

# Content Manager

Software Version 10.1

## OpenID Connect authentication



Document Release Date: October 2024  
Software Release Date: October 2024

## Legal notices

### Copyright notice

© Copyright 2008-2024 Micro Focus or one of its affiliates.

The only warranties for products and services of Micro Focus and its affiliates and licensors ("Micro Focus") are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Micro Focus shall not be liable for technical or editorial errors or omissions contained herein. The information contained herein is subject to change without notice.

Contains Confidential Information. Except as specifically indicated otherwise, a valid license is required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

## Documentation updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

You can check for more recent versions of a document through the [MySupport portal](#). Many areas of the portal, including the one for documentation, require you to sign in with a Software Passport. If you need a Passport, you can create one when prompted to sign in.

Additionally, if you subscribe to the appropriate product support service, you will receive new or updated editions of documentation. Contact your Micro Focus sales representative for details.

To check for recent updates or to verify that you are using the most recent edition of a document, contact your Client Director.

## Support

Visit the [MySupport portal](#) to access contact information and details about the products, services, and support that Micro Focus offers.

This portal also provides customer self-solve capabilities. It gives you a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the MySupport portal to:

- Search for knowledge documents of interest
- Access product documentation
- View software vulnerability alerts
- Enter into discussions with other software customers
- Download software patches
- Manage software licenses, downloads, and support contracts
- Submit and track service requests
- Contact customer support
- View information about all services that Support offers

Many areas of the portal require you to sign in with a Software Passport. If you need a Passport, you can create one when prompted to sign in. To learn about the different access levels the portal uses, see the [Access Levels descriptions](#).

For information and details about the products, services, and support that Micro Focus offers, contact your Client Director.

# Contents

|   |    |
|---|----|
| Introduction .....  | 5  |
| ADFS for Native client .....  | 6  |
| Configure ADFS .....  | 6  |
| Configure OpenID settings in Content Manager .....                  | 13 |
| ADFS for Web Client and Service API .....                           | 16 |
| Create the ADFS application .....                                   | 16 |
| Add the settings to the Web Client .....                            | 18 |
| Configure ADFS for the Office integration access .....              | 18 |
| Add Office integration to the settings of the Web Client .....      | 20 |
| Azure AD for WebClient, Mobile App and Service API .....            | 21 |
| Create the Azure AD application .....                               | 21 |
| Configure for mobile .....  | 23 |
| Add the mobile redirect URI .....                                   | 23 |
| Add the mobile redirect URI to the Service API .....                | 23 |
| Configure authentication in hptrim.config .....                     | 23 |
| Enable redirect .....   | 24 |
| Allow users (Web Client only) .....                                 | 24 |
| Logout .....  | 25 |
| Allow anonymous access in IIS .....                                 | 25 |
| Azure AD for Content Manager desktop .....                          | 26 |
| Create the Azure AD application .....                               | 26 |
| Configure authentication in Content Manager Enterprise Studio ..... | 27 |
| Configure Azure AD for Office integration access .....              | 28 |
| Troubleshooting Azure AD for Office integration access .....        | 30 |
| Google authentication .....   | 31 |
| Create the Google credentials .....                                 | 31 |
| Configure authentication in hptrim.config .....                     | 31 |

Enable redirect .....32

Logout ..... 32

Allow anonymous access in the IIS .....32

# Introduction

This document provides high level information for you to configure OpenID Connect authentication with Content Manager.

OAuth authentication is managed via OpenID Connect authentication. The authentication is configured in your Identity Provider (e.g. Azure AD) and then the appropriate details are stored in Content Manager in the **hptrim.config** file for the Web Client and Service API, and in Content Manager Enterprise Studio for the desktop client.

# ADFS for Native client

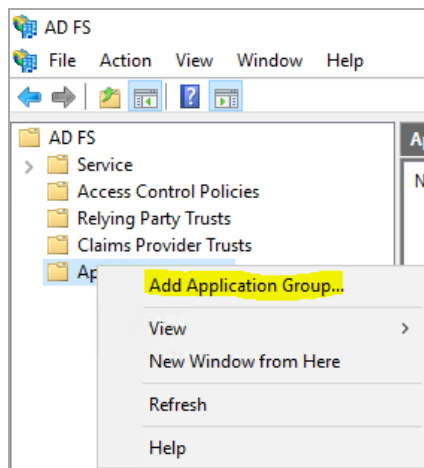
As of Content Manager 10 the Content Manager web applications (Service API, WebDrawer and Web Client) have an OpenID Connect authentication provider built in.

This section describes creating an ADFS application and configuring the Windows native Content Manager client.

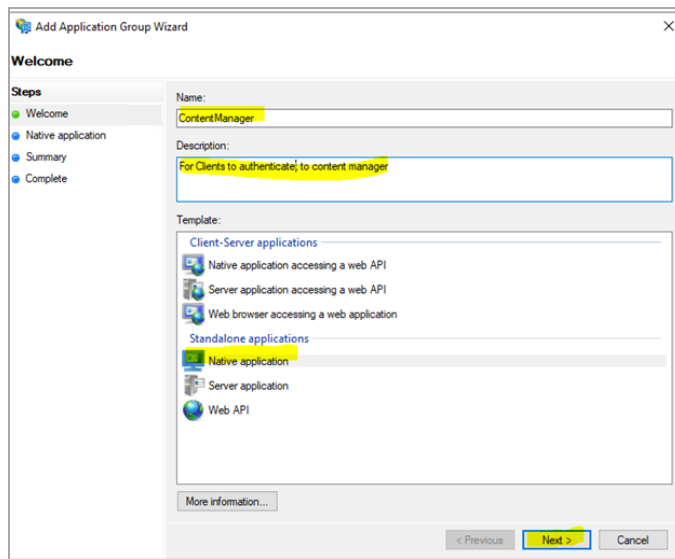
## Configure ADFS

To setup ADFS to support OpenID for Content Manager native, perform the following steps:

1. Log on to your ADFS Server and create an application group.



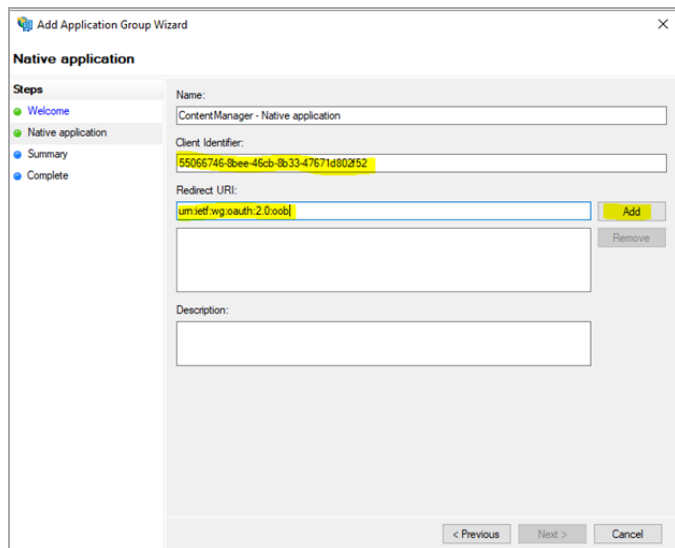
2. Enter the name and the description for the group, then select **Native application**. Click **Next**.



