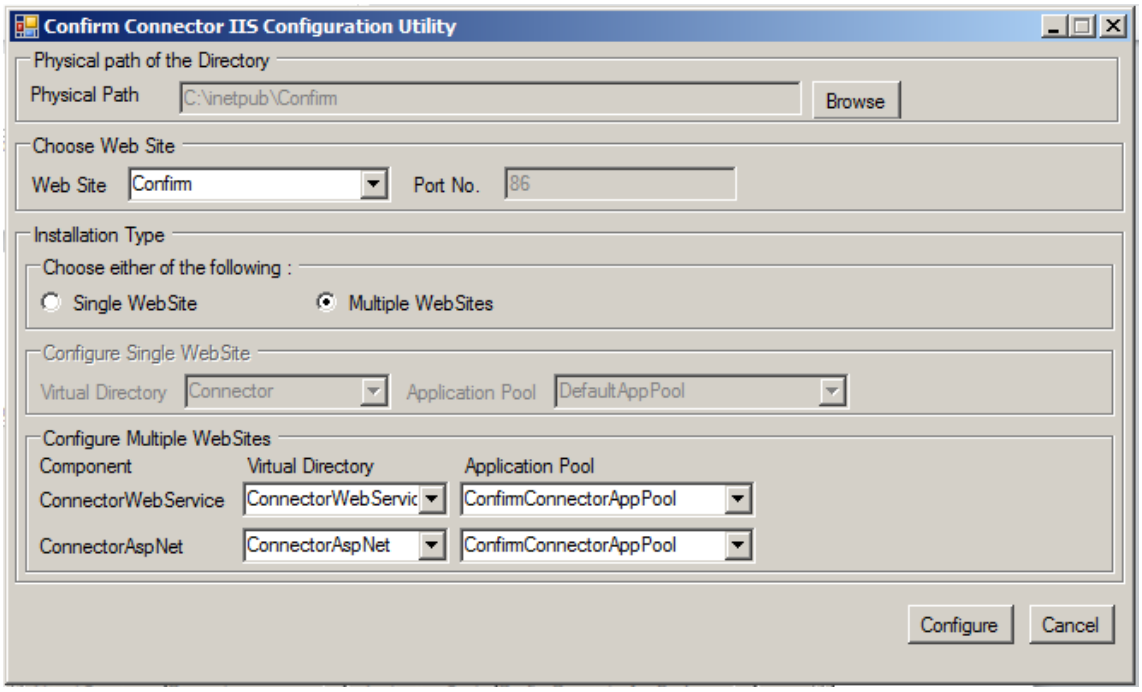


- Clicking on the hyperlink in the message box will open the test page of the Confirm Connector in web browser.
- Click OK to close the utility.

Connector has been deployed successfully.

ConnectorWebService and ConnectorAspNet pages can now be browsed using the same URL by replacing the 'TestConnector.aspx' with 'ConfirmConnector.asmx' for Confirm Connector web service or with 'ConnectorAspNet.aspx' for Confirm Mobile connector.

Steps to deploy Confirm Connector in Multiple web Sites:



Select Multiple WebSites and then follow steps 1 to 5, but repeat steps 3 and 4 for components ConnectorWebService and ConnectorAspNet.

Troubleshooting Confirm Connector IIS Configuration Utility:

In case, there is any issue while running this utility, you will get a message pop up stating the reason and also the log file details. The log file can be found at

"<InstallLocation>\Confirm\Connector\IISConfiguration.log ".

Connector Logging and Folder Permissions

In order to create and update log files, Read/Write permissions must be granted to the directory specified in the Connectors configuration (see section 'Connector Configuration Settings' and the note about DumpData).

