iService Administrators Guide

A guide for users of the iService Customer Interaction Solution.

iService Administrators Guide

This user guide is intended for users of the iService system. It is not intended to provide information relating to software customization or integration. Feedback regarding this guide should be sent to support@1to1service.com.

PO Box 2231 • Champaign, IL 61825
Phone 217.398.MAIL (6245) • Fax (815) 366-8167

Table of Contents

1 Welcome

This Administrator guide describes how to install and upgrade the iService customer interaction solution. It is intended for network engineers familiar with Microsoft Windows Server, Microsoft SQL Server, and basic network administration concepts. It is not intended for people responsible for configuring an iService tenant or business segment.

For information regarding configuring an iService tenant that has already been created, please see the iService Setup Guide.

This guide is available in the following formats.

Online Help - This HTML help can be browsed online. You will be automatically directed to the standard, smartphone, or iPad version that is most appropriate for your browser.

Compiled HTML Help - You can download this Windows Compiled Help file and view it in your browser while offline.

PDF - Acrobat file for printing

iPad / iBook - This is an ePub file for viewing on your iPad. To view it on your iPad, you much transfer the ePub file to your iTunes account. You can right click the link above and save it to your machine or click here to download a .zip version of the file.

2 iService Technical Components

iService can be installed on a single server, or multiple servers depending upon the number of users within your installation. It is a .Net application that requires a Microsoft Internet information service (IIS) web server and a Microsoft SQL Server database. This user guide does not make any distinction between a single server or multiple server configuration.

The components of iService are briefly described below.

SQL Server Database

- iService stores all of its information for the tenants and the overall installation within a set of Microsoft SQL Server databases. Microsoft SQL Server 2008, 2008R2, 2012, and 2014 are certified with iService.
- SQL Server Express can be used, but is not recommended for installations with more than 10 users.
- The reports used with iService business intelligence are built on Microsoft SQL Server reporting services (SSRS). Although SSRS can be run on your primary database server, for security purposes it is recommended that SSRS operate on a separate server. This is because the SSRS server is directly exposed to users and has a larger attack surface than the SQL Server database, which can be located on internal IPs in a more secure location. Reports are compatible with SSRS 2008, 2008R2, 2012, or 2014. The SSRS Web Services should operate on a separate server, but the SSRS database may reside on the database server.

iService Website

NOTE: The IIS Web Server must be running at least 4.6.x.


The installer will detect your version of .NET and prompt for updating if you are not running the correct version. If you are running other applications on the web server, ensure the new version is compatible with your other applications.

- Users configure their iService tenant and access all of its features through the iService website. The website requires Microsoft IIS 7 or higher, but IIS 8 is recommended for best performance (supports Websockets).
- iService is built upon a set of Web services that interact with the iService database. The Web services file (iService.asmx) is typically located on the same Web server as the iService website, but can operate on a separate Web server if desired.
- Any error log files generated by the web services will be written into the folder that contains the web services. You must ensure that appropriate access is granted for writing to this folder. The deploy script 32 will set permission automatically.

MailPopper Service

NOTE: The server running the Windows service must be running at least 4.6.x.

- iService uses a Windows service that gathers mail from monitored mailboxes, launches mail processing threads, and performs a variety of other functions. This service can be installed on any Windows server (we do not recommend running the mailpopper service on the database server).
- The MailPopper service interacts heavily with the iService Web services, and the best performance is usually found by installing the service on your Web server that contains the iService Web services file.
- The DLLs for this service are by default stored in c:\Program Files\1to1Service\iService. Any error log files generated by the mail popper service will be written to this folder.

3 Installing iService

You must have administrator access to the servers on which you will be installing iService. You will also need the SQL user that has the create database permission to create the iService database. For security purposes, it is best to have a new SQL account created by your DBA for this purpose rather than use the default SA user account shipped with SQL Server.

This installation guide assumes you have already installed MS SQL Server on your database server, and MS IIS on your web server.

Preliminary Considerations

The iService Web Application must be installed on a Windows 2008 or newer web server running IIS 7.x or newer. We recommend running IIS 8.0 or higher with the Web Sockets feature installed. See https://www.iis.net/configreference/system.webserver/websocket. The web server must have the .NET 4.6.1 or higher run-time package installed (If it is not preset during installation, iService will automatically download and install it during the setup process). The iService Mail Popper Service should be installed on a Windows 2008 or newer Windows Server, and also requires that the .NET 4.6.1 or higher run-time package be installed. The Master and Tenant databases require a SQL Server 2008 or newer database server.

Installation requirements for SQL Server are outside the scope of this document. Note that it is required that the server be configured to allow Windows and SQL Server Authentication. This is because you will create a SQL Server User that the iService web services and mail popper will use to access the iService database. In addition, it is recommended that you install the SQL Server Reporting Services (SSRS) when you setup your SQL server as this is the engine that iService Reporting will utilize when the reports are formally released. For best security practices, it is recommended to run SSRS on a separate server from your actual database.

When determining what servers to use for the installation of each component, keep in mind the following requirements:

- The Web Server must be able to be reached by your agents and must be able to open a network connection to your SQL server. This server must also be able to access the outgoing mail server via SMTP for delivery of agent responses. If you plan on utilizing Secure Messaging to your customers or the Find Answers Knowledge Base, your customers will also need to be able see web pages served by this server.
- The iService Mail Popper Service must be able to access the mail server where incoming messages may be retrieved via POP3, SSL-POP3, or IMAP. This server must also be able to access the outgoing mail server via SMTP for delivery of agent and alert notifications. These services must also be able to open a network connection to your SQL server.
- The SQL server must be reachable by both the Web server and the Mail Popper Service.

Web Server Prerequisites

As indicated above, .NET 4.6.1 or higher is required for the web application to function. Installation of .NET is beyond the scope of this document, but it can generally be added as a feature during the Windows Server setup. Otherwise, the iService installer will detect that it is not present, and will prompt to automatically download the pre-requisite .NET libraries.

The Web Application pool used by iService needs to be set to use .NET CLR 4.0.

3.1 The Installer Archive

The iService Installer Archive

The installation files and installer are included within an archive that typically has a name such as:

iServiceCRM-7.6.044-2017.02.16-151658.zip

The naming convention is as follow:

7.6 - the version number

044 - the build number

2017.02.16 - the build date

151658 - the build time (15:16:58 in 24 hour time)

The zip file contains the following.

For New Installation Folder

This folder contains configuration files that generally do not change during an update. You will configure them when you first install iService, but do not modify them during release updates unless specifically instructed to make changes.

Web.config - This is the configuration file used by the iService web services to connect to the iService database. It is located in the root of the iService website.

iService.MailPopper.config - This is the configuration file used by the iService mail popper to connect to the iService database. It is located in the folder that contains the mail-popper and management console code (usually C:\Program Files\1To1Service\iService).

Website Folder

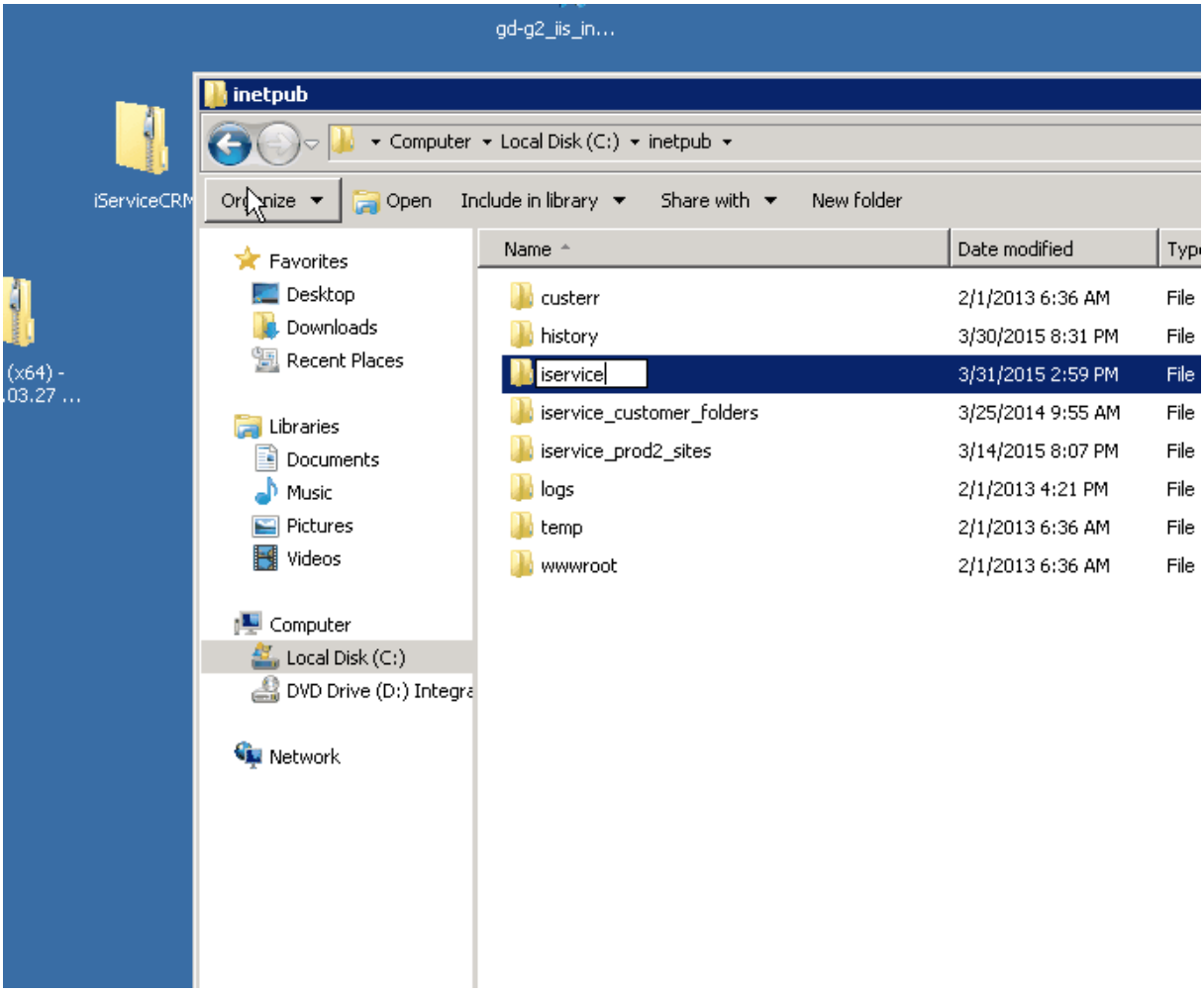
This folder contains the iService website documents to be installed on the web server. If you are performing an update, these files are used to replace your existing website. It does not contain a web.config file.

iServiceSetup.exe file

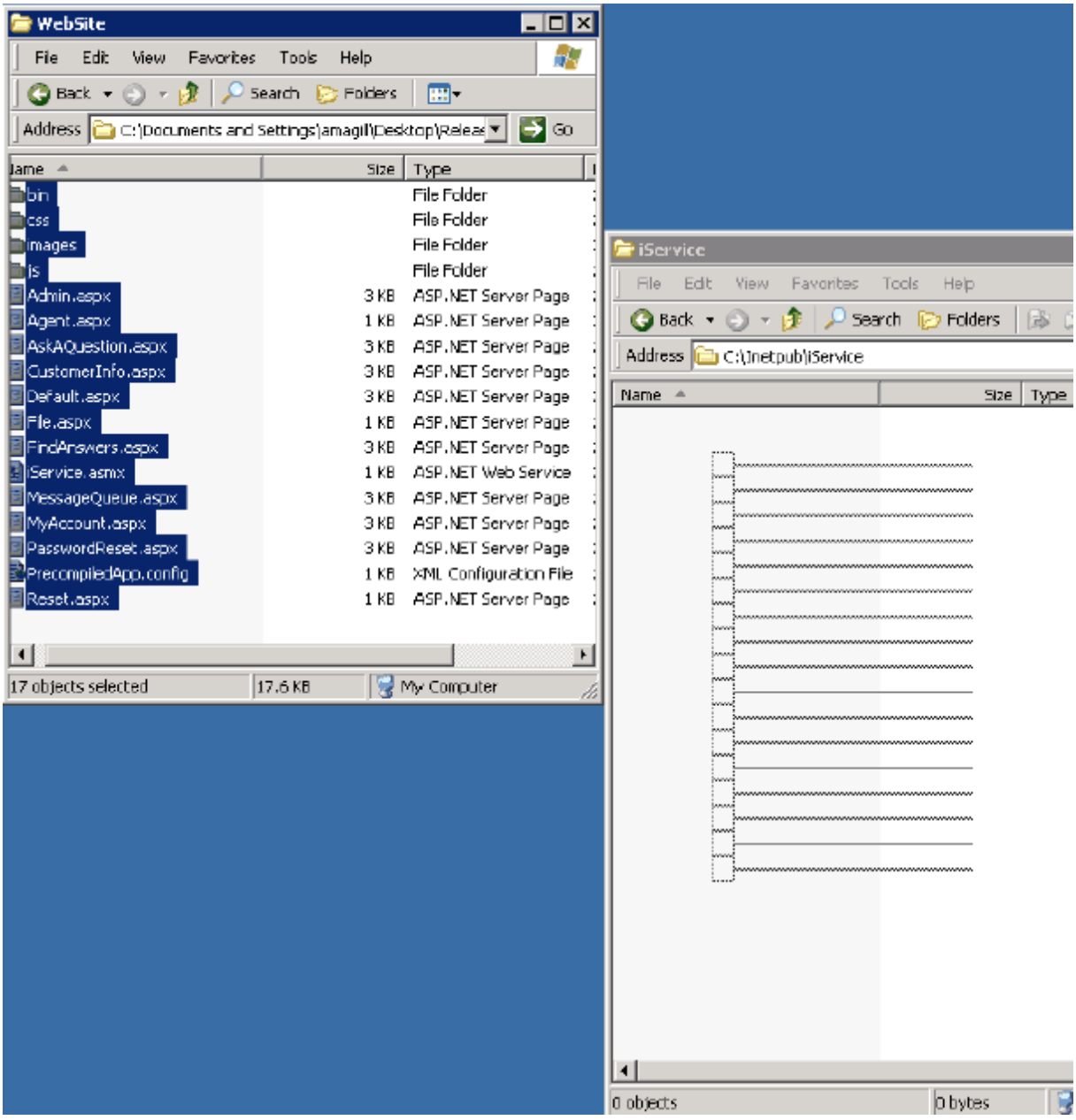
The .exe file is the files that the deploy scripts will use to install iService. You will NOT run this installer directly. Instead, you will run the installer scripts, and those scripts will launch this ext at the appropriate point of the installation.

3.2 Website Installation

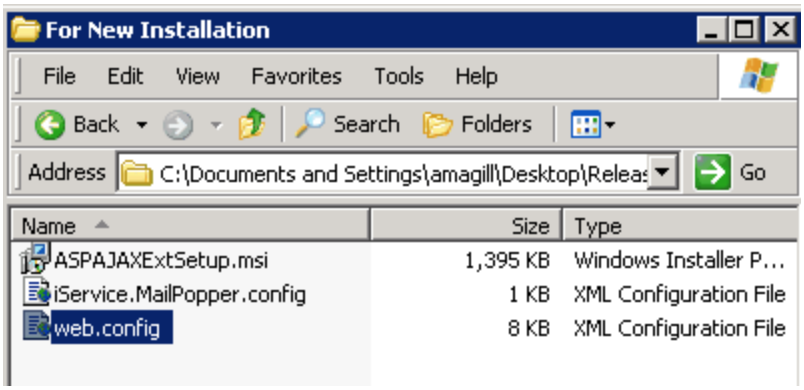
To install the web application, you must determine where to store the web site on your web server. By convention, web sites are usually installed in the C:\Inetpub directory on the web server itself. In our example, we create a new folder named iService at this location.



Once this folder has been created, copy all of the files within the Website directory of our release package into this new folder, shown here:



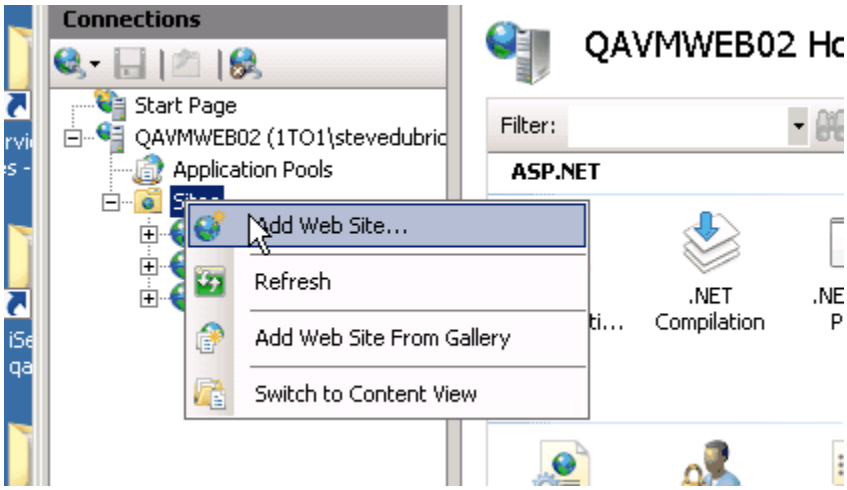
While leaving the C:\Inetpub\iService window open, click Back in the Website folder window and open up the For New Installation folder. Copy the web.config file from this directory into the C:\Inetpub\iService folder.



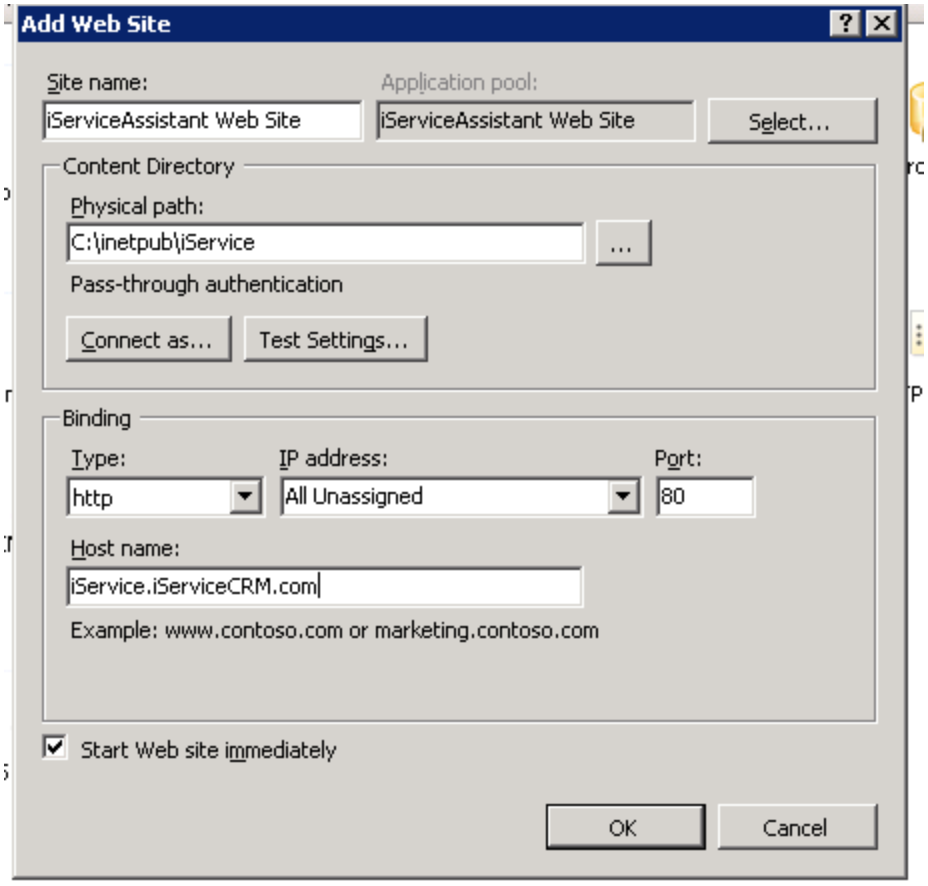
You can close these windows now.