The 'Add Application Group Wizard' window is shown at the 'Welcome' step. The 'Steps' pane on the left lists 'Welcome', 'Native application', 'Summary', and 'Complete'. The main area contains fields for 'Name' (ContentManager) and 'Description' (For Clients to authenticate to content manager). Below these is a 'Template' section with two categories: 'Client-Server applications' (Native application accessing a web API, Server application accessing a web API, Web browser accessing a web application) and 'Standalone applications' (Native application, Server application, Web API). The 'Native application' option is selected. At the bottom are 'More information...', '< Previous', 'Next >', and 'Cancel' buttons.

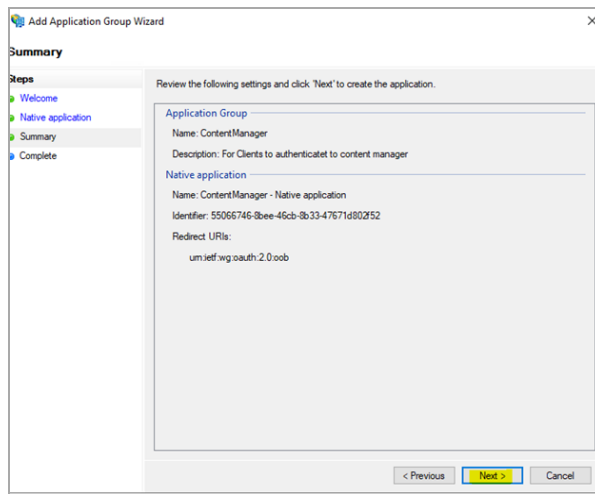
3. Configure the group.

Note the client identifier for later use and enter the value 'https://127.0.0.1' in Redirect URI, click **Add** and then click **Next**.



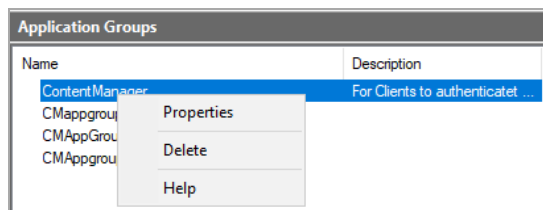
The 'Add Application Group Wizard' window is shown at the 'Native application' step. The 'Steps' pane on the left lists 'Welcome', 'Native application', 'Summary', and 'Complete'. The main area contains fields for 'Name' (ContentManager - Native application), 'Client Identifier' (55065745-8bee-46cb-8b33-47671d802f52), 'Redirect URI' (um.ietf.vig.oauth.2.0.oob), and 'Description'. There are 'Add' and 'Remove' buttons next to the 'Redirect URI' field. At the bottom are '< Previous', 'Next >', and 'Cancel' buttons.

4. Review the settings and click **Next**.

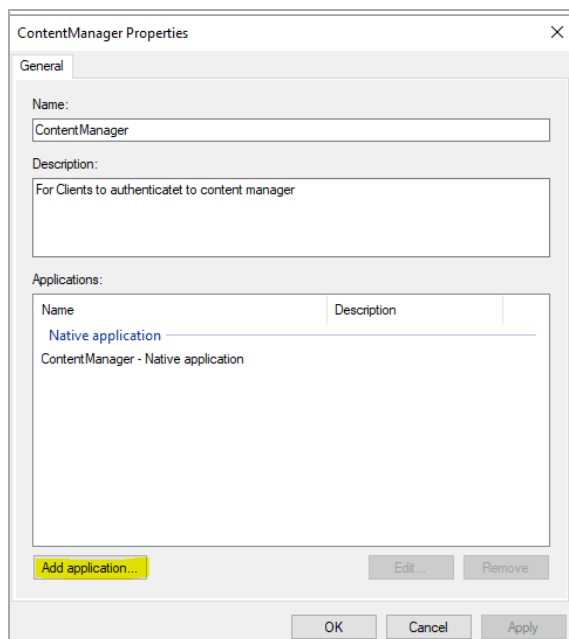


5. Modify application group properties.

Once this has been completed, right-click the application group and select **Properties**.

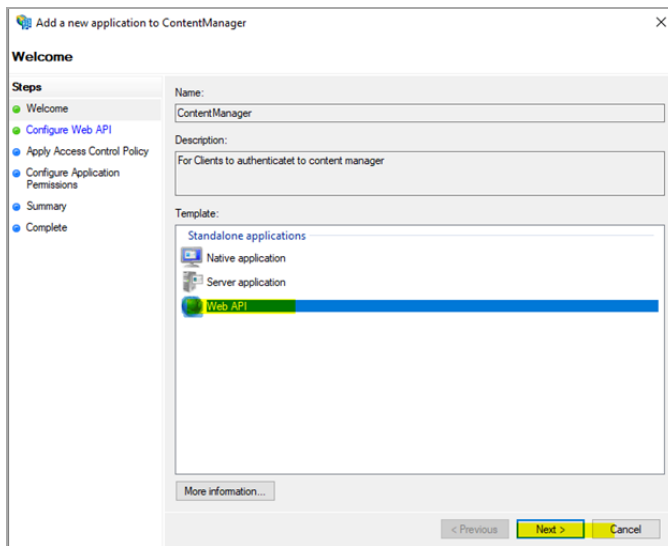


6. Click **Add application**.



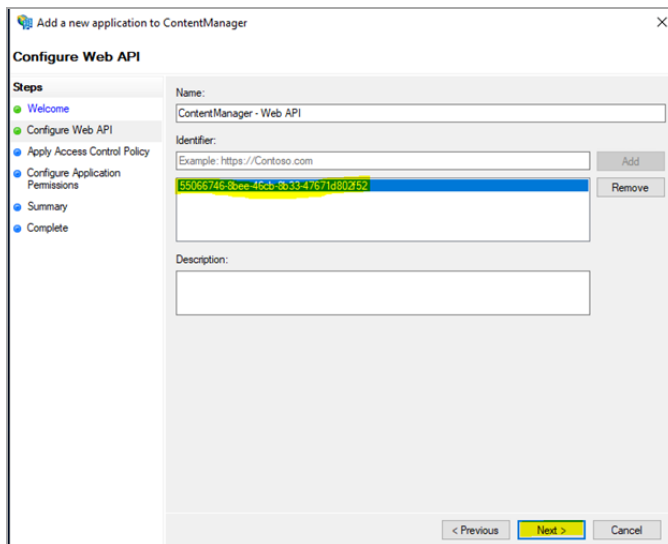
7. Select **Web API** as the **Template**.



