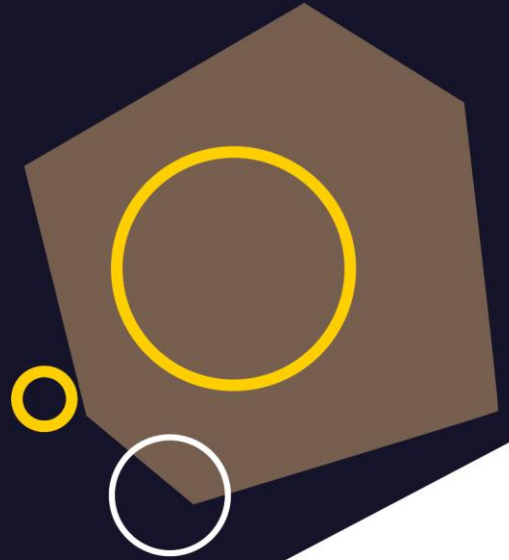


**censornet.**



Date: 28/04/2022

Version: 1.0

## **Censornet Archive – OAuth with ADFS Configuration Guide**

## Contents

<i>Introduction.....</i>	<i>3</i>
<i>ADFS side Settings - Step by Step.....</i>	<i>4</i>
<i>Step 1: Register an ADFS Client for OAuth .....</i>	<i>4</i>
<i>Step 2: Create a Relying Party Trust .....</i>	<i>5</i>
<i>Step 3: Create a Rule to Send LDAP Attributes as Claims for a Relying Party Trust .....</i>	<i>9</i>
<i>Step 4: Set the option "EnableJWT" to true .....</i>	<i>12</i>
<i>Step 5: Grant permission to registered client app to access relying party.....</i>	<i>12</i>
<i>Censornet Side Settings .....</i>	<i>13</i>
<i>Create the OAuth connection for Microsoft ADFS provider type .....</i>	<i>13</i>
<i>Access Control.....</i>	<i>15</i>
<i>Local User Accounts for OAuth .....</i>	<i>15</i>
<i>Censornet Archiving OAuth Configuration .....</i>	<i>16</i>
<i>Modify registration settings for an OAuth client registered with ADFS.....</i>	<i>16</i>
<i>Check the registration settings for an OAuth client registered with ADFS .....</i>	<i>16</i>
<i>Remove OAuth client registered with ADFS.....</i>	<i>16</i>

## **Introduction**

---

This document shows how to configure OAuth for user's authentication against on premise Microsoft Active Directory via Active Directory Federated Service (ADFS).

## ADFS side Settings - Step by Step

A quick run through of the steps involved in supporting the Active Directory Federation Services (ADFS) for authentication using OAuth2 in Windows Server 2012 R2 and above.

### Step 1: Register an ADFS Client for OAuth

Register the OAuth client with ADFS by using calling PowerShell cmdlet. A template for the command is shown here – showing key elements that you will need to supply:

```
Add-AdfsClient -Name <ClientApp> -ClientId <ClientId> -RedirectUri  
https://archive.clouduss.com/uss/microsoftoauth.do -Description "ADFS OAuth"
```

Where:

**-ClientId:** Client identifier for the OAuth client to register with ADFS. It can be set as any \*random string that is likely to be unique for this client. Note that the same string needs to be specified under OAuth connection created at Censornet Archive end. Replace the **<ClientId>** with a reasonably long text string (e.g. "71d3cf488b4bf413547e83410e047885f3148ec5bcd90913") that cannot be easily guessed.

**\*Note – You can use an online UUID tool for the creation of the ClientId**

**-RedirectUri:** Specifies your Censornet Archive redirection URIs to register with ADFS against this OAuth client. Except URIs registered here, no other URI can be used as target endpoint for successful authentication by ADFS.

\*Replace the <https://archive.clouduss.com/uss/microsoftoauth.do> with your Censornet Archive region hostname.

\*Check Censornet support site for the URL's for Censornet's another Archive regions.

**-Description:** Any suitable description for the OAuth client

**-Name:** Any appropriate name for the OAuth client

Replace the **<ClientApp>** with a string value like "Client App", this is likely to be used as the target name for the app.

