

# Azure Active Directory Integration with Amazon Cognito User Pools

Amazon Cognito lets us add user sign-up, sign-in, and access control to the web access within Data.all. Amazon Cognito supports sign-in with social identity providers via SAML 2.0.

We would like to integrate Azure AD with Cognito via authentication and redirection URLs/Tokens that will be entered into Azure AD. These are generated securely within Cognito before deployment to Azure AD.

To set up an AWS Cognito User Pool with an Azure AD identity provider and perform single sign-on (SSO) authentication with Azure AD account to access AWS services in the data.all serverless application, we need to follow these steps:

## Get the ACS endpoint and the Entity ID from AWS Cognito

With the built-in hosted web UI, Amazon Cognito provides token handling and management for all authenticated users, so that Data.all backend systems can standardise on one set of user pool tokens.

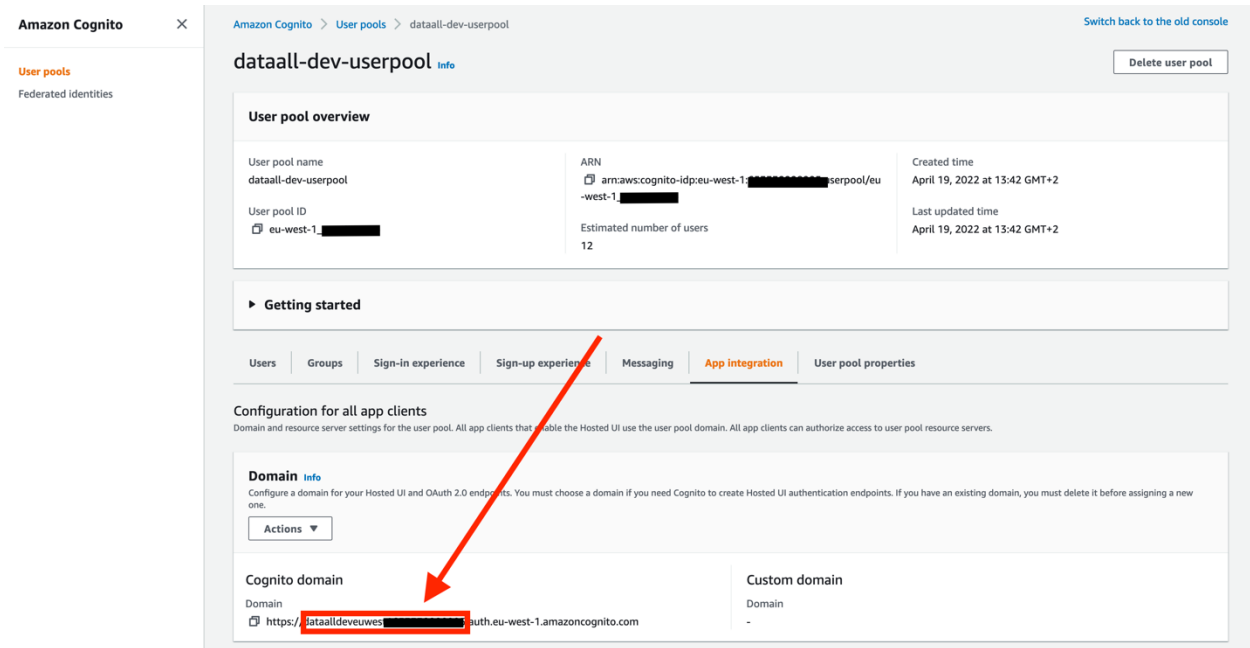


As part of its deployment, data.all already creates a Cognito User Pool, an App Client, and an Amazon Cognito domain. To build your Azure AD application that will integrate with the Cognito User Pool, you will need to provide two things:

- Assertion Consumer Service (ACS) endpoint - This corresponds to the location which the SSO tokens are sent. Configure this endpoint for SAML 2.0 POST binding in your SAML identity provider:

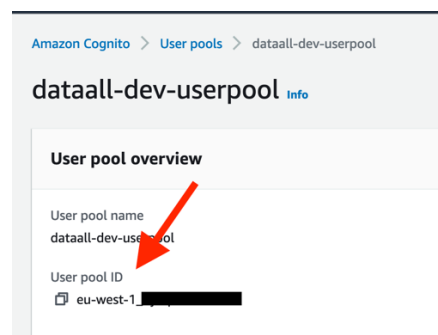
<https://<yourDomainPrefix>.auth.<region>.amazoncognito.com/saml2/idpresponse>

Use the region where data.all is deployed in this link. You can find your domain prefix for your user pool on the **App Integration** tab of the Amazon Cognito console.