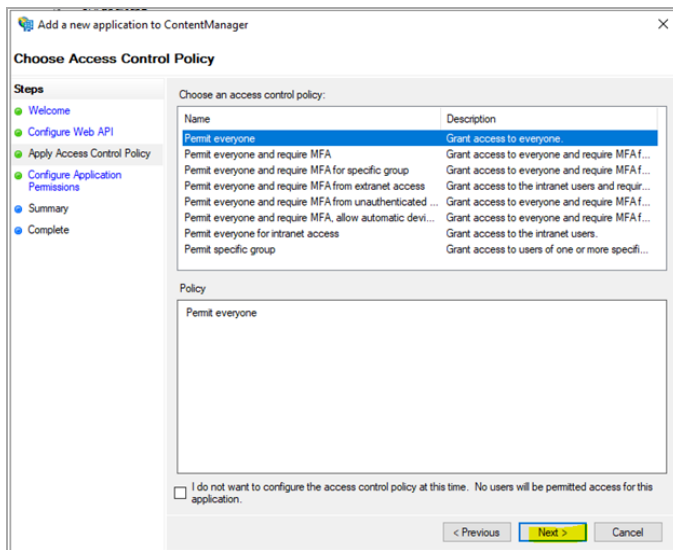


8. Configure the Web API.

Add the Redirect to be the same as the client ID of the Native application that you previously copied. Click **Next**.

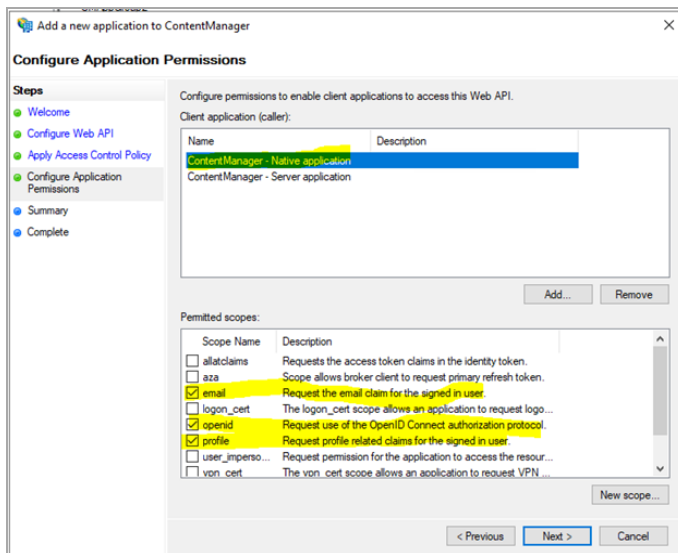


9. Select an appropriate Access Control Policy.

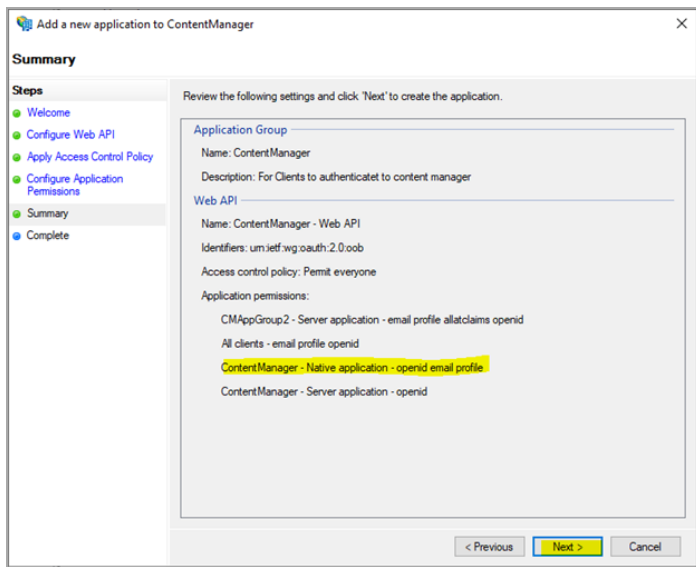


10. Set the scopes.

Select the Native Application and ensure **email**, **openid**, and **profile** are checked.

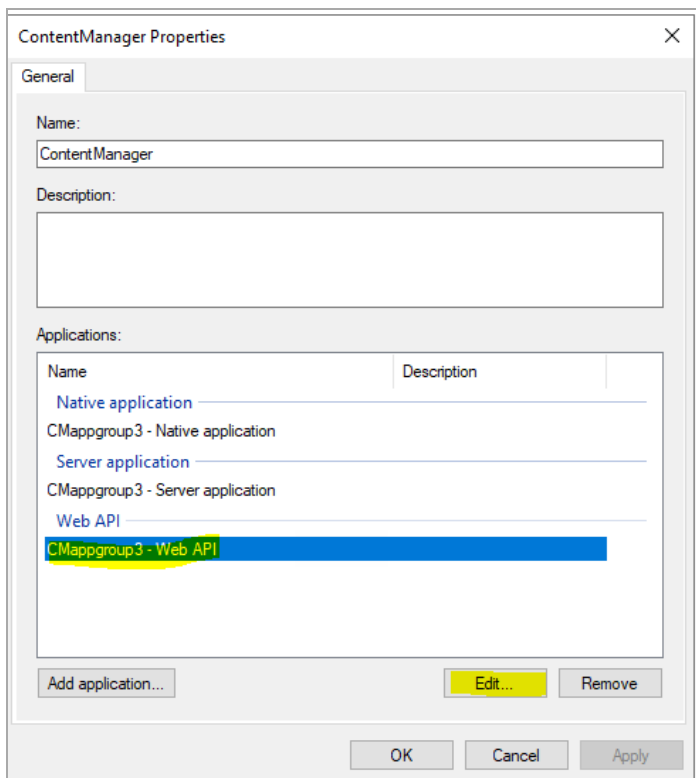


11. Review the configuration, click **Next** and then **Close**.



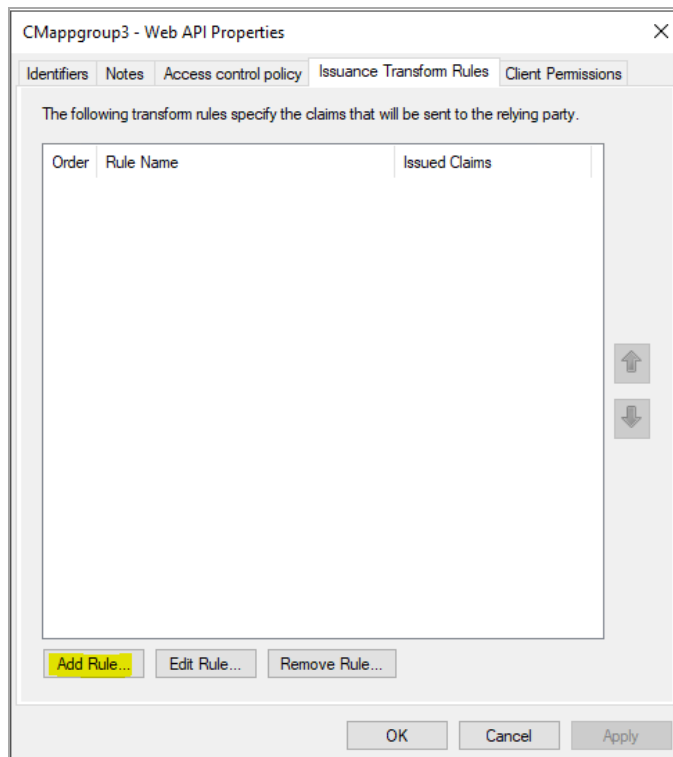
12. Set issuance rules.

