



## **AWLRS Product Installation and Upgrade Guide v1.3**

**22-Nov-2021**

**Issue: 1.3**

*Submitted by:*

Bentley Systems (UK) Ltd.,  
9th Floor,  
20 Gracechurch Street,  
London  
EC3V 0BG

### Document Version History

Issue	Owner	Update Source	Date	Description
1.0	Peter Bibby		17-Jun-2020	Initial Draft
1.1	Barbara O'Driscoll		04-Mar-2021	Brought in line with v1.2 changes
1.2	Andrew Briscoe		01-Apr-2021	SSO OIDC configuration
1.3	Andrew Briscoe		19-May-2021	Encryption key settings
1.4	Barbara ODriscoll		22-Nov-2021	MapServer version changes
For review:				For approval:
Rob May				Pano Voudouris
Mike Huitson				

## Table of Contents

1	Introduction.....	1
2	Prerequisites.....	2
2.1	Overview.....	2
2.2	Staging .....	3
3	Upgrade.....	5
3.1	Database.....	5
3.2	AWLRS and AWLRS Web API applications.....	6
4	Installation.....	8
4.1	Apply AWLRS Database Scripts.....	8
4.2	Apply Map Builder Configuration.....	9
4.3	Deploy AWLRS Web API.....	10
4.4	Secure the Web Service .....	18
4.5	Deploy AWLRS.....	20
4.6	Application Pool Separation .....	27
4.7	Permissions .....	29
4.7.1	CGI Bin.....	29
4.7.2	Log Files .....	29
4.7.3	Configure Web Service Origins.....	31
5	Post Install/Upgrade Tasks.....	32
5.1	Product Options .....	32
5.2	Help file .....	32
5.3	Fontset file .....	32
5.4	Write Map Files .....	33
6	AWLRS Local Culture Setting.....	35
7	AWLRS Local Password Policy and Email Address Criteria.....	37

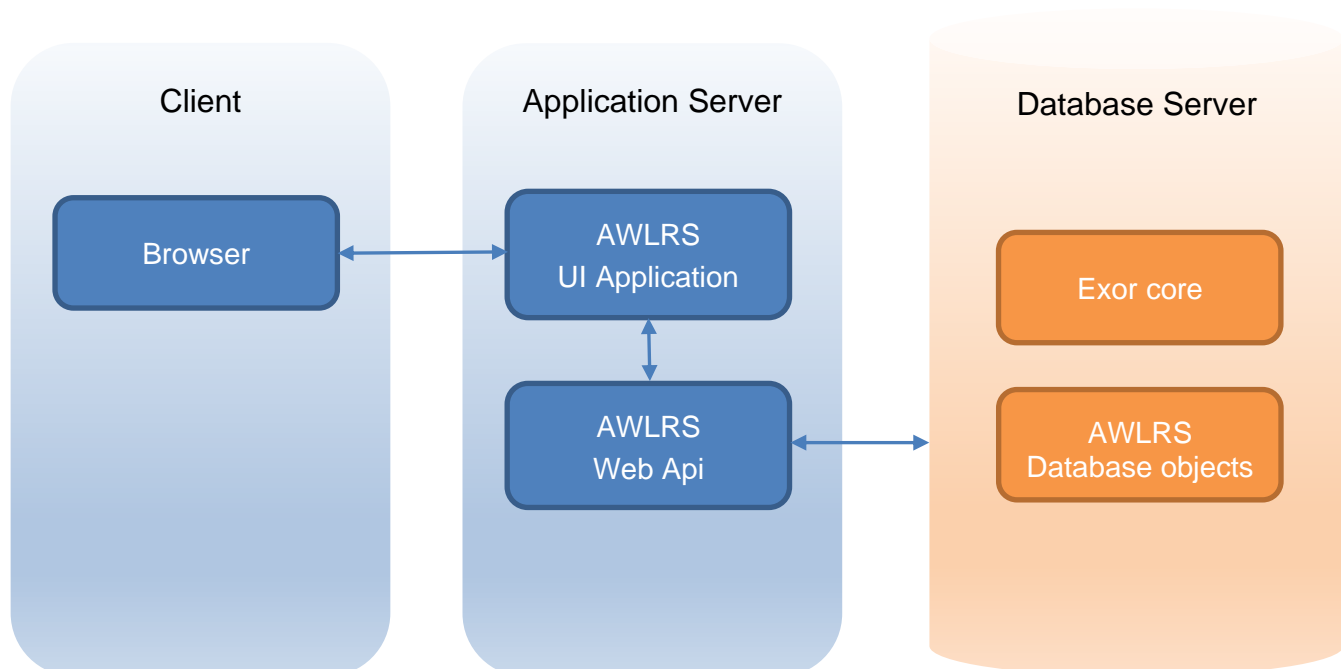
# 1 Introduction

This guide provides instructions for deploying the Bentley AWLRS product and is intended to be read either by customers or Bentley personnel carrying out an AWLRS deployment.

AWLRS is a web based application that is made up of 3 components that need to be installed or upgraded:

- Database Objects
- Web API
- UI application

The end user accesses the AWLRS application via a client browser. This client browser connects to the AWLRS UI application. The AWLRS UI application communicates directly with the AWLRS Web API. The AWLRS WebAPI in turn communicates with the Database.



**NB. To upgrade an existing AWLRS deployment please follow the instructions given from section 3 onwards.**

For a new installation of AWLRS please follow the instructions given from **section 4** onwards.

This document should be read in association with all supporting documentation, notably the Bentley Release Configuration available via [Be Communities](#) which details the versions of third-party client and server software supported for this release.

## 2 Prerequisites

### 2.1 Overview

AssetWise ALIM Linear Referencing Services (AWLRS) is an application that allows an organization to manage their linear referenced network and associated assets.

This document assumes that a copy of the AWLRS Release ZIP file, for example: **exawlr01031901en.zip**, has been downloaded from the Bentley Select web site.

You will also need to have the appropriate infrastructure in place to support the deployment of the Bentley AWLRS product, namely:

1. An Oracle Database Server with a suitable exor 4800/4900 database and <exor\_base> folder with all released fixes applied.
2. This release of AWLRS can support single sign on (SSO) authentication using OIDC. This is currently not mandatory and will not be installed by default. To enable an SSO authenticated user to access the AWLRS protected resources the exor database must be enabled for SSO and provisioned with suitable SSO user accounts, which known by the approved identity provider (e.g. Bentley IMS OIDC. If you wish to take advantage of SSO you should first discuss this with your Bentley account/product management team.
3. An application server to which the AWLRS web components will be deployed, with the following third-party products installed and correctly configured:
  - a. Microsoft IIS 10 Server
  - b. .NET 4.7.2
  - c. .NET Core 3.1 Hosting Bundle
  - d. Oracle Client (a version appropriate to the application server operating system and database)
  - e. Oracle Map Builder
4. An application server with MapServer 7.6.4 installed and correctly configured. There is a dependency here on IIS being installed, so MapServer can either be installed on the same IIS server as the AWLRS web component as mentioned above or on its own dedicated IIS server. This decision is likely to be dependent upon predicted utilization and performance requirements.

**Note:** please refer to the Bentley Exor Release Configuration document available via [Be Communities](#) for the latest information on supported third-party components such as Oracle Database Server, Oracle WebLogic, browser/JRE combinations etc.

---

PREREQUISITES

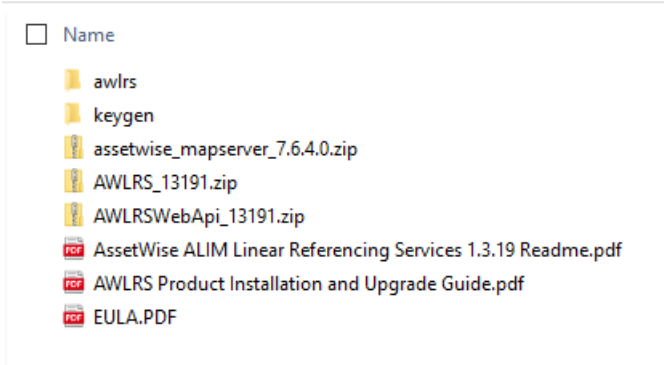
---

## 2.2 Staging

Before you start the installation, you should stage the files from the AWLRS Release Build to the correct locations on the relevant servers.

Unzip the AWLRS release to a staging folder, for example C:\stage\exawlr01031901en

PC > Windows (C:) > stage > exawlr01031901en >



Inside the unzipped AWLRS folder structure file there will be a folder called 'awlrs' for example C:\stage\exawlr01031901en\awlrs

Copy this folder to the <exor\_base> folder on your **database server**, for example c:\exor

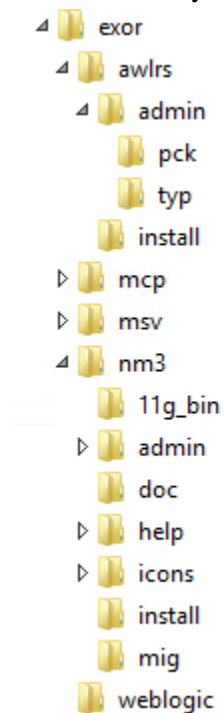
**NB: The <exor base> must also include the unzipped exor core (nm3) release folders.**

**NB2: If this is an upgrade then you will need to rename the existing <exor base>\awlrs folder before copying the new version in. This ensures that a copy is available for backup or reference purposes should any issues arise during the upgrade.**

## PREREQUISITES

---

As a minimum, your <exor base> folder structure should now look something like this:



Inside the unzipped AWLRS folder structure there will be a folder named **keygen** and two ZIP files called:

- **AWLRS\_[revision].zip**
- **AWLRSWebAPI\_[revision]**

for example:

- C:\stage\keygen
- C:\stage\exawlr01031901en\AWLRS\_13191.zip
- C:\stage\exawlr01031901en\AWLRSWebApi\_13191.zip

Copy the **keygen** folder with its contents and the two zip files to a staging folder on the **application server**.

You are now ready to install the Bentley AWLRS product.

## 3 Upgrade

This section describes the steps necessary to upgrade AWLRS from a previous version. This includes the upgrade of the database objects and the AWLRS web applications.

Please Note if you are upgrading from a 1.0, 1.1 or 1.2 release of AWLRS then significant infrastructure and configuration changes are needed to the database and application server prior to performing this upgrade. The most notable configuration change is amendments to the handlers “Restrict Map Files” and “Mapserver via FastCGI” moving to the cgi-bin level.

If you are performing a new installation this section can be skipped.

### 3.1 Database

Change directory to <exor\_base>\awlr\install

- Login to SQL\*PLUS as the highways owner on the client PC
- Select and run the correct upgrade script based on the current version and required target version.

```
start awlrs<xxxx>_awlrs<yyyy>.sql _
```

where <xxxx> refers to the current version of AWLRS and <yyyy> refers to the required version for upgrade. Note that for current installed version numbers <xxxx> between 1.0.0.x and 1.3.13.x inclusive the actual value of <xxxx> will always be 1010 as the same script will work for each of those.

For example, an upgrade to 1.3.18.x from any version 1.0.1.x to 1.3.13.x i.e. 1.1.7.x, 1.3.18.x would require:

```
start awlrs1010_awlrs13190.sql
```

Any version proceeding 1.0.1.x would require first upgrading to 1.0.1.x version. And then the above upgrade can be carried out

- You will be prompted to enter the path of the location of your highways software. This should be name of the directory, including disk identifier and a trailing slash character, referred to as <exor\_base>.

For example, if you unzipped your highways software in a directory called exor on your C drive, you would enter the following when prompted.

```
C:\exor\
```

**NB. The path must include a trailing directory separator.**

- When you have supplied this value, you will be prompted to confirm that it is correct and asked whether you wish to continue.

If the value specified is not correct or does not end with a slash character, you will be given an error message and the installation script will abort. You will then need to login to SQL\*PLUS again and rerun the script.

- When the script has completed, all the AWLRS objects and data will have been upgraded.



- The following log files are produced in the working directory. At the end of the upgrade, they can be viewed to check for any errors that could have occurred during the upgrade process.

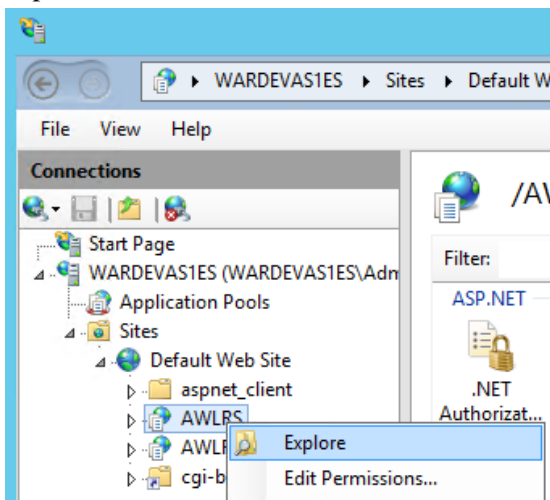
```
awlrs<xx>_awlrs<yy>_1_<date&time>.LOG
awlrs<xx>_awlrs<yy>_2_<date&time>.LOG
```

where <xx> refers to the original release number (i.e. 1010) and <yy> refers to the new release number (i.e. 1160)

## 3.2 AWLRS and AWLRS Web API applications

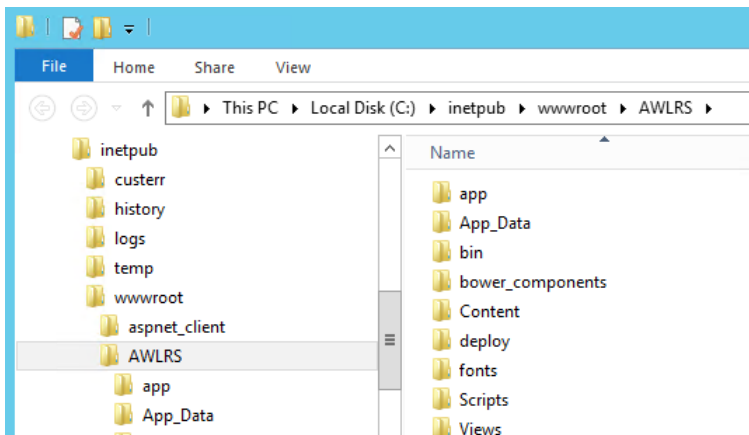
Before deploying the new version of AWLRS Web API and AWLRS UI application you should make a copy of the existing deployments for reference.