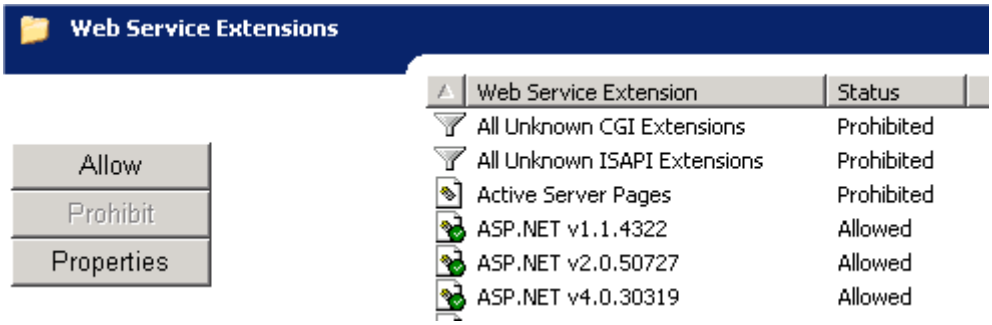
Confirm Connector Web Service Test

To test the Connector Web Service you will need to browse to TestConnector.aspx from the Virtual directory configured on IIS.

Mobile Connector as a Web Service Test

To test the mobile connector you will need to browse to ConnectorAspNet.aspx from the Virtual Directory configured on IIS. An additional step is required for ASP pages to run in IIS on a Windows 2003 server, follow the steps below to allow ASP pages to work: -

- Open IIS and navigate to the ‘Web Service Extension’ folder.
- Select the ASP.NET v4.0.30319 entry and click the ‘Allow’ button.
- Select Active Server Pages entry and click the ‘Allow’ button.



Confirm Connector User

It is strongly recommended that a new user called ‘CONNECTOR’ is created within Confirm specifically for use with the Connector Web Service. Speak to your Confirm administrator to help set this up.

Do not use the common ‘SBS’ user because Work Group policy changes can cause conflicts which prevent the Connector Web Service from working properly. The ‘CONNECTOR’ user should be a member of a Work Group with Seed Numbers enabled, but no special permissions are necessary.

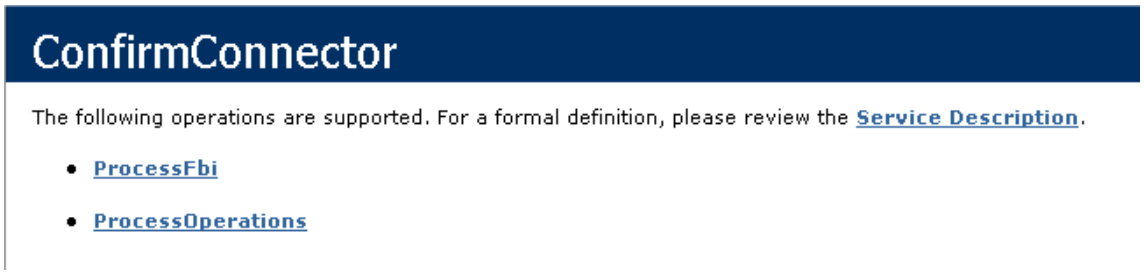
Testing Web Services

This section details how to test the Confirm and Mobile Connectors Web Service deployments.

Testing the Confirm Connector Web Service

To test the Confirm Connector Web Service, from within IIS navigate to the virtual directory as specified by you when deployed using Configuration utility <Virtual Directory>/Application and browse the ConfirmConnector.asmx file.

A correct response will look like the below image:



This proves that the Confirm Connector is running and ready to accept requests via a Web Service.

The TestConnector.aspx page can be accessed from :

ConnectorWebService/TestConnector.aspx

See troublehsooting section should an error occur.

Testing Requests using the Confirm Connector Web Service

Follow the steps below to perform a test request against the database using the Confirm Connector deployed as a Web Service.