Right-click the app group and select **Properties**, highlight the WEB API application and select **Edit**.



13. Add a rule.

Navigate to the **Issuance Transform Rules** Tab and click **Add Rule**.



14. Map attributes.

Give the rule a name, select **Active Directory** as the **Attribute Store**, and add the following mapping of LDAP attributes and click **OK**:

- **E-Mail-Addresses - E-mail Address**
- **User-Principal-Name - UPN**

**Edit Rule - Claim rule**

You can configure this rule to send the values of LDAP attributes as claims. Select an attribute store from which to extract LDAP attributes. Specify how the attributes will map to the outgoing claim types that will be issued from the rule.

Claim rule name:

Rule template: Send LDAP Attributes as Claims

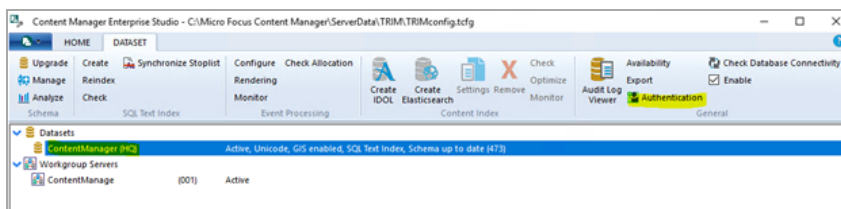
Attribute store:

Mapping of LDAP attributes to outgoing claim types:

|   | LDAP Attribute (Select or type to add more) | Outgoing Claim Type (Select or type to add more) |
|---|---|--|
| ▶ | E-Mail-Addresses                            | E-Mail Address                                   |
|   | User-Principal-Name                         | UPN  |
| * |   |  |

## Configure OpenID settings in Content Manager

1. Navigate to the Content Manager server and run the Content Manager Enterprise Studio as an Administrator. Right-click the dataset and select **Authentication**.



2. Enter the ADFS URL for the ADFS server, the client identifier which was noted earlier (Native Application) and click **Test Authentication**.

Configure Authentication Mechanisms for Dataset: ContentManager (HQ)

Windows OpenID Connect OneDrive Connect

☒ Enable OpenID Connect authentication

OAuth 2.0 Configuration

OpenID Issuer URL:

Authorize Endpoint URL:

Token Endpoint URL:

Client ID:

Client App Scope:

Client Redirect URI:

OpenID Connect Confidential Clients

| Client ID | Scope |
|-----------|-------|
|-----------|-------|

3. Enter the user details for a test user and click **Sign in**.

Content Manager Third Party Authentication

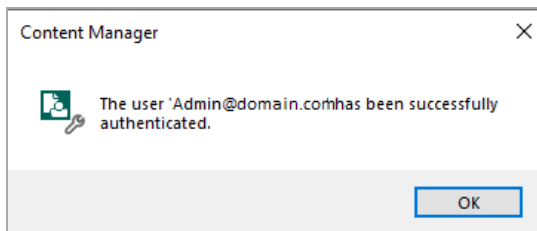
Resource Domain

Sign in

Sign in as current user

© 2018 Microsoft

The users email in active directory should appear, if successful.



# ADFS for Web Client and Service API

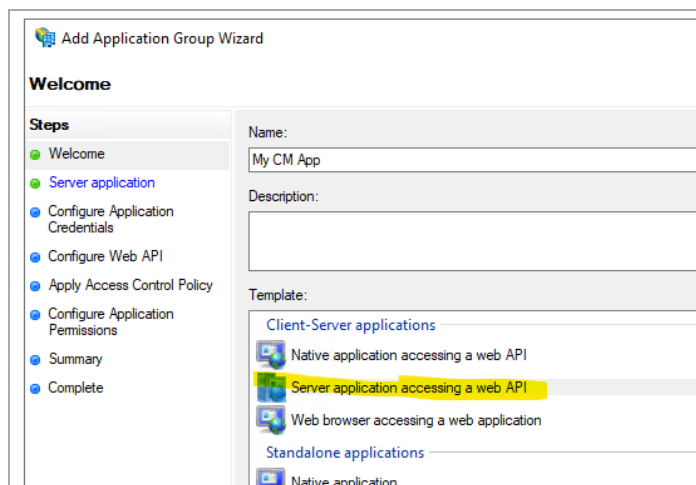
As of Content Manager 10 the web applications (Service API, WebDrawer and Web Client) have a built in OpenID Connect authentication provider.

This section describes creating an ADFS application and configuring the Content Manager web applications.

## Create the ADFS application

To create the ADFS application,

1. Create a new **Application Group**.
2. Select **Server application accessing a web API** as the template.



3. Enter a name and click **Next**.

Note the client identifier.

4. Add a Redirect URI.

The redirect URI must be in lowercase and be the URL of the Content Manager web site with the suffix `/serviceapi/auth/openid`.

For example, `https://myserver/contentmanager/serviceapi/auth/openid`.



The screenshot shows the 'CM Web Client DC - Server application Properties' dialog box with the 'General' tab selected. The 'Name' field contains 'CM Web Client DC - Server application'. The 'Client Id' field contains '9eeda8c0b3a2-43c5-aa45-34930f0dc8ff'. The 'Redirect URI' field contains 'https://auchurchla02/cm/serviceapi/auth/openid'. There are 'Add' and 'Remove' buttons next to the 'Redirect URI' field. The 'Description' field is empty.

5. Click **Next**.
6. Generate a shared secret and note the secret.

The screenshot shows the 'Configure Application Credentials' dialog box. On the left, there is a 'Steps' pane with 'Configure Application Credentials' selected. The main area has the following options:

- ☐ Register a key used to sign JSON Web Tokens for authentication. Below it is a 'Configure...' button.
- ☐ Windows Integrated Authentication. Below it is a 'Select the AD Account:' label, an example text 'Example: CONTOSO\expensercv', and a 'Select...' button.
- ☒ Generate a shared secret. Below it is a 'Secret:' label, a text field containing 'uEa-ZY4-SNLwAW9mFoLD23ax8sWso2bZxvilm\_', and a 'Copy to clipboard' button.