Next, open up the **Internet Information Services (IIS) Manager**. This can be found in the Administrative tools folder usually located in the Control Panels or All Applications links in the Start Menu.

Once this tool is open, right click on the **Sites folder** and select **Add Web Site**…from the popup menu.



Fill in the Site Name with the friendly name you'd like it to appear as in IIS. The physical path should match the location that you just placed the iService web site files (in our example, c:\inetpub\iService). Depending on your desired configuration the Bindings section can be modified to your requirements, but the most straightforward setup is to have a normal http connection on port 80 bind to a host name across all unassigned IP addresses for the server (this is shown in the example screenshot).



If SSL is required for connections, https should be selected with port 443. Though out of scope of this guide, you will additionally need to import a signed certificate and assign it to the binding, as well as set up http to https redirection (such as by using URL rewriting) if required.

Next click on **Application Settings**

And edit the values appropriately.

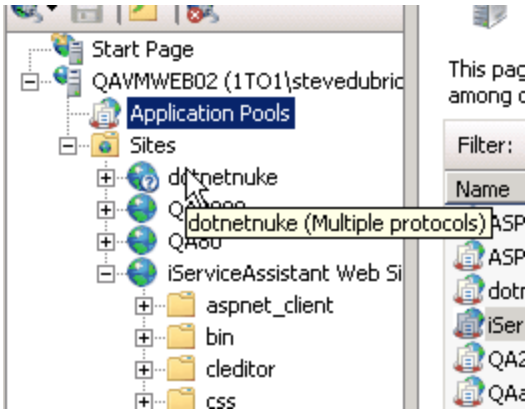
asmxUrl is the location of the iService web services, and if installing the web services on the same web server as the web site, would be http://ServerURL/iService.asmx. In our example above, it would be http://iService.iServiceCRM.com/iService.asmx.

databaseDatabaseName is the name of the iService *master* database

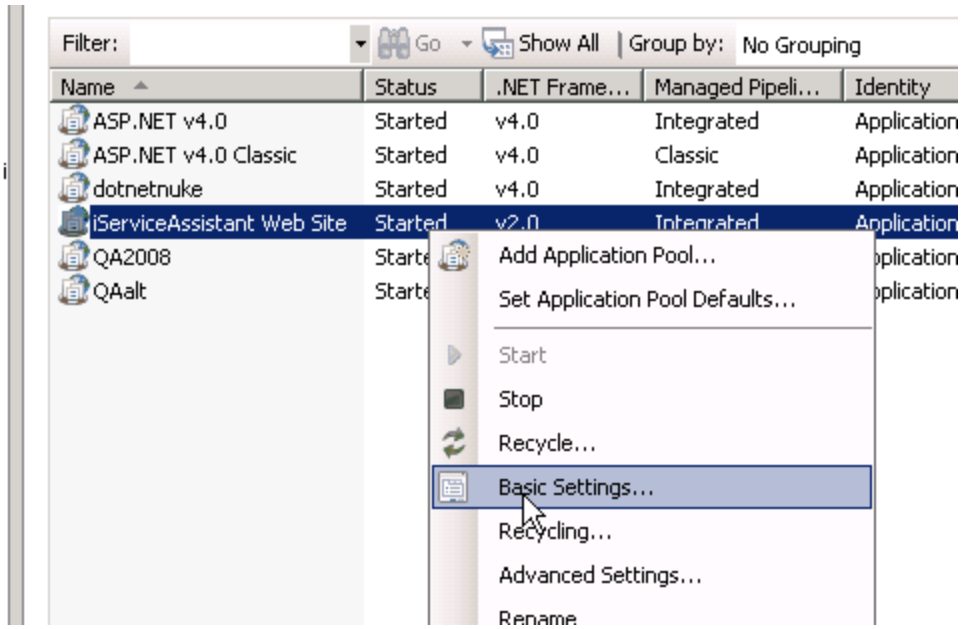
databaseServerName is the name of the SQL server instance where iService databases reside
databaseUserName and databasePassword are the connection properties for the SQL login that has appropriate access to the iService master database.

Name	Value	Entry Type
asmxUrl	http://[redacted]/iService.asmx	Local
databaseDatabaseName	qa2008	Local
databasePassword	[redacted]	Local
databaseServerName	qavmsql01	Local
databaseUserName	[redacted]	Local

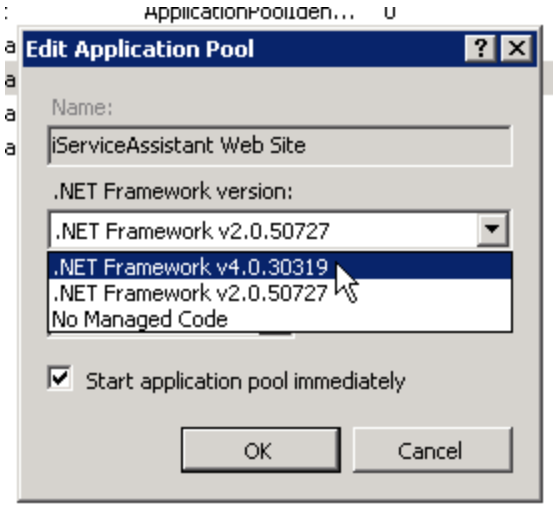
Next click on the **Application Pool** option on the left side of IIS:



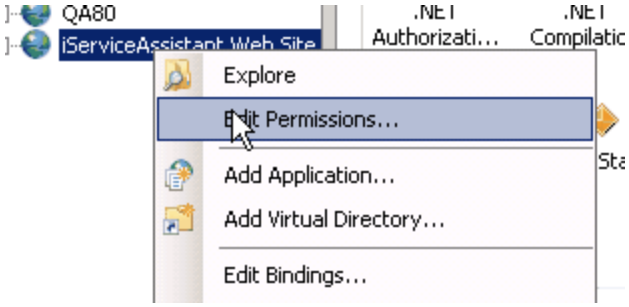
You should see a new application with the friendly name you gave the web site in the previous step. Right click this and choose **Basic Settings**.



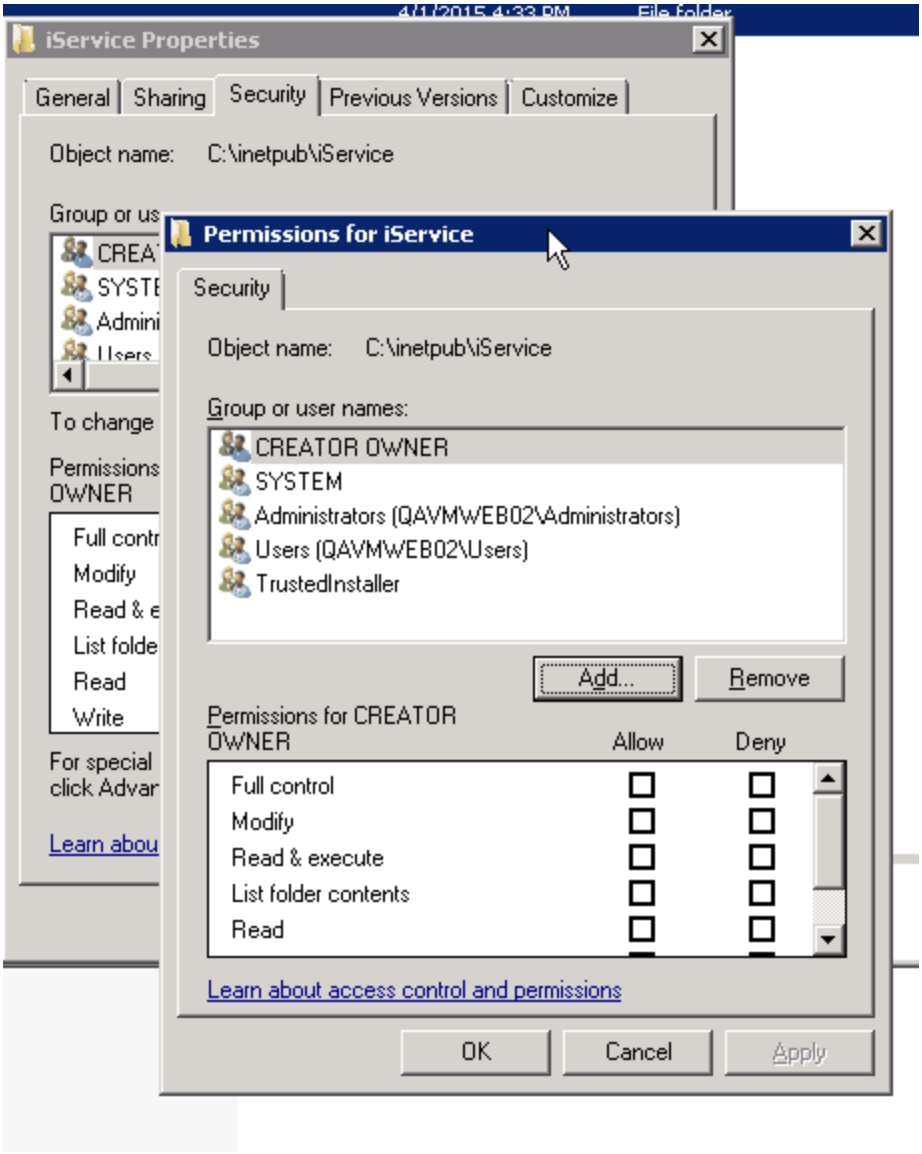
iService uses .NET 4.x, so change the Framework version to the 4.x version, and click OK.



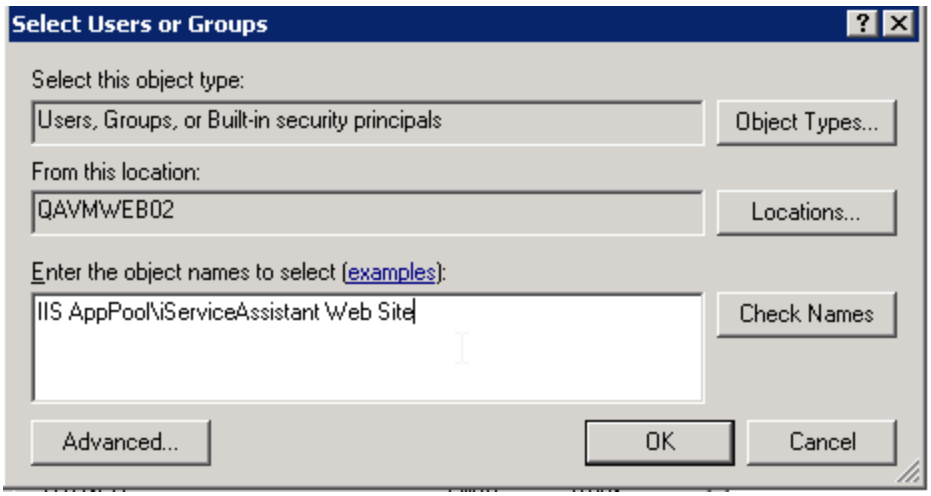
Now we need to be sure that the new Application Pool has the proper permissions to access the folder you made in the earlier step. Right click on the web site in IIS and choose **Edit Permissions**.



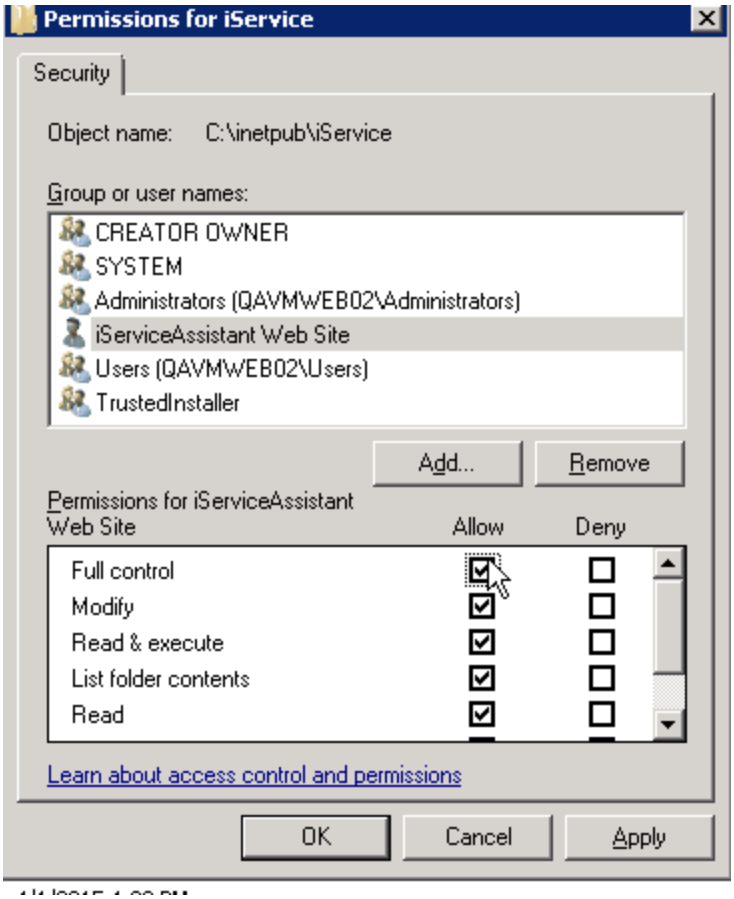
Choose the **Security t**ab, and press the **Edit**.. button. In the new window click **Add**.



In the text area, type **IIS AppPool\<appname>** where <appname> is the friendly name you gave the website in the earlier step. In our example, it would be **IIS AppPool\iServiceAssistant** Web Site



Press OK. Give the app pool Full Control permissions of the folder and Press OK through both windows.



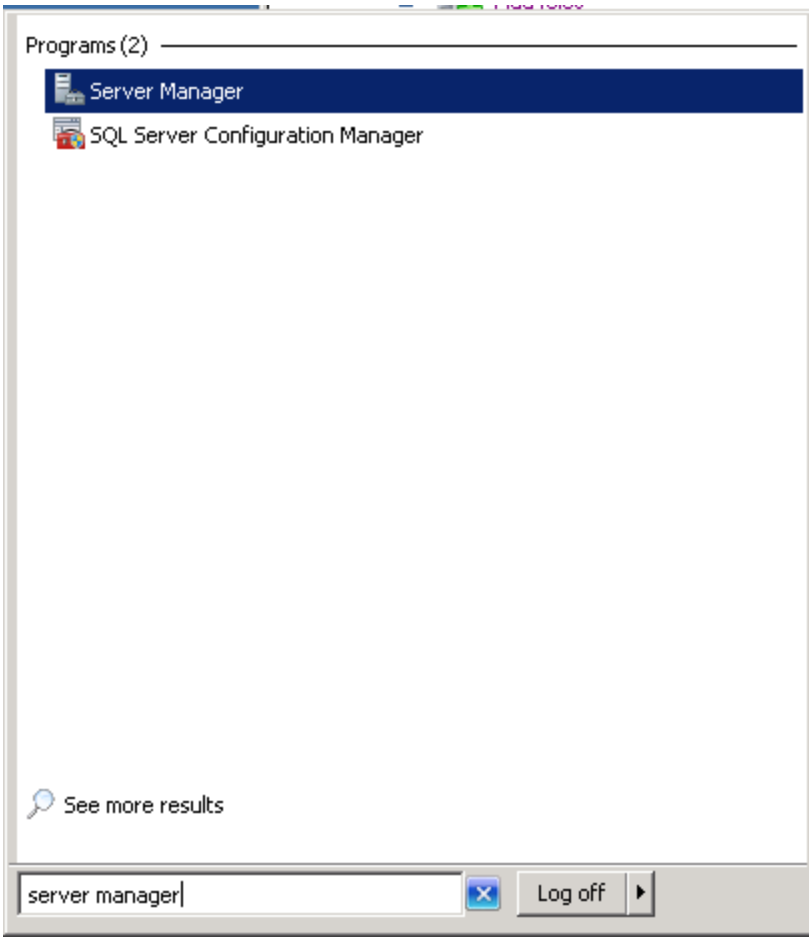
3.2.1 Setting up a Web/Application Windows Server

In order for iService to function as a web application on a Windows server, your server must be setup as both an Application Server and as an IIS Web Server. This can be accomplished by adding these roles to the set of roles your server performs.

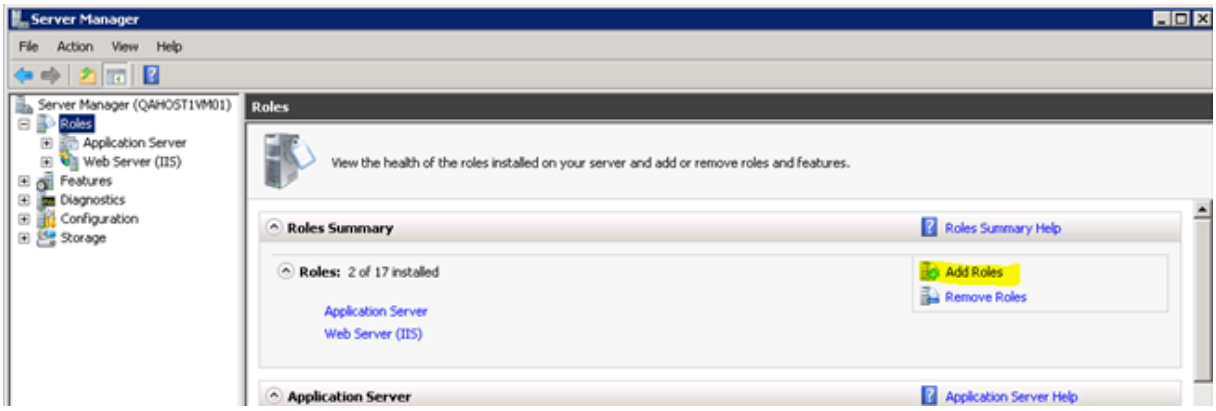
If your server already has these roles installed, then you may safely skip this section and proceed to "Setting up IIS for iService"

To add new roles, you need to first log into your server as an Administrator. This need not be the actual Administrator account, but you may see fewer security confirmation dialog boxes if you are. However, Roles require administrative rights to be installed.