To do this open IIS manager on the **application server** and select the AWLRS UI application (for example “AWLRS”), right click on it and select “Explore” from the context menu to open the deployment folder in windows explorer:



This will open the root folder of the application for example:

## UPGRADE



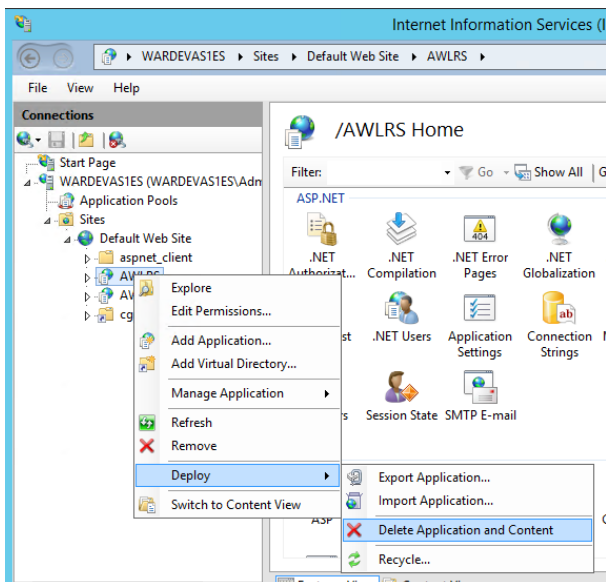
Copy the contents of this folder to a safe place as a backup.

Repeat the above for the AWLRS Web Api.

Before deploying the new versions, the previous deployments need to be removed.

In IIS manager select the AWLRS UI application (for example “AWLRS”), right click on it and select “Deploy” → “Delete Application and Content” from the context menu

**NB: This is not reversible!**



Repeat the above for the AWLRS Web API.

Once both components have been successfully removed then the new versions need to be deployed, to do this follow the instructions given from Section 4 (step 4.3) onwards.

## 4 Installation

### 4.1 Apply AWLRS Database Scripts

On your **database server**, start a command prompt in the <exor base>\awlrs\install folder.

Run SQL\*Plus, logging in as the highways owner:


```
Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.7.0.0.0

SQL> _
```

Run the following script:

```
start awlrs_inst.sql
```

You will be prompted to enter the path to the <exor base> directory:

 Administrator: Windows PowerShell

```
Please enter the value for exor base. This is the directory under which
the exor software resides (eg c:\exor\ on a client PC). If the value
entered is not correct, the process will not proceed.
There is no default value for this value.
```

```
IMPORTANT: Please ensure that the exor base value is terminated with
the directory separator for your operating system
(eg \ in Windows or / in UNIX).
```

```
Enter exor base directory now : _
```

## INSTALLATION

---

Enter the path to the <exor base> directory where you unzipped the 'awlr' folder from the AWLRS Release Build ZIP (see the Prerequisites section):

**NB. The path must include a trailing directory separator.**

```
Please enter the value for exor base. This is the directory under which
the exor software resides (eg c:\exor\ on a client PC). If the value
entered is not correct, the process will not proceed.
There is no default value for this value.
```

```
IMPORTANT: Please ensure that the exor base value is terminated with
the directory separator for your operating system
(eg \ in windows or / in UNIX).
```

```
Enter exor base directory now : C:\exor\
```

You should be prompted whether to continue, once accepted the installation will begin:

SQL\*Plus will exit once the script has finished:

Check the log files for errors and take action to resolve any issues before continuing. Two log files are produced by the install script, both are created in the awlrs\install folder and their file names are in the following format:

- awlrs\_install\_1\_[date in the format DDMMYYYY\_HHmmSS].LOG
- awlrs\_install\_2\_[date in the format DDMMYYYY\_HHmmSS].LOG

For example:

- awlrs\_install\_1\_17JUN2020\_095953.LOG
- awlrs\_install\_2\_17JUN2020\_095953.LOG

## 4.2 Apply Map Builder Configuration

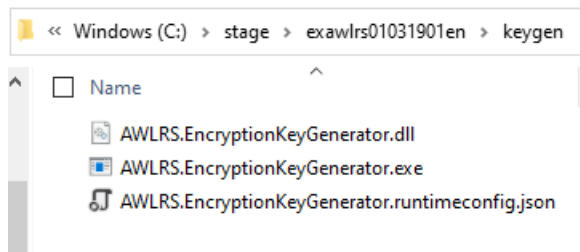
Please refer to the appropriate version of the "AWLRS Administration Guide" and the exor core document "Spatial Server Admin" for details on how to use Oracle Map Builder to configure a Base Map for AWLRS.

**AWLRS allows shape files to be displayed as layers in the map, please refer to the AWLRS Admin Guide for details of how to implement this functionality.**

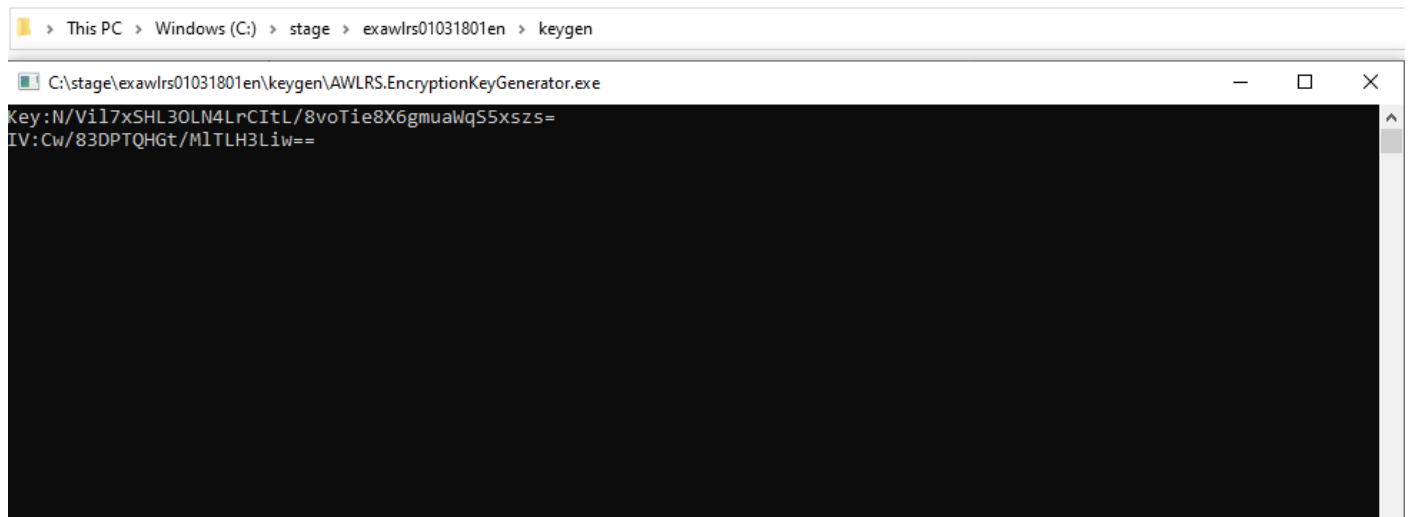
## 4.3 Deploy AWLRS Web API

The following instructions assume a default non-SSO installation of AWLRS. If you wish to install AWLRS with SSO then additional instructions will be required. In this case you should discuss this with your Bentley account team.

Earlier in step 2.2 a folder name keygen was copied to a staging area on the **application server**. Locate the keygen folder and run the executable named **AWLRS.EncryptionKeyGenerator** from that folder. You can either do this by double clicking the executable in File Explorer or by opening a command prompt in that folder and running the executable by name:



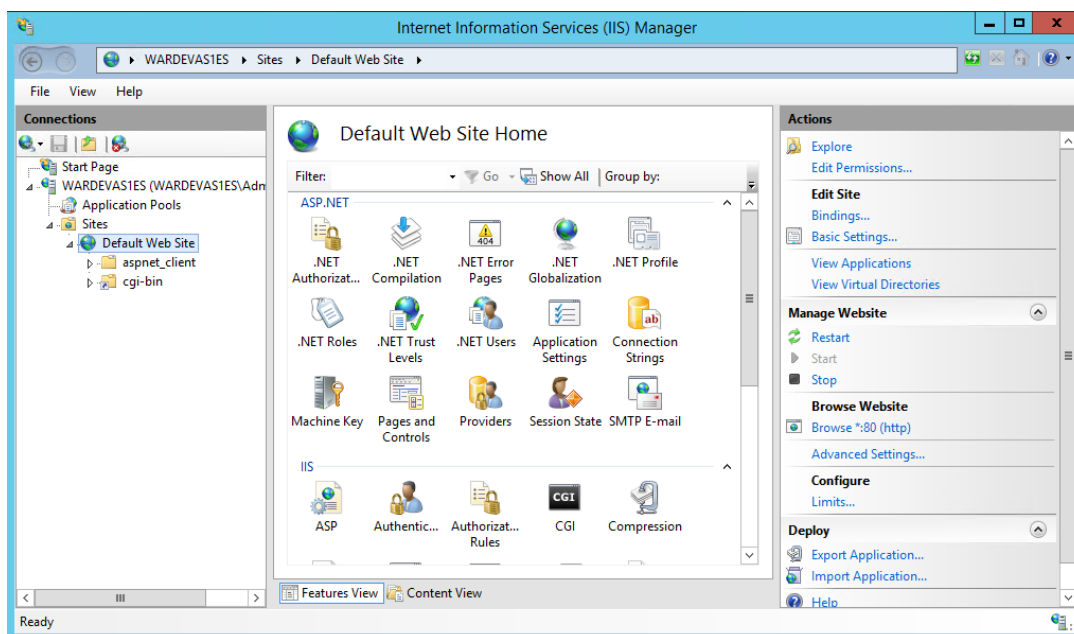
Execution of **AWLRS.EncryptionKeyGenerator** will result in the creation of two values (a Key and an IV) as seen below. You will need these values when you configure the Web API during then next few steps, so make sure to copy them carefully to a text editor for reference using cut and paste. Note that only the values after the colon (:) to the end of the line are needed, the **Key:** and **IV:** just identify the actual value relevance.



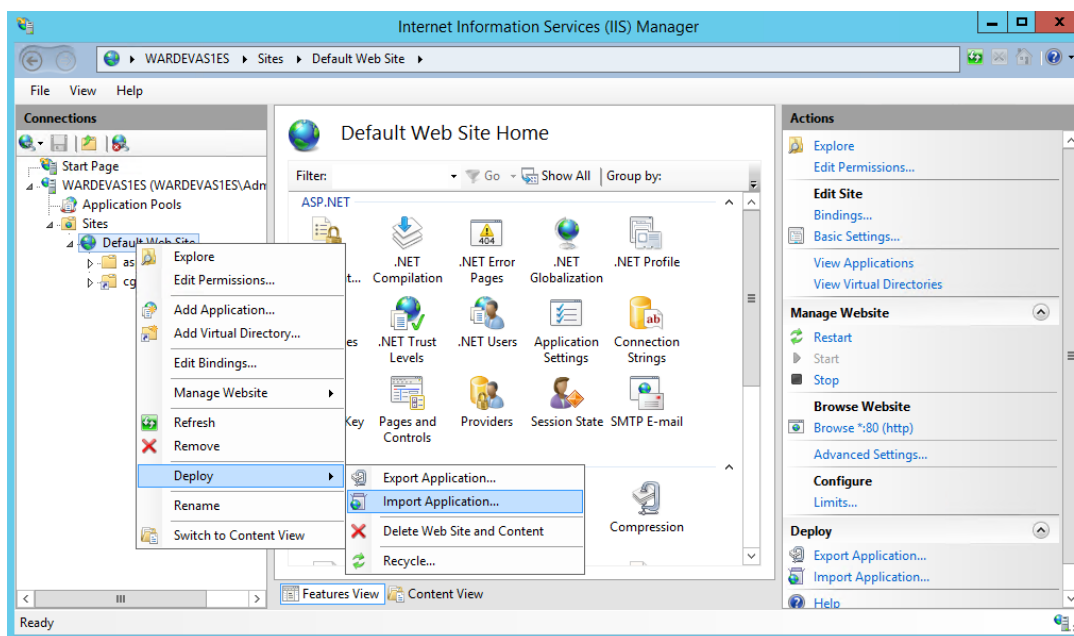
It is okay to call this multiple times, just ensure to use a Key and IV pair of values from an individual run of **AWLRS.EncryptionKeyGenerator**. Once you have the values recorded you may hit return and close the command window.