### Example 1: Add a client

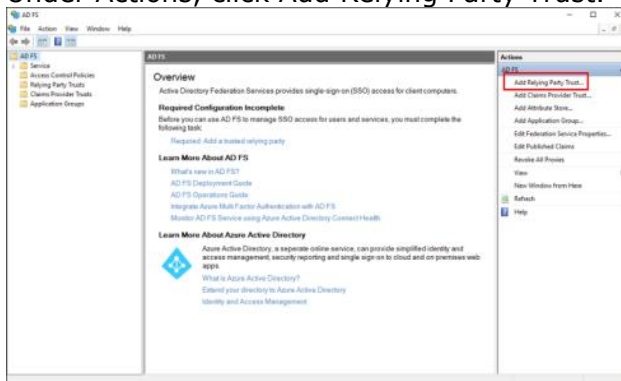
```
Add-AdfsClient -Name "CensornetADFS" -ClientId  
"71d3cf488b4bf413547e83410e047885f3148ec5bcd90913" -RedirectUri  
https://archive.clouduss.com/uss/microsoftoauth.do -Description "ADFS OAuth"
```

## Step 2: Create a Relying Party Trust

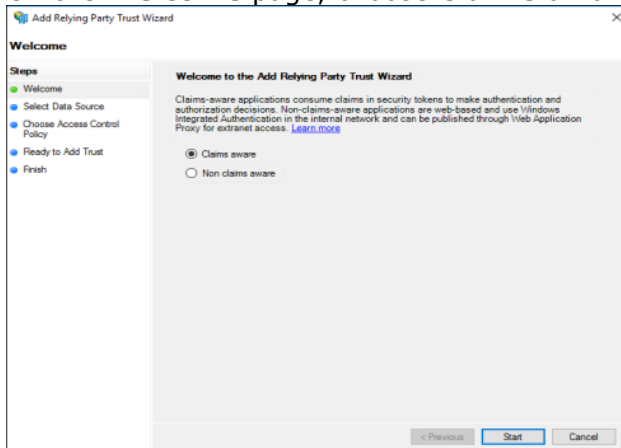
A new relying party trust needs to be added by using the ADFS Management snap-in. This requires following procedure.

Membership in Administrators or equivalent, on the local computer is the minimum required to complete this procedure.

1. In Server Manager, click Tools, and then select **AD FS Management**.
2. Under Actions, click Add Relying Party Trust.



3. On the **Welcome** page, choose **Claims aware** and click **Start**.



4. On the Select Data Source page, click Enter data about the relying party manually, and then click Next.

Add Relying Party Trust Wizard

**Select Data Source**

Steps

- Welcome
- Select Data Source
- Specify Display Name
- Configure Certificate
- Configure URL
- Configure Identifiers
- Choose Access Control Policy
- Ready to Add Trust
- Finish

Select an option that this wizard will use to obtain data about this relying party:

☐ Import data about the relying party published online or on a local network

Use this option to import the necessary data and certificates from a relying party organization that publishes its federation metadata online or on a local network.

Federation metadata address (host name or URL):

Example: fs.cortoso.com or https://www.cortoso.com/app

☐ Import data about the relying party from a file

Use this option to import the necessary data and certificates from a relying party organization that has exported its federation metadata to a file. Ensure that this file is from a trusted source. This wizard will not validate the source of the file.

Federation metadata file location:

Browse

☒ Enter data about the relying party manually

Use this option to manually input the necessary data about this relying party organization.

< Previous Next > Cancel

5. On the **Specify Display Name** page, type any appropriate name in **Display name**, under **Notes** type a description for this relying party trust, and then click **Next**. This name is to be used as the Target Name for this Relying Party.

Add Relying Party Trust Wizard

**Specify Display Name**

Steps

- Welcome
- Select Data Source
- Specify Display Name
- Configure Certificate
- Configure URL
- Configure Identifiers
- Choose Access Control Policy
- Ready to Add Trust
- Finish

Enter the display name and any optional notes for this relying party.

Display name:

Censornet Archive

Notes:

< Previous Next > Cancel

6. On the **Configure Certificate** page, just click on the **Next** button.

The screenshot shows the 'Configure Certificate' step of the 'Add Relying Party Trust Wizard'. On the left, a 'Steps' list includes: Welcome, Select Data Source, Specify Display Name, **Configure Certificate** (highlighted), Configure URL, Configure Identifiers, Choose Access Control Policy, Ready to Add Trust, and Finish. The main area contains instructions: 'Specify an optional token encryption certificate. The token encryption certificate is used to encrypt the claims that are sent to this relying party. The relying party will use the private key of this certificate to decrypt the claims that are sent to it. To specify the certificate, click Browse.' Below this is a form with fields for 'Issuer:', 'Subject:', 'Effective date:', and 'Expiration date:'. There are 'View...', 'Browse...', and 'Remove' buttons. At the bottom are '< Previous', 'Next >', and 'Cancel' buttons.