A yellow information box at the bottom states: 'Copy and save the secret. You will not be able to view the secret after the application group is created. You can reset the secret later if required.'

7. Click **Next**.
8. Add an identifier. For example, <https://MyServer/contentmanager/>.

The screenshot shows the 'Add Application Group Wizard' dialog box, 'Configure Web API' step. On the left, there is a 'Steps' pane with 'Configure Web API' selected. The main area has the following fields:

- 'Name:' field containing 'My CM App - Web API'.
- 'Identifier:' field containing 'https://btpvm2237.myexch19.com/contentmanager/'.
- 'Description:' field is empty.

9. Click **Next**.
10. Choose an access control policy. For example, Give access to everyone.
11. In the **Configure Application permissions**, select **email**, **openid**, and **profile**.
12. Complete rest of the steps in the Application Group.

## Add the settings to the Web Client

To configure the Web Client, edit the **hprmServiceApi.config** file and add (or edit) the authentication element to look similar to the example below.

- Client ID and secret (noted in the previous section)
- issuerUri is found in the ADFS console - Endpoints, in the OpenID Connect section.
- Name must be openid.

For example,

```
<authentication allowAnonymous="false"slidingSessionMinutes="30">
<openIdConnect>
<add
name="openid"
clientId="CLIENT_ID"
clientSecret="SECRET"
issuerURI="https://MyServer/adfs/.well-known/openid-configuration"/>
</openIdConnect>
</authentication>
```

## Configure ADFS for the Office integration access

The office integration requires an access token to allow it to authenticate with the Web Client, this can be configured in ADFS.

1. Go to the **Application group** configured in the above section.
2. Click **Add Application** to add a native application.

Add a new application to CM Web Client Sample

**Native application**

**Steps**

- Welcome
- Native application
- Summary
- Complete

Name: CM Web Client Sample - Native application 1

Client Identifier: b93b7949-4715-4238-9a66-01b9663b4e75

Redirect URI: Example: https://Contoso.com  
https://myserver/contentmanager

Ac Rem

Preserve the Client ID for later use.

3. Complete rest of the steps for the native application.
4. Edit the Web API application and in the **Client Permissions**, add the new client application selecting the **scopes, email, openid, and profile**.

**CM Web Client Sample - Web API Properties**

Identifiers Notes Access control policy Issuance Transform Rules **Client Permissions**

Configure application permissions to enable client applications to access this Web API

Client application (caller):

| Name   | Description |
|--|-------------|
| CM Web Client Sample - Server application        |             |
| <b>CM Web Client Sample - Native application</b> |             |

< Add... Remove

Permitted scopes:

| Scope Name                                  | Description   |
|---|---|
| <input type="checkbox"/> allatclaims        | Requests the access token claims in the identity token.       |
| <input type="checkbox"/> aza                | Scope allows broker client to request primary refresh token.  |
| <input checked="" type="checkbox"/> email   | Request the email claim for the signed in user.               |
| <input type="checkbox"/> logon_cert         | The logon_cert scope allows an application to request logo... |
| <input checked="" type="checkbox"/> openid  | Request use of the OpenID Connect authorization protocol.     |
| <input checked="" type="checkbox"/> profile | Request profile related claims for the signed in user.        |

5. In the **Issuance Transform Rules**, add a new Rule.
6. Select the **Send LDAP Attributes** and click **Next**.
7. Choose the **Active Directory** as the attribute store.
8. Map the following two claims:
  - **Display Name - Name**
  - **User-Principal-Name - UPN**

You can configure this rule to send the values of LDAP attributes as claims. Select an attribute store from which to extract LDAP attributes. Specify how the attributes will map to the outgoing claim types that will be issued from the rule.

Claim rule name:  
**MyRule**

Rule template: Send LDAP Attributes as Claims

Attribute store:  
Active Directory

Mapping of LDAP attributes to outgoing claim types:

|   | LDAP Attribute (Select or type to add more) | Outgoing Claim Type (Select or type to add more) |
|---|---|--|
| ▶ | Display-Name                                | Name   |
|   | User-Principal-Name                         | UPN  |
| * |   |  |

9. Click **Finish**.

## Add Office integration to the settings of the Web Client

The Office integration authentication settings are stored in the file ADFS\config.xml in the Web Client install directory:

- clientAuthority - the URL to your ADFS server
- clientResourceUri - the relying party identifier from the ADFS Web API
- clientID - the Client ID from the ADFS native application
- The Redirect URI from the ADFS native application

For example,

```
<adfsClient>  
<clientAuthority>https://test.com/adfs</clientAuthority>  
<clientResourceUri>https://test.com/contentmanager/</clientResourceUri>  
<clientID>ab999999-999d-9aeb-a999-999b999999a99</clientID>  
<clientReturnUri>https://test.com/contentmanager/</clientReturnUri>  
</adfsClient>
```

# Azure AD for WebClient, Mobile App and Service API

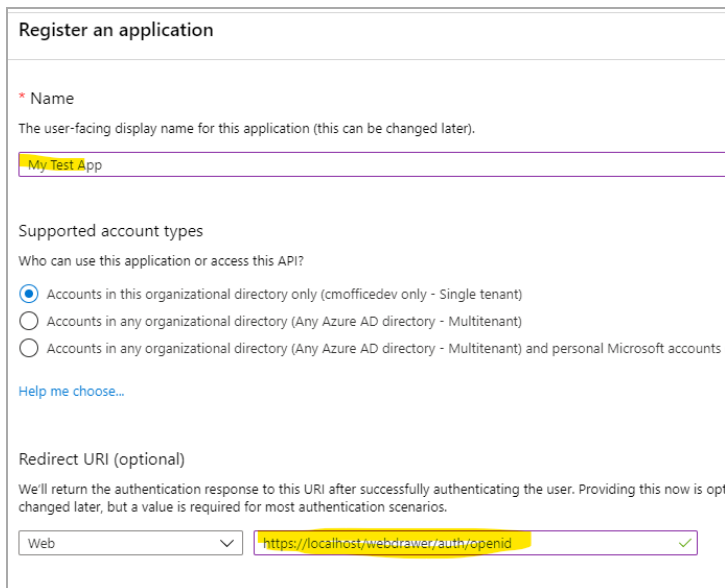
As of Content Manager 10 the web applications (Service API, WebDrawer and Web Client) have a built in OpenID Connect authentication provider.

This section describes creating an Azure AD application and configuring the web application. This document also provides detailed steps required to allow the Content Manager Mobile App to authenticate.

## Create the Azure AD application

To create the Azure AD application:

1. From the [portal.azure.com](https://portal.azure.com), go to Azure AD.
2. Go to **App Registrations** and select **New Registration**.



**Register an application**

\* Name  
The user-facing display name for this application (this can be changed later).

My Test App

Supported account types  
Who can use this application or access this API?