- **Entity ID** - The globally-unique identifier for your Cognito User Pool. This ID has the following form: `urn:amazon:cognito:sp:<yourUserPoolID>`

You can find your **user pool ID** on Overview box in the Amazon Cognito console.

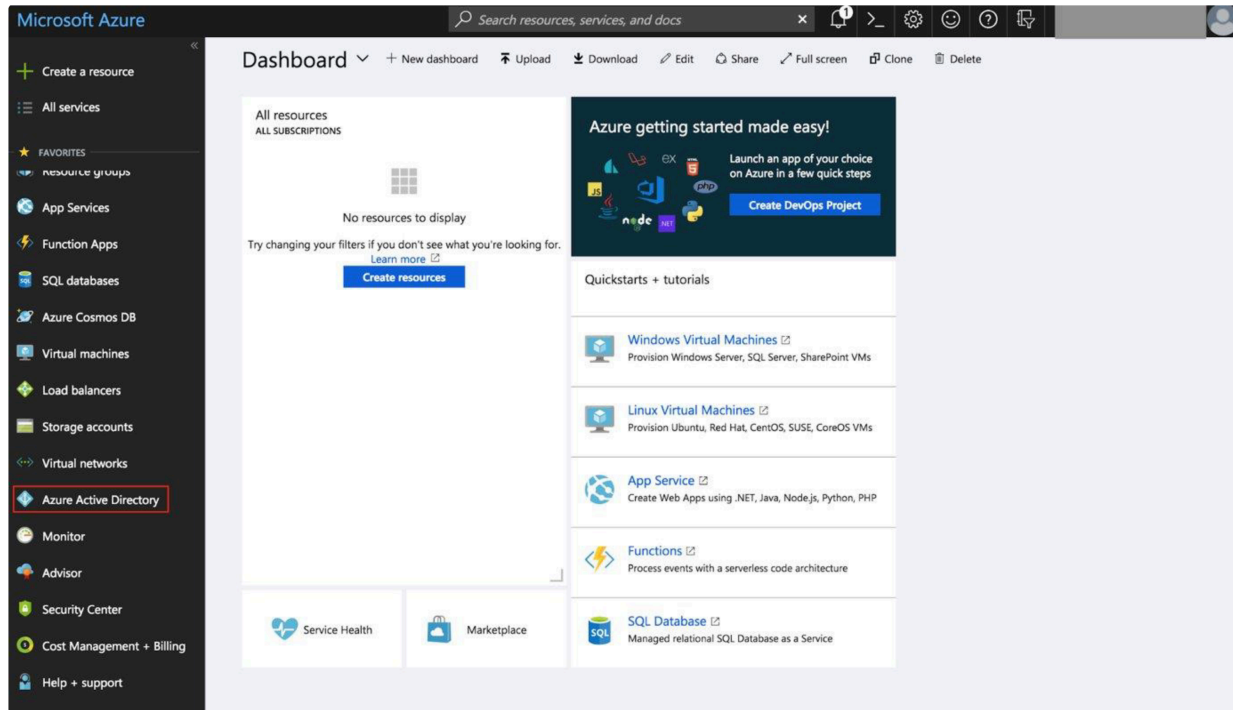


Store your ACS endpoint and your Entity ID somewhere as they will be required when creating the SAML application in Azure AD.