- From within IIS navigate to the Virtual Directory/Application and browse the TestConnector.aspx file.
- Enter a Confirm username in the <Username> section. It is strongly recommended that a new Confirm user called 'CONNECTOR' is created specifically for use with the Connector Web Service. The 'CONNECTOR' user should be a member of a Work Group with Seed Numbers enabled, but no special permissions are necessary.
- Enter a password (if applicable) in the <Password>
- Enter a profile name in the <DatabaseId> section which represents a profile name stored in the configuration file.

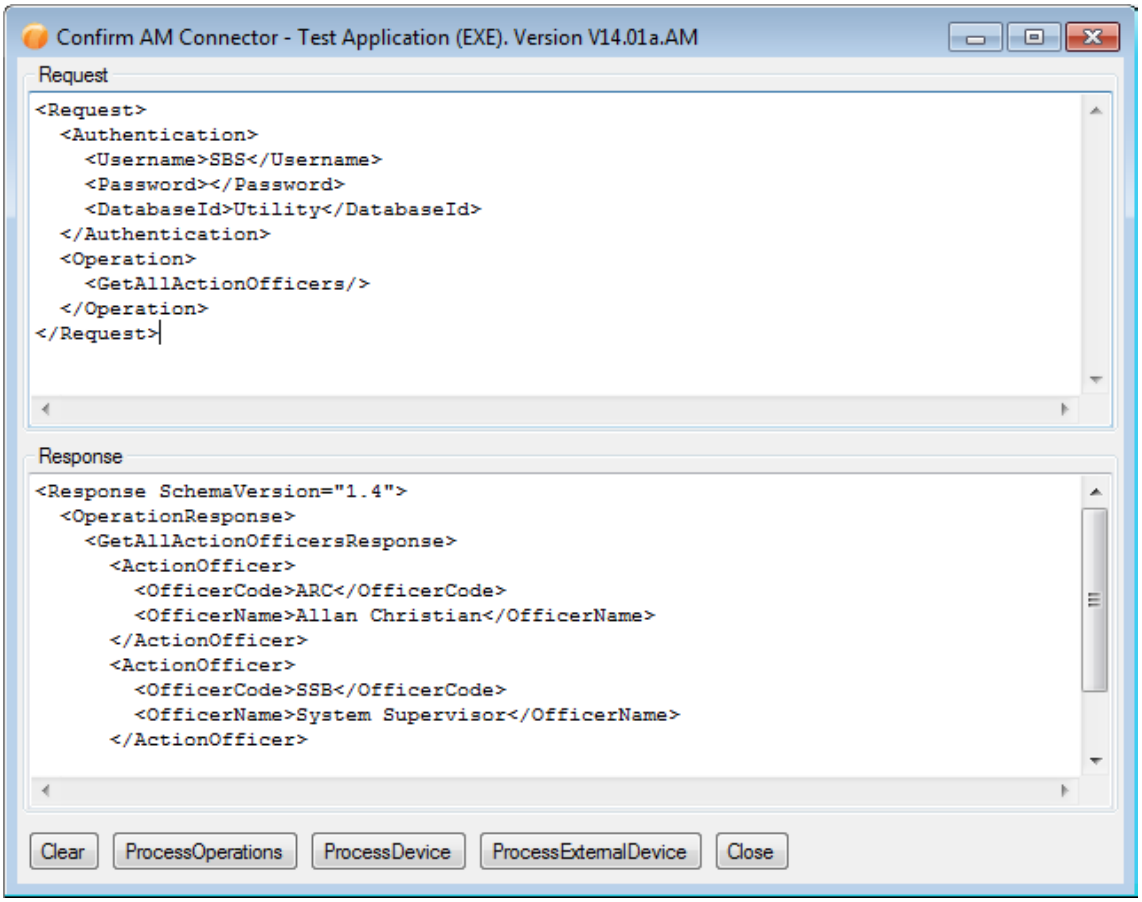
Testing Requests using the test application

A test application is located in the Connector application directory called 'ConnectorConsole.exe'. This can be used to run requests without using the Web Service.