7. On the **Configure URL** page, select the **Enable support for the WS-Federation Passive protocol** check box. Under **Relying party WS-Federation Passive protocol URL**, type the URL for this relying party trust, and then click **Next**.

This URL can be like `https://<your-adfs.fqdn>/adfs/services/trust` this must be unique amongst the Relying Parties. Please note that, for the purposes of OAuth, this URL does not need to be a real web address – it is just used as a unique identifier name. It will be used later in Censornet’s Archive OAuth configuration and in the ***Grant-AdfsApplicationPermission PowerShell command***.

The screenshot shows the 'Configure URL' step of the 'Add Relying Party Trust Wizard'. The 'Steps' list on the left is the same as in the previous screenshot, with 'Configure URL' highlighted. The main area contains instructions: 'AD FS supports the WS-Trust, WS-Federation and SAML 2.0 WebSSO protocols for relying parties. If WS-Federation, SAML, or both are used by the relying party, select the check boxes for them and specify the URLs to use. Support for the WS-Trust protocol is always enabled for a relying party.' There are two sections: 1) 'Enable support for the WS-Federation Passive protocol' (checked), with a text box for 'Relying party WS-Federation Passive protocol URL:' containing 'https://your-adfs.fqdn/adfs/services/trust' and an example 'https://fs.contoso.com/adfs/fs/'. 2) 'Enable support for the SAML 2.0 WebSSO protocol' (unchecked), with a text box for 'Relying party SAML 2.0 SSO service URL:' and an example 'https://www.contoso.com/adfs/fs/'. At the bottom are '< Previous', 'Next >', and 'Cancel' buttons.

8. On the **Configure Identifiers** page, click **Next**.

The screenshot shows the 'Configure Identifiers' step of the 'Add Relying Party Trust Wizard'. The left sidebar lists the steps: Welcome, Select Data Source, Specify Display Name, Configure Certificate, Configure URL, Configure Identifiers (selected), Choose Access Control Policy, Ready to Add Trust, and Finish. The main area has a title bar 'Add Relying Party Trust Wizard' and a close button. Below the title bar is the section 'Configure Identifiers'. A message states: 'Relying parties may be identified by one or more unique identifier strings. Specify the identifiers for this relying party trust.' There are two input fields: 'Relying party trust identifier:' with an 'Add' button, and 'Relying party trust identifiers:' with a 'Remove' button. An example URL is provided: 'https://fs.contoso.com/adfs/services/trust'. At the bottom are buttons for '< Previous', 'Next >', and 'Cancel'.

9. On the **Choose Access Control Policy** select a policy and click **Next**.

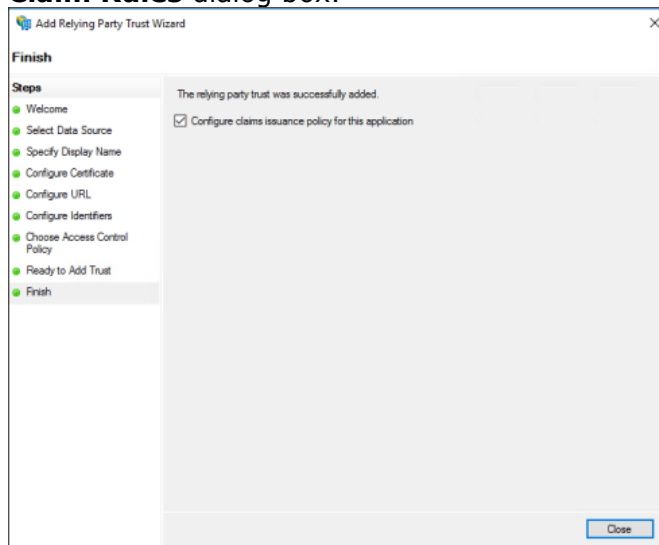
The screenshot shows the 'Choose Access Control Policy' step of the 'Add Relying Party Trust Wizard'. The left sidebar is the same as in the previous step, with 'Choose Access Control Policy' selected. The main area has a title bar 'Add Relying Party Trust Wizard' and a close button. Below the title bar is the section 'Choose Access Control Policy'. A message states: 'Choose an access control policy:'. There is a table with two columns: 'Name' and 'Description'. The first row is 'Permit everyone' with the description 'Grant access to everyone...'. Below the table is a 'Policy' section with a text area containing 'Permit everyone'. At the bottom is a checkbox labeled 'I do not want to configure access control policies at this time. No user will be permitted access for this application.' At the bottom are buttons for '< Previous', 'Next >', and 'Cancel'.