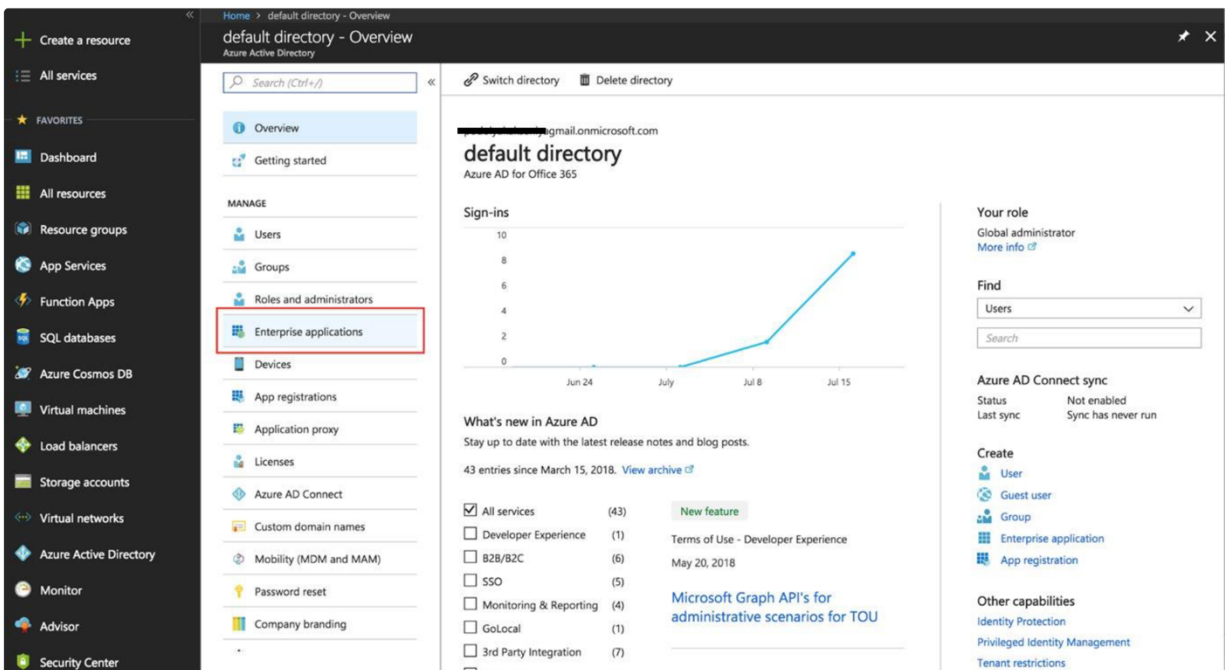
## Create an Azure AD enterprise application and set up Azure AD identity provider to the Cognito User Pool

If you don't have access to Azure AD, you can skip this section and raise a ticket to team responsible for your Identity Provider. They will ask you the **Entity ID** and the **ACS endpoint** you recovered from the previous steps. When they send you the XML file, go to the [next section of this guide](#).

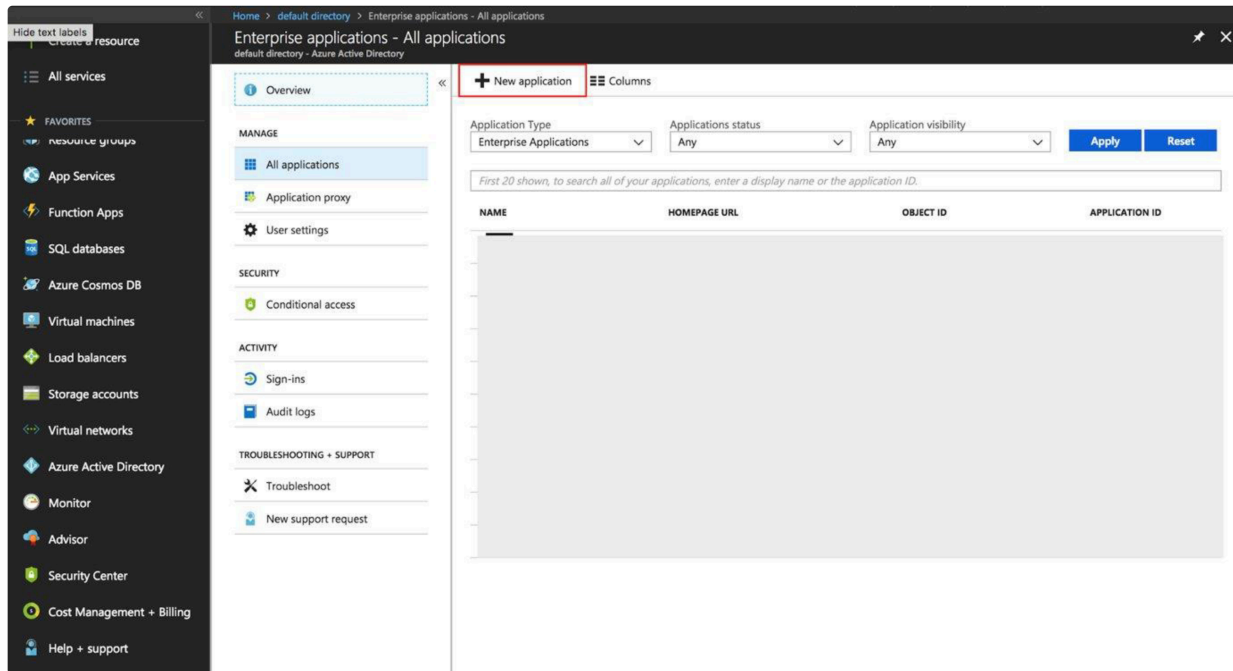
Open Azure Portal, on the right side menu choose **Azure Active Directory**.