To use this application follow the steps below: -

- Launch the ConnectorConsole.exe from the Connector's application directory (e.g. C:\Program Files\Confirm\Connector).

  Note: Windows 2008 or above requires the application to be run in Administrator mode (right click the ConnectorConsole.exe and select 'Run as Administrator').



This example shows a ProcessOperations request to retrieve all Action Officers within the database using the 'Utility' profile name.

Note:<DatabaseId> is the profile name stored in your configuration file and not the database name.

- Enter a valid request into the 'Request' window. Use the request below if you need to, changing the Username and Database ID as appropriate.
- Click the ProcessOperations button to retrieve the details, which will appear in the Response window.

ProcessOperation Sample: -

<Request>

<Authentication>

<Username>CONNECTOR</Username>

<Password></Password>

<DatabaseId>ConLive</DatabaseId>

<Authentication>

<Operation>

<GetAllActionOfficers/>

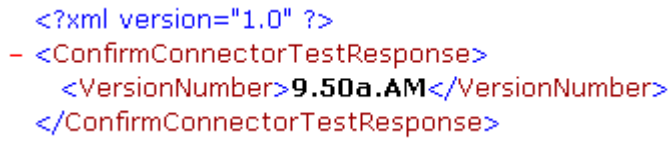
</Operation>

</Request>

Testing Mobile Connector

Follow the steps below to perform a test request against the database using the Mobile Connector via a Web Service.

- Launch Internet Explorer.
- Enter the following URL 'http://localhost/<VirtualDirectory>/ConnectorAspNet.aspx'.
- If a screen similar to the below appears then the Mobile Connector component is working and will be ready for testing on the device.



Confirm Connector (Optional Settings)

The Confirm Connector has optional settings as detailed below, should you wish to use these settings edit the web.config file in the Confirm Connectors application directory. Default location is "C:\inet-pub\Confirm\Connector\".

Navigate to the <appSettings> section.

<add key="ReturnFaultInResponse" value="true" /> (default = false)

This setting will allow a fault response to be returned embedded within the normal connector response, rather than raising a soap exception containing the fault details.

<add key="SetDefaultNamespace" value="true" />(default = false)

This will set the default namespace on the <Response> element and all child elements to be the same as the default namespace on its parent element; <ProcessOperationResult> (xmlns="http://www.con-

firm.co.uk/schema/am/connector/webservice”). The default value for the <Response> element default namespace is blank (xmlns=””).

Deploying an additional Confirm Connector on the same IIS Server

It is possible to configure multiple instances of the Confirm Connector on an IIS server using separate configuration options for each. When configured in this way, the registry entry for locating a ‘.cfg’ file will not be used (detailed in sections 'Connector Shared Profile (configuration) file' and 'Local Security Policy').

Prerequisite: Before starting deployment of a separate connector instance on same IIS Server, please make sure you have already configured one Connector instance on that machine following this installation guide and testing it runs successfully.

Please follow the following steps:

- Follow the steps in section 'Deploying Confirm Connector as a Web Service using IIS Configuration Utility' to deploy an another version of Confirm Connector on same IIS, the only IMPORTANT step is that please make sure you choose(browse) a different physical path for the new connector and give a new name of Virtual Directory.
- Once successful, please go the location where new connector has been created(physical path), rename “TemplateCfgFilePath.config” to “CfgFilePath.config”.
- Open “CfgFilePath.config”.
- Give the path of the configuration file (.CFG) that you want to be used by this instance.If a new file is required, please refer to section 'Connector Shared Profile (configuration) file'.