☒ Accounts in this organizational directory only (cmofficedev only - Single tenant)  
☐ Accounts in any organizational directory (Any Azure AD directory - Multitenant)  
☐ Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts

[Help me choose...](#)

Redirect URI (optional)  
We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional, but a value is required for most authentication scenarios.

Web

3. Enter a name.
4. Under Redirect URI leave **Web** selected.

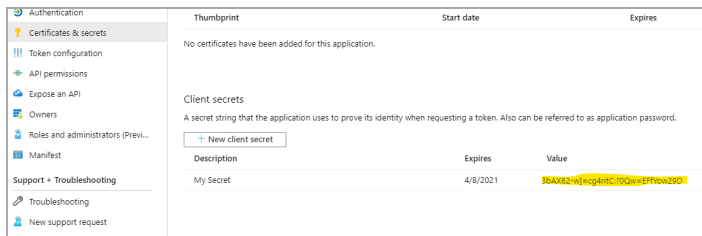
The value in the Redirect URI is important. It must be lowercase and must be the URL to your application. For example, <https://mydomain.com/cmwebdrawer> followed by the path to the authentication provider, for example, `/auth/openid`.

The /auth/ component is fixed but the 'openid' is the name you will supply in **hptrim.config** later and so can be any string, as long as it matches the value in **hptrim.config**.

For the Web Client the path must include the path to the Service API, for example, <https://mydomain.com/contentManager/serviceapi/auth/openid>.

5. Add a secret.

From **Certificates and Secrets**, add a secret and note the secret.



6. Configure Tokens.

From the **Authentication** section, select the tokens to be issued and check the option **ID Tokens**.

7. Configure permissions.

From the **API Permissions**, add the following delegated Microsoft Graph permissions:

- email
- offline\_access
- openid
- profile
- User.Read
- Files.Read
- Files.Read.All
- Files.ReadWrite
- Sites.Read.All
- Sites.ReadWrite.All
- User.Read

Select **Grant admin consent** to grant access to all permissions.

| + Add a permission ✓ Grant admin consent for cmtrunk |             |   |                        |
|--|-------------|---|------------------------|
| API / Permissions name                               | Type        | Description   | Admin consent required |
| ▼ Microsoft Graph (11)                               |             |   |                        |
| email  | Delegated   | View users' email address                           | No                     |
| Files.Read   | Delegated   | Read user files                                     | No                     |
| Files.Read.All                                       | Application | Read files in all site collections                  | Yes                    |
| Files.ReadWrite                                      | Delegated   | Have full access to user files                      | No                     |
| Files.ReadWrite.All                                  | Delegated   | Have full access to all files user can access       | No                     |
| offline_access                                       | Delegated   | Maintain access to data you have given it access to | No                     |
| openid   | Delegated   | Sign users in                                       | No                     |
| profile  | Delegated   | View users' basic profile                           | No                     |
| Sites.Read.All                                       | Delegated   | Read items in all site collections                  | No                     |
| Sites.ReadWrite.All                                  | Delegated   | Edit or delete items in all site collections        | No                     |
| User.Read  | Delegated   | Sign in and read user profile                       | No                     |

## Configure for mobile

### Add the mobile redirect URI

If this Azure app is to be used to connect to the Content Manager Mobile App then,

1. Go to the **Authentication** section and select **Add a platform**.
2. Choose **Mobile and desktop applications** and enter `trimapp://mobile` in the **Custom Redirect URIs** field.

### Add the mobile redirect URI to the Service API

The redirect URI set above must be included in the Service API `hptrim.config`. To do this, add the **authentication** and **openIdConnect** elements.

For example,

```
<authentication allowAnonymous="false"slidingSessionMinutes="60">
  <openIdConnect>
    <addname="mobile"
      clientId="[CLIENT ID from Azure App]"
      clientSecret="[SECRET from Azure APP]"
      issuerUri="[OpenID Connect metadata document from endpoints in Azure App]"
      redirectUri="trimapp://mobile" />
  </openIdConnect>
</authentication>
```

## Configure authentication in hptrim.config

To use the Azure AD app created above, edit the `hptrim.config` (or `hprmServiceAPI.config` in the Web Client) so that it has an authentication similar to the one below:

1. The name must match the last segment of the Redirect URI path.
2. Client ID is the application ID from the Azure AD Overview.
3. The secret is the one saved when creating the App. If it was not saved, created a new one in Certificates and Secrets.
4. Get the issuerUri from **Overview > Endpoints > OpenID Connect metadata document**.

For example,

```
<authentication allowAnonymous="false"slidingSessionMinutes="60">
<openIdConnect>
<add
name="openid"
clientId="ae99999d-99e9-9ecc-b9eb-99d9d999dd"
clientSecret="_MqXXXXXXXXXXXXXG[sp3GrMfD:"
issuerURI="https://login.microsoftonline.com/09999ee9-9999-9999-9d9a-
999999999/v2.0/.well-known/openid-configuration"/>
</openIdConnect>
</authentication>
```

## Enable redirect

The Web Client will not re-direct the authentication endpoint unless the Html feature is enabled in **hprmServiceAPI.config**, perform the following steps:

1. Edit the **hprmServiceAPI.config** file.
2. Find the property named **serviceFeatures**.
3. Add the feature **Html**.

## Allow users (Web Client only)

For the Web Client, find the **web.config** file and find the **Location** element with the path **serviceapi** in the **web.config** file. It should contain the element `<allow users="*" />` within its authorization element.

For example,

```
<location path="serviceapi">
<system.web>
<httpHandlers>
<add path="*" type="ServiceStack.WebHost.Endpoints.ServiceStackHttpHandlerFactory,
ServiceStack" verb="*" />
</httpHandlers>
<authorization>
<allow users="*" />
</authorization>
```



```
</system.web>  
...  
</location>
```

## Logout

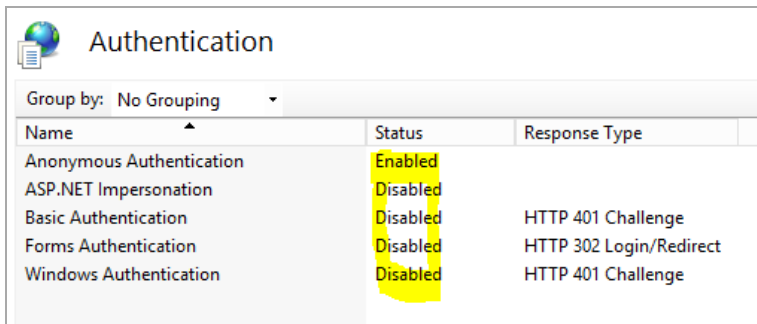
For WebDrawer the logout link is configured in the uiSettings. It should contain ~/auth/logout. In the Web Client a logout link will be displayed automatically when OpenID Connect authentication is enabled.

For example,

```
<uiSettings  
logoutLink="~/auth/logout"  
...  
>
```

## Allow anonymous access in IIS

The IIS will not handle authentication, so use IIS Manager to allow anonymous access only.