## INSTALLATION

On the **application server** run IIS Manager and navigate to the “Default Web Site”:



Right-click on “Default Web Site” and select Deploy > Import Application:



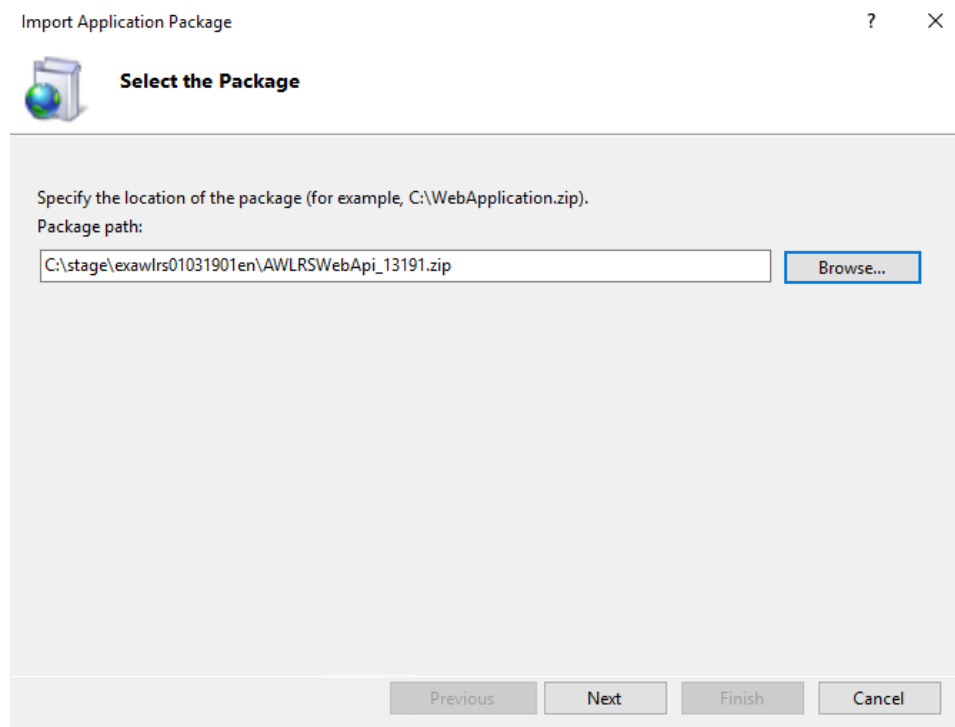
## INSTALLATION

---

Use the “Browse...” button to select the **AWLRSWebAPI\_[revision].zip** file located in the staging area on the server (see Prerequisites section):

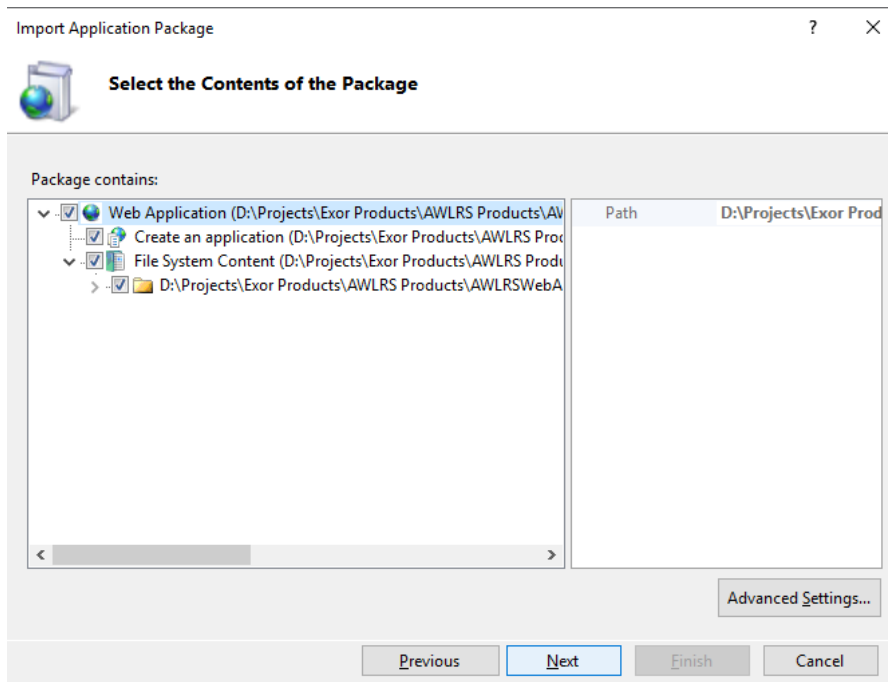
For example:

C:\stage\exawls01031901en\AWLRSWebApi\_13191.zip



Click Next. You should not need to make any changes on the following screen:

## INSTALLATION



Click Next. On this screen you should enter the following information in the named fields as follows:

- **Application Path** - The path to be used for the deployed application, this will form the later part of the application URL and is also the “deployment name” displayed under the web site in IIS Manager.
- **Allowed Origins (CORS)** – You must set this to a comma separated list of server URLs that are allowed to access the Web API data (<https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS>). This is expected to be the addresses of the AWLRS application servers and could be as simple as just the current server in a single IIS/AWLRS deployment. With two servers named server1 and server2 in the domain named domain.com the value would look like this **https://server1.domain.com,https://server2.domain.com**. As this is a security setting, if you get the value wrong you can deny AWLRS access to the resources it needs to function so take care. If you need assistance contact your account support.
- **Connection String** - The database connection string for the AWLRS Application, please change [Database Alias] to the database alias
- **Encryption Key** - The **Key:** value generated by **AWLRS.EncryptionKeyGenerator** called earlier
- **Encryption IV** - The **IV:** value generated by **AWLRS.EncryptionKeyGenerator** called earlier
- **TNS Admin Path** – The path to the TNS to be used. Double slash \\ should be used on the path.
- **AWLRS\_mapName** – Name of the Map created in Map Builder for AWLRS, Please NOTE a corresponding product option must also be set.
- **SDL\_mapName** – Name of the Map created in Map Builder for SDL, Please NOTE a corresponding product option must also be set.



- **MapServerProjLib** – The MapServer Projection Library Path. Double slash \\ should be used on the path.
- **MapServerPath** – The path to the MapServer cgi-bin folder. Double slash \\ should be used on the path.
- **MapServerOGRPath** – The path to the MapServer OGR folder. Double slash \\ should be used on the path.
- **MapServerURL** – URL for the MapServer cgi-bin, please change [iisservername] to the fully qualified name of the IIS Server.
- **LogFilePath** – The path for the Web API Log file
- **SdlMapServerOGRPath** - The path to the SDL MapServer OGR folder. Double slash \\ should be used on the path.

The following fields are all optional settings and relate to extended features of AWLRS that may not be available (SSO and Log Dispatcher). If the values are left at the defaults, then the feature will not be operational. If you do not know what to set for the following fields than they can be safely ignored.


- **Mid-tier Connection String – OPTIONAL** - The database connection string for the Oracle database mid-tier user for proxy connections. Please change [Database Alias] to the database alias and provide the correct mid-tier user account and credentials (see the relevant Exor documentation).
- **OIDC Authority – OPTIONAL** - The URL of the OIDC JWT token provider for validation. An example for Bentley production environments would be <https://imsoic.bentley.com/>. This is used in validation of the token used to access the API. If the received token was not issued by the defined authority access will not be granted. If not set then AWLRS will use the original username and password authentication by default.
- **OIDC Audience – OPTIONAL** - The URL of the audience for the token issued by the authority. This value is used to validate the token. If this value is not a match for the relevant field in the token access will not be granted. Only set this if a value has been provided for **OIDC Authority** above.

The following settings are more commonly used in Bentley managed service deployments. They should be left unset if unsure of the values needed and you have not been directed to set them.

- **Log Dispatcher Service URI – OPTIONAL** - The URI of the Bentley Log Dispatcher service that is to be used by the web API. An example for Bentley production environments would be **https://connect-logdispatcher.bentley.com**.
- **Bentley UDDI (BUDDI) code – OPTIONAL** - The numeric UDDI (Universal Description, Discovery, and Integration) code for service selection of the correct Log Dispatcher service. This value depends on your deployment and region so you may need to contact support or your account manager for the correct value to use.

## INSTALLATION

Import Application Package ? X

 **Enter Application Package Information**

Enter information that is required to install this package:


**Application Path**  
Full site path where you want to install your application (for example, Default Web Site/Application).  
Default Web Site/

**Allowed Origins (CORS)**  
Provide comma separated URL list of origins allowed to access these resources

**ConnectionString**  
Please provide AWLRS connection string...

**Encryption Key**  
Please provide the key value from the key generator tool

Import Application Package ? X

 **Enter Application Package Information**

Enter information that is required to install this package:

**Encryption IV**  
Please provide the IV value from the key generator tool

**TNSAdminPath**  
Please provide TNS Admin Path... (Use '\\' in the path)

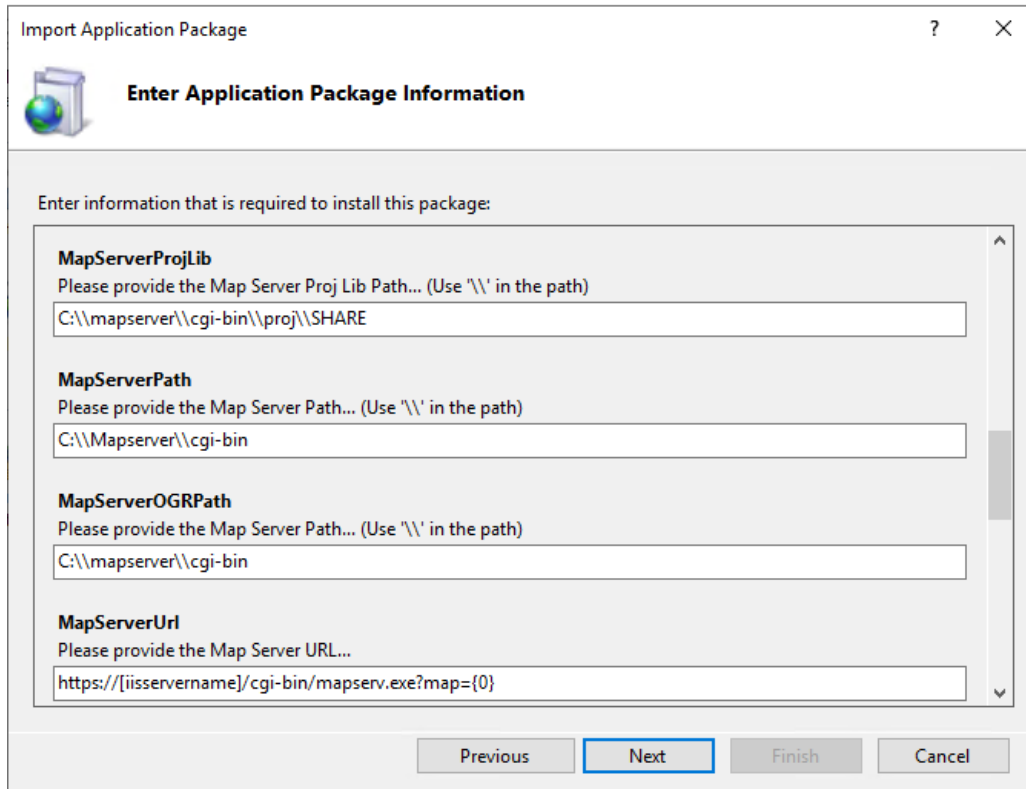
**AWLRS\_mapName**  
Please provide the Map Name...

**SDL\_mapName**  
Please provide the Map Name...

## CONFIDENTIALITY STATEMENT

The contents of this document, including system ideas and concepts, are confidential and proprietary in nature and are not to be distributed in any form without the prior written consent of Bentley, Inc.

## INSTALLATION



**Import Application Package**

**Enter Application Package Information**

Enter information that is required to install this package:

**MapServerProjLib**  
Please provide the Map Server Proj Lib Path... (Use '\\' in the path)  
C:\\mapserver\\cgi-bin\\proj\\SHARE

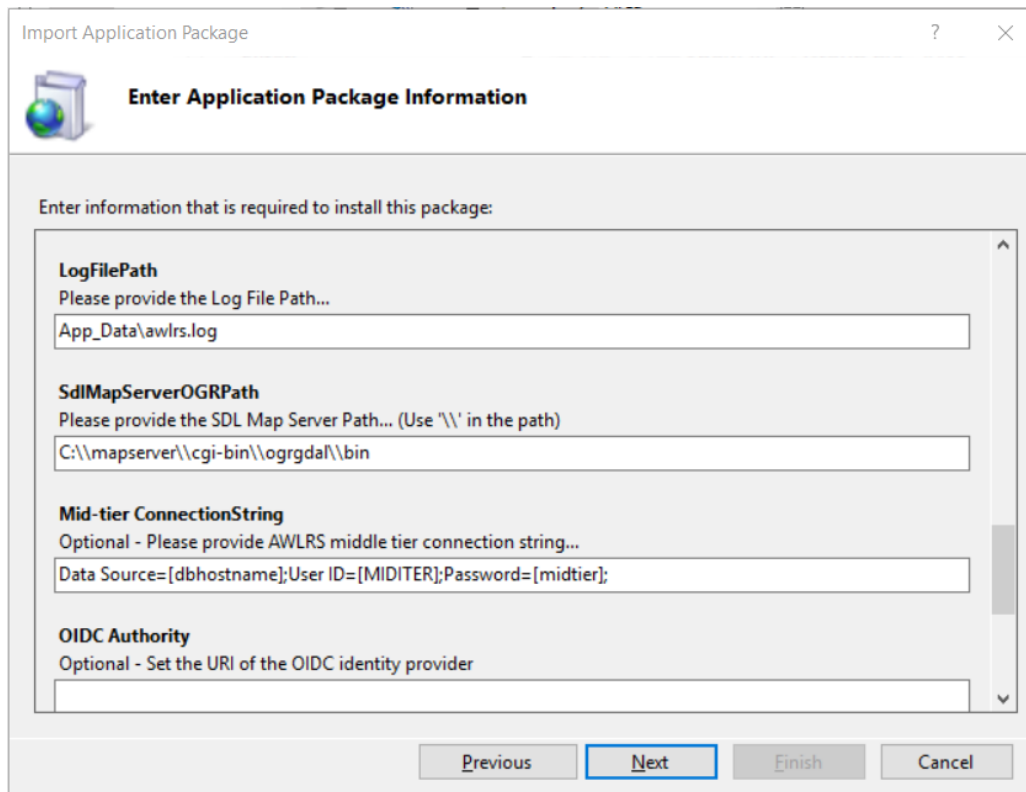
**MapServerPath**  
Please provide the Map Server Path... (Use '\\' in the path)  
C:\\Mapserver\\cgi-bin

**MapServerOGRPath**  
Please provide the Map Server Path... (Use '\\' in the path)  
C:\\mapserver\\cgi-bin

**MapServerUrl**  
Please provide the Map Server URL...  
https://[iisservername]/cgi-bin/mapserv.exe?map={0}

Previous Next Finish Cancel

Settings after and including **Mid-tier ConnectionString** are all optional (see descriptions above).