10. On the **Ready to Add Trust** page, review the settings, and then click **Next** to save your relying party trust information.

The screenshot shows the 'Ready to Add Trust' step of the 'Add Relying Party Trust Wizard'. The left sidebar is the same as in the previous steps, with 'Ready to Add Trust' selected. The main area has a title bar 'Add Relying Party Trust Wizard' and a close button. Below the title bar is the section 'Ready to Add Trust'. A message states: 'The relying party trust has been configured. Review the following settings, and then click Next to add the relying party trust to the AD FS configuration database.' There are tabs for 'Monitoring', 'Identifiers', 'Encryption', 'Signature', 'Accepted Claims', 'Organization', 'Endpoints', and 'Not'. The 'Monitoring' tab is selected. Below the tabs is a section 'Specify the monitoring settings for this relying party trust.' with a text area for 'Relying party's federation metadata URL:'. There are two checkboxes: 'Monitor relying party' and 'Automatically update relying party'. Below these are two labels: 'This relying party's federation metadata data was last checked on:' and 'This relying party was last updated from federation metadata on:', both with a dropdown menu showing '< never >'. At the bottom are buttons for '< Previous', 'Next >', and 'Cancel'.



11. On the **Finish** page, click **Close**. This action automatically displays the **Edit Claim Rules** dialog box.



### Step 3: Create a Rule to Send LDAP Attributes as Claims for a Relying Party Trust

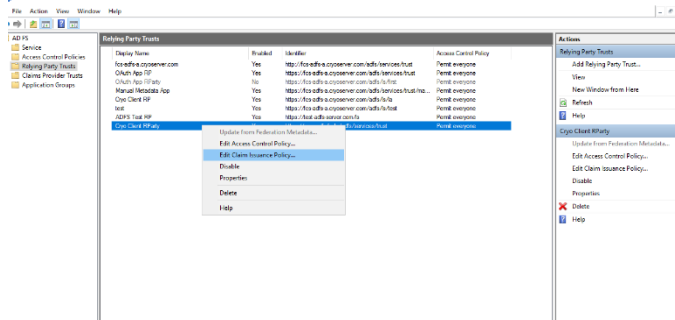
Using the “Send LDAP Attributes as Claims rule” template in AD FS, we can create a rule that will select required attributes from a LDAP attribute store, to send as claims to the relying party.

Membership in **Administrators**, or equivalent, on the local computer is the minimum required to complete this procedure.

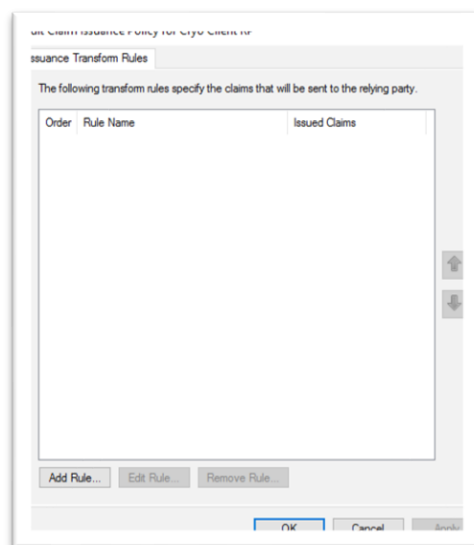
1. In Server Manager, click **Tools**, and then select **AD FS Management**.
2. In the console tree, under **AD FS**, click **Relying Party Trusts**.



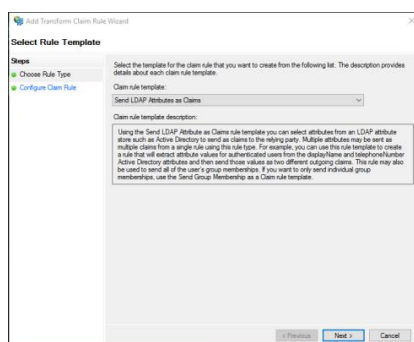
- Right-click the selected trust, and then click **Edit Claim Issuance Policy**.