The screenshot shows the 'Authentication' section of the IIS Manager console. A table lists various authentication methods and their status. 'Anonymous Authentication' is highlighted in yellow and is 'Enabled'. All other methods are 'Disabled'.

| Name                     | Status   | Response Type           |
|--------------------------|----------|-------------------------|
| Anonymous Authentication | Enabled  |                         |
| ASP.NET Impersonation    | Disabled |                         |
| Basic Authentication     | Disabled | HTTP 401 Challenge      |
| Forms Authentication     | Disabled | HTTP 302 Login/Redirect |
| Windows Authentication   | Disabled | HTTP 401 Challenge      |

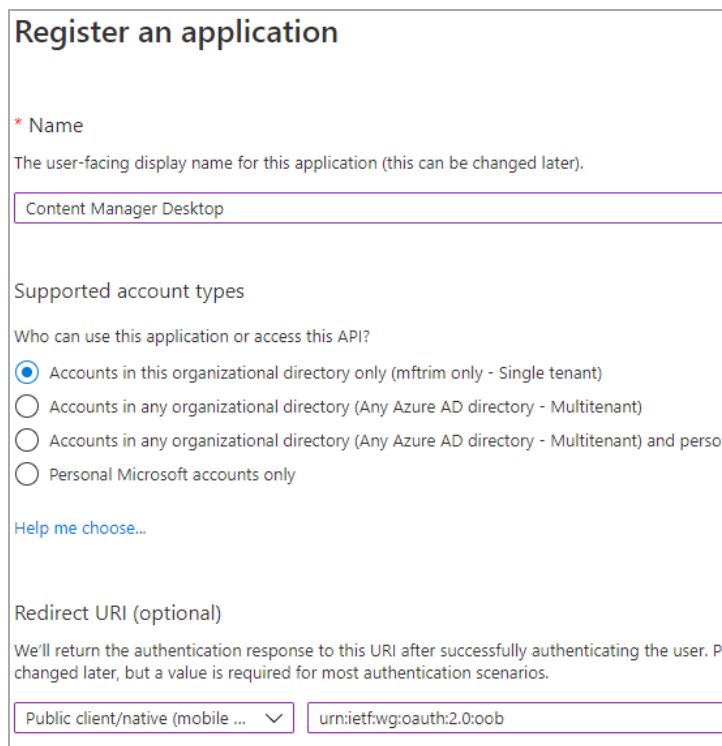
# Azure AD for Content Manager desktop

OpenID Connect may be used to authenticate with the Content Manager desktop client, this section describes how to configure this.

## Create the Azure AD application

To create the Azure AD application:

1. From the [portal.azure.com](https://portal.azure.com), go to Azure AD.
2. Go to **App Registrations** and select **New Registration**.



The screenshot shows the 'Register an application' form in the Azure portal. The form has the following sections:

- Name:** A text input field with the value 'Content Manager Desktop'. Above the field is a red asterisk and the label 'Name'. Below the field is a note: 'The user-facing display name for this application (this can be changed later).'
- Supported account types:** A section titled 'Who can use this application or access this API?'. It contains four radio button options:
  - ☒ Accounts in this organizational directory only (mftrim only - Single tenant)
  - ☐ Accounts in any organizational directory (Any Azure AD directory - Multitenant)
  - ☐ Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts
  - ☐ Personal Microsoft accounts onlyBelow these options is a link: 'Help me choose...'
- Redirect URI (optional):** A section with a note: 'We'll return the authentication response to this URI after successfully authenticating the user. This can be changed later, but a value is required for most authentication scenarios.' It contains a dropdown menu with 'Public client/native (mobile ...)' selected and a text input field with the value 'urn:ietf:wg:oauth:2.0:oob'.

3. Enter a name.
4. Select **Public native client** in the Redirect URI.
5. Enter the redirect Uri as 'urn:ietf:wg:oauth:2.0:oob'.
6. Configure permissions.

From the **API Permissions**, add the following delegated Microsoft Graph permissions:

- email
- offline\_access

- openid
- profile
- User.Read
- Files.Read
- Files.Read.All
- Files.ReadWrite
- Sites.Read.All
- Sites.ReadWrite.All
- User.Read

Select **Grant Admin Access** to grant access to all permissions.

| API / Permissions name | Type      | Description   | Admin |
|------------------------|-----------|---|-------|
| Microsoft Graph (11)   |           |   |       |
| email                  | Delegated | View users' email address                           | No    |
| Files.Read             | Delegated | Read user files                                     | No    |
| Files.Read.All         | Delegated | Read all files that user can access                 | No    |
| Files.ReadWrite        | Delegated | Have full access to user files                      | No    |
| Files.ReadWrite.All    | Delegated | Have full access to all files user can access       | No    |
| offline_access         | Delegated | Maintain access to data you have given it access to | No    |
| openid                 | Delegated | Sign users in                                       | No    |
| profile                | Delegated | View users' basic profile                           | No    |
| Sites.Read.All         | Delegated | Read items in all site collections                  | No    |
| Sites.ReadWrite.All    | Delegated | Edit or delete items in all site collections        | No    |
| User.Read              | Delegated | Sign in and read user profile                       | No    |

## Configure authentication in Content Manager Enterprise Studio

To use the Azure AD app created above, open the Content Manager Enterprise Studio and perform the following:

1. From the database, select **Authentication > OpenID**.
2. The OpenID Issuer URL is taken from the **Azure App - Overview > Endpoints > OpenID Connect metadata document**.
3. The Client ID is taken from the **Azure App - Overview > Application ID**.
4. Client secret should be empty.

5. Client app scope should contain **openid email offline\_access**.

The screenshot shows a dialog box titled 'Properties - DB244\_14' with a close button (X) in the top right corner. The dialog has four tabs: 'Authentication', 'Connection', 'Document Storage', and 'Options'. The 'Options' tab is selected. Under the 'Options' tab, there are four checkboxes: 'Enable Integrated Windows authentication (Active Directory)' (checked), 'Enable Explicit Windows authentication' (unchecked), 'Enable OpenID Connect authentication' (checked), and 'Enable OpenID Connect authentication' (checked). Below these checkboxes are several text input fields: 'OpenID Issuer URL' (empty), 'Client ID' (empty), 'Set Client Secret' (button), 'Client Scope' (containing 'openid email office\_access'), 'Identity Claim' (containing 'email'), and 'Redirect URL' (containing 'https://127.0.0.1'). There is a 'Test Authentication' button. At the bottom, there is a section for 'OpenID Connect Confidential Clients' with a table containing 'Client ID' and 'Scopes' columns, and buttons for 'Add', 'Remove', and 'Test'. The 'OK', 'Cancel', and 'Help' buttons are at the bottom right.