  For example:

  <settings>

  <add key="CfgFilePath"value="C:\NewDbConfig.CFG"/>"

  </settings>

- Restart IIS.
- Open IIS Go to the website and then to the virtual directory as selected/created in Step 1 above. Browse the TestConnector.aspx.
- Supply a sample request to the new Connector instance to ensure that the configuration is correct.
- Now the URL of newly created instance of the Connector can be used as separate Connector instance, with different configuration, irrespective of the local registry entries

TROUBLESHOOTING

Mobile Connector Photo Upload

By default Confirm Mobile photo uploads are sent directly into the Confirm database using a blob table. When the first user tries to view the photo, Confirm will then attempt to move the photo from the blob table to the location defined in the ‘Document Store Folder’ field of Confirm ‘Document Link System Settings’ screen.

Alternatively, it is possible to change this functionality to upload the photo directly to the ‘Document Store Folder’ by setting the following value in the Mobile Connectors web.config file to false.

<add key="UseDocLinkBlobData"value="true"

UNC paths are permitted in either implementation.

However, when setting this value to false, the Connector will attempt to place the files into that location as the account set in the application pool identity. So please make sure relevant permission should be

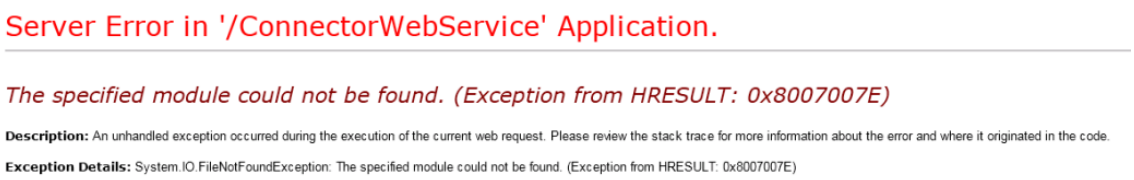
given to the group as per the application pool identity. It will not be possible to grant permissions on a folder on the network, this will be fine if the shared location is on the server running the Connector.

Connector Log file not being populated

You have configured the Confirm Connector logging correctly yet the log file isn't being populated.

This can happen if permissions to the Connector Log File directory have not been granted to the application pool identity user of the application pool in use. See section 'Connector Configuration Settings' for details.

Specified Module could not be found.



This error also contains a large stack trace and can usually be resolved by rebooting the server.

Restarting IIS does not resolve the problem.

It can occur when modifications have been made to files referenced by IIS whilst IIS is still running but there is no consistent cause.

Fault: CONNECTOR_ERR_MISSING_CFG_SETTING

This error indicates that the Connector cannot find the requested database profile in the Confirm database configuration file. This may be because the file cannot be found in the location specified in the CFGPath registry setting (see 'Installation' and 'Connector Shared Profile (configuration) file') or the database profile name is invalid in the configuration file.

Fault: DA_ERR_CONNECT

The Connector is unable to connect to the specified database. See the <reason> value for more details and consult with the Confirm database specific installation notes.

- Reason: SQL_DRIVER_NOPROMPT

The above error may occur if the Connector is being hosted by a non-visual Service, such as Component Services or IIS. To rectify it, the DBParm setting for the appropriate database in the Confirm Database Configuration file must be modified as described in section 'Connector Shared Profile (configuration) file'.

Fault: CONNECTOR_ERR_XML_LOAD

If this error occurs, enable logging (section 'Connector Configuration Settings') and view the log file for further information.

Fault: CONNECTOR _ERR_MISSING_DATA

This message will be displayed if an essential element is missing from the XML supplied to the Connector.

Fault: CONNECTOR _ERR_USER_NOT_FOUND

This error can occur if you have entered an invalid Confirm User.

Unable to Browse Web Service

If, after deploying the Web Service wrapper it is not possible to browse to the Web Service URL or the following message or similar is displayed in the browser:

<%@ WebService Language="vb" Codebehind="ConfirmConnector.asmx.vb" Class="ConnectorWebService.ConfirmConnector" %>

It may be that the .NET ASP extensions have not been properly configured. First, verify that the .NET framework has been installed. If it has, then it may be necessary to run the aspnet_regiis utility which can be found in the .NET framework version directory, e.g.