**Import Application Package**

**Enter Application Package Information**

Enter information that is required to install this package:

**LogFilePath**  
Please provide the Log File Path...  
App\_Data\\awlrs.log

**SdlMapServerOGRPath**  
Please provide the SDL Map Server Path... (Use '\\' in the path)  
C:\\mapserver\\cgi-bin\\ogr\\gdal\\bin

**Mid-tier ConnectionString**  
Optional - Please provide AWLRS middle tier connection string...  
Data Source=[dbhostname];User ID=[MIDITER];Password=[midtier];

**OIDC Authority**  
Optional - Set the URI of the OIDC identity provider

Previous Next Finish Cancel

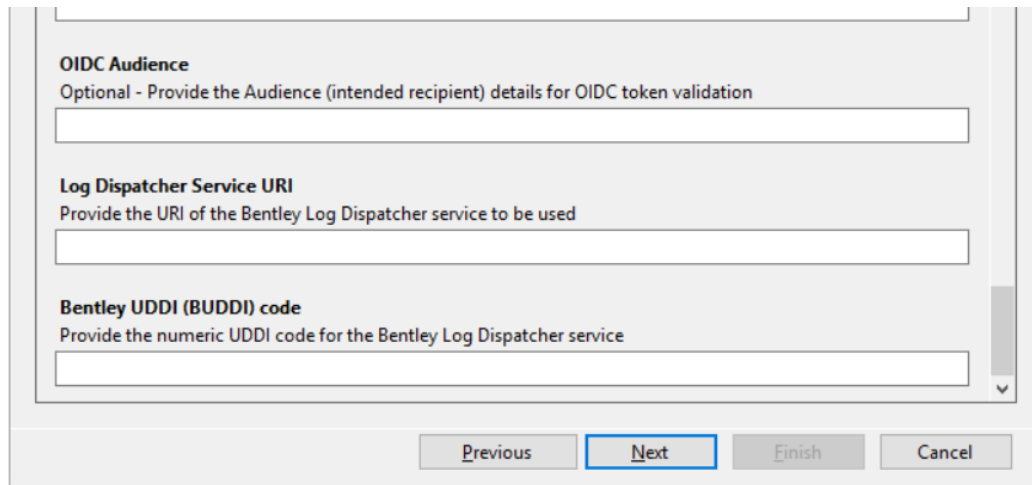
### CONFIDENTIALITY STATEMENT

The contents of this document, including system ideas and concepts, are confidential and proprietary in nature and are not to be distributed in any form without the prior written consent of Bentley, Inc.

INSTALLATION

---

And finally, if needed:

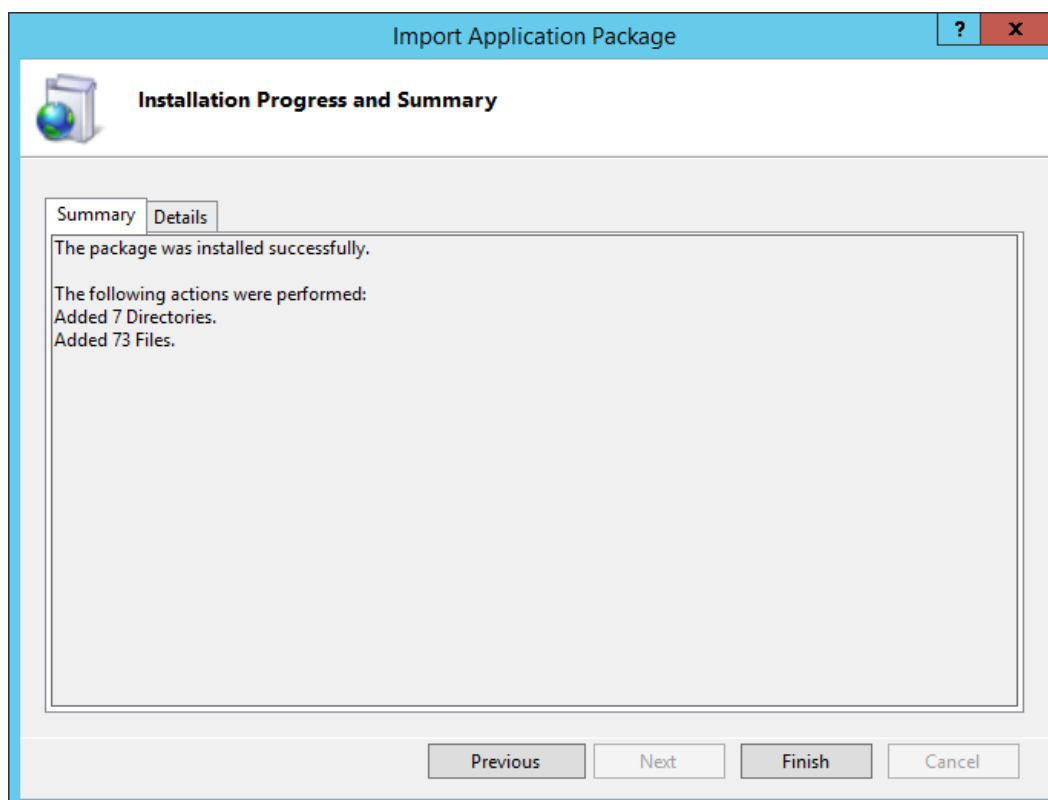


The screenshot shows a dialog box titled "Import Application Package" with a "Next" button highlighted. The dialog contains three sections:

- OIDC Audience**  
Optional - Provide the Audience (intended recipient) details for OIDC token validation
- Log Dispatcher Service URI**  
Provide the URI of the Bentley Log Dispatcher service to be used
- Bentley UDDI (BUDDI) code**  
Provide the numeric UDDI code for the Bentley Log Dispatcher service

At the bottom, there are four buttons: "Previous", "Next" (highlighted), "Finish", and "Cancel".

Click Next. Deployment might take a few seconds then a summary screen will be displayed:



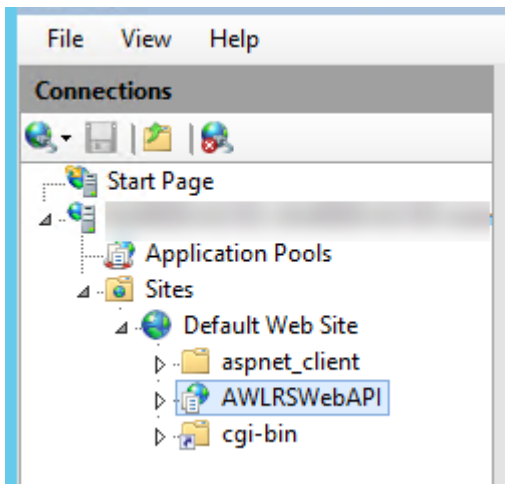
The screenshot shows the same dialog box, but now the "Summary" tab is selected. The "Details" tab is also visible. The "Summary" tab displays the following text:

The package was installed successfully.

The following actions were performed:  
Added 7 Directories.  
Added 73 Files.

At the bottom, there are four buttons: "Previous", "Next", "Finish" (highlighted), and "Cancel".

Click Finish. In IIS Manager, you should now see a new application listed under Default Web Site:

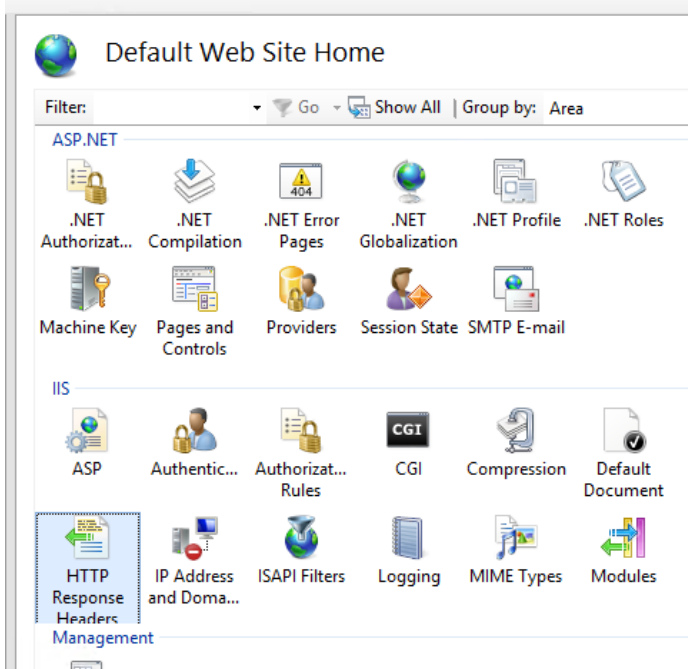


## 4.4 Secure the Web Service

All networked web services expose actions to the network be that an intranet or internet. It is important to secure the services against attack. To this end there are some simple IIS settings that can be made to ensure the IIS instance is secured against more common attack vectors. We recommend you make the following changes at the web site level, so they are inherited by web services within that site. Alternatively, you can secure the web service itself by making the changes at the web service level. The following describe making the settings at the web site level.

In IIS Manager select the Default Web Site node (where AWLRSWebAPI has been installed)

Then locate and open the HTTP Response Headers feature under the IIS Area



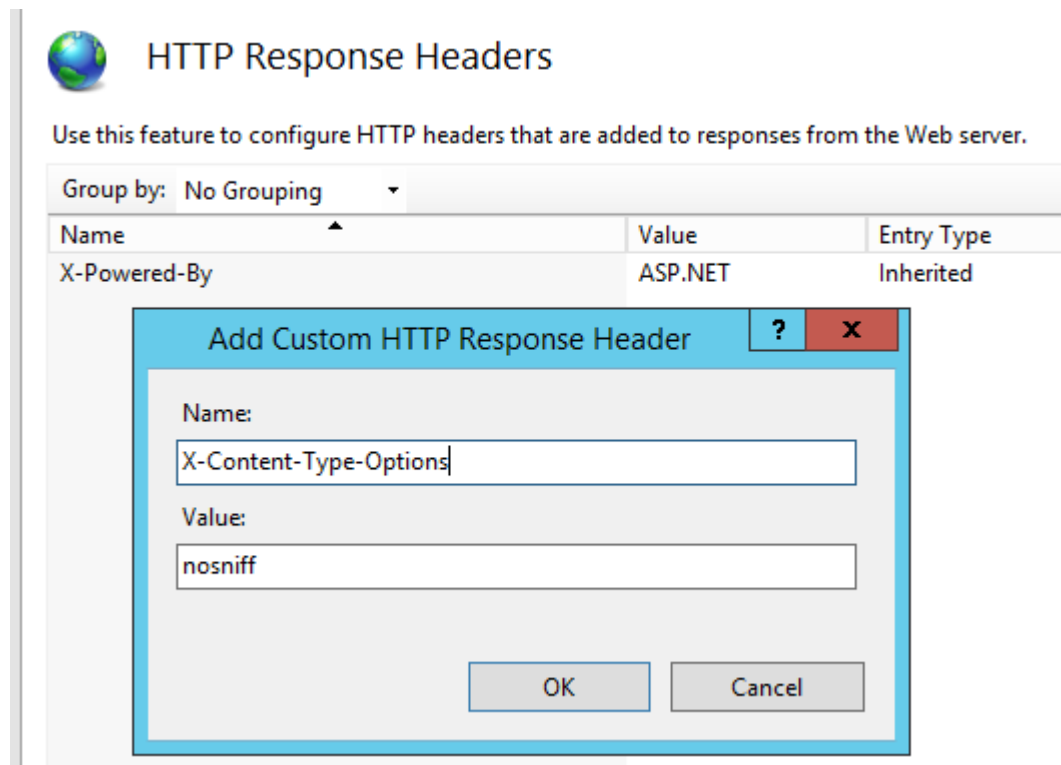
The feature will open

## INSTALLATION

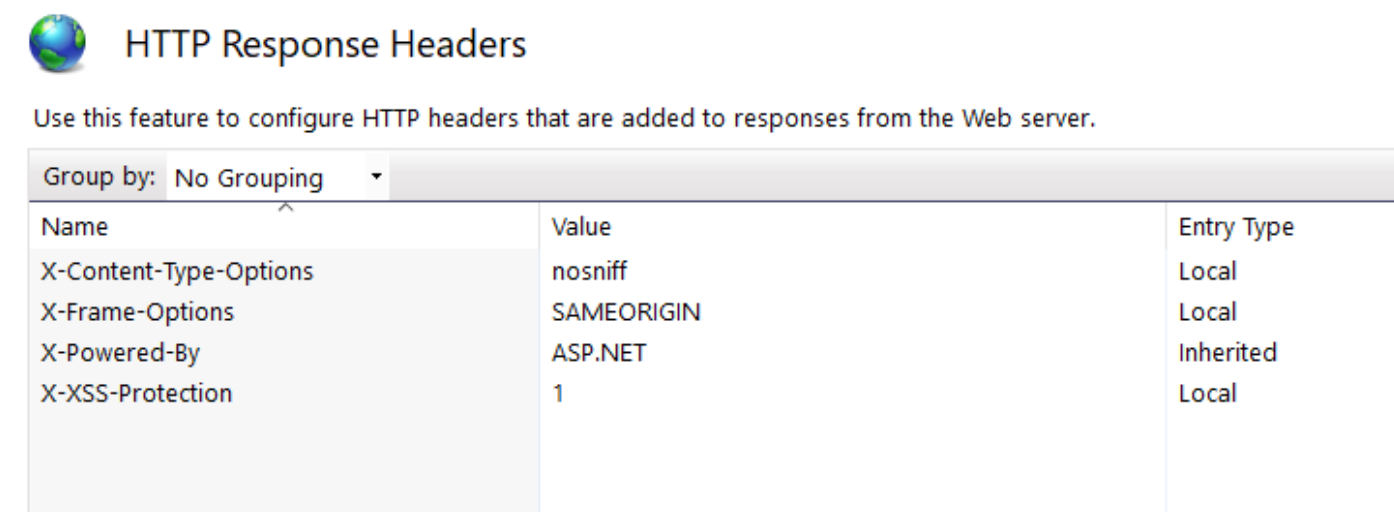
Now use the Add action to add the following HTTP Response Headers if they are not already present:

Response Header Name	Value
<b>X-Content-Type-Options</b>	nosniff
<b>X-XSS-Protection</b>	1
<b>X-Frame-Options</b>	SAMEORIGIN

For example, to add the first of the above fill in the name and value fields and click OK:



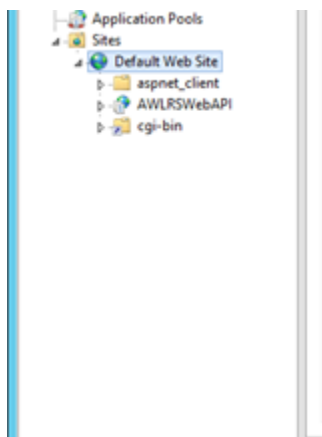
Once all the values have been added the Feature will look similar to:



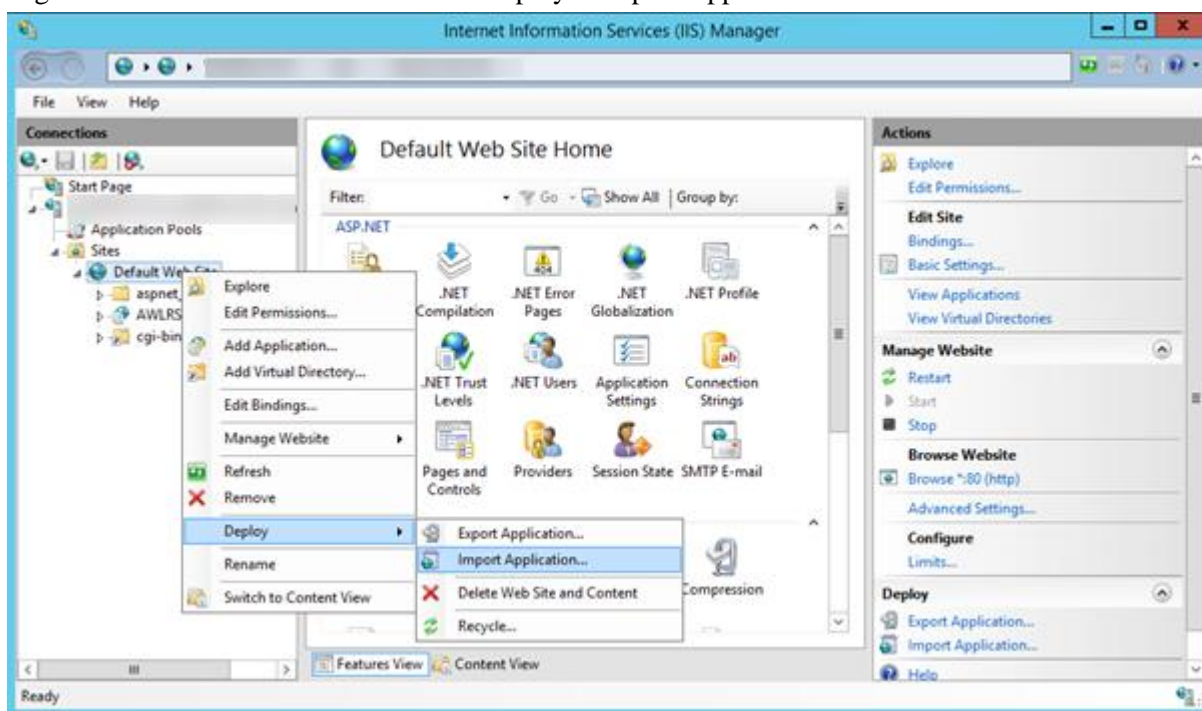
## INSTALLATION

### 4.5 Deploy AWLRS

On the **application server**, open IIS Manager and navigate to Default Web Site:



Right-click Default Web Site and select Deploy > Import Application:

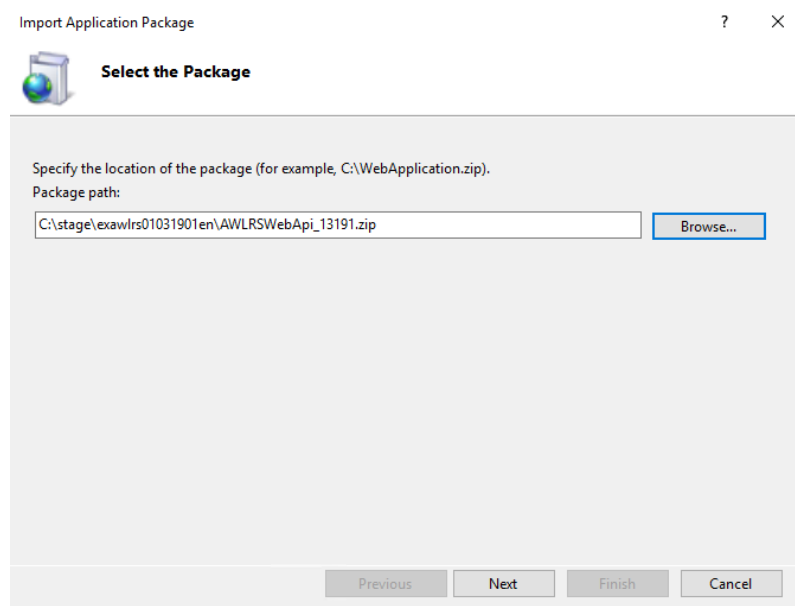


## INSTALLATION

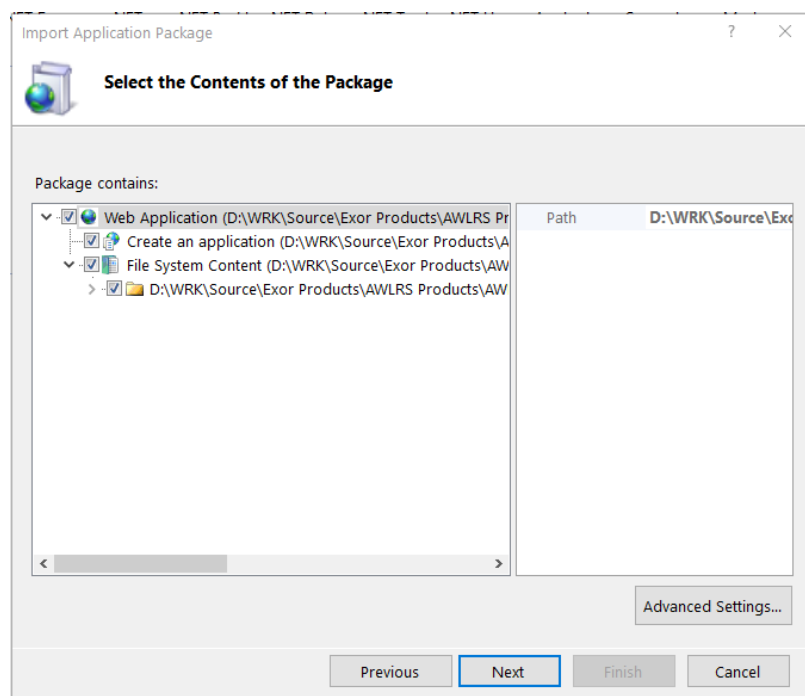
Use the “Browse...” button to select the **AWLRS\_[revision].zip** file located in the staging area on the server (see Prerequisites section):

For example:

C:\stage\exawls01031901en\AWLRS\_13191.zip



Click Next. You should not need to make any changes on the following screen:





## INSTALLATION

---

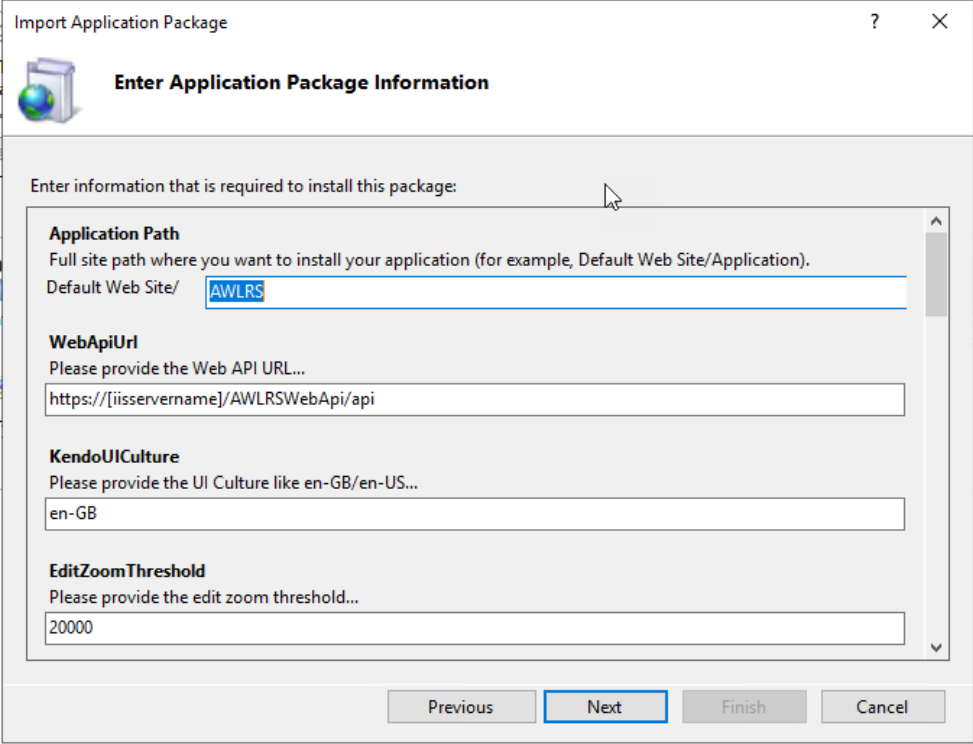
Click Next. On this screen you should enter the following information

- **Application Path** - The path where you want to install the application
- **WebApiUrl** – The location of the deployed AWLRSWebAPI set in section 4.3, please change [iisservername] to the fully qualified name of the IIS Server.
- **KendoUICulture** – The browser displayed culture type for example ‘en-GB’, ‘en-US’ etc
- **EditZoomThreshold** – The minimum zoom level before editing is allowed
- **KeepAliveInterval** – Number of milli-seconds after which a ping request is sent to WebApi to keep the session alive
- **externalIdp** – Defaulted to false. This value should be false unless IMS OIDC is to be used, for details on IMS configuration please refer to your Bentley account team for further assistance.
- **IncludeEndDatedForQuickSearch** – True/False on whether end dated included in quick search
- **MaxRowsForQuickSearch** – The max number of returned rows on quick search
- **SnapTolerance** – The default snap tolerance
- **DecimalFormat** – Default decimal format to be used for none layer specific numeric fields. For example, coordinate displayed in map, center map to coordinates and tolerance displayed in preferences dialog. An example entry would be ‘#.#####’
- **DecimalPlaces** – Default decimal places for none layer specific numeric fields. For example, coordinate displayed in map, center map to coordinates and tolerance displayed in preferences dialog.
- **EnableAwlrNSGPlugIn** – Defaulted to False. To enable the NSG (Street Gazetteer Manager) plug-in functionality, simply set this value to True. Please refer to the AWLRS Admin Guide v1.3 for more information regarding this plugin.
- **EnableAwlrPlmPlugIn** – Defaulted to False. Please refer to the AWLRS Admin Guide v1.3 for more information regarding this plugin.
- **EnableAwlrTMAPPlugIn** – Defaulted to False. Please refer to the AWLRS Admin Guide v1.3 for more information regarding this plugin.
- **EnableAwlrSDLPlugIn** – Defaulted to False. Please refer to the AWLRS TDL Guides v1.3 for more information regarding this plugin.

The following fields are all optional settings and relate to extended features of AWLRS that may not be available (SSO). If the values are left at the defaults, then the feature will not be operational. If you do not know what to set for the following fields then they can be safely ignored.