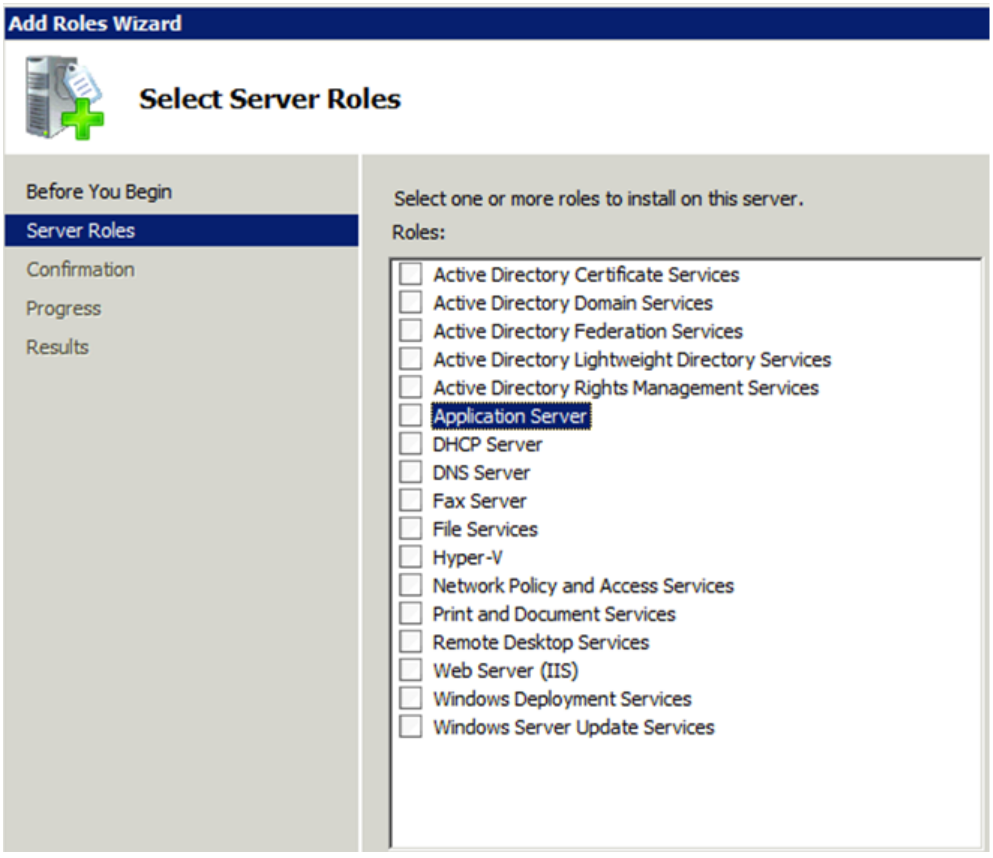
Run the Server Manager application to pull up the Role Management tools. This is most commonly accessed from the toolbar at the bottom of the server screen as the computer icon next to the Start button, but if this icon is not present, you can find it by using the Start button and typing in Server Manager in the "Search Programs and Files" field as shown here:



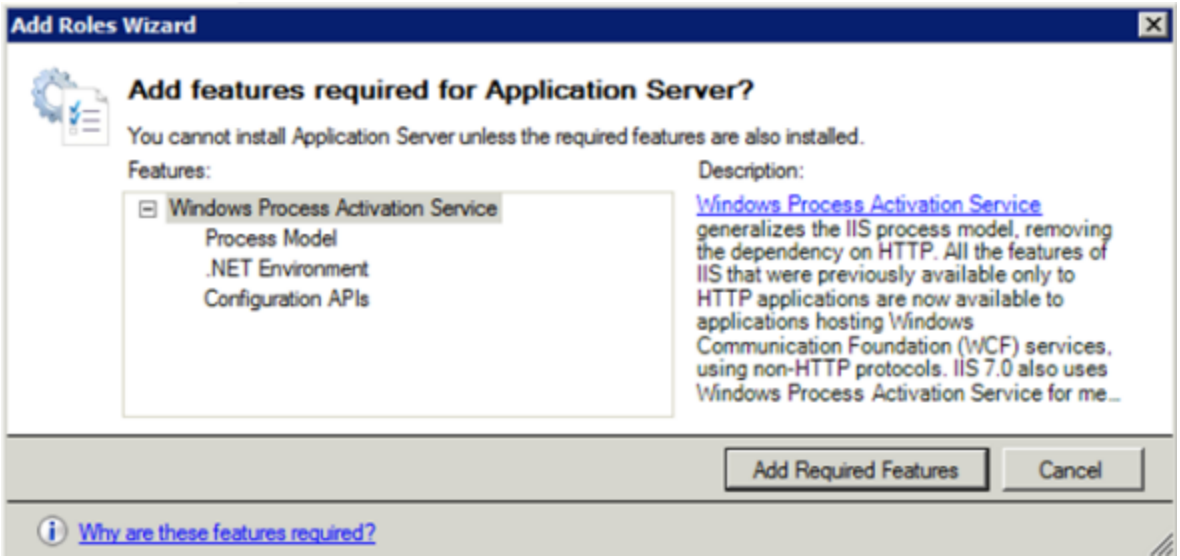
Once the Server Manager is running, select the Roles item in the left pane and then click on Add Roles, which will appear in the far right of the newly opened pane describing existing roles.



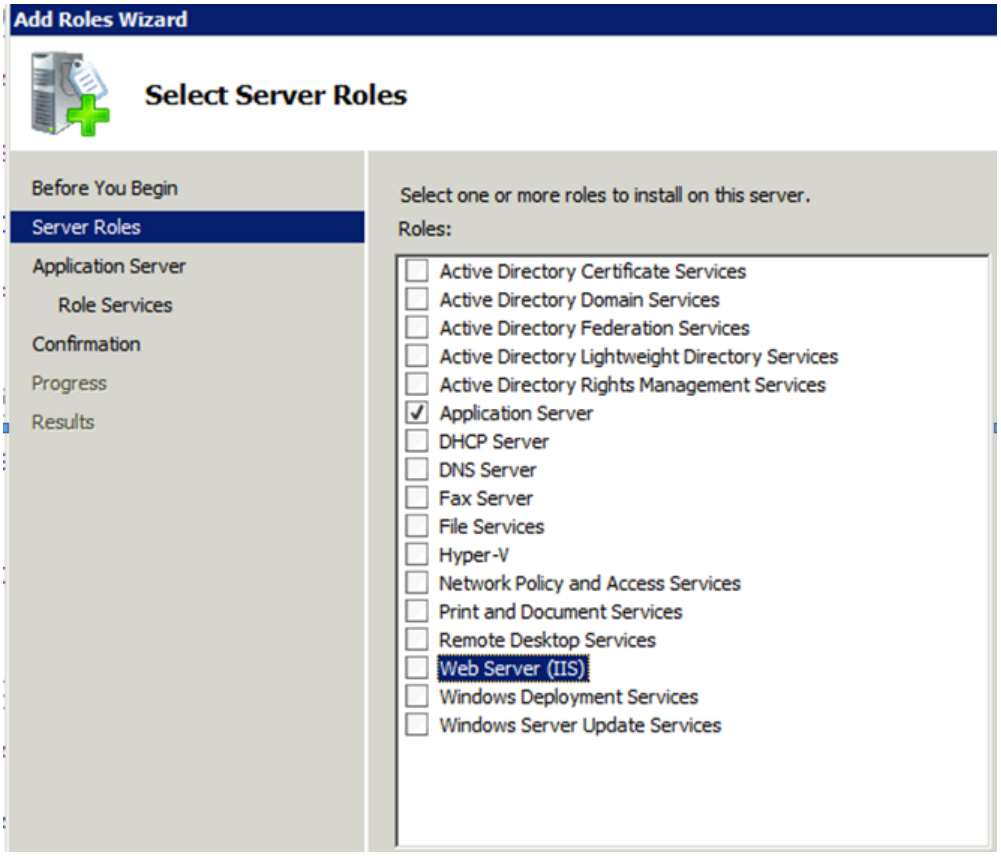
Once you've clicked on this, you will be presented with a list of possible roles which can be installed. Select Application Server and Web Server (IIS).



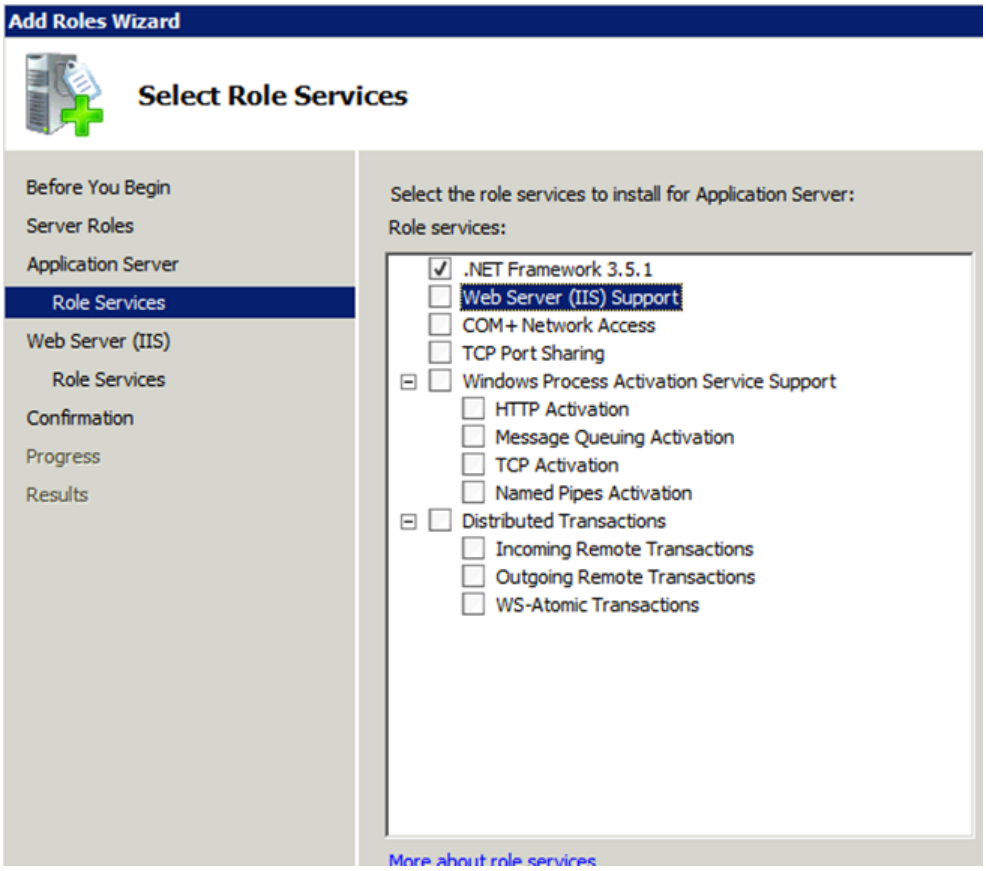
When you select Application Server, you will likely be presented with a dialog listing additional features required for the role to perform properly. Click on the Add Required Features button, as shown here:



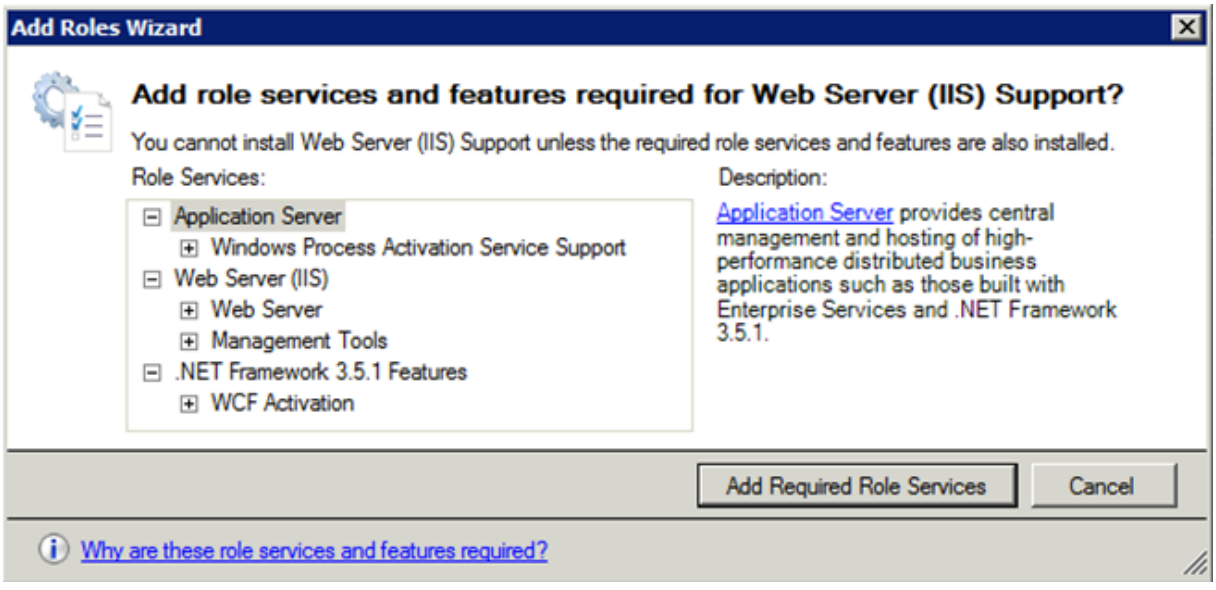
Now, add the Web Server IIS role and check it, as shown here:



When you check the box for Web Server (IIS), you will be asked about additional setting which are required for the Roles to be installed as you require them. Select the Web Server (IIS) Support feature for the Application Server roles to ensure proper coordination between the two roles. (**Note: for iService 6.0 and later you should see .NET 4**)



Check this box and then Next to continue.



Like before with the Application Server Role, these additional features are required for IIS to function properly. Click on Add Required Role Features and then click Next. You will be presented with a confirmation dialog confirming that you wish to install the selected items. Proceed, and your server will be updated to provide the required services.

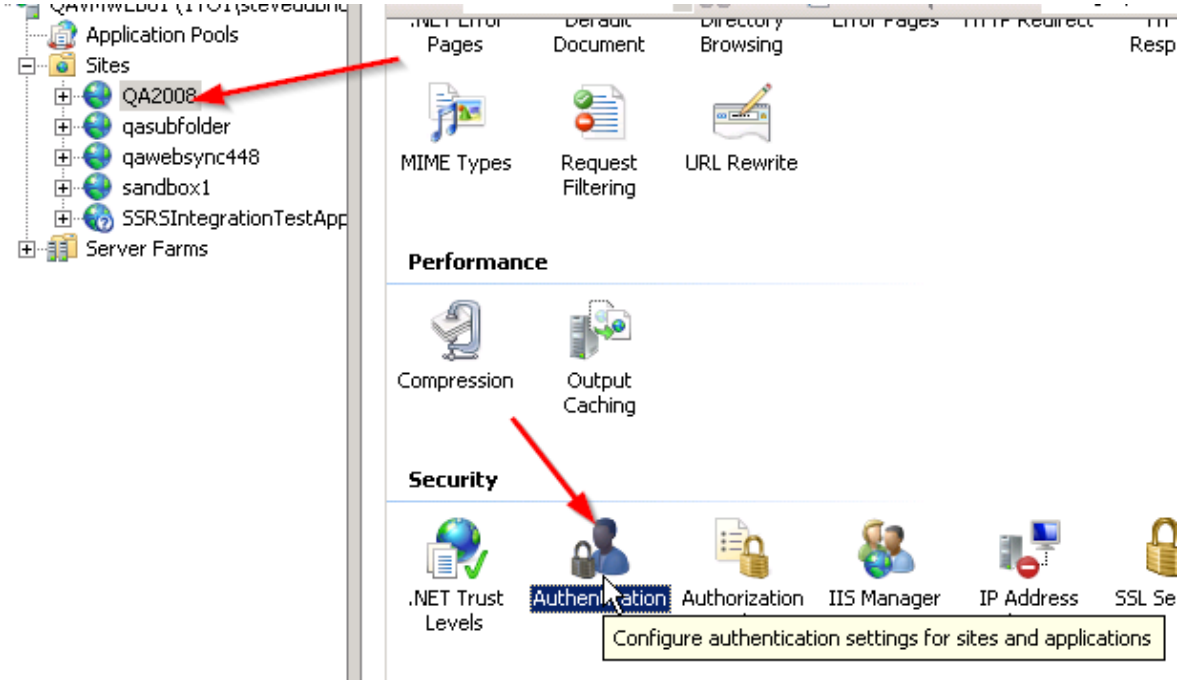
3.2.2 Using Active Directory Integration

You can configure iService to use Active Directory integration. This allows users to be logged directly into iService if they are already authenticated on your Windows domain, and their domain credentials are associate with iService in the Admin Tools interface. Setting up AD integration requires two steps: Configure IIS and Configure the users browser.

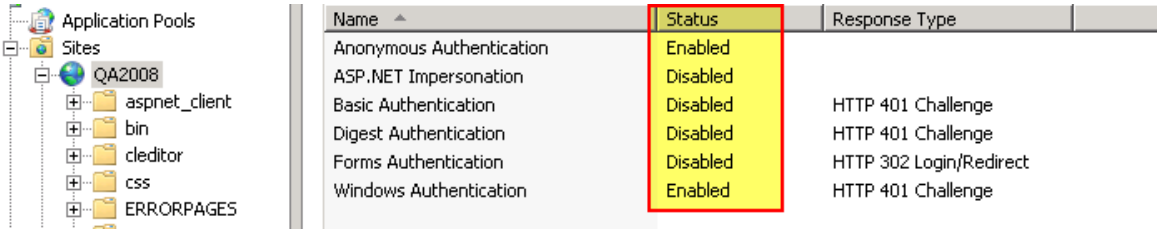
Configuring IIS for AD Integration

If it is desired to have Active Directory used as the authentication method, you will need to configure IIS to support NTML.

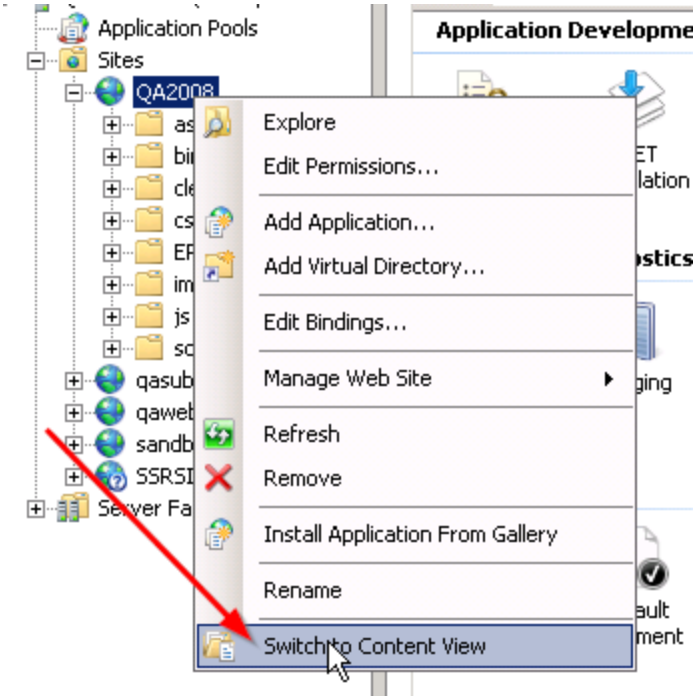
First, you will need to verify that the default IIS authentication methods are in place for your iService website. In the IIS Manager, select the relevant iService website in the left column, then double-click on the Authentication icon in the main window as shown below.



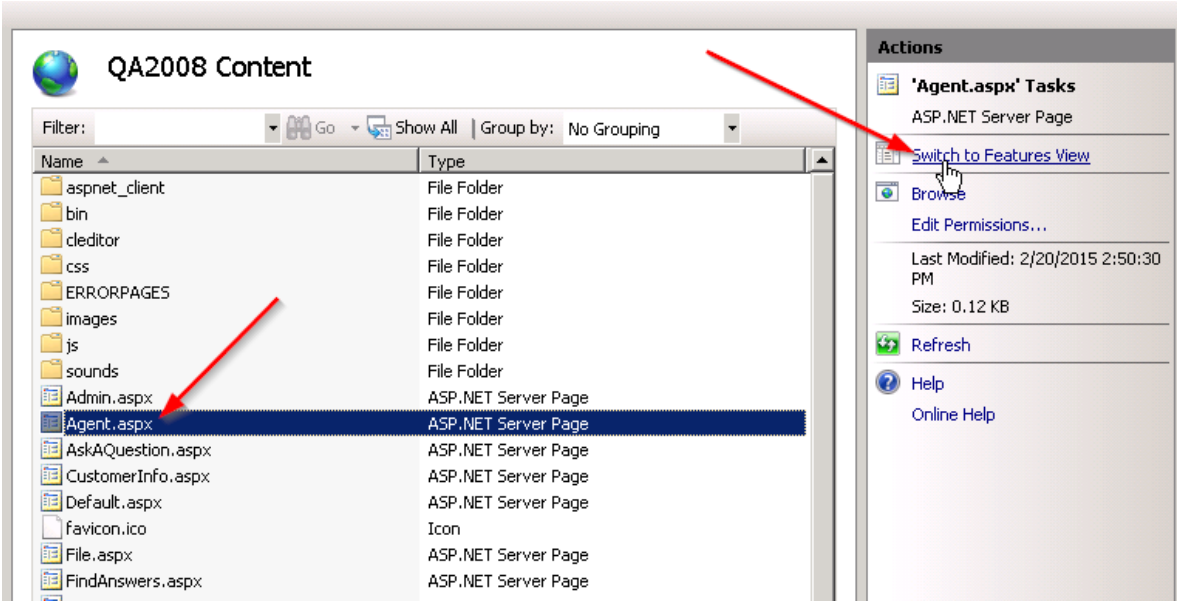
The currently enabled authentication methods will be listed. The methods shown match below.