- In the **Edit Claim Issuance Policy** dialog box, under **Issuance Transform Rules** click **Add Rule** to start the rule wizard.



- On the Select Rule Template page, under Claim rule template, select **Send LDAP Attributes as Claims** from the list, and then click **Next**.



- On the **Configure Rule** page under **Claim rule name** type the display name for this rule, select the **Attribute Store** (Active Directory) and then select each LDAP attribute required by Censornet Archive, and map it to the outgoing claim type. Add the Claims with the same Outgoing Claim Type name corresponding to the LDAP Attributes as shown:

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)
User-Principal-Name	userPrincipalName
E-Mail-Addresses	mail
Proxy-Addresses	proxyAddresses
Surname	Surname
Given-Name	givenName

View Rule Language...

LDAP Attribute	Outgoing Claim Type
<b>User-Principal-Name</b>	<i>userPrincipalName</i>
<b>E-Mail-Addresses</b>	<i>mail</i>
<b>Proxy-Addresses</b>	<i>proxyAddresses</i>
<b>Surname</b>	<i>Surname</i>
<b>Given-Name</b>	<i>GivenName</i>

- Click the **Finish** button.
- In the **Edit Claim Rules** dialog box, click **OK** to save the rule.

#### Step 4: Set the option "EnableJWT" to true

Set the option "EnableJWT" to true on the Relying Party Trust you configured.

Execute the PowerShell Set-AdfsRelyingPartyTrust command. The template of this command is like this:

```
Set-AdfsRelyingPartyTrust -TargetName "<Relying Party Display Name>" -EnableJWT $true
```

Replace the **<Relying Party Trust Name>** with the display name of the Relying Party, that you specified in the **Step 2, point 5**.

*Example:*

```
Set-AdfsRelyingPartyTrust -TargetName "Censornet Archive" -EnableJWT $true
```

#### Step 5: Grant permission to registered client app to access relying party

Run this command to explicitly grant clients permission to a resource

```
Grant-AdfsApplicationPermission -ClientRoleIdentifier "<ClientId>" -  
ServerRoleIdentifier "<Relying Party Identifier>"
```

Where,

**-ClientRoleIdentifier Client ID**, that is set while adding the client.

Refer to the **Step 1**.

**-ServerRoleIdentifier RP Identifier**. It should be the trust URL associated with Relying Party Trust as shown below

Relying Party Trusts			
Display Name	Enabled	Identifier	Access Control Policy
fcs-adfs-a.cryoserver.com	Yes	http://fcs-adfs-a.cryoserver.com/adfs/services/trust	Permit everyone
OAuth App RP	Yes	https://fcs-adfs-a.cryoserver.com/adfs/services/trust	Permit everyone
OAuth App RParty	No	https://fcs-adfs-a.cryoserver.com/adfs/ls/first	Permit everyone
Manual Metadata App	Yes	https://fcs-adfs-a.cryoserver.com/adfs/services/trust/ma...	Permit everyone
Cryo Client RP	Yes	https://fcs-adfs-a.cryoserver.com/adfs/ls/la	Permit everyone
test	Yes	https://fcs-adfs-a.cryoserver.com/adfs/ls/test	Permit everyone
ADFS Test RP	Yes	https://test-adfs-server.com/ls	Permit everyone
Cryo Client RParty	Yes	https://your-adfs.fqdn/adfs/services/trust	Permit everyone