## INSTALLATION

- **OIDC Authority – OPTIONAL** - The URL of the OIDC JWT token provider for validation. An example for Bentley production environments would be <https://imsoidc.bentley.com/>. This is used in validation of the token used to access the API. If the received token was not issued by the defined authority access will not be granted. If not set, then AWLRS will use the original username and password authentication by default.
- **OIDC Client Id – OPTIONAL** - The client identifier for this application as defined with the OIDC provider. Only set this if a value has been set for **OIDC Authority** above.
- **OIDC Web API Scope – OPTIONAL** – Name the scope required for access to the Web API protected resources as defined with the OIDC provider. Only set this if a value has been set for **OIDC Authority** above.



Import Application Package

**Enter Application Package Information**

Enter information that is required to install this package:

**Application Path**  
Full site path where you want to install your application (for example, Default Web Site/Application).  
Default Web Site/

**WebApiUrl**  
Please provide the Web API URL...


**KendoUICulture**  
Please provide the UI Culture like en-GB/en-US...

**EditZoomThreshold**  
Please provide the edit zoom threshold...

Previous Next Finish Cancel

## INSTALLATION

Import Application Package ? X

 **Enter Application Package Information**

Enter information that is required to install this package:

**KeepAliveInterval**  
Please provide the Keep Alive Interval...  
300000


**externalIdP**  
Set true if an external IdP (e.g. IMS) will be used...  
false

**IncludeEndDatedForQuickSearch**  
Please provide the Include End Dated for Quick Search...  
false

**MaxRowsForQuickSearch**  
Please provide the Max Rows returned for Quick Search...  
100

Previous Next Finish Cancel

Import Application Package ? X

 **Enter Application Package Information**

Enter information that is required to install this package:

**SnapTolerance**  
Please provide the default snap tolerance...  
5

**DecimalFormat**  
Please provide the default decimal format...  
#,###

**DecimalPlaces**  
Please provide the default decimal places...  
4

**EnableAwlrnsNgPlugin**  
Enable the NSG Plug-in functionality...  
false

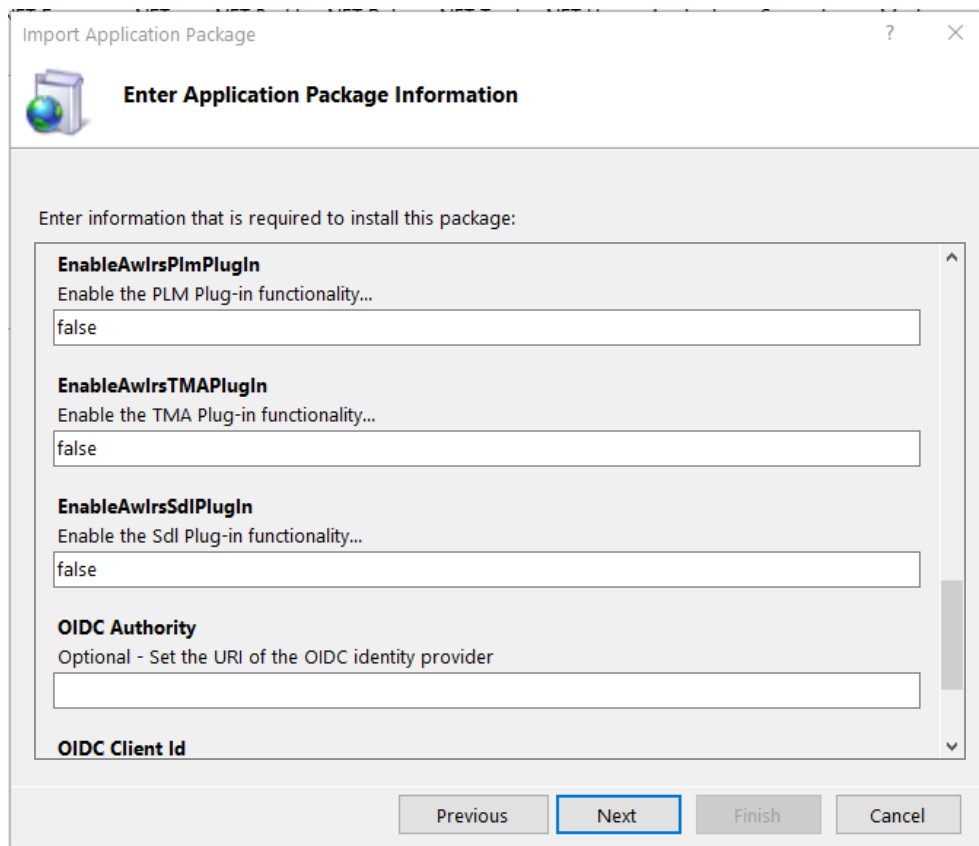
Previous Next Finish Cancel

## CONFIDENTIALITY STATEMENT

The contents of this document, including system ideas and concepts, are confidential and proprietary in nature and are not to be distributed in any form without the prior written consent of Bentley, Inc.

## INSTALLATION

*The final settings related to OIDC are all optional (see descriptions above).*



Import Application Package

**Enter Application Package Information**

Enter information that is required to install this package:

**EnableAwlrPlmPlugin**  
Enable the PLM Plug-in functionality...

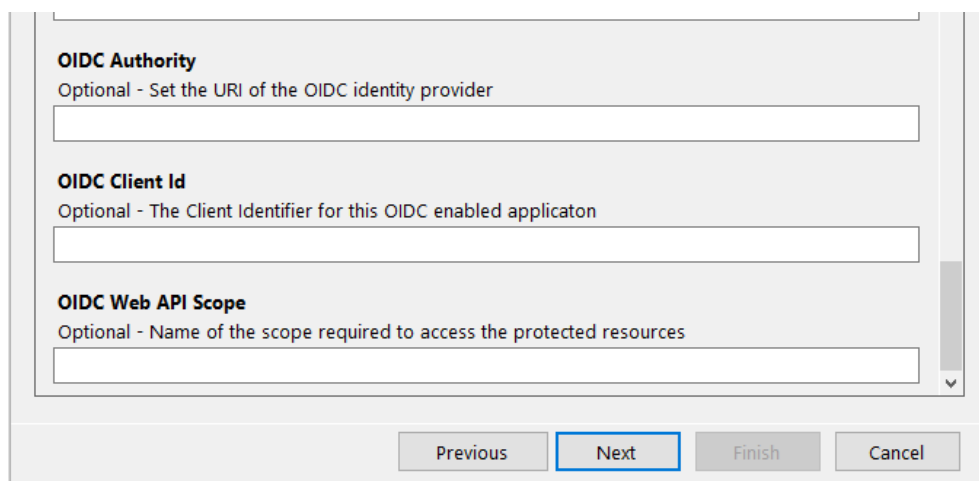
**EnableAwlrTMAPlugin**  
Enable the TMA Plug-in functionality...

**EnableAwlrSdlPlugin**  
Enable the Sdl Plug-in functionality...

**OIDC Authority**  
Optional - Set the URI of the OIDC identity provider

**OIDC Client Id**

Previous Next Finish Cancel



**OIDC Authority**  
Optional - Set the URI of the OIDC identity provider

**OIDC Client Id**  
Optional - The Client Identifier for this OIDC enabled application

**OIDC Web API Scope**  
Optional - Name of the scope required to access the protected resources

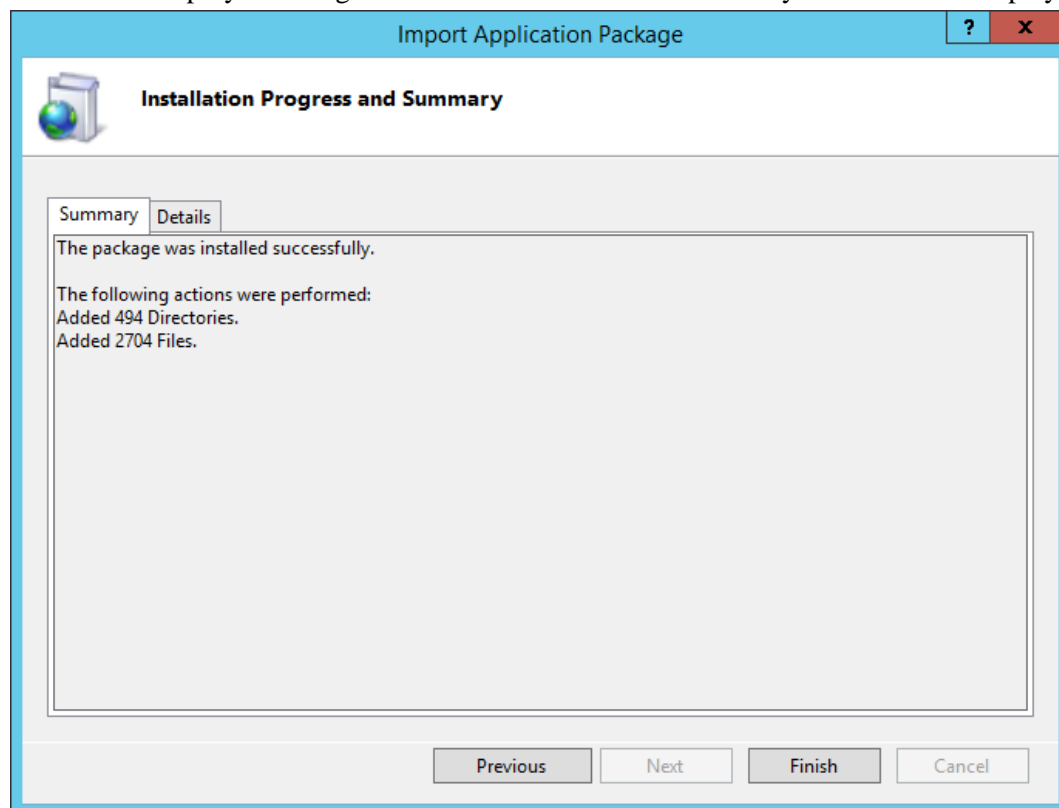
Previous Next Finish Cancel

## CONFIDENTIALITY STATEMENT

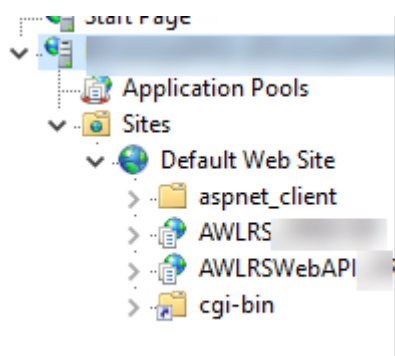
The contents of this document, including system ideas and concepts, are confidential and proprietary in nature and are not to be distributed in any form without the prior written consent of Bentley, Inc.

## INSTALLATION

Click Next. Deployment might take a few seconds then a summary screen will be displayed:



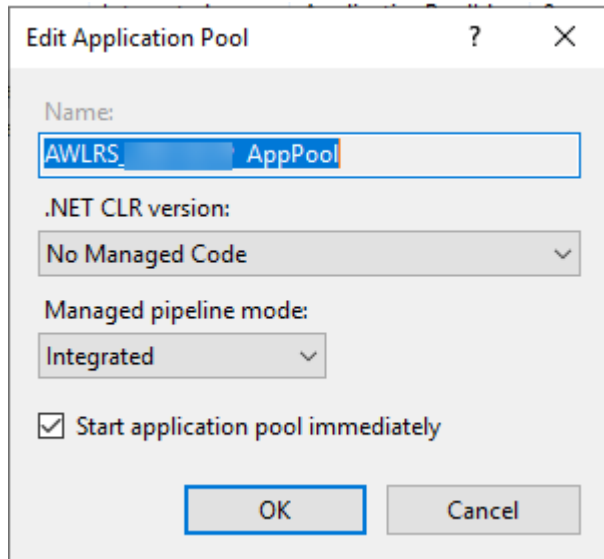
Click Finish. In IIS Manager, you should now see a new application listed under Default Web Site:



## INSTALLATION

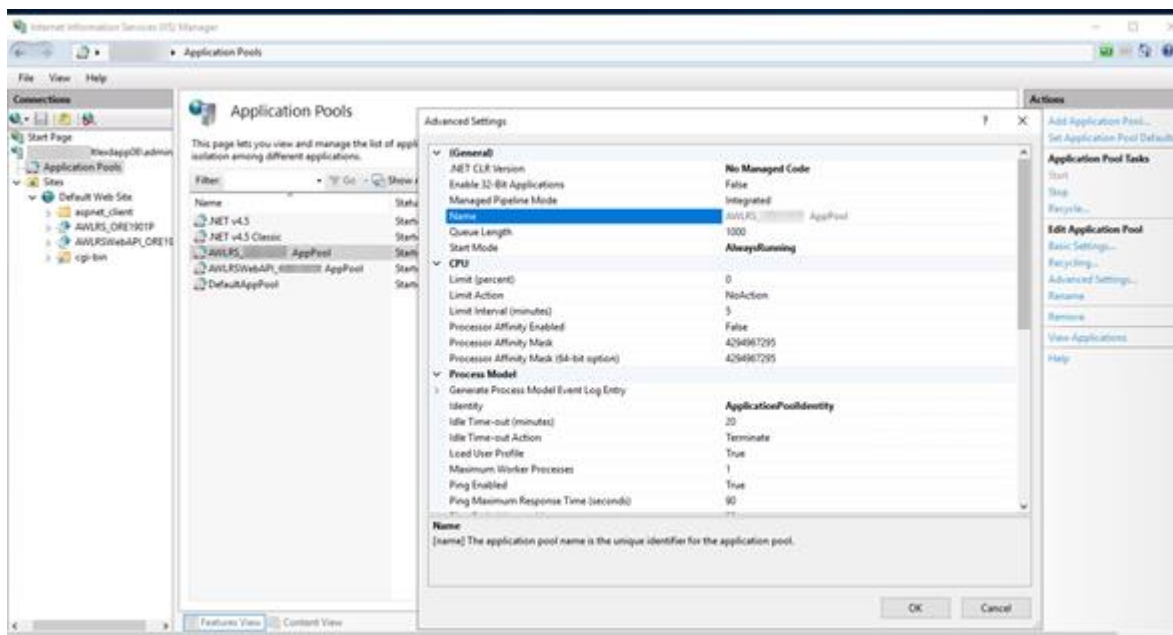
### 4.6 Application Pool Separation

The AWLRS applications need to sit in their own application Pools. Create a new application pool for the AWLRS application by navigating to application pools and selecting ‘Add Application Pool’.