C:\WINNT\Microsoft.NET\Framework\v2.0.50727.

To register ASP.NET, run the aspnet_regiis utility with the “-i” switch, i.e.:

aspnet_regiis –i

Note: the switches can be ‘/i’ instead ‘–i‘ and in some cases this is required.

The request failed with HTTP status 401: Access Denied.

This error can occur when using Web Services and Oracle with the Connector. The Web Service account (usually ASPNET) is trying to execute the Oracle drivers to connect to the database and had been denied access. The following steps should be taken in order to resolve this issue: -

- Launch Windows Explorer and navigate to the ORACLE_HOME directory. This is typically the “Ora92” folder under the “Oracle” folder (i.e. C:\Oracle\Ora92).
- Right click on the ORACLE_HOME folder and choose the “Properties” option from the drop down list.
- Click on the “Security” tab of the “Properties” window.
- Click “Authenticated Users” item in the “Name” list (on Windows XP the “Name” list is called “Group” or “User Names”)
- Uncheck the “Read and Execute” box in the “Permissions” list called “Permissions for Authenticated Users”
- Re-check the “Read and Execute” box under the “Allow” column (this is the box you have just unchecked)
- Click the advanced button and in the “Permissions Entries” list make sure you see the “Authenticated Users” listed there with:
  - Permission = Read and Execute
  - Apply To = This folder, subfolders and files
- If this is not the case, edit that line and make sure the “Apply onto” drop down is set to “This folder, subfolders and files”. This should already be set properly but it is important you verify this.
- Click the “Ok” button until you close out all of the security properties windows.

It has been known for this process to not fix the problem, if this is the case then follow the steps below: -

- Use Windows explorer and navigate to the Oracle home directory (i.e. C:\Oracle\ora92)right click then select properties, select the security tab.
- Tick “allow inheritable permissions from parent to propagate to this object”
- Click copy, when the security window appears click apply.

- Click the advanced button and tick the “allow inheritable permissions from parent to propagate to this object”, click apply then click ok and finally click ok.

The user does not have permission to perform this action.

This SQL Server 2005 only error occurs because the SQL Login users do not have permissions to view the syslockinfo table in the master database. Follow the instructions in section 'SQL Server 2005/2008/2012' to resolve this issue.

Page cannot be Found

Http Error 404

This error can occur if ASP pages are being prevented from running on your server, most commonly seen in Windows 2003 server where the ASP Web Extension has not been enabled. See section 'Local Security Policy' for details on how to resolve this issue.

Parser Error

Errors similar to the below can be caused by a corrupted web service / virtual directory, these can usually be resolved by un-installing the Connector and then manually ensuring the ‘C:\inetpub\Confirm\Connector’ directory has been removed.

Then perform the installation as per section 'Installation'.



Incorrect Version Number

The version number of the Connector is checked against the version number of the Confirm database from Connector 8.00zx.AM (GA) and 8.05m.AM (NB) onwards. Validation is only against the numeric portion of the version number. Therefore, Connector 8.05m.AM is compatible with database version 8.05z.AM but not 8.06a.AM.

Oracle databases with the Web Service

If you use Oracle database and observe an error “System.data.oracle client requires Oracle client software version 8.1.7 or greater”, and Oracle client software has been installed, Authenticated Users must be granted Read/Execute permissions to the Oracle Home directory.

Note:A reboot may be required for the changes to take effect.

Mobile Device Photo upload failing with message ‘Photo path is not valid …’

This will happen if:

- The ‘Document Store Folder’ setting has not been done in System Administration/System Configuration/Document Link system Settings in the Confirm Client running on the Confirm Connector host machine.
- The specified “Document Store Folder” path does not exist.

Mobile Device Photo upload failing reporting that it is unable to write to the path

This will happen if identity under which application pool is running has not been given permission to write to the “Document Store Folder” path.