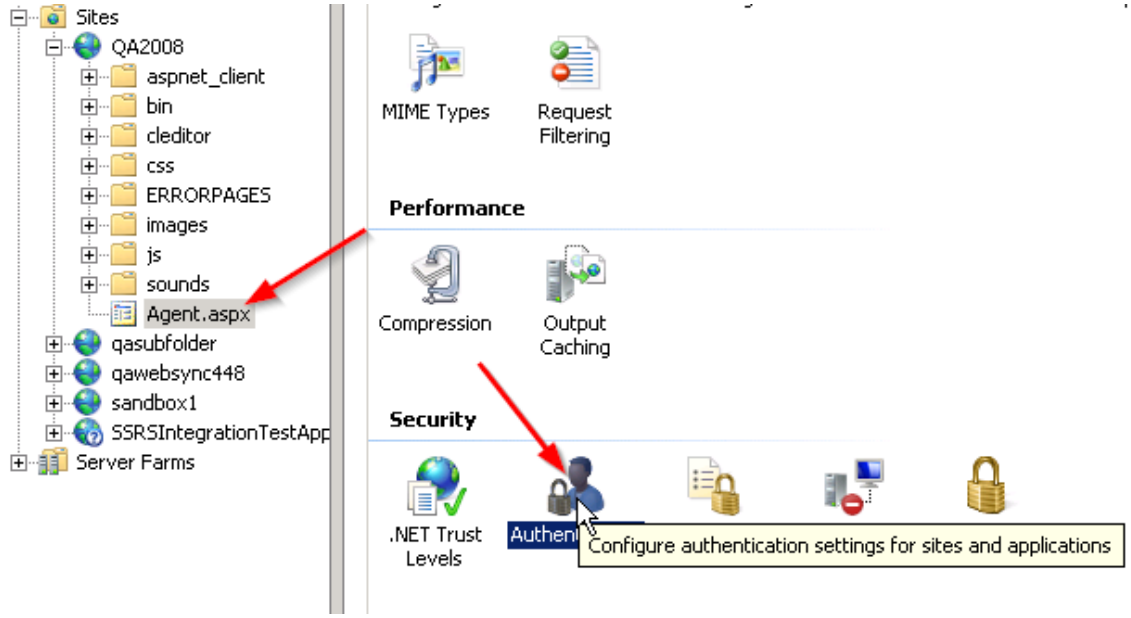
Next, we need to specifically set NTLM-only authentication for the Agent.aspx page. To do this, right click on the relevant iService website (shown as QA2008 below) and select “Switch to Content View”



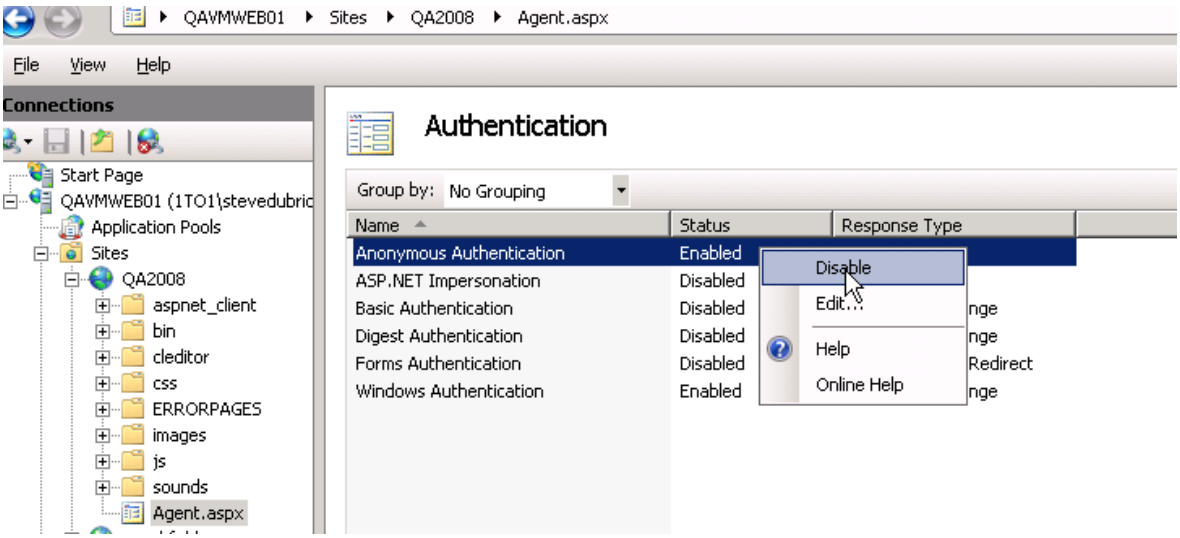
This will list the file contents of the website. Highlight the Agent.aspx file, and then click on the "Switch to Features View" as shown below.



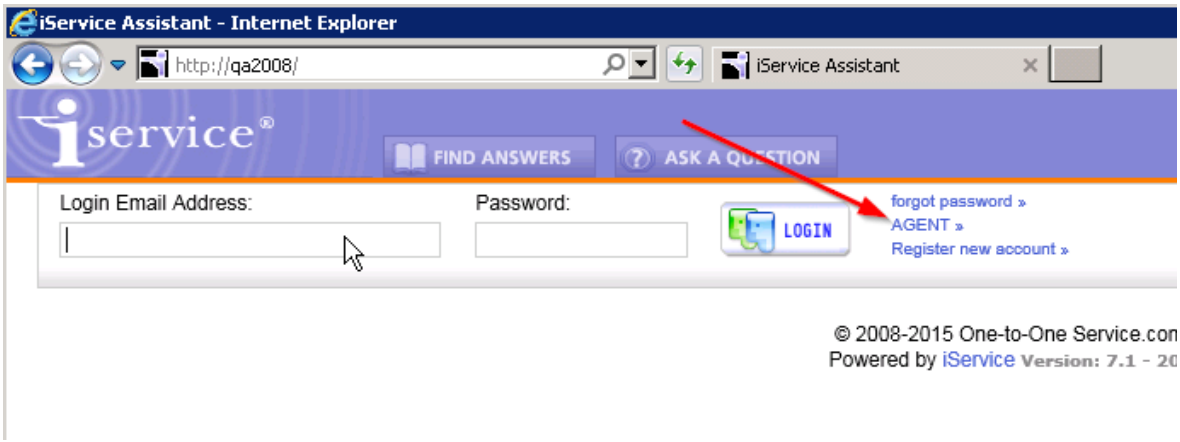
The Agent.aspx page will now be listed in the left panel under the iService website hierarchy. While it remains highlighted, again double click on the Authentication icon in the main panel as shown below.



In the authentication method list, disable "Anonymous Authentication" for this Agent.aspx file as shown below.

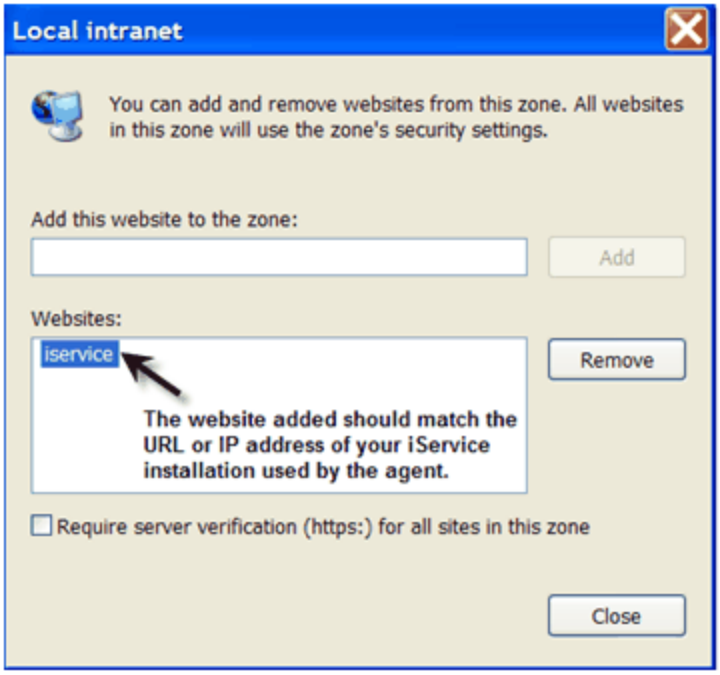


Now when visiting an iService page, you will have a new "AGENT" link on the login panel. If the iService agent has been linked to an AD login in the Admin panel, they will automatically be logged in when clicking this link or directly going to http://yourdomain.com/Agent.aspx.

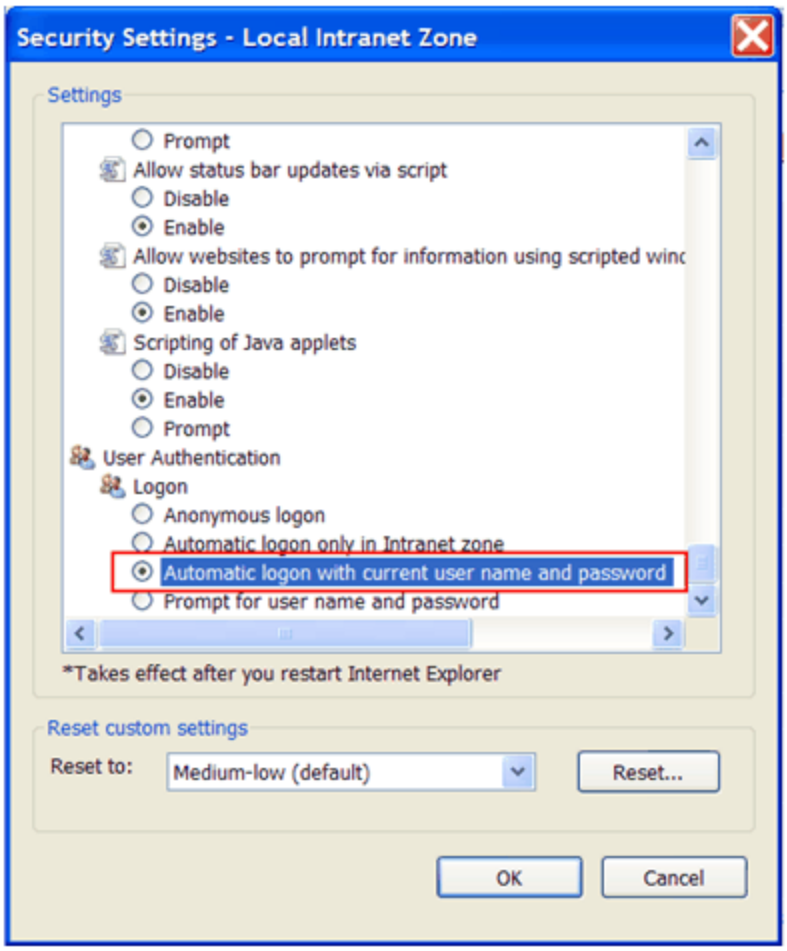


Browser Configuration - Internet Explorer

Go to Tools->Internet Options, then the Security Tab, Select "Local Intranet" and click "Sites" then click "Advanced" and add the site " http://yourdomain" and click "Close" then "OK".

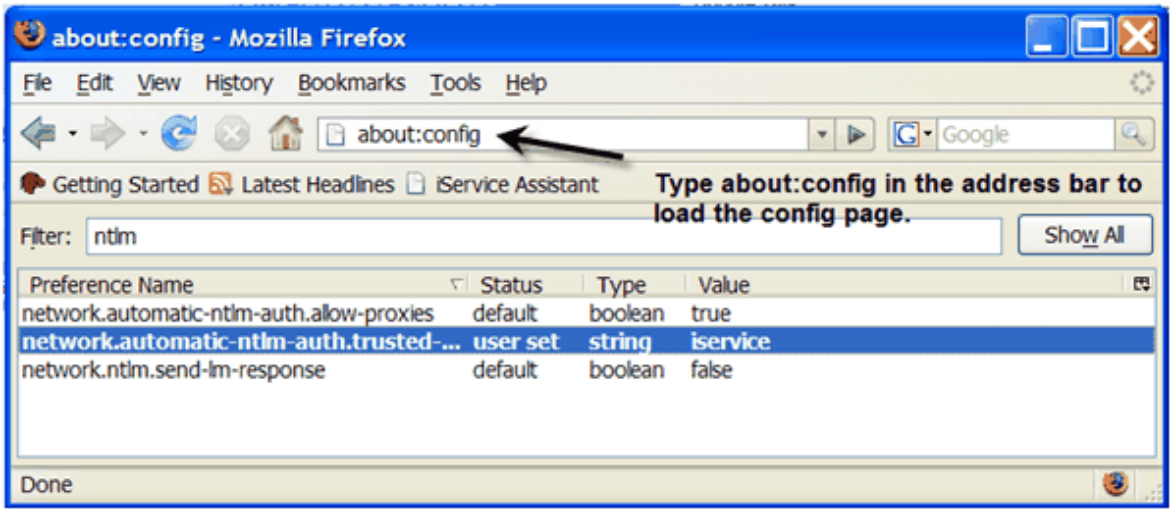


Then in the Security tab, with "Local Intranet" still selected, click "Custom Level" and at the very bottom of the list, choose "Automatic logon with current username and password", then click "OK" and "Yes" until changes are saved.



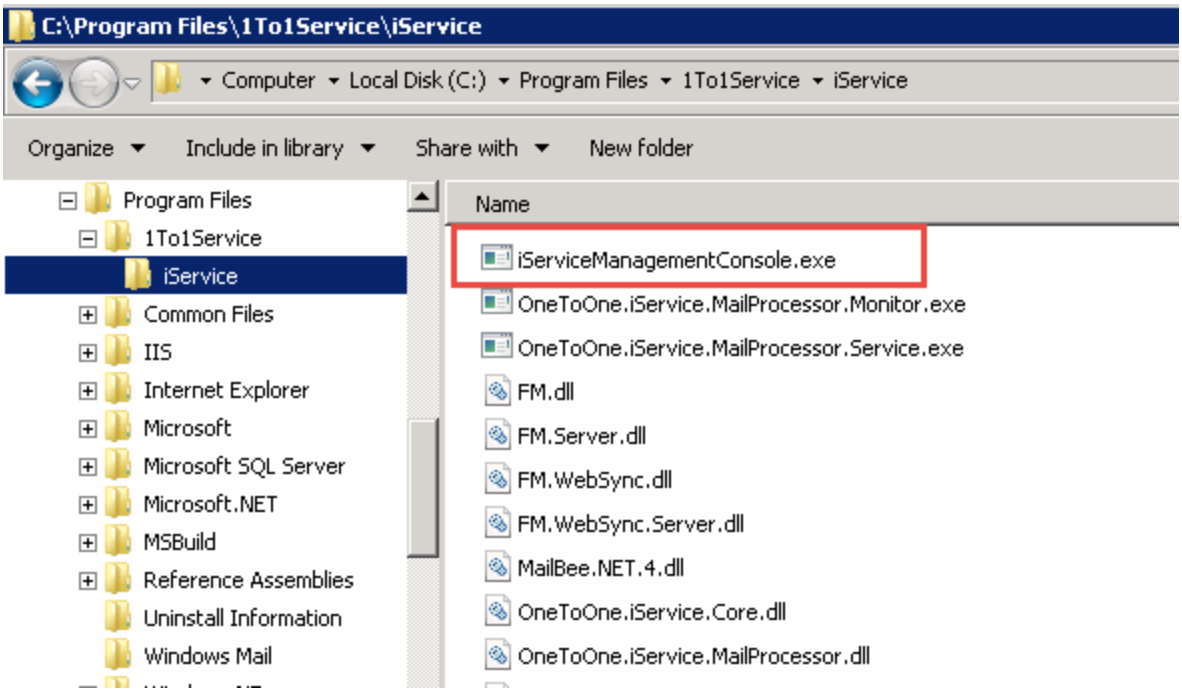
Browser Configuration - Firefox

Go to "about:config", filter for "ntlm" and in the " network.automatic-ntlm-auth.trusted-uris", enter a value for the iService website or address (or add that to whatever is already there with comma separator).



3.3 Create Initial Database

By default, the iService Windows service, management console, and mailpopper monitor is installed to c:\Program Files\1to1service\iService. Inside this folder you will find the Management Console application used to create your initial master database and tenants. Launch the iServiceManagementConsole to begin the process for creating these databases.



Usage of the iService Management Console is described within the iService Management Console User guide. Please review this user guide when you are ready to create your new master database and iService tenants.

4 Upgrading iService After Initial Installation

Beginning with release 7.6, iService is shipped with a new installer and slightly different upgrade procedure.

4.1 iService 7.8 Prerequisites

Duplicate Logins (Very Important Prerequisite)

Changes were made to the iService database to prevent duplicate logins within release 7.8. The database update process will fail when installing 7.8 if your database has any of these duplicate values (most common if you are using iService forms).

The following query should to be run against the tenant db before upgrading to version 7.8 to avoid potential issues, especially if the tenant uses a contact creation action in any form.

```
select count(name) as 'Occurrences', name from ContactsLogins group by name
having count(name) > 1
```

If any results are returned, this means there are contacts with conflicting email addresses. If no results are returned, there are no issues and the rest of this can be ignored.

For each row returned in the previous statement, note the email address and run:

```
select contactID from ContactsLogins where name = 'email address'
```

This will show the contact ID's that are conflicting. You'll need to change the email address of one or more of the conflicting contacts so that the email address is unique per contact. One way is to simply add a #7.8upgrade hashtag at the end:

```
update contactslogins set name = email address#7.8upgrade' where contactid
= 12345
```

iService Forms Within iFrames

In order to mitigate clickjacking attacks, version 7.8+ by default prevents iService HTML forms from being embedded as iframes on external sites unless explicitly allowed on a form by form basis. In order to allow a form to be embedded as an iFrame on an external site, the following value must be specified anywhere within the HTML of your form:

```
<!-- $header -allowframeall$ -->
```

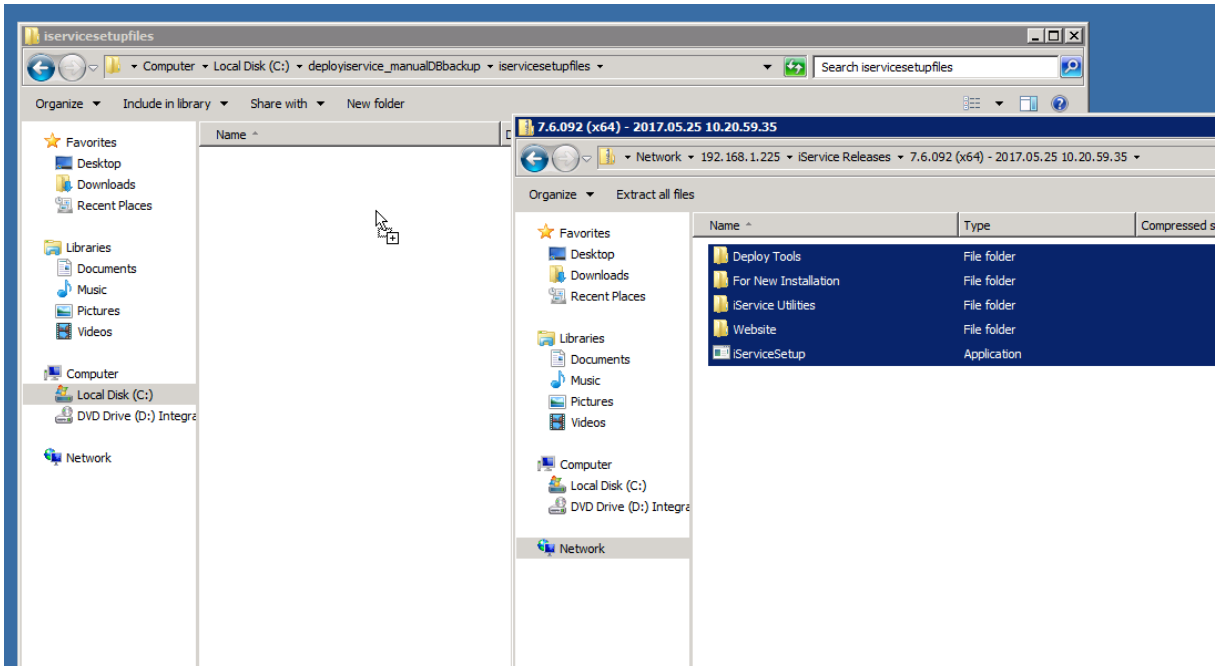
4.2 Upgrading to iService 7.6 Using Deploy Scripts

We strongly encourage the use of our automated deploy script to streamline the update process for iService. **The following description assumes you are using the deploy scripts to update to version 7.6.**

Always backup your iService databases before performing an upgrade.