Set ‘.NET CLR Version’ to ‘No Managed Code’

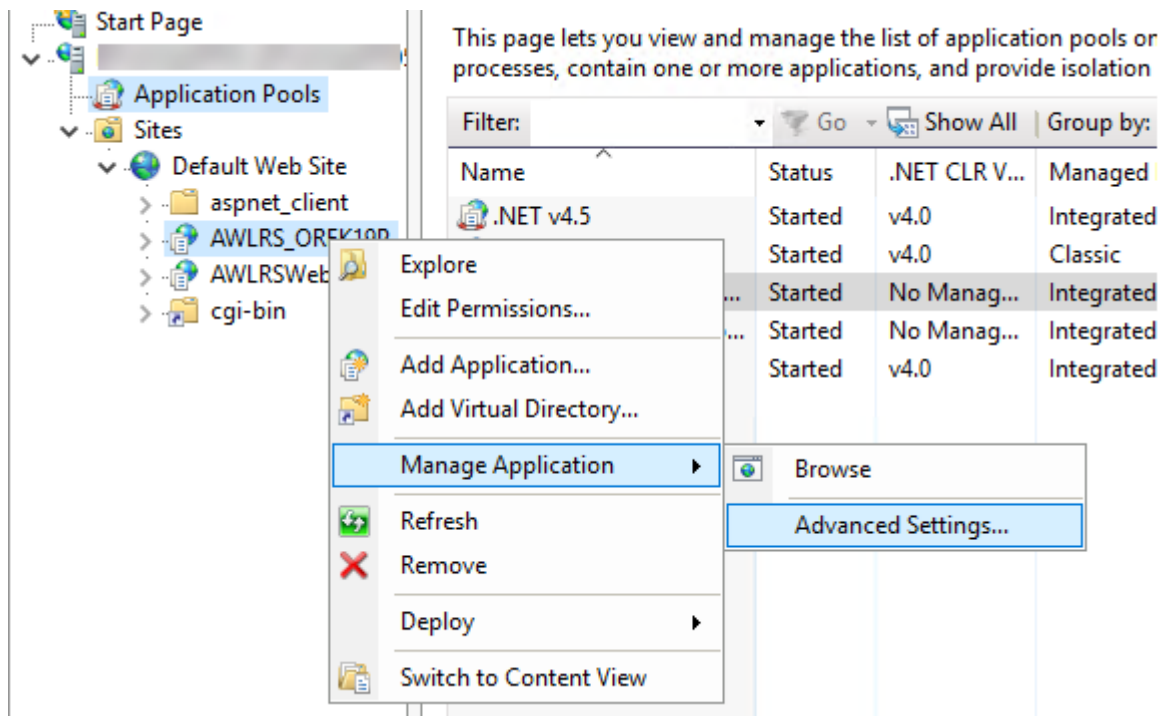
Once created navigate to the newly created pool and select advanced properties and ensure Start Mode is set to ‘AlwaysRunning’



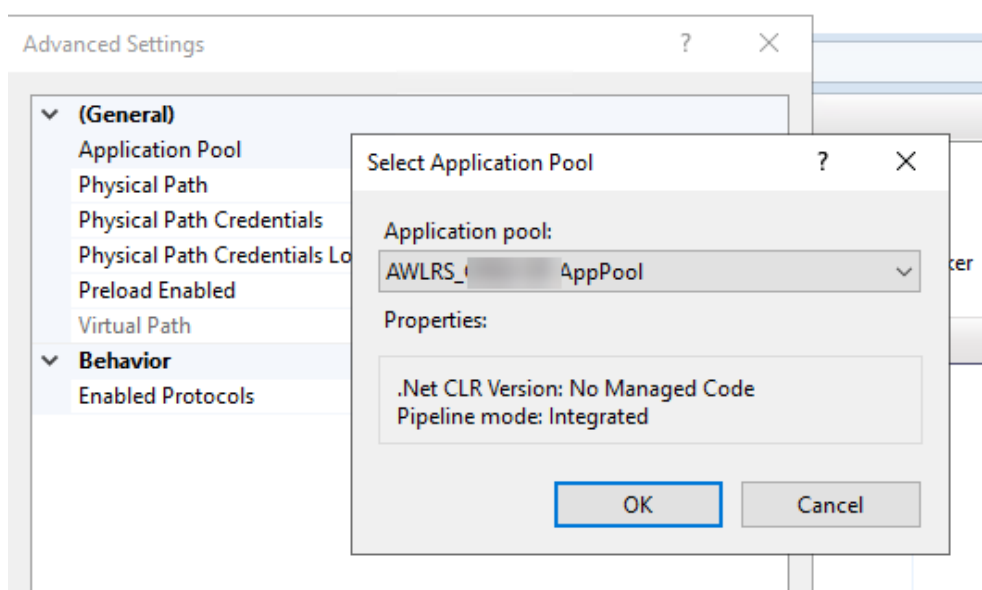
Repeat this process for the AWLRS WebAPI application pool.

## INSTALLATION

Once these application pools are created then the AWLRS and AWLRS WebAPI deployments should be assigned to use them. Select the application in the tree for example AWLRS. Right Click > Manage application > Advanced Settings



Select the application pool the deployment should be assigned to and click ok. This should be done for both AWLRS and AWLRS WebAPI.

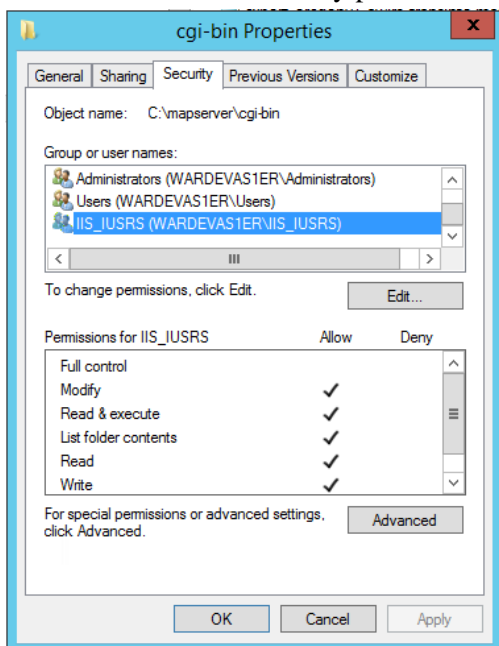


## 4.7 Permissions

The following permissions must be assigned to allow AWLRS to function correctly. The installation instructions are based on installing the product to execute within the default app pool. If there is a requirement to use a custom application pool, then additional care for permissions may be needed. In addition, local security policies may add further restrictions that must be observed, such detail is beyond the scope of this document.

### 4.7.1 CGI Bin

Add IIS\_IUSRS with modify permission to relevant CGI\_BIN location.



This grants sufficient permissions to all automatically generated application pools such as the default application pool and custom application pools using operating system managed identities. As a result, any application pool identity added to IIS\_IUSRS (either automatically or manually) will have permissions to the CGI\_BIN folder.

### 4.7.2 Log Files

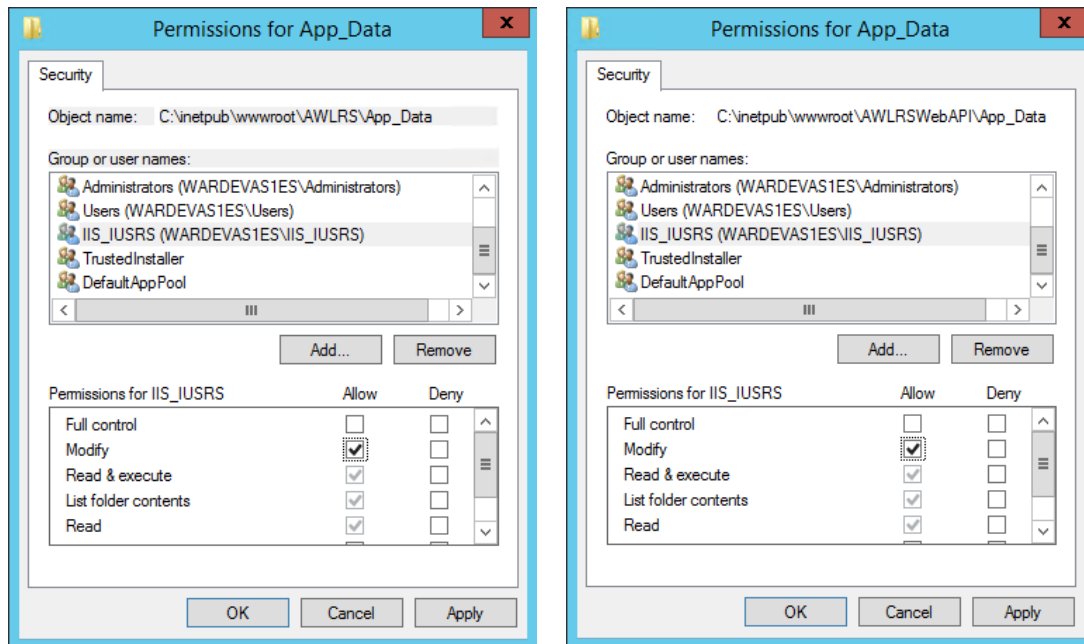
Following a standard installation the product will make use of two predefined folders to write any log information that may be generated, these folders are located on the **application server** and are named as follows:

- [AWLRS UI application deployment path]\App\_Data  
For example, C:\inetpub\wwwroot\AWLRS\App\_Data
- [AWLRS Web API deployment path]\App\_Data  
For example, C:\inetpub\wwwroot\AWLRSWebAPI\App\_Data

The relevant permissions must be granted for the application pool identity. In general it may be appropriate the “modify” windows security privilege for “IIS\_IUSRS” to the chosen logging folders on the **application server** as follows:



## INSTALLATION



As this will grant all identities that belong to the IIS\_IUSRS group permissions to the folders then you might choose to assign the permissions to the application pool identity rather than the group to restrict access to the folders as needed, for instance when not using the default application pool and you wish to ensure restricted access (such as might be needed in a multi-tenant configuration).

If choosing to grant permissions to the relevant folder by application pool identity, ensure to use the actual app pool identity (as named by the application pool name) e.g.

**IIS AppPool\AwlrAppPool**) or whatever this was called when the application pool was set up.

If you wish to use folders other than the defaults then a new log folder can be specified. To do this the following changes are required:

Open the AWLRS UI application Web.config file (for example, C:\inetpub\wwwroot\AWLRS\Web.config).

Find the <configuration><log4net><file> element and change the “value” attribute to the path and filename required (note it is important to specify the filename as well as the path):

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <!--
6  <configuration>
7    <configSections>
10   <appSettings>
17   <log4net>
18     <appender name="FileAppender" type="log4net.Appender.FileAppender">
19       <file value="C:\awlrlogs\awlrssitelog.txt" />
20       <lockingModel type="log4net.Appender.FileAppender+MinimalLock" />

```

Open the AWLRS Web Api log4net.config file (for example C:\inetpub\wwwroot\AWLRSWebAPI\Configs\log4net.config)

## INSTALLATION

Find the `<log4net><appender><file>` element and change the “value” attribute to the path and filename required (note it is important to specify the filename as well as the path):

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <log4net>
3    <appender name="FileAppender" type="log4net.Appender.RollingFileAppender">
4      <file value="C:\awlrlogs\awlr.log" />
5      <lockingModel type="log4net.Appender.FileAppender+MinimalLock" />

```

Clearly if you change the logging folders from those of the default configuration then the permissions granted to the application identities (as described earlier in this section) must be adjusted accordingly.

### 4.7.3 Configure Web Service Origins

**Note:** This is for information only. You should have set this during installation in section 4.3 and can skip to the next section. The information provided is to allow correction or update of CORS settings after an installation has already been carried out.

To limit the potential for a CORS attack the AWLRS Web API allows you to set the Access-Control-Allow-Origin CORS response Header. By default, it would be set to “\*” to allow all domains. You can restrict this to the domains you want to allow (e.g. your AWLRS application servers) by adding their addresses to the CorsOrigins setting in the AWLRS Web API web.config file.