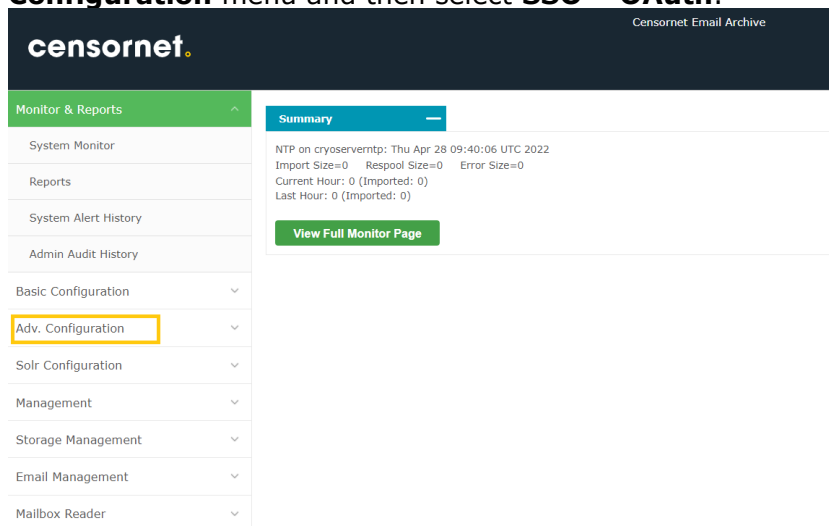
*Example:*

```
Grant-AdfsApplicationPermission -ClientRoleIdentifier  
"71d3cf488b4bf413547e83410e047885f3148ec5bcd90913" -ServerRoleIdentifier  
https://<your-adfs.fqdn>/adfs/services/trust
```

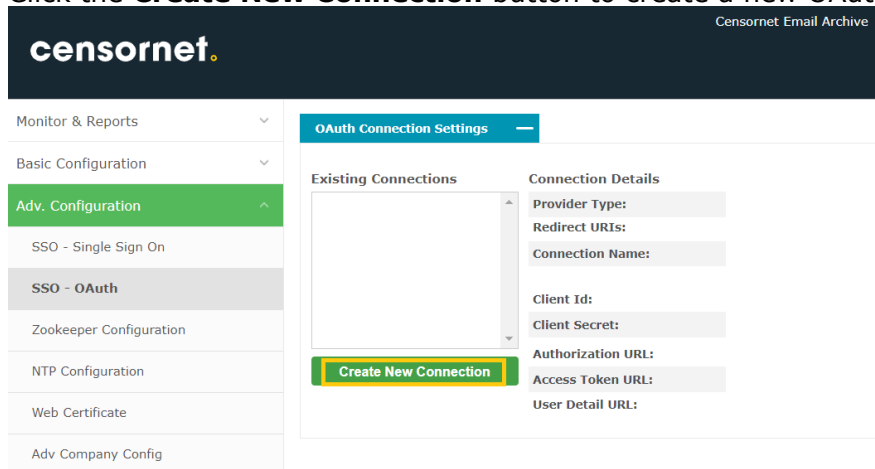
## Censornet Side Settings

### Create the OAuth connection for Microsoft ADFS provider type

1. Login as admin user to your Censornet Archive enviroment, click on **Adv. Configuration** menu and then select **SSO – OAuth**.



2. Click the **Create New Connection** button to create a new OAuth Connection.



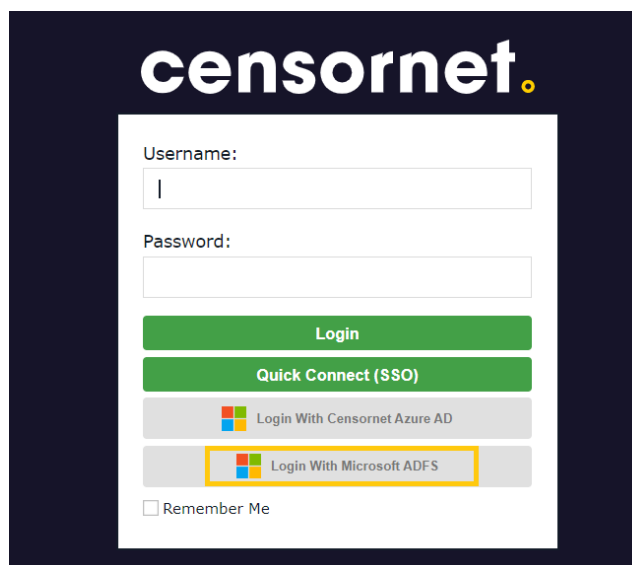
- Under the Create New Connection, select **Microsoft ADFS** from the provider type drop down and enter the following details

The screenshot shows the 'OAuth Connection Settings' form. On the left, there is a section for 'Existing Connections' with a list box and 'Save Connection' and 'Cancel' buttons. The main section is 'Connection Details' with the following fields:

Provider Type:	Microsoft ADFS
Redirect URIs:	https://archive.clouduss.com/uss/microsoftoauth.do https://archive.clouduss.com/ussv9/microsoftoauth.do
Connection Name:	Microsoft ADFS
Client Id:	Your Client ID
NOTE:	Authorization URL, Access Token URL and User Detail URL are mandatory.
Authorization URL:	https://<adfs-server-hostname>/adfs/oauth2/authorize
Access Token URL:	https://<adfs-server-hostname>/adfs/oauth2/token
User Detail URL:	https://<your-adfs.fqdn>/adfs/services/trust

where,

- **Connection Name:** A relevant connection name that will also get displayed with label of OAuth login button on the login page.
  - **Client Id:** Identity of the registered OAuth Client App. This should be the same "clientId" string as specified under step 1 of ADFS side settings.
  - **Authorization URL:** URL where frontend will redirect the user for authorization, It is of the form "https://<adfs-server-hostname>/adfs/oauth2/authorize"
  - **Access Token URL:** URL for getting access token against authorization code. It is of the form "https://<adfs-server-hostname>/adfs/oauth2/token"
  - **User Detail URL:** Specify the Relying party trust identifier. It should be the same URL as first provided in **Step 2: Create a Relying Party Trust, part 7 (Relying party WS-Federation Passive protocol URL)** and later as the ServerRoleIdentifier value specified under Step 5 of ADFS side settings.
- Now save the connection and you are done!
  - The login page should now display a new button with the label "Login with <Connection Name>", to allow user login via **Microsoft ADFS** as shown



### *Access Control*

For security, URL links pointing to Censornet's Archive will only be allowed for registered domains. The domains that are unexpected, the system will respond with an "Access barred" error message:

**censornet.**

Access from host

is barred. Please ask a Censornet administrator to add this referrer to the list of valid names, if the link appears to be genuine.

To fix this, you must contact **Censornet's Support Team** and ask them to add your OAuth service as a valid "referrer" to Censornet's Archive solutions.

### *Local User Accounts for OAuth*

When a user accesses Censornet's Archive for the first time using OAuth, it will create a Local User Account entry within Censornet's Archive solution.

To review the accounts created by OAuth logins, visit the **Basic Configuration > Local User Accounts**. The users accounts will show the user's email address as their username. They will be set to "external authorization" (meaning that their password is not held in Censornet's Archive solution so must be validated with some external system).

## Censornet Archiving OAuth Configuration

### *Modify registration settings for an OAuth client registered with ADFS*

The **Set-AdfsClient** cmdlet modifies registration settings for an OAuth 2.0 client registered with Active Directory Federation Services (AD FS). Use this cmdlet to modify the settings, including the client identifier, redirection URI, name, or description of the OAuth 2.0 client. You can also use this cmdlet to register additional redirection URIs for the OAuth 2.0 client.

```
Set-AdfsClient -TargetName "<ClientApp>" -RedirectUri  
@("https://<archive.clouduss.com>/uss/microsoftoauth.do",  
"https://<archive.uk.clouduss.com>/uss/microsoftoauth.do")
```

*Example:*

```
Set-AdfsClient -TargetName "Client App" -RedirectUri @(""  
https://archive.clouduss.com/uss/microsoftoauth.do",  
"https://archive.uk.clouduss.com/uss/microsoftoauth.do")
```

### *Check the registration settings for an OAuth client registered with ADFS*

Use the following cmdlet template

```
Get-AdfsClient -name "<ClientApp>"
```

*Example:*

```
Get-AdfsClient -name "CensornetADFS"
```

### *Remove OAuth client registered with ADFS*

The **Revoke-AdfsApplicationPermission** cmdlet revokes permission for an application in Active Directory Federation Services (AD FS).

Use the following cmdlet template

```
Revoke-AdfsApplicationPermission -TargetClientRoleIdentifier "clientID " -TargetServerRoleIdentifier  
"https://<your-adfs.fqdn>/adfs/services/trust"
```

*Example:*

```
Revoke-AdfsApplicationPermission -TargetClientRoleIdentifier "2960ba77-37fc-4c91-a8dd-f6e5093b1ea2" -  
TargetServerRoleIdentifier "https://ADFS.Censornet.com/adfs/services/trust/censornetarchive"
```

*References:*

<https://docs.microsoft.com/en-us/windows-server/identity/ad-fs/operations/create-a-relying-party-trust#to-create-a-claims-aware-relying-party-trust-using-federation-metadata>