Extract the zip contents

Open and copy the entire contents of the release .zip file to the "iservicesetupfiles" sub-folder located in your previous deploy script folder. Generally this is something like c:\deployiservice_manualDBbackup\iservicesetupfiles.



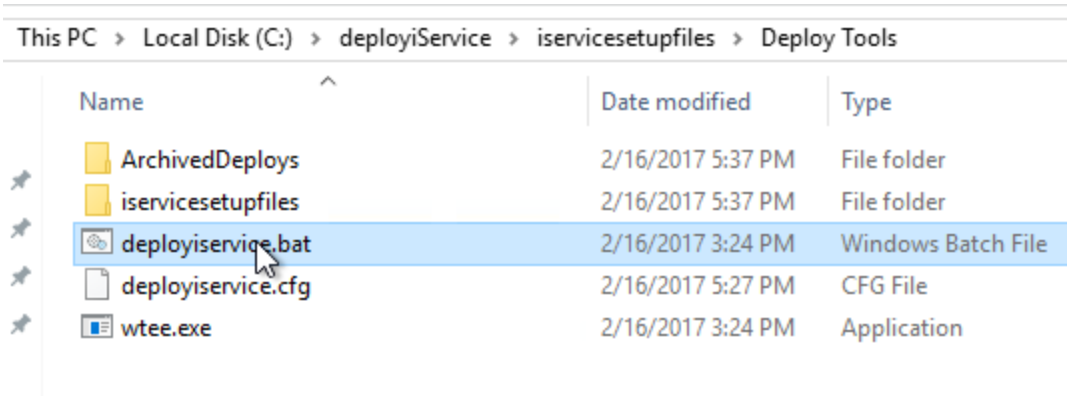
Click to expand

Copy the new .bat file (during update to 7.6 only)

***If you are upgrading to v7.6 for the first* time**, the "Deploy Tools" folder in the zip package includes a new deployiservice.bat file. This new version handles upgrading to iService versions 7.6 and higher. Copy this to your original deployiservice folder, replacing the old .bat file.

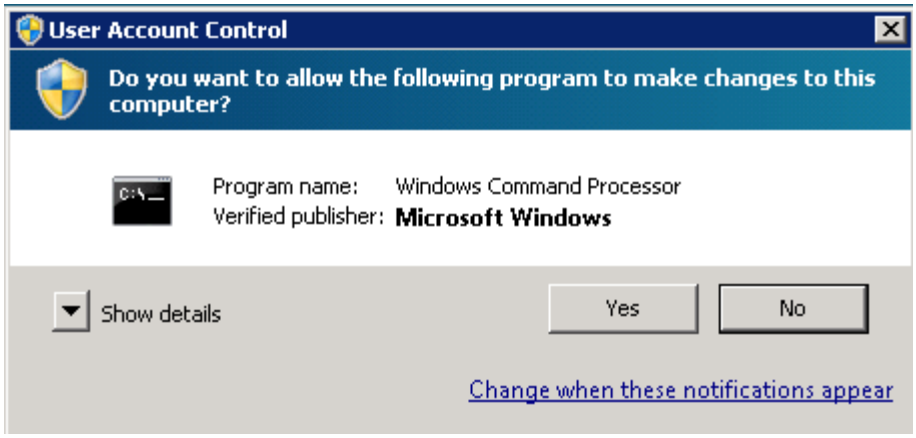
It's a good idea to keep a copy of the 7.5 and prior installation batch file in case you need to rollback because of an install issue.

*Note that this step is not necessary for subsequent upgrades.

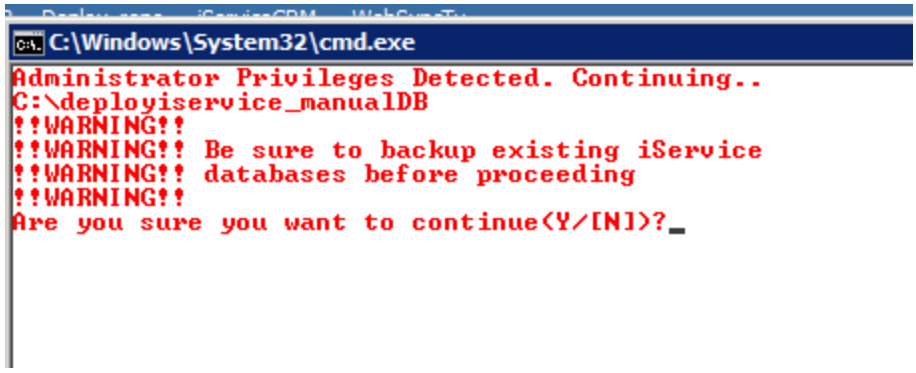


Run the deployiservice.bat

Right click and run the new deployiservice.bat file as Administrator.



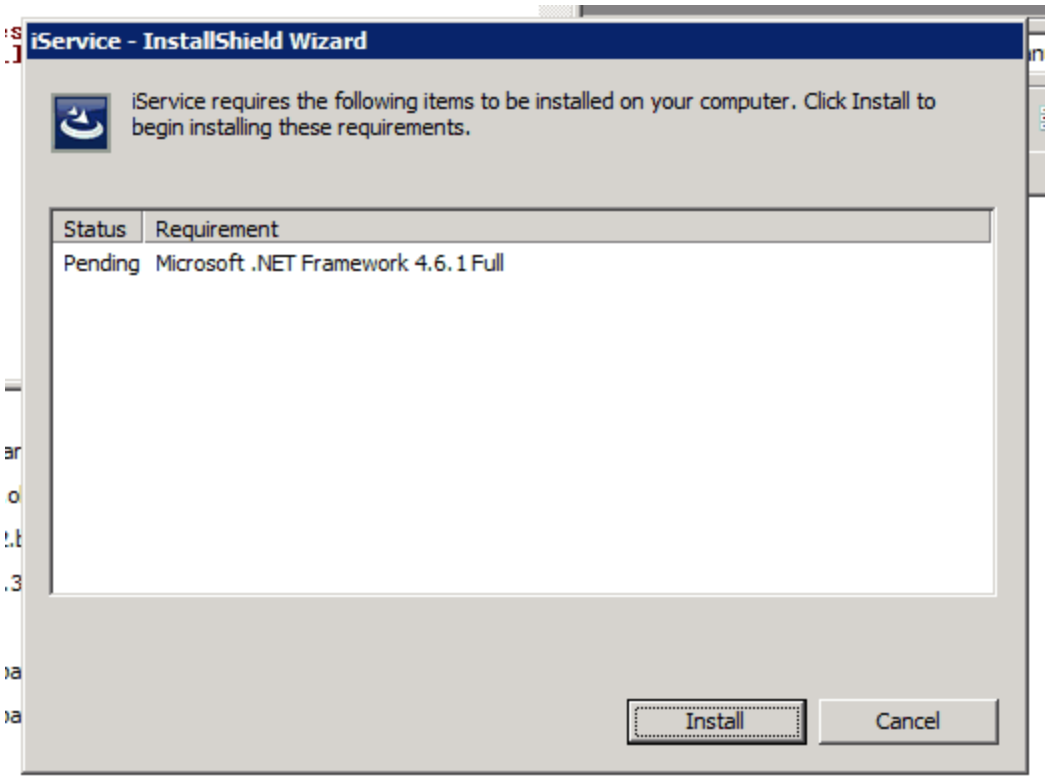
As a safety precaution. You will be reminded to backup your databases before continuing. Press Y and enter to continue.



Install .NET 4.6.1 (first install only)

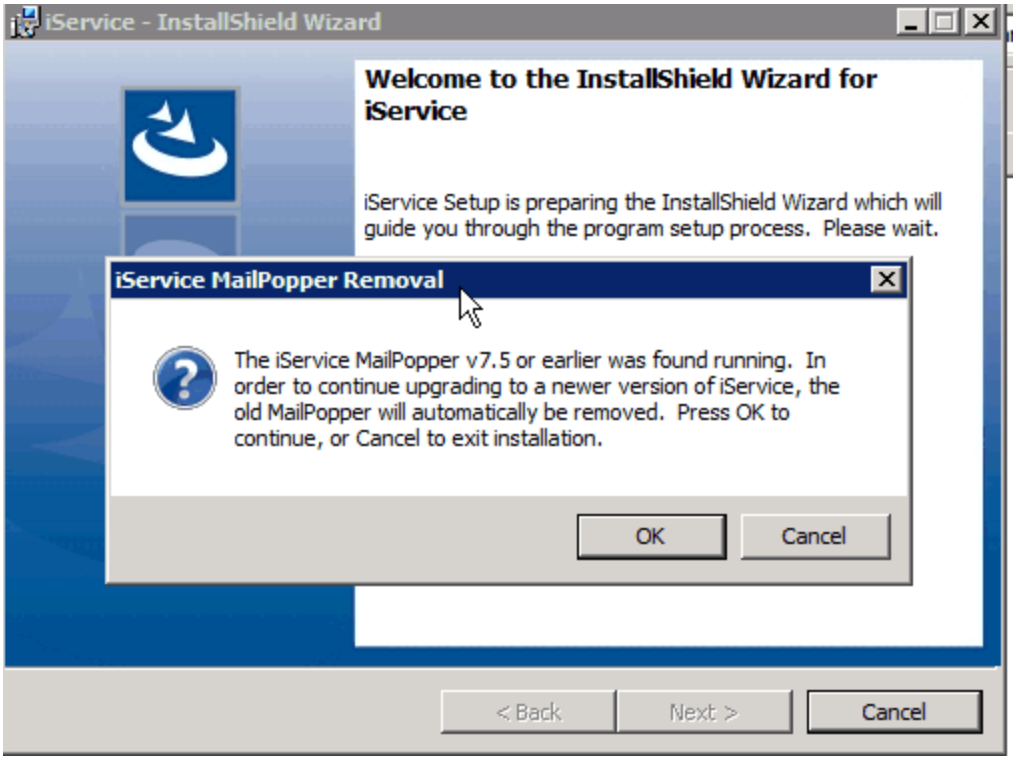
If you are upgrading to v7.6 for the first time and you haven't already installed the latest .NET runtime, you will be prompted to install v4.6.1. Click Install to continue. **This may take a while depending on your system.**

When you install a new version of the .NET Framework, you will need to reboot your server. Allow the iService installation to finish and then reboot your server.



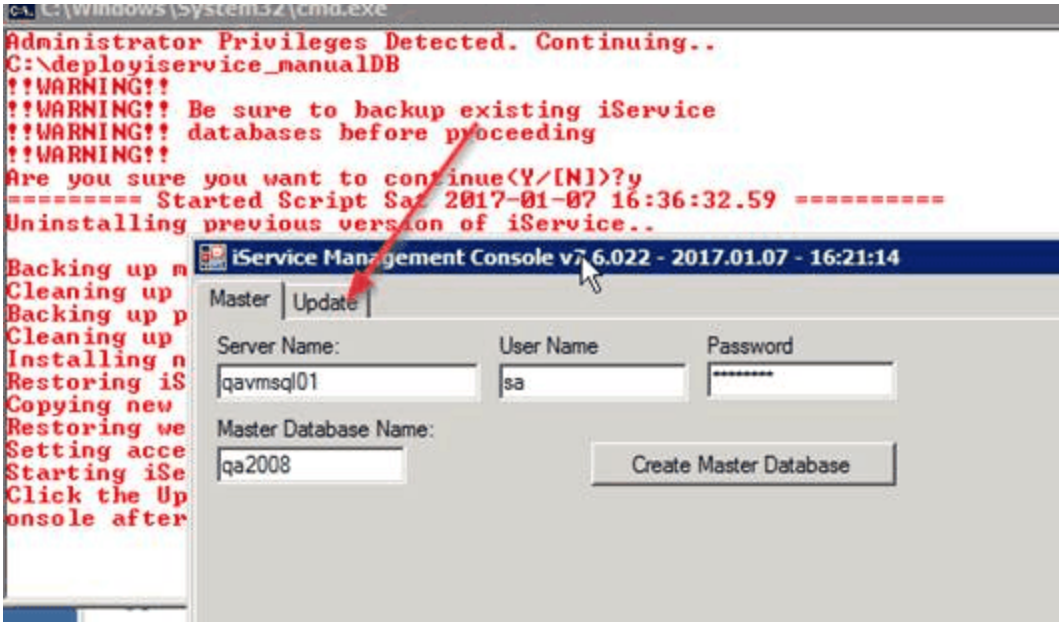
Allow removal of mailpopper service (during update to 7.6 only)

If you are upgrading to v7.6 for the first time, you will be prompted to remove the mailpopper service from the old version. Press OK to continue.

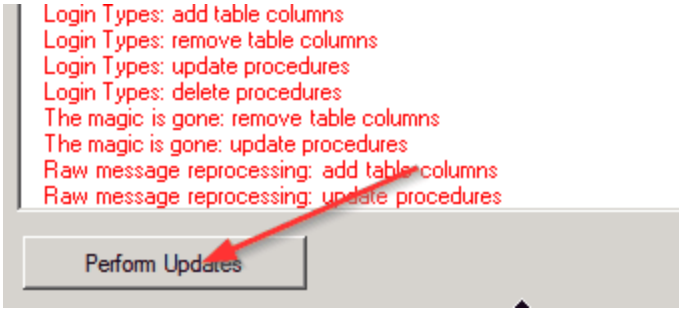


Upgrade database via the management console

Next, the management console will appear, which guides you through the database upgrade process. Press the "Update" tab.



If any database updates are needed, they will be listed in red text. Click the "Perform Updates" button to apply them:



After they are applied, close the Management Console.

Complete upgrade

When the deploy script says "Done. Press any key to continue..", the upgrade is finished and you can close the deploy script window.

4.3 Upgrading to iService 7.5 and Prior

iService 7.5 IMPORTANT NOTE:

There are rare conditions where interactions may be resolved while they are still assigned to an agent. For example, you may have an iService alert that resolves old interactions without him assigning them first from the agent. Prior to version 7.5, these scenarios were not noticeable but beginning with 7.5 these old interactions may appear in the agent's queue. To correct the situation you need to run a script against your database to remove the agent assignment.

```
--To check if there are 'bad' interactions:
select *
from dbo.Interactions as i inner join dbo.InteractionTypes as it on it.id = i.type in
where assignedToContactID IS NOT NULL
and s.name not in ('Queued','Pending','SentWaitingReply')
and it.name in ('Customer Email','Private Email','Ask A Question','Ticket')

--If there are any, use this to fix them:
UPDATE i
SET i.assignedtoContactID = NULL
from Interactions as i inner join InteractionTypes as it on it.id = i.type inner join
where assignedToContactID IS NOT NULL
and s.name not in ('Queued','Pending','SentWaitingReply')
and it.name in ('Customer Email','Private Email','Ask A Question','Ticket')
```

Windows Server 2008R2 IMPORTANT NOTE:

Versions 6.0 and later of iService require .NET 4. You should verify that your server running the mailpopper service has .NET 4 installed by viewing the Add/Remove Programs tab. If you are updating an existing site to .NET 4, you must change the App Pool used by the iService website to .NET 4.

On Windows 2008R2 servers we've encountered this error in IIS after installing .NET 4.0 and switching the app pool to use .NET 4.0:

```
Handler "PageHandlerFactory-Integrated" has a bad module "ManagedPipelineHand
```

This is a bug in the .NET installation, and if you get this error the easiest way to fix it is to run this at the command prompt:

```
%windir%\Microsoft.NET\Framework\v4.0.21006\aspnet_regiis.exe -i
```

As updates become available to the core iService code, you will be provided with an installation folder with a similar layout to the original installation. For updates, you can ignore the For New Installation folder.

As with any database intensive application, it is highly recommended that you update your iService installation during a maintenance window when no agent or customer is logged into the web site. This can be most easily accomplished by stopping the iService.Mailpopper service on the Applications and Services server and by stopping the web site on the Web Server.

After shutting down access to the site and database, perform a database backup of your master database and the tenant databases on your SQL Server. The master database will have the name chosen during your initial installation (iSA4 if you used the default) and each tenant will have a database name in the format of [master].[tenant] (for the example in these documents, the tenant database name would be iSA4.malcomdev).

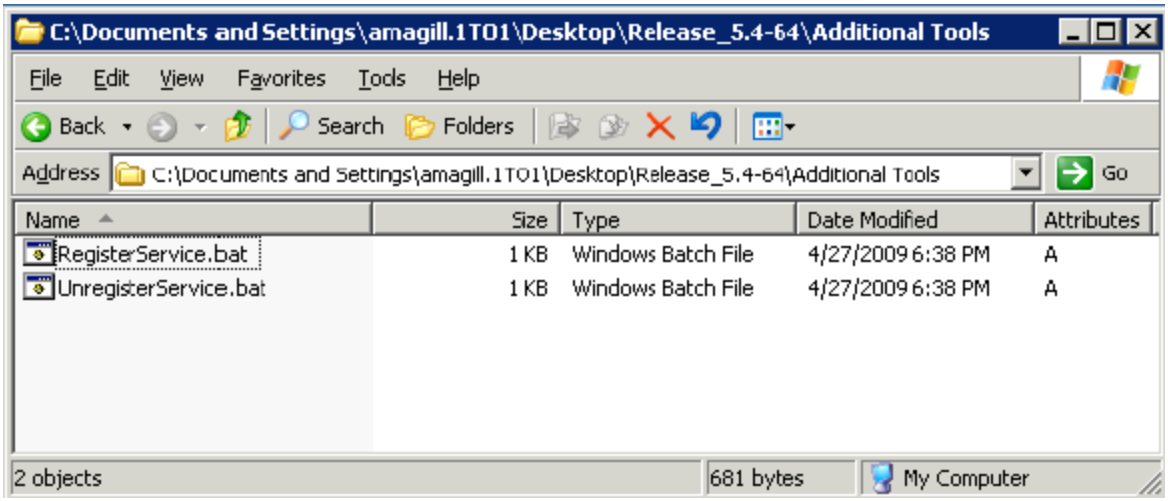
Making a backup folder of the web site directory (c:\Inetpub\iService on the web server) and the Applications folder (C:\Program Files\1to1Service\iService on the Applications and Services server) will aid in recovery in the event that the upgrade fails for any reason; this is less important that the databases, though, as these can also be restored from an earlier installation package – the data already in your databases cannot.

After creating the database backup files, you are ready to perform the update.

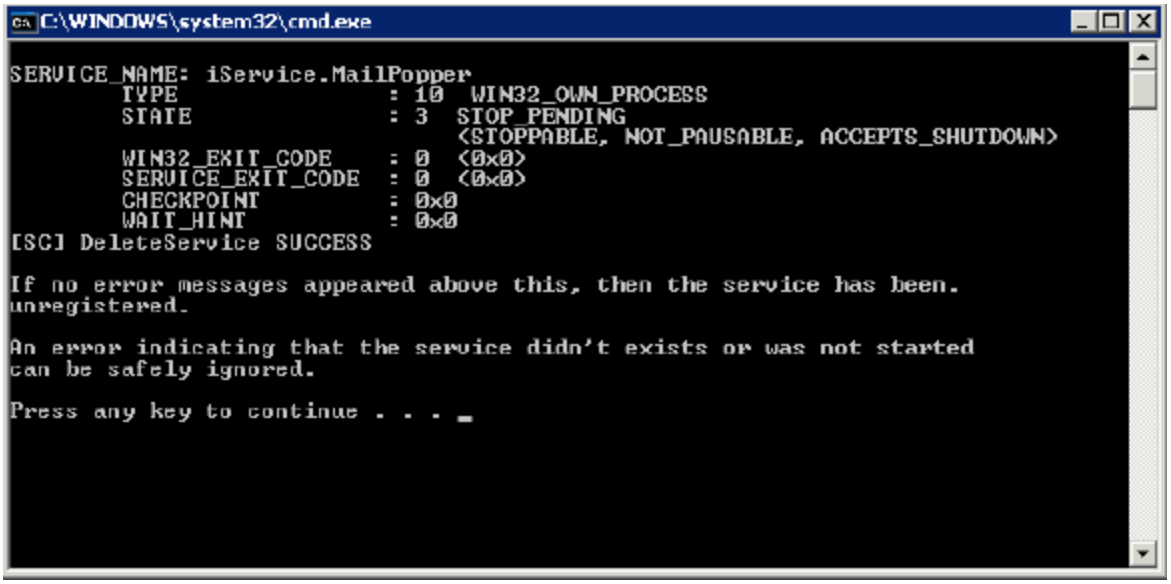
If you are updating the Mail Popper Services on a 64-bit server, you will need to take the following additional steps (these are not required for a 32-bit installation).