The CorsOrigins is an Application Setting in the **AWLRS Web API** Web.config file (for example, C:\inetpub\wwwroot\AWLRSWebAPI\Web.config). Open this file in a suitable text editor (ensure you have permissions to edit and save the file, make a backup prior to editing if you wish). Then in the **appSettings** section of the file add the **CorsOrigins** key with a value that contains a comma separated list of the origins you want to limit access to. The example on line 14 in the image below shows a setting to allow access from two fictional servers:

[https://AWLRS\\_Server1.your\\_domain.com](https://AWLRS_Server1.your_domain.com) and [https://AWLRS\\_Server2.your\\_domain.com](https://AWLRS_Server2.your_domain.com)

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <!--
3    For more information on how to configure your ASP.NET application, please visit
4    http://go.microsoft.com/fwlink/?LinkId=301879
5  -->
6  <configuration>
7    <configSections>
8      <section name="log4net" type="log4net.Config.Log4NetConfigurationSectionHandler,log4net,Version=1.2.12.0,Culture=n
9      <section name="unity" type="Microsoft.Practices.Unity.Configuration.UnityConfigurationSection, Microsoft.Practices
10     </configSections>
11     <unity configSource=".\\Configs\\unity.config" />
12     <log4net configSource=".\\Configs\\log4net.config" />
13     <appSettings>
14       <add key="CorsOrigins" value="https://AWLRS_Server1.your_domain.com,https://AWLRS_Server2.your_domain.com" />
15       <add key="AppConnectionName" value="AWLRSConnection" />

```

Note: if this setting is not made correctly you may lose access to AWLRS. If you have problems with access to AWLRS (e.g. HTTP 400/500 error) you should remove the CorsOrigins line from **AWLRS Web API** Web.config file and try again. If access is restored, then this would indicate an incorrect setting that requires correction before applying again.

## 5 Post Install/Upgrade Tasks

### 5.1 Product Options

AWLRS has several product options that need to be set prior to starting the application. For example:

Option Id	Description	Default Value
<b>AWLMAPEPSG</b>	AWLRS Map Output Projection	<PLEASE SET>
<b>AWLMAPSRIID</b>	AWLRS Map Default SRID	<PLEASE SET>

For more information on all AWLRS product options please refer to the appropriate version of the “AWLRS Administration Guide”.

### 5.2 Help file

A default Help file is provided as part of the AWLRS deployment.

For more information regarding updating the Help file with the latest release please refer to the appropriate version of the “AWLRS Administration Guide”.

### 5.3 Fontset file

MapServer users a Fontset file to locate any fonts used for Text Labels, Legend Graphics etc used in the applications’ map.

A default Fontset file is provided as part of the AWLRS Web Api deployment, please copy the fonts.list file:

From:

[**AWLRS Web Api**]\App\_Data\

For example, C:\inetpub\wwwroot\AWLRSWebAPI\AppData

To:

<**Mapserver**>\cgi-bin\

For example, C:\mapserver\cgi-bin

For more information regarding editing the Fontset file please refer to the appropriate version of the “AWLRS Administration Guide”.

**NB. If AWLRS is deployed to several load balanced servers this needs to be done on each of them.**

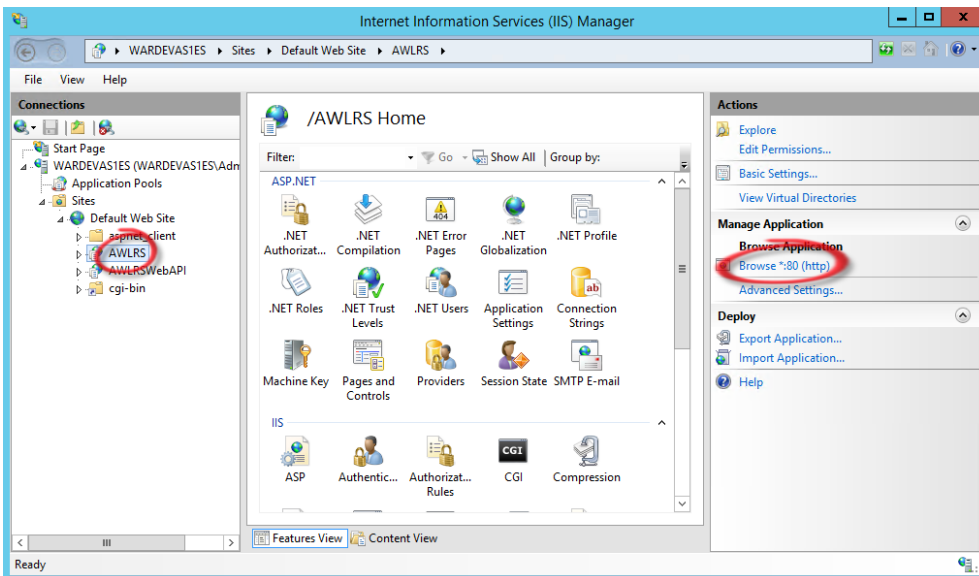
## 5.4 Write Map Files

Any image files used as the symbol for point layers need to be written to the Mapserver cgi-bin folder before running the application for the first time **or** when any changes involving image files have been made to the AWLRS Map in Oracle Map Builder.

**NB. Please refer to the appropriate version of the “AWLRS Administration Guide” and the exor core document “Spatial Server Admin” for details on how to use Oracle Map Builder to configure a Base Map for AWLRS.**

To write the image files to the mapserver cgi-bin you must first login to the application, the URL for the application can be derived in the following way:

On the IIS server run IIS Manager, select the AWLRS UI application (for example “AWLRS”) and click on “Browse \*:80 (https)” in the right-hand panel:



This will open a browser window, if all is well the AWLRS login page will be displayed, copy the URL from the address bar of the browser and replace “localhost” with the fully qualified name of the server, for example if the server name is awlrserver.bentley.com, the application path used in section 4.4 was “AWLRS” and the application was deployed to the Default Web Site then the starting URL will be:

<http://localhost/AWLRS/#/login>

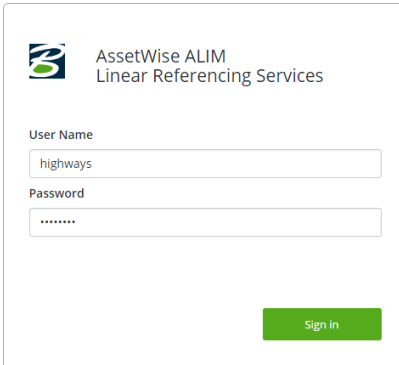
and should be changed to:

<http://awlrserver.bentley.com/AWLRS/#/login>

This is the URL that should be used to launch the AWLRS UI application.

## POST INSTALL/UPGRADE TASKS

Paste the new URL into the browser's address bar and login as the Highways Owner:

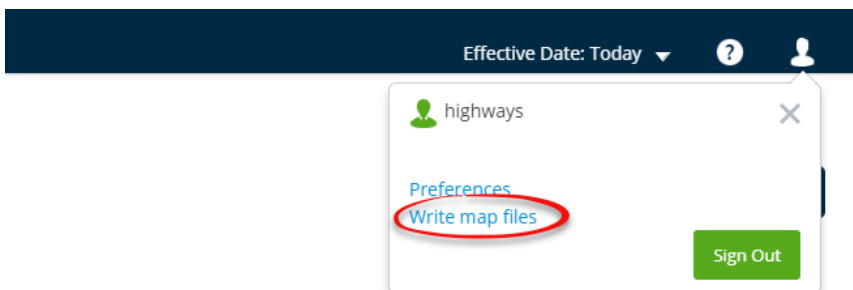


AssetWise ALIM  
Linear Referencing Services

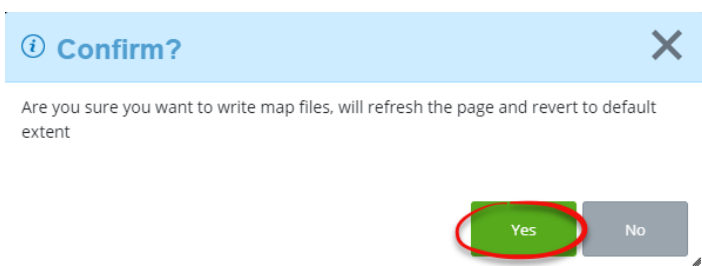
User Name

Password

Once logged in open the user menu located at the top right hand side to the application and click on the Write map files link.



When prompted for confirmation click on the “Yes” button to continue.



The application will write the required image files to the `<mapserver>\cgi-bin` folder and refresh the map, any layers that use the image files should now display correctly.

**NB. If AWLRS is deployed to several load balanced servers this needs to be done on each of them.  
i.e. Apply the above steps to each server using the direct (non load balanced) URLs.**

## 6 AWLRS Local Culture Setting

AWLRS makes use of a language tag to localize messages and labels shown to the user. This language tag will be referred to in this section as the Kendo UI culture code. As well as ensuring correct language specific messages are used, the Kendo UI culture code will ensure local regional settings are appropriate for such things like date formats, currency format etc.

A variety of Kendo UI culture codes are distributed within the application but if the required language is **not** distributed within the application use the following steps to install and configure the additional code.

- 1) Identify the required code of kendo-ui-culture. The following URL displays all available culture codes provided by kendo:

<https://github.com/telerik/kendo-ui-core/tree/2016.R3/src/cultures>

The file names will be of the pattern as shown below

kendo.culture.<language code>[-<country/region code>].js

For example

kendo.culture.en-GB.js

The code used within AWLRS config detailed below in this example will be “en-GB”

- 2) Configure AWLRS for the required culture.
  - a. Edit [AWLRS UI application deployment path]\app\config\config.json
  - b. Edit the variable kendoUICulture and change its value to required code which we got in Step 1.

For example:

```
{
  "appConfig": {
    "sysConfig": {
      "webApiUrl": "http://[redacted]/AWLRSWebApi/api",
      "kendoUICulture": "en-GB",
      "projectTitle": "AssetWise ALIM Linear Referencing Services",
      "version": "01.01.06.01",
      "keepAliveInterval": 300000,
      "externalIdP": false,
      "decimalFormat": "#.####",
      "decimalPlaces": 4,
    }
  }
}
```

- 3) Add culture files to AWLRS .
  1. Get the required kendo.culture.<code>.js file from <https://github.com/telerik/kendo-ui-core/tree/2016.R3/src/cultures>
  2. Copy the required kendo.culture.<code>.js file to [AWLRS UI application deployment path]\Scripts/kendocultures\ folder.

- 4) Add the resources file to AWLRS.
  1. Create the required resources file by copying any existing resource file of any language and change the values within it to the appropriate language (where required).
  2. The file name pattern should be as below  
resources-locale\_<code>.js Eg. resources-locale\_&b>en-GB.js
  3. If we create for English and for Australia, the code will be en-AU. The file name will be resources-locale\_en-AU.js
  4. Copy this to [AWLRS UI application deployment path]\app\plugins\map\config\i18n folder.
- 5) Restart the IIS and AWLRS and the application will start using new kendo-ui-culture.

**Note:**

If a new code is added and there is no kendo.culture.<code>.js file, then by default, kendo will use en-US .

For every <code> added, the following files are needed:

kendo.culture.<code>.js within:

[AWLRS UI application deployment path]\Scripts\kendocultures\

resources-locale\_<code>.js file within:

## 7 AWLRS Local Password Policy and Email Address Criteria

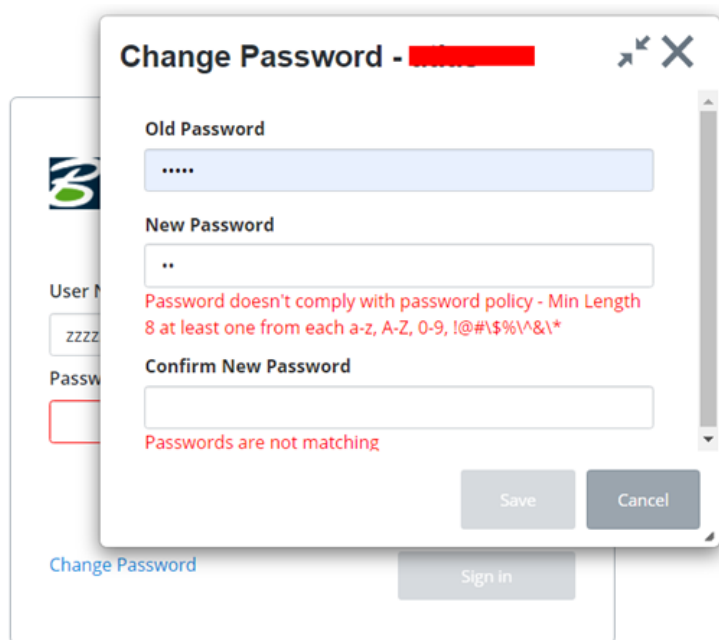
AWLRS can impose a password policy in the UI when users attempt to modify or create a new password. Email address creation is also tested to ensure the email address proposed conforms to basic RFC 822 structure. Both can be configured to better suit local policies or can be removed entirely. To alter the settings the following file must be edited (you are advised to backup the file prior to changing it):

[AWLRS UI application deployment path]\app\config\config.json

Towards the end of the file is a section like the following:

```
"configurationConfig": {
  "regExp": "^(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*[!@#\\$%\\^&\\*]) (?=.{8,})",
  "passwordHint": "Min Length 8 at least one from each a-z, A-Z, 0-9, !@#\\$%\\^&\\*",
  "regExpEmail": "([A-Za-z0-9][._]?)+[A-Za-z0-9]@[A-Za-z0-9_\\-]+\\.([A-Za-z0-9]{2,})",
  "emailHint": "You have entered an invalid e-mail address."
},
```

The value of the **regExp** setting is a regular expression controlling the password policy. The value of **passwordHint** describes the policy in a message that will be shown to the user to assist them in setting a suitable password, for example:



The value of the **regExpEmail** setting and value of **emailHint** set and prompt for the email address in a similar fashion.

Altering the values of those keys will allow control over the policy applied by that instance of AWLRS. Removal of the **regExp** or **regExpEmail** will turn off the respective policy in the UI allowing the user to provide untested data which may be refused by the lower layers of the application if the value fails further tests during later processing.