## Configure Azure AD for Office integration access

The office integration requires an access token to allow it to authenticate with the Web Client, this can be configured in Azure AD:

1. Create an Azure App for Web Authentication, you may use the one you created to authenticate with the Content Manager Web Client.
2. Create (or edit) the file ADFS\config.xml in your Content Manager Web Client installation folder and set it as follows:
  - **clientAuthority** - 'https://login.windows.net/' followed by your domain. For example, https://login.windows.net/cmofficetest.onmicrosoft.com
  - **clientResourceUri** - the Application ID URI from your Content Manager Web Client Azure App.
  - **clientID** - the Application ID from your Content Manager Desktop Azure App.
  - **clientReturnUri** - urn:ietf:wg:oauth:2.0:oob
3. From the Azure App for Web Authentication,

- a. Go to **Expose an API** and select **Add a Scope**.
  - b. Enter the following values:
    - **Scope name**: access\_as\_user
    - **Who can consent**: Admins and Users
    - **Admin consent display name**: Office can act as the user
    - **Admin consent description**: Enable Office to call the add-in's web APIs with the same rights as the current user
    - **User consent display name**: Office can act as you
    - **User consent description**: Enable Office to call the add-in's web APIs with the same rights that you have
  - c. Click **Add scope**.
4. From the Content Manager Desktop Azure App, go to **API Permissions** and perform the following:
- a. Select **Add a Permission** and choose **My APIs**.
  - b. Select the Web Client Application.
  - c. Select the **access\_as\_user permissions**.
  - d. Select **Add permission**.
5. The Web Client authentication information needs to be updated to be made aware of the new client, perform the following steps:
- a. From the Content Manager Desktop Azure App, copy the **Application ID URI**. For example, api://cf2501bd-19f9-4ad6-96dc-f5cf7b2b3bf9.
  - b. Edit the Web Client **hprmServiceAPI.config** file.
  - c. In the **add** element of the openIDConnect element, add a new attribute called **appIdURI**. This is a case sensitive name.
  - d. The value of **appIdURI** should be the **Application ID URI** from the Content Manager Desktop Azure App.

For example,

```
<adfsClient>
```

```
<clientAuthority>https://login.windows.net/cmofficetest.onmicrosoft.com</clientAuthority>
```

```
<clientResourceUri>api://testqa99:3000/09f0ec5c-87e9-4568-8b60-4eb3e20de75e</clientResourceUri>
```

```
<clientID>cf9999bd-10f0-4ad6-99dc-f5cf7b2b3bf5</clientID>
```

```
<clientReturnUri>http://MyWebClient</clientReturnUri>
```

```
</adfsClient>
```

## Troubleshooting Azure AD for Office integration access

Error AADSTS50011: The resource principal named `https://MYSERVER/contentmanager/` was not found in the tenant named `XXXX-XXXX-XXXXX-XXXXXX`.

If you get this error, try using the Client ID in `clientResourceUri`, rather than Application ID URI.

# Google authentication

As of Content Manager 10 the web applications (Service API, WebDrawer and Web Client) have a built in OpenID Connect authentication provider.

This section describes creating Google credentials and configuring the web application.

## Create the Google credentials

To create the Google credentials:

1. Go to <https://console.developers.google.com/>.
2. Select **Credentials > OAuth Client ID**.
3. Set Application type as **Web Application**.
4. Add your domain in the **Authorized JavaScript origins**.

The value in the Redirect URI is important. It must be lowercase and must be the URL to your application. For example, <https://mydomain.com/cmwebdrawer> followed by the path to the authentication provider, for example, </auth/openid>.

The </auth/> component is fixed but the 'openid' is the name you will supply in **hptrim.config** later and so can be any string, as long as it matches the value in **hptrim.config**.

For the Web Client the path must include the path to the Service API, for example, <https://mydomain.com/contentManager/serviceapi/auth/openid>.

5. On saving, the Client ID and Client Secret will be displayed, note them for later use.

## Configure authentication in hptrim.config

To use the Google credentials created above, edit the **hptrim.config** ( or **hprmServiceAPI.config** in the Web Client) so that it has an authentication as follows:

1. The name must match the last segment of the Redirect URI path.
2. Client ID and secret as noted in above section.
3. The issuerURI is: <https://accounts.google.com>

For example,

```
<authentication allowAnonymous="false" slidingSessionMinutes="2">
<openidConnect>
<addname="openid"
clientID="999999999999-abcdefghijklmnopqrstuvwxyz.apps.googleusercontent.com"
clientSecret="j1-BiX7685hjgf99999y"
```

```
issuerURI="https://accounts.google.com"  
</>  
</openIdConnect>  
</authentication>
```

## Enable redirect

The Web Client will not re-direct the authentication endpoint unless the **Html** feature is enabled in **hprmServiceAPI.config**. To do this:

1. Edit **hprmServiceAPI.config**
2. Find the property named **serviceFeatures**.
3. Add the feature **Html**.

## Logout

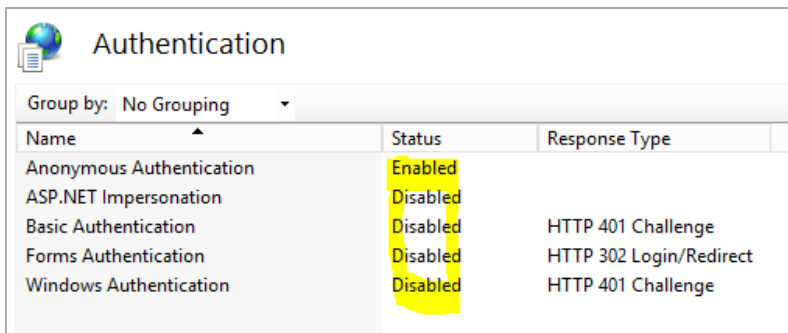
For WebDrawer the logout link is configured in the uiSettings. It should contain ~/auth/logout. In the Web Client a logout link will be displayed automatically when OpenID Connect authentication is enabled.

For example,

```
<uiSettings  
logoutLink="~/auth/logout"  
...  
</>
```

## Allow anonymous access in the IIS

The IIS will not handle authentication, so use IIS Manager to allow anonymous access only.

The screenshot shows the 'Authentication' section of the IIS Manager console. It features a table with columns for 'Name', 'Status', and 'Response Type'. The 'Status' column for 'Anonymous Authentication' is highlighted in yellow and shows 'Enabled'. Other authentication methods like 'ASP.NET Impersonation', 'Basic Authentication', 'Forms Authentication', and 'Windows Authentication' are listed with 'Disabled' status.

| Name                     | Status   | Response Type           |
|--------------------------|----------|-------------------------|
| Anonymous Authentication | Enabled  |                         |
| ASP.NET Impersonation    | Disabled |                         |
| Basic Authentication     | Disabled | HTTP 401 Challenge      |
| Forms Authentication     | Disabled | HTTP 302 Login/Redirect |
| Windows Authentication   | Disabled | HTTP 401 Challenge      |