In the release archive, open the "Additional Tools" folder and locate the UnregisterService.bat file.

iService6 Note: If you are updating from iService5 to iService6, be sure to use the unregister batch file that came with the iService file installation files. you can also get it from the iService software download site.

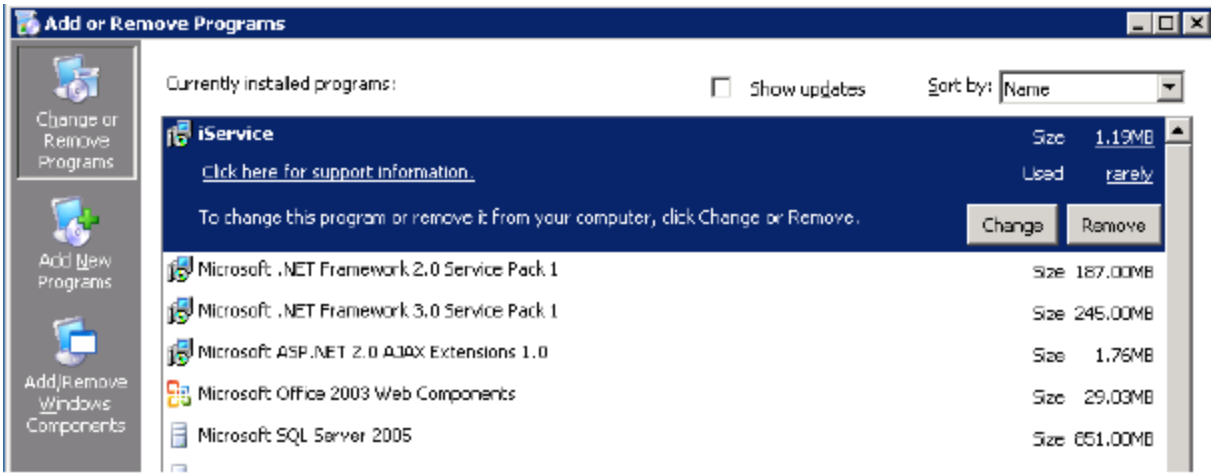


Double click on this file, and you should see a screen similar to the following:

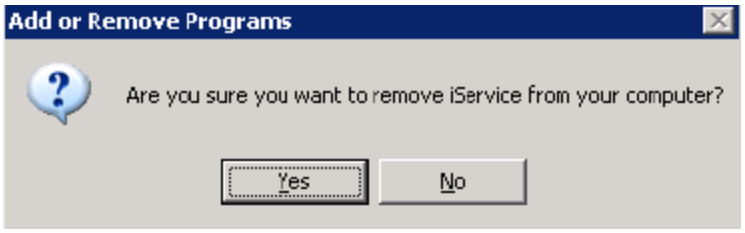


The information displayed above [SC] DeleteServices SUCCESS is caused by a command to first shut-down the service. If you have already stopped the service, you will instead see a message stating that the service is not running – this is fine and does not indicate an error.

Now, for both 32-bit and 64-bit installations, go to the "Add/Remove Programs" item in your Control Panels folder. Locate the iService application and click on the Remove button.

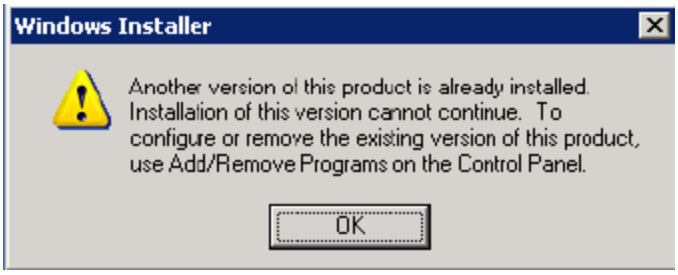


Click on Yes to confirm that you wish to remove the old code. Follow any instructions given until you are returned to the Add/Remove Programs window. Close this window.



After removing the old version, double click on the iService.MailPopper.Setup.msi or iService.MailPopper.Setup64.msi package in the new release folder provided and follow the installation instructions. Ensure that you choose the same installation directory as the previous installation, or you will have to copy/remake the mailpopper.config file as per the instructions in the initial installation section.

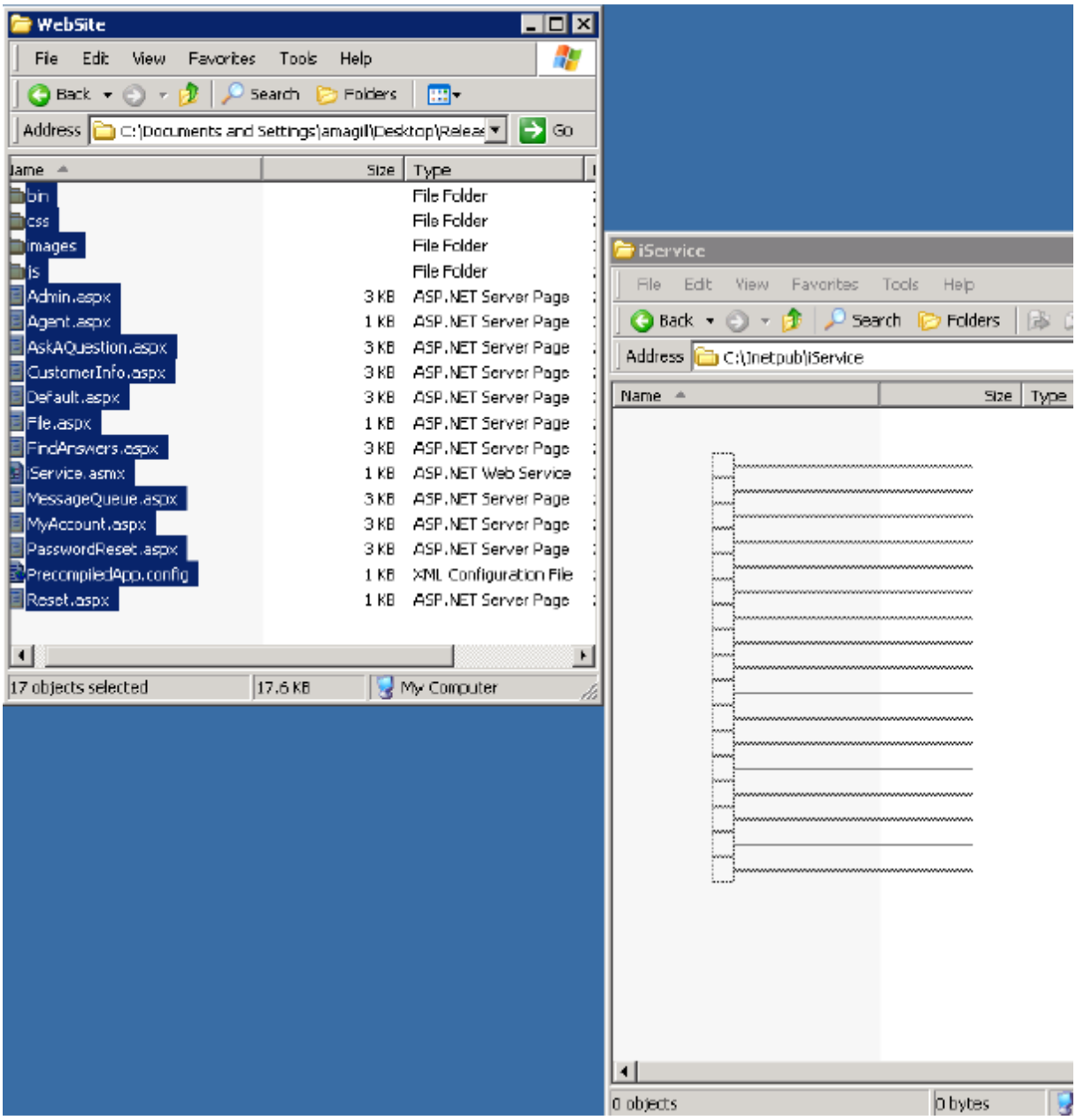
Note that if you fail to uninstall the iService Applications and Services before installation, you will be given the following error:



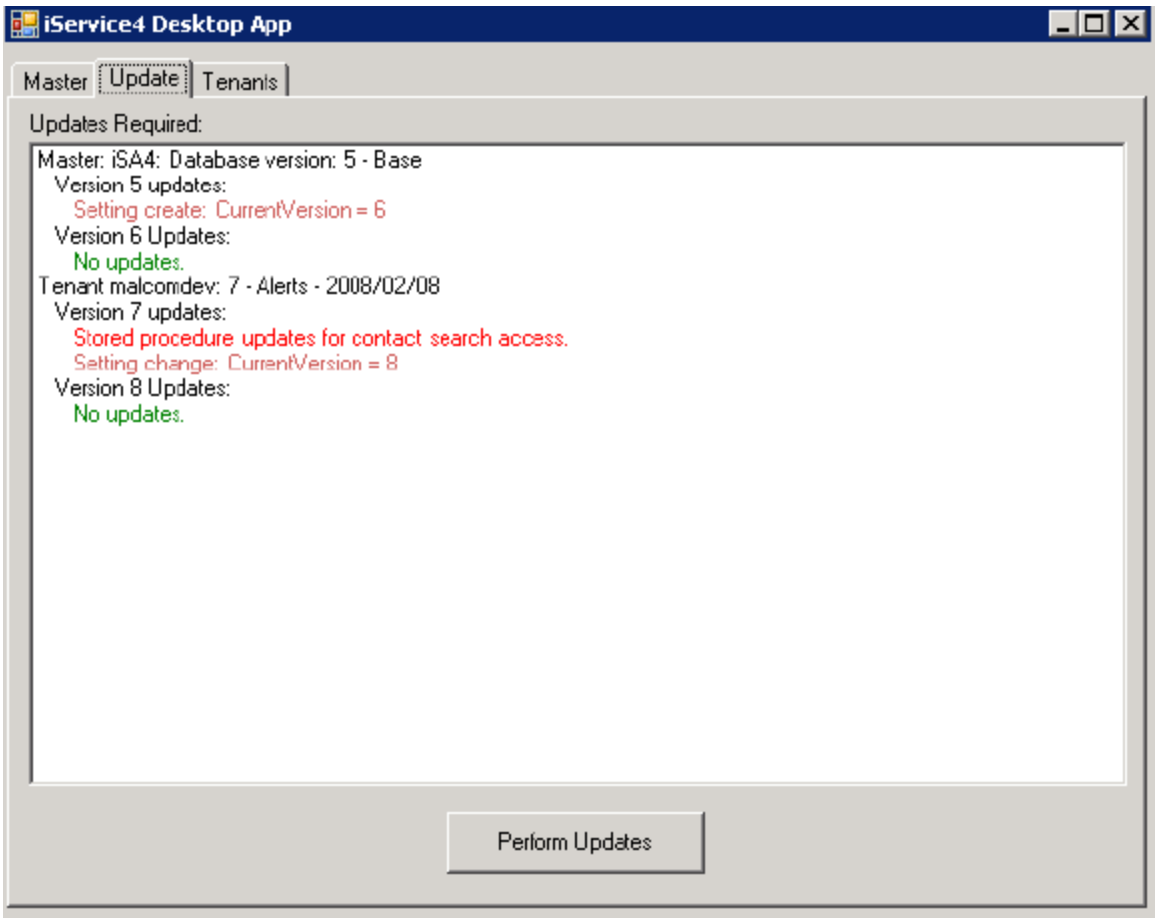
If this occurs, please go back to the Add/Remove Applications Control Panel and try again.

If you have copied the Desktop Application to another workstation for use by administrators, make sure that they know a new version is available and they will need to re-copy the application from the server.

To update the web site, on your web server, copy all of the files from the new releases Website folder into the installation folder for the iService website (C:\Inetpub\iService). Click Yes To All when prompted about overwriting files.

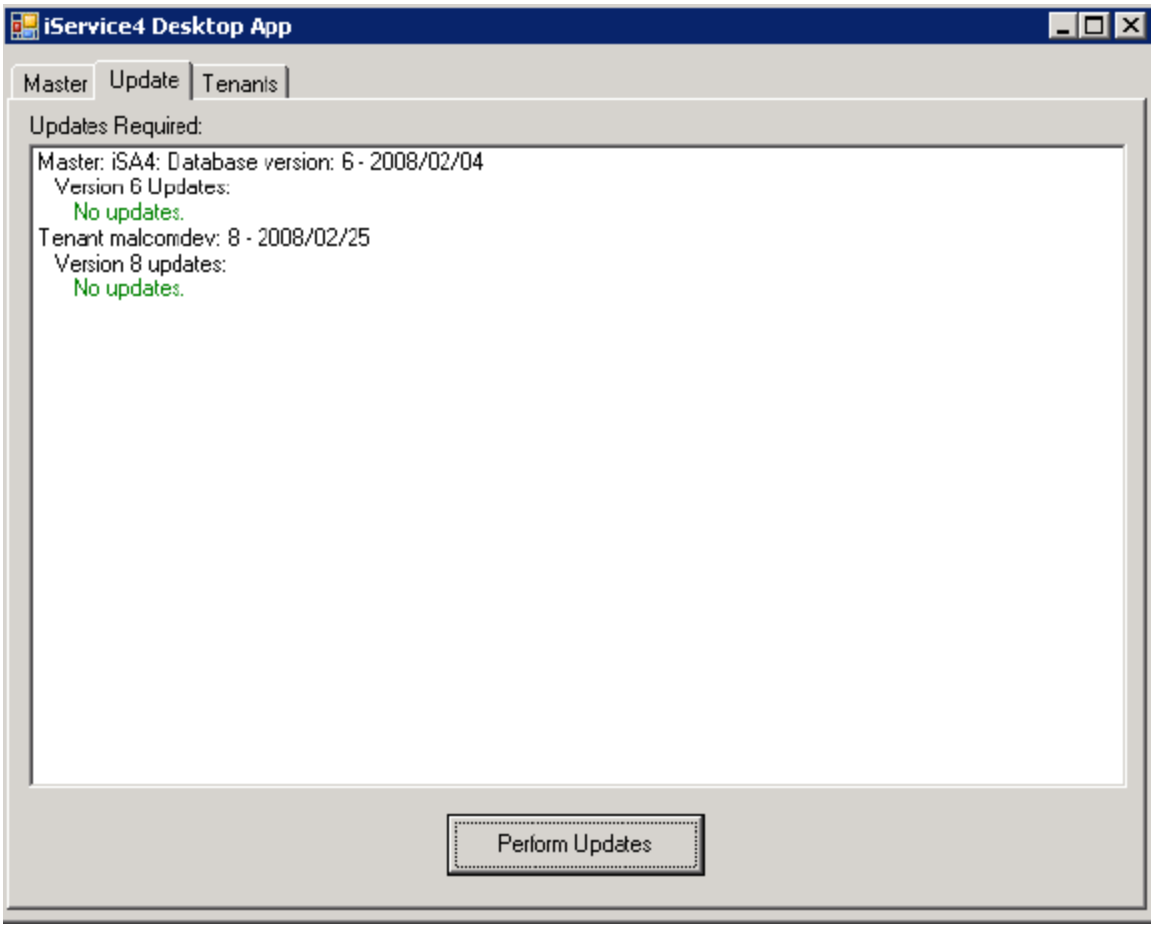


To update the databases themselves, run the new Desktop Application. Verify that the logon information in the Master tab is correct and click on the Update tab.



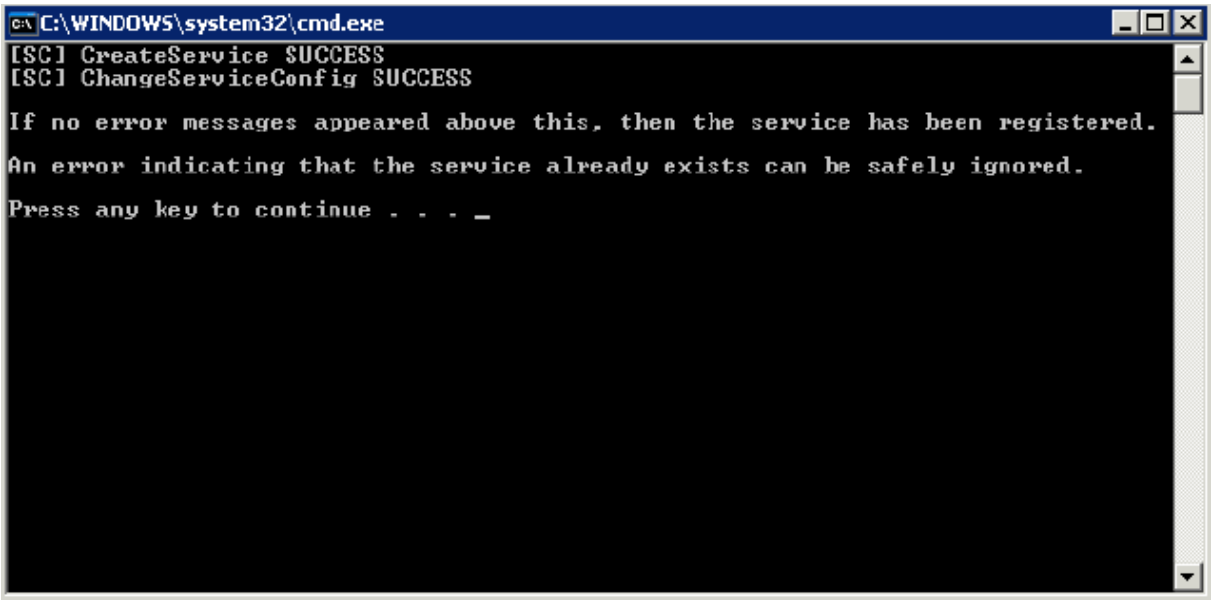
You will see a display similar to this, though your tenants and number of tenants may differ. Lines in any color other than Green indicate that database updates are required. Click on the Perform Updates button.

After the update is completed, your display should change to something similar to the following:



If you are running the updated Mail Popper on a 64-bit server, you will now need to re-register the service using the RegisterService.bat file located in the Additional Tools folder in the 64-bit archive. This step is not required if the mail popper is running on a 32-bit server.

Double click on this file to complete installing the service definition into the registry. You should see the following results:



The Mail Processor/Mail Popper application is now configured and ready for use. You may now **re-start the mailpopper service** and web site to bring the updated Web Application back on-line.

NOTE: Your email will be queued on the mail server until you restart the mail popper service. However, you must manually restart the mail popper service after you complete the update.

4.3.1 Upgrade Checklist for 7.5 and Prior

Below is a simple checklist that summarizes the steps described in the full Upgrading iService section. This is useful if you are already familiar with the iService Upgrade process and just need a short list as a reminder.

NOTE: We suggest always comparing your web.config file in user prior to an upgrade with the web.config file in the For New Installations folder of the release .zip. From time to time, the web.config will change and you will need to either copy the changes into your existing file, or use the standard web.config and update with your connection details.

1. Stop /Unregister service with bat file from current running version
2. Uninstall iservice
3. delete all (execept webconfig) from website dir (NOTE: for 6.0 you will also remove web.config since its changed)
4. delete all (except iService.MailPopper.config) from mailpopper dir (no change to the mailpopper config, so it does still remain)
5. install iservice from .msi
6. Copy website files to website directory (this now has the new web.config)
7. register service with bat file from the upgrade zip (ensure the path specified is correct)
8. start iservice service in services.msc
9. Run management console and perform any updates necessary

10. Run mailpopper viewer to confirm service is running
11. Login to site and confirm successful deploy

5 Maintenance Tasks

From time to time, there are various maintenance tasks you need to perform on your iService installation to keep it optimized. Most of these relate to your database.

5.1 Purging Interactions

Interactions may need to be deleted in an iService tenant. For example, in some cases sensitive data like credit card numbers must be removed for compliance reasons. Or, your database might grow large and require you to purge interaction data to reduce its size.

iService contains a stored procedure to delete interactions, but it cannot be accessed from the web interface and must be run by a system administrator or DBA with access to execute the iServiceInteractionsDelete stored procedure in the tenant database.

Notes:

- Deleting an Interaction is unrecoverable and removes all relational data connected to it, including:
  - Audit trails of the interactions status and properties
  - If interaction was sourced from an email, all related raw email and post-processed email data
- Other interactions in the same thread are NOT deleted. For example, if a customer emails in sensitive data and an agent replied to them with the original email quoted, both the customer email AND the agent reply must be deleted seperately.
- If the parent of a child interaction is deleted, the child's grandparent interaction will be inherited as its parent, recursively. If the parent of a child interaction was the thread root, the child interaction and its siblings will all become thread roots.

The procedure simply takes one argument: the interaction ID

Example:
EXECUTE [dbo].[iServiceInteractionsDelete] 123456


iService includes a stored procedure to mass deleted interactions, but you must first construct a query to derive the list of interactions to purge. For assistance with performing a mass delete on your database please contact One-to-One Service.com at support@1to1service.com.

5.2 Moving Raw Messages Table

Background

Most of the data storage in iService databases tends to be concentrated in 3 LOB (Large Object) tables:

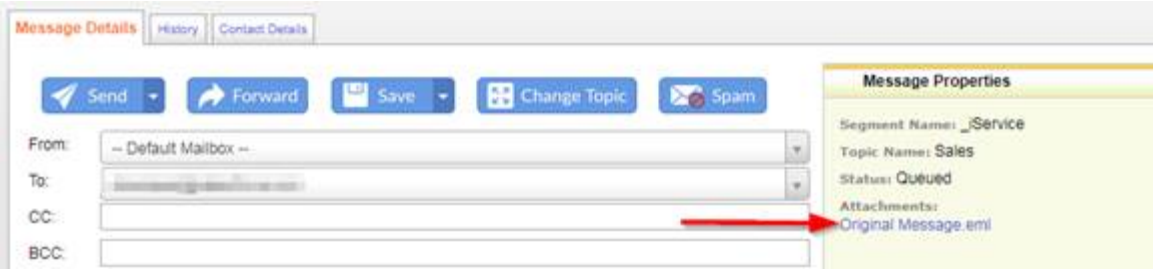
- FilesRawMessages
- FilesText

- Files

For example, it's not unusual to see table sizes such as this list of the largest tables in an iService tenant. In a tenant with a large number of incoming emails, we've seen these 3 LOB tables account for over 90% of a 163GB database:

Table Name	Size (GB)
FilesRawMessages	104.61
FilesText	24.98
Files	18.23
InteractionsStatusAudit	0.75
InteractionsEmails	0.35
Interactions	0.28
InteractionsAttachments	0.18
InteractionsCompositions	0.10
SmtpOut	0.08
ContactsPropertiesAudit	0.08

The largest, FilesRawMessages, contains data which is infrequently accessed. It is used to store raw emails as they come into the tenant before Customer Email interaction types are generated from the raw email. After this initial action, they are seldom, if ever, accessed again. However, they do remain in the database for archival purposes and can be retrieved when an iService agent clicks on a Customer Email's "Original Message.eml" link:



The combination of this table's large size and infrequent access make it a good candidate for moving to a lower tier (cheaper/slower) storage solution than the rest of the tenant. This guide demonstrates how to move the data in the FilesRawMessages on a live tenant to a filegroup located on lower tier storage.

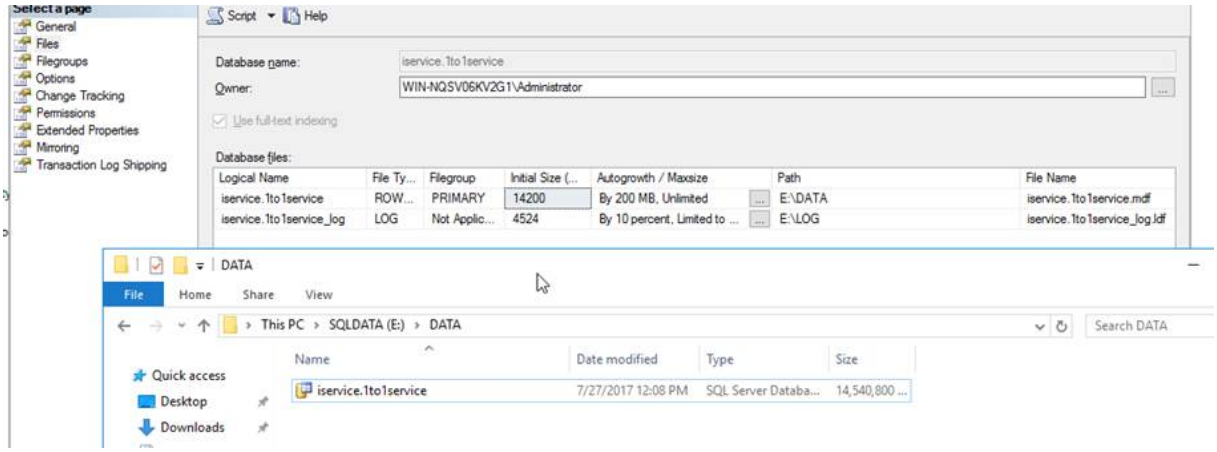
Assumptions, Prerequisites, and Caveats

- This guide was tested with iService 7.6 and will work with versions 6.0 until 7.6. While it may work, it has not been tested with later versions.

- This will work with SQL Server 2008r2 to SQL Server 2016 It may work with later versions, but has not been tested. In all versions of SQL Server, this must be performed with Raw Messages unavailable (offline) because although 2012 and higher support online LOB index rebuilds, they cannot be performed on ntext() columns.
- In highly-available systems where 1. mail processing cannot be offline for the time it takes to rebuild and 2. Old interactions are purged on a regular schedule, we suggest simply doing a standard table partition at the current FilesRawMessages row ID and skip performing the index rebuilds which move the old data. Over time, new Raw Messages will enter into the new partition's filegroup, while old data is purged out of the old partition.
- SQL Server Enterprise Edition is required, which allows table partitioning.
- This guide will focus on moving the FilesRawMessages table, but the same process applies for any of the LOB tables.
- This process may take a very long time and can be very IO intensive. It's recommended to be done during low-usage periods.
- Your database's log file may grow since there are probably many more transactions occurring during this operation than normal. You may want to note the initial size, and shrink the log back to that size when you're finished with the process.
- Thanks to SQL Mag for coming up with the original process this is based on.

Guide

In our example, the original iService filegroup is located on the E: drive, and we'll be moving FilesRawMessages to the lower-tier F: drive. Our tenant database is around 14.5 GB:



A query for the FilesRawMessages clustered index size tells us that about 4.5GB of space is allocated to it:

```
USE [iservice.1to1service]
GO
SELECT i.[name] AS IndexName
      ,Sum(s.[used_page_count]) * 8 AS IndexSizeKB
FROM sys.dm_db_partition_stats AS s
INNER JOIN sys.indexes AS i ON s.[object_id] = i.[object_id]
      AND s.[index_id] = i.[index_id]
WHERE i.NAME = 'PK_FilesRawMessages'
GROUP BY i.[name]
ORDER BY i.[name]
GO
```

Results | Messages

	IndexName	IndexSizeKB
1	PK_FilesRawMessages	4438624

First, create a new filegroup in the tenant database. We'll call it LOBdata.

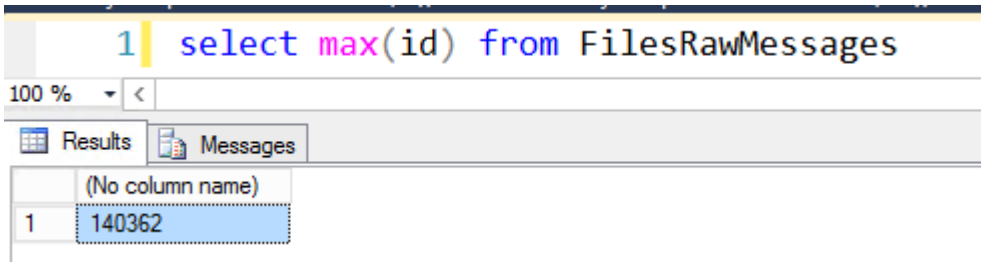
```
USE [master]
GO
ALTER DATABASE [iservice.1to1service] ADD FILEGROUP [LOBdata]
GO
```

Add a new file located on the F: drive, inside the new LOBdata filegroup. Earlier we noted that FilesRawMessages was 4.5GB, so we're going to make the initial size 5120000KB to hold it all.

```
USE [master]
GO
ALTER DATABASE [iservice.1to1service] ADD FILE (
      NAME = N'iservice.1to1service_LOBdata'
      ,FILENAME = N'F:\DATA\iservice.1to1service_lobdata.ndf'
      ,SIZE = 5120000 KB
      ,FILEGROWTH = 204800 KB
      ) TO FILEGROUP [LOBdata]
GO
```

Next, we're going to partition the FilesRawMessages table, with the first partition residing in the LOBdata filegroup and actually containing all the rows in the table. The second partition will contain no rows and will reside in the original Primary filegroup. This will allow us to later re-index the table, moving the data to the LOBdata filegroup while keeping the database online.

We need to find what the highest row ID currently is in the FilesRawMessages table so that we can create a RIGHT partition range much higher than it, so that any new rows created during the move also are within range:



Our highest ID is 140362, so a right range of 200000 should be enough. Create the partition function:

```
USE [iservice.1to1service]
GO
CREATE PARTITION FUNCTION rawmessagemove (INT) AS RANGE RIGHT
FOR
VALUES (200000);
```

Then apply the partition to the FilesRawMessages table, with partition #1 mapped to LOBdata, and partition #2 mapped to Primary:

```
USE [iservice.1to1service]
GO
CREATE PARTITION SCHEME rawmessagemove AS PARTITION rawmessagemove TO (
    lobdata
    ,[primary]
    );
```

Next, we'll perform the actual move of the data, by dropping and recreating the indexes of FilesRawMessages onto the new filegroup. Let's get a list of the indexes we'll be working with:

```
USE [iservice.1to1service]
GO
SELECT dbschemas.[name] AS 'Schema'
    ,dbtables.[name] AS 'Table'
    ,dbindexes.[name] AS 'Index'
    ,indexstats.alloc_unit_type_desc
    ,indexstats.avg_fragmentation_in_percent
    ,indexstats.page_count
FROM sys.dm_db_index_physical_stats(DB_ID(), NULL, NULL, NULL, NULL) AS indexstats
INNER JOIN sys.tables dbtables ON dbtables.[object_id] = indexstats.[object_id]
INNER JOIN sys.schemas dbschemas ON dbtables.[schema_id] = dbschemas.[schema_id]
INNER JOIN sys.indexes AS dbindexes ON dbindexes.[object_id] = indexstats.[object_id]
    AND indexstats.index_id = dbindexes.index_id
WHERE indexstats.database_id = DB_ID()
    AND dbtables.[name] = 'FilesRawMessages'
ORDER BY indexstats.avg_fragmentation_in_percent DESC
```

	Schema	Table	Index	alloc_unit_type_desc	avg_fragmentation_in_percent	page_count
1	dbo	FilesRawMessages	UQ__FilesRaw__43B15FA02EF4C8F4	IN_ROW_DATA	1.69753086419753	648
2	dbo	FilesRawMessages	PK_FilesRawMessages	IN_ROW_DATA	0.701402805611222	998
3	dbo	FilesRawMessages	PK_FilesRawMessages	LOB_DATA	0	554081

There are 3 indexes we'll need to move - the row data clustered index, the LOB data clustered index, and the unique unclustered index. As an aside, you might want to take note of the fragmentation shown, and compare it to the fragmentation after the move is complete.

First we'll drop and recreate the row data clustered index on the new partition (this may take a few minutes, and mail won't be processed during this time if doing it in OFFLINE mode):

```
USE [iservice.1to1service]
GO
CREATE UNIQUE CLUSTERED INDEX PK_FilesRawMessages ON dbo.FilesRawMessages ([ID] ASC)
    WITH (
            DROP_EXISTING = ON
            ,ONLINE = OFF
            ) ON rawmessagemove(ID)
```

Next we'll drop and recreate the lob data clustered index, on the new filegroup. This will move all the LOB data (this again may take a few minutes, and mail won't be processed during this time if doing it in OFFLINE mode):

```
USE [iservice.1to1service]
GO
CREATE UNIQUE CLUSTERED INDEX PK_FilesRawMessages ON dbo.FilesRawMessages ([ID] ASC)
    WITH (
            DROP_EXISTING = ON
            ,ONLINE = OFF
            ) ON [lobdata]
```

Finally, we'll do the same for the unclustered 'unique' index. Note that the name of the index will be different on your tenant. This should be quick compared to the other 2, and can be done ONLINE.

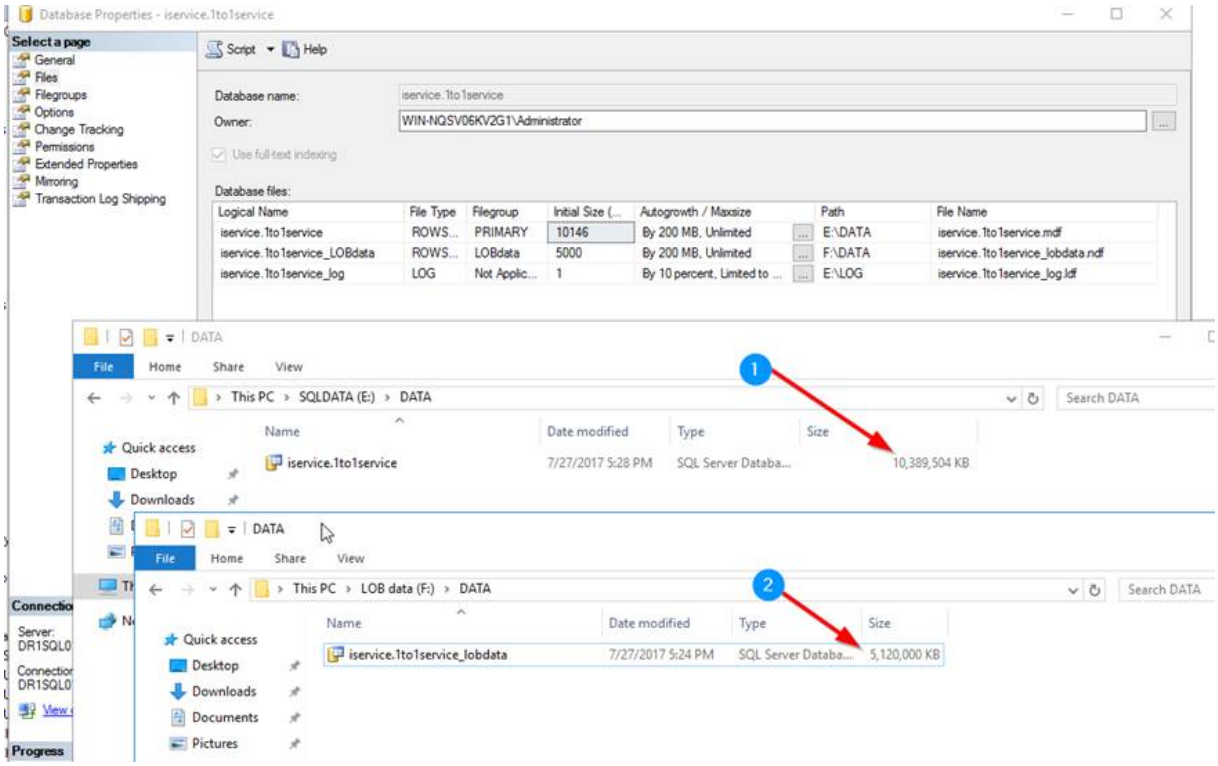
```
USE [iservice.1to1service]
GO
CREATE UNIQUE INDEX UQ__FilesRaw__43B15FA02EF4C8F4 ON dbo.FilesRawMessages ([hash] ASC)
    WITH (
            DROP_EXISTING = ON
            ,ONLINE = ON
            ) ON [lobdata]
```

That's it! Your FilesRawMessages LOB have been moved to the other storage device. Now that your LOB data has been moved, depending on your future expected growth patterns you may or may not want to shrink the database to free up the newly unallocated space.

Note that this is completely optional, and there are some very negative consequences for doing so, one of which is fragmenting your freshly created indexes.

In this example, we'll do a database shrink simply to demonstrate that the FilesRawMessages table was moved to the new storage.

1. Our original file in the primary filegroup on E: has been reduced from 14.5GB to around 10GB without FilesRawMessages.
2. The new file in the LOBdata filegroup on F: holds the 4.5GB of FilesRawMessages data.



6 Troubleshooting

This chapter describes some common issues we've seen for on-premise users.

6.1 Current Identify Does Not Have Write Access

When you attempt to access the web site, you receive the following in your browser window:

Server Error in '/' Application.

The current identity (NT AUTHORITY\NETWORK SERVICE) does not have write access to 'c:\WINDOWS\Microsoft.NET\Framework\v2.0.50727\Temporary ASP.NET Files'.

Description: An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code.

Exception Details: System.Web.HttpException: The current identity (NT AUTHORITY\NETWORK SERVICE) does not have write access to 'c:\WINDOWS\Microsoft.NET\Framework\v2.0.50727\Temporary ASP.NET Files'.

Source Error:

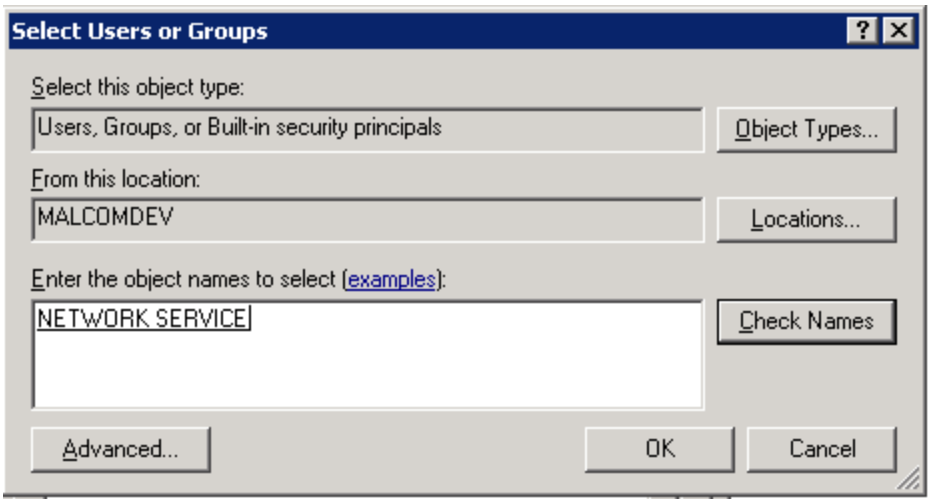
Click to expand

Note the specific identity (NT AUTHORITY\NETWORK SERVICE in this example) and the directory referenced. Depending upon how and when IIS and ASP.NET 2.0 were installed/added to your server, the expected rights may not be correct. To address this issue, open up the parent folder of the directory in question. In this case:

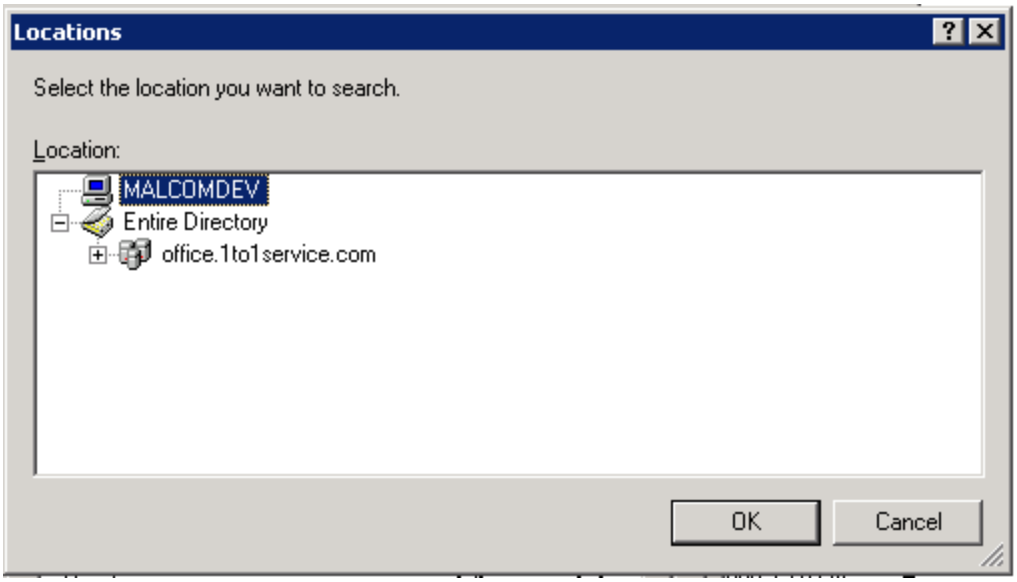
```
C:\WINDOWS\Microsoft.NET\Framework\v2.0.50727
```

Once you are there, select the folder “Temporary ASP.NET Files” and right click on it. Select Properties and then select the Security Tab.

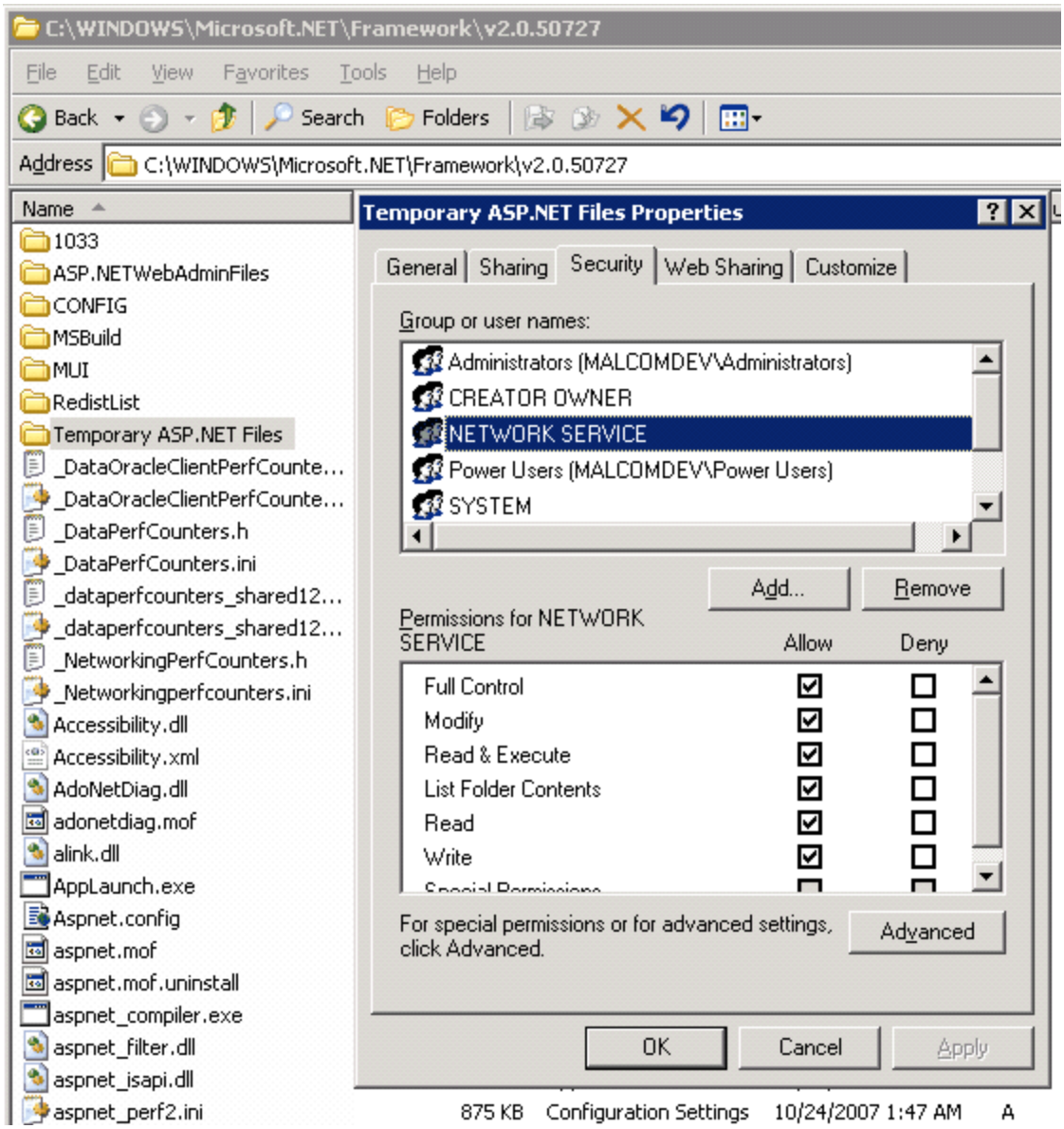
If the identity (NETWORK SERVICE) exists in the list, ensure that it is set to “Full Control”. If the identity does not exist, click on the Add button to add it.



If the web server is on your Windows Active Directory Domain, you will likely need to click on the Locations button and select the local machine, as the default search location is the Domain.



Once you've selected the correct location, type in NETWORK SERVICE and click on Check Names to ensure that it is properly identified. Click on OK to be returned to the Security Properties dialog. Select NETWORK SERVICE and check Full Control for this new entity.



Once you have done this, click OK to apply the changes.

6.2 Unable to Generate a Temporary Class

Server Error in '/' Application.

Unable to generate a temporary class (result=1).
error CS2001: Source file 'C:\WINDOWS\TEMP\f7ni3e2f.0.cs' could not be found
error CS2008: No inputs specified

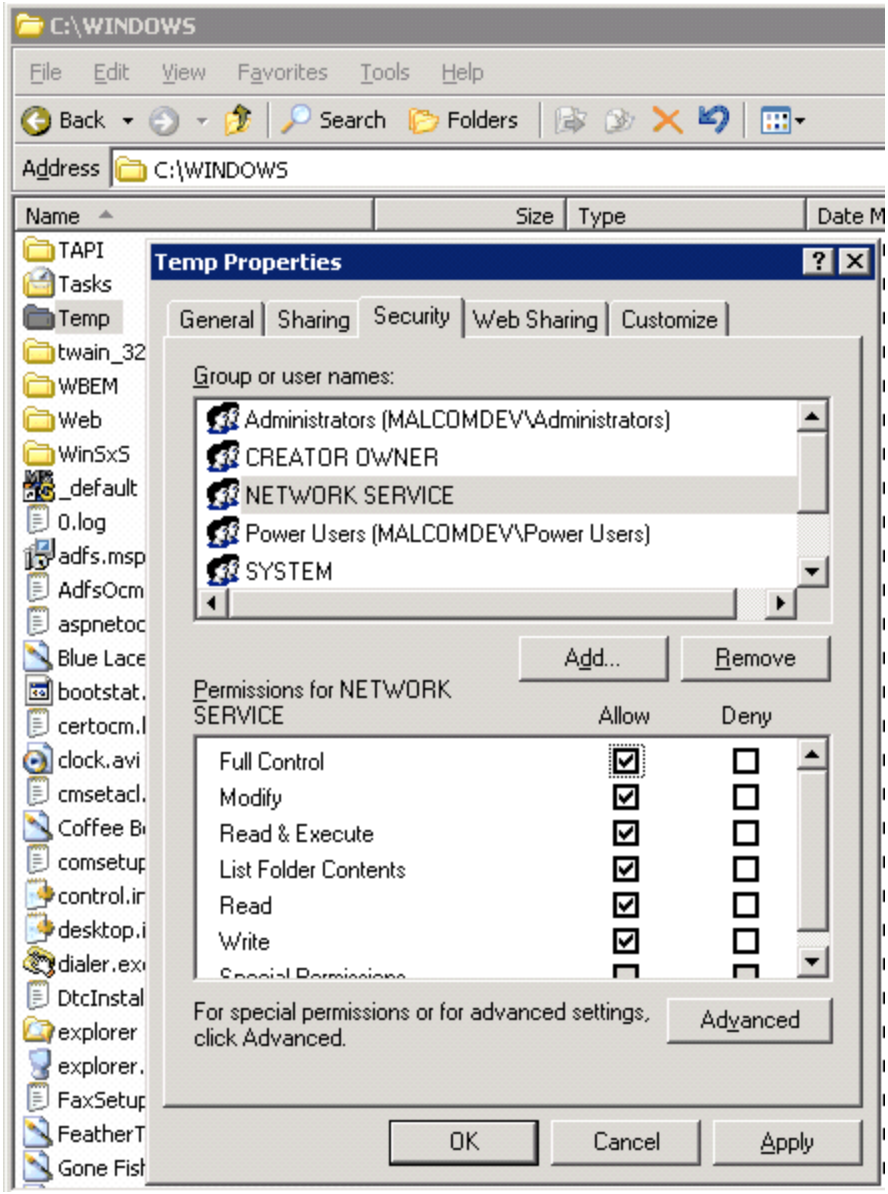
Description: An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code.

Exception Details: System.InvalidOperationException: Unable to generate a temporary class (result=1).
error CS2001: Source file 'C:\WINDOWS\TEMP\f7ni3e2f.0.cs' could not be found
error CS2008: No inputs specified

Click to expand

In this case, the problem is that the web server cannot pre-compile the ASP.NET objects for your web site because it lacks the rights to the C:\WINDOWS\TEMP directory.

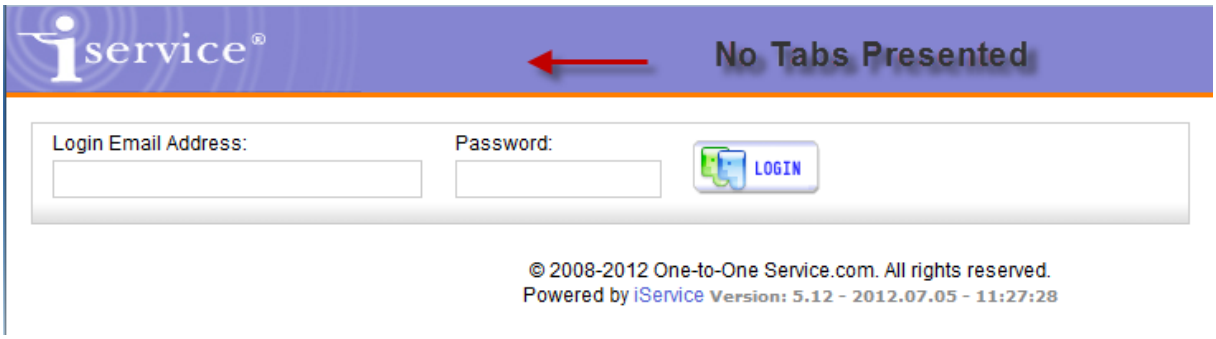
Add the NETWORK SERVICE identity to the Security properties for this folder as above, and give it full rights.



Once you have done this, click OK and try the web site again (to ensure that your browser does not show cached data, it is recommended that you close your browser completely and then reopen it before trying the web site again.

6.3 Navigation Tabs Do Not Appear

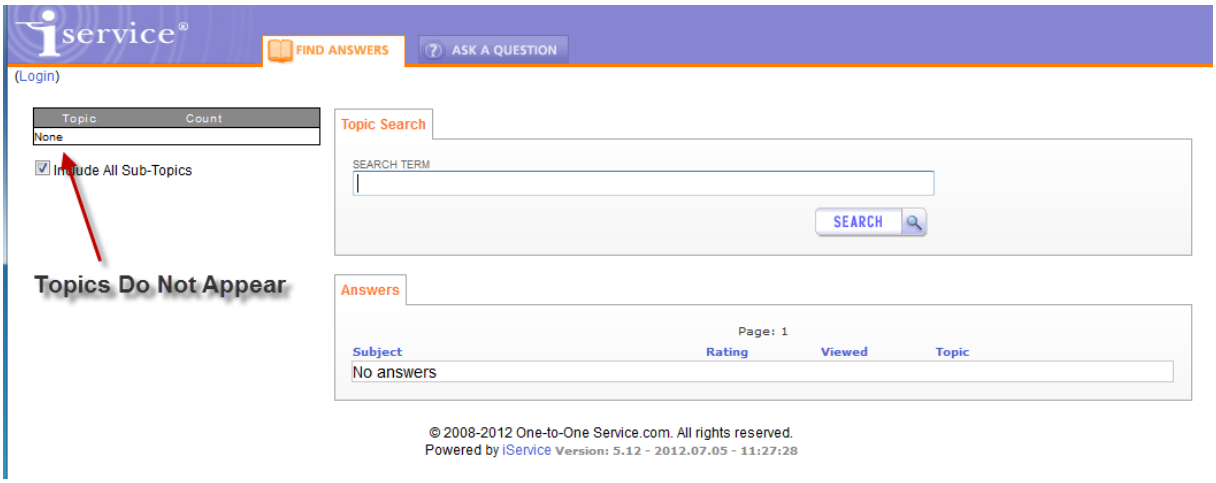
If you create or update a tenant and the navigation tabs do not appear (Find Answers and Ask a Question), this indicates that the website was not able to connect to the database. The web page will load with the iService logo and login box because the website is properly configured. However, since the web.config does not contain valid database information, the site will not know which tenant to load or which tabs to present.



Click to expand image

6.4 Topics Do Not Appear in Find Answers

If the tenant website loads but there are no topics displayed in the Find Answers page **to anonymous users**, this is often the result of a conflict in tenant URLs. If you have made a copy of an existing database and renamed it for this tenant you must be sure to change the URL of the website. This is done either in the WebsiteDomains table of the database or the Admin Tools - Websites page.



Click to expand image

6.5 Messages Not Leaving iService

The iService.MailPopper Windows service must be running for the system to send and receive email. As part of the upgrade process, this service is typically turned off (or removed if there is a DB update). So, if your system is not sending or receiving email but there are not error log files, this would be the first thing to check.

When the mail processor is not running all mail will be queued until it is turned back on. As soon as you restart the service all queued mail will be delivered and all mail waiting to be popped in your monitored mailboxes will be processed.

Step 1 - Confirm your mail popper service is running

Step 2 - Look in the Admin Tools - SMTPOut tab to see is messages are being rejected from the mail server

Step 3 - Look in Admin Tools - Segments - Mailboxes for errors on mailboxes

6.6 Checking table sizes in iService

From time to time, its important to evaluate the size of the iService database and evaluate the number of records in certain tables. For instance, if you are planning to run an interaction purge you will want to know how big your tables are.

The query below returns details about the tables within iService and can be helpful in understanding where your database is growing. It outputs the size of all tables in a database. You should run this query against the iService tenant database, not the iService Master database.

```
SELECT
    t.NAME AS TableName,
    p.rows AS RowCounts,
    SUM(a.total_pages) * 8 AS TotalSpaceKB,
    SUM(a.used_pages) * 8 AS UsedSpaceKB,
    (SUM(a.total_pages) - SUM(a.used_pages)) * 8 AS UnusedSpaceKB
FROM
    sys.tables t
INNER JOIN
    sys.indexes i ON t.OBJECT_ID = i.object_id
INNER JOIN
    sys.partitions p ON i.object_id = p.OBJECT_ID AND i.index_id = p.in-
dex_id
INNER JOIN
    sys.allocation_units a ON p.partition_id = a.container_id
WHERE
    t.NAME NOT LIKE 'dt%'
    AND t.is_ms_shipped = 0
    AND i.OBJECT_ID > 255
GROUP BY
    t.Name, p.Rows
ORDER BY
    t.Name
```

6.7 Purging Interactions

Doing a mass delete involves two separate steps/queries, which are fairly straightforward. They can be run manually or set up as scheduled jobs. Generally the idea is to run #1 as a scheduled job, and then #2 manually as needed. Here are the details if you'd like to review them before:

1. **Mass Interaction Delete** – This stored procedure deletes all interactions that match a specific clause, and can be called using the following script below. From our experience this procedure can take a while if you are deleting a large number of interactions. A typical installation on SQL Server 2008 R2 with 4 cpu cores and 8GB ram can be 35,000 interactions per hour, but obviously depends on overall server specs and other activities. The <delete clause> below specifies the range of interactions you want to delete, which could be anything beyond a specific date, anything in a topic, etc, or a combination of multiple clauses.

```
DECLARE @RC int
DECLARE @WhereClause nvarchar(512) = N' <Delete Clause>'
DECLARE @Max int = <MaxInteractionsToDelete>

EXECUTE @RC = [<iservicedbname>].[dbo].[iServiceInteraction-
sMassDelete]
   @WhereClause
  ,@Max
   GO
```

2. **Cleanup –** This is a separate stored procedure that can be used to cleanup emails and other files unused after an interaction delete. It tends to be much more process intensive, and will likely have an impact on users if they are trying to use iService at the same time. There aren't any variables to be called, it is simply executed as shown below.

```
DECLARE @RC int
EXECUTE @RC = [<iservicedbname>].[dbo].[iServiceFilesTex-
tCleanup]
GO
```

6.8 iService Error Log Files

iService generates log files when various errors occur. These log files can be generated from either the iService Web Services or the iService MailPopper service.

Web Service Log Files

Log files generated by the iService web services are written into the same directory as the iService Web Site. These files have two types of extensions: successfulretry and log. Log files with the successretry extension indicate that an initial attempt at a web service failed, but a second attempt was successful. These can usually be ignored.

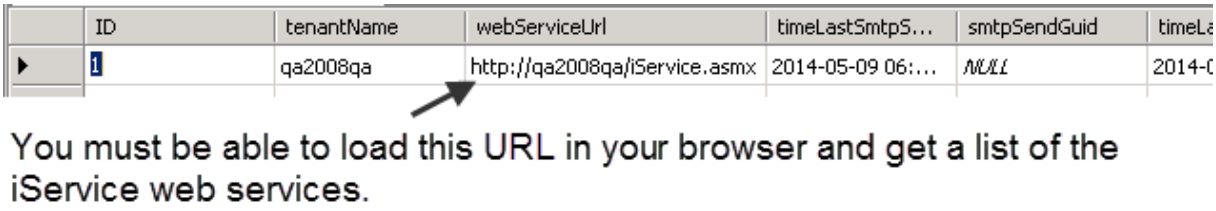
Log files that have the .log file extension indicate a web service failure that was not successful on retry. These errors should be investigated as they indicate some type of problem with your installation.

MailPopper Service Log Files

Log files generated by the iService MailPopper Windows Service are recorded in the same folder as the service files. The default path for the web service is \Program Files\1to1Service\iService.

6.9 Incorrect URL for Web Services

If you see a 404 error within your log files, this often indicates that the path specified in your master database for the iService.asmx files is incorrect. You can check this by opening the master database for your installation and inspecting the Tenants table. Ensure the url listed for the iService Web Services is correct.



6.10 Handler "PageHandlerFactory-Integrated" has a bad module

On Windows 2008R2 servers we've encountered this error in IIS after installing .NET 4.0 and switching the app pool to use .NET 4.0:

Handler "PageHandlerFactory-Integrated" has a bad module "ManagedPipelineHandler" in its module list.

This is a bug in the .NET installation, and if you get this error the easiest way to fix it is to run this at the command prompt:

```
%windir%\Microsoft.NET\Framework\v4.0.21006\aspnet_regiis.exe -i
```

Index

To view this documentation in other formats and access additional support resources, visit the iService support and documentation website.

http://www.iService.info

To learn more about One-to-One Service.com, visit our corporate website.

http://www.1to1Service.com