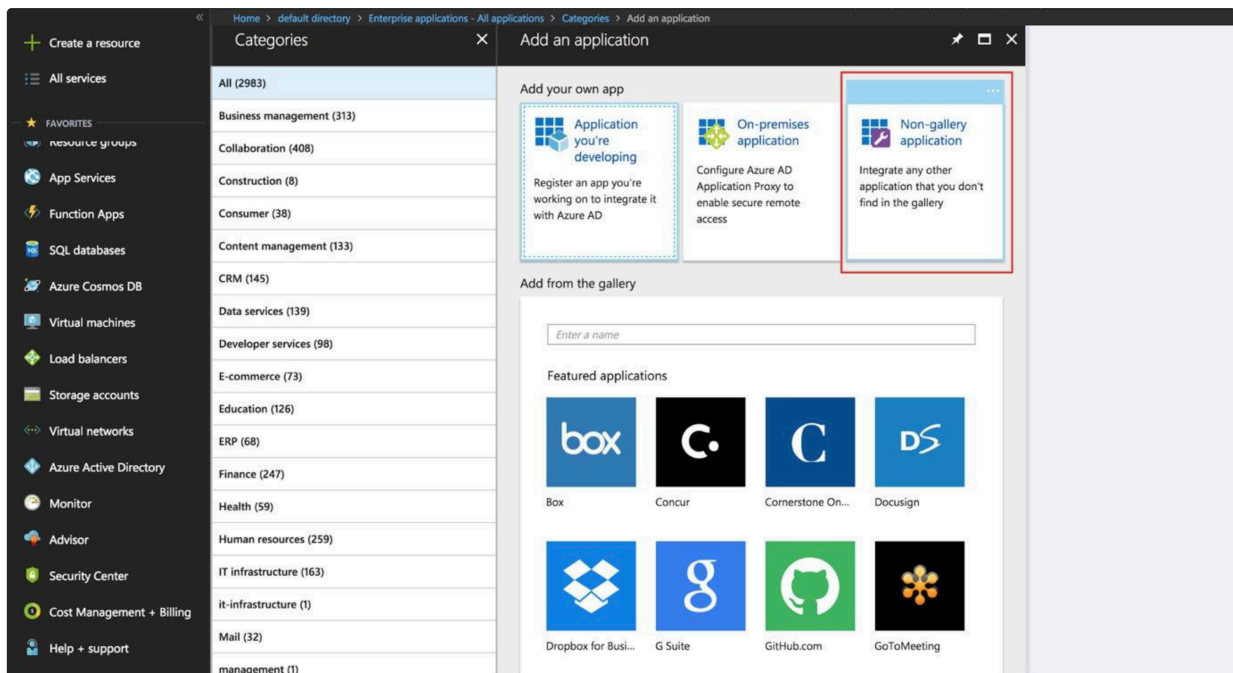
In Active Directory menu choose **Enterprise applications**:



In opened section choose **New Application**:

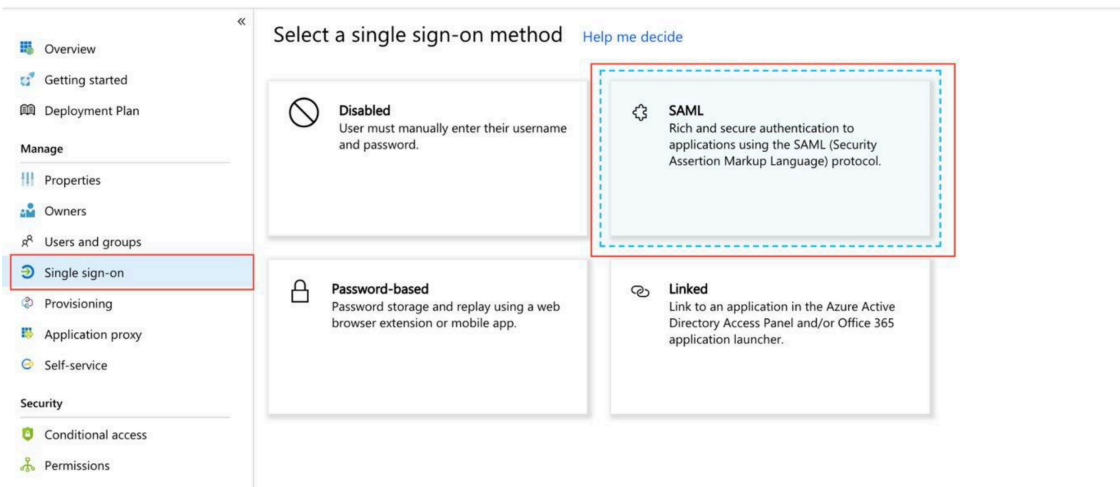


Pick **Non-gallery application** type for your application:



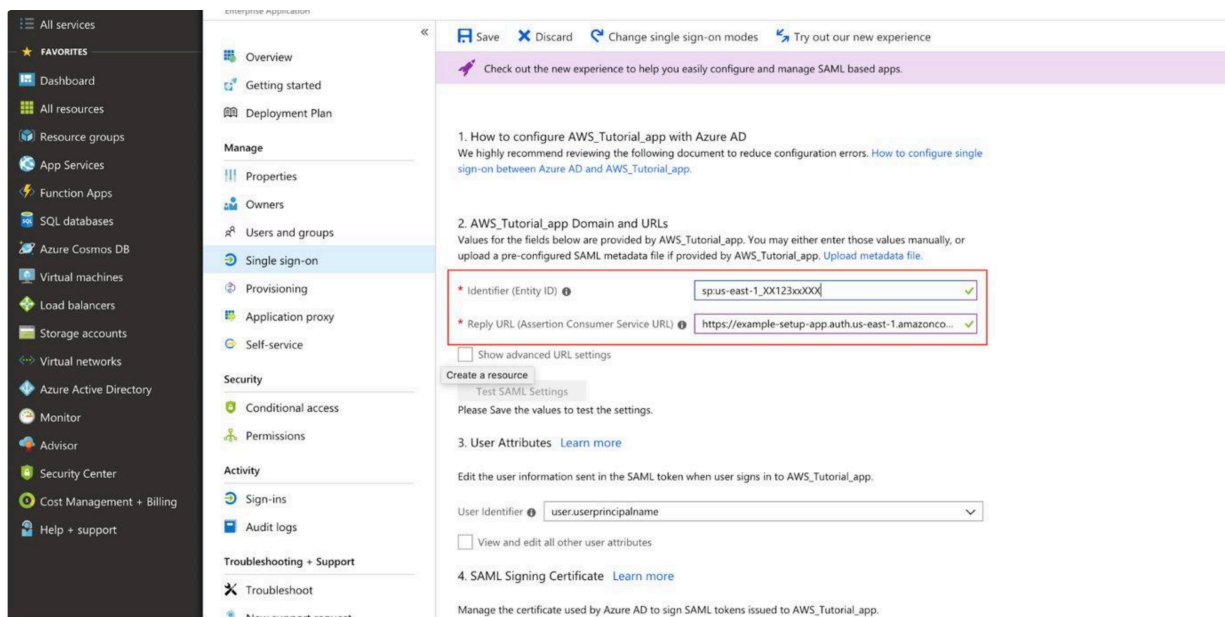
Type the name of your application and press **Add**. Now your application is created and it is time to connect it to AWS User Pool.

In your Azure AD enterprise application choose section **Single sign-on**, in dropdown list choose **SAML-based Sign-on**:



In section **Domain and URLs**, set the following information (refer to previous section of this document) :

- **Identifier** : This is your Cognito Entity ID - `urn:amazon:cognito:sp:<yourUserPoolID>`
- **Reply URL** : This is the link from where your Azure AD application expects to receive the authentication link token. This is your Cognito ACS - `https://<yourDomainPrefix>.auth.<region>.amazoncognito.com/saml2/idpresponse`



## Save your changes and **download SAML File**:

Microsoft Azure

Home > default directory > Enterprise applications - All applications > Categories > Add an application > ExampleCognitoAzureApp - Single sign-on

ExampleCognitoAzureApp - Single sign-on

Enterprise Application

Getting started

MANAGE

- Properties
- DeploymentPlan
- Owners
- Users and groups
- Single sign-on
- Provisioning
- Application proxy
- Self-service

SECURITY

- Conditional access
- Permissions

ACTIVITY

- Sign-ins
- Audit logs

Save Discard

Add attribute

### 4. SAML Signing Certificate [Learn more](#)

Manage the certificate used by Azure AD to sign SAML tokens issued to ExampleCognitoAzureApp.

App Federation Metadata Url

STATUS	EXPIRATION	THUMBPRINT	DOWNLOAD
Active	7/17/2021	[Redacted]	<a href="#">Certificate (Base64)</a> <a href="#">Certificate (Raw)</a> <a href="#">Metadata XML</a>

Create new certificate

Show advanced certificate signing settings [Learn more](#)

\* Notification Email

### 5. ExampleCognitoAzureApp Configuration

ExampleCognitoAzureApp must be configured to use Azure AD as a SAML identity provider. Click below to view instructions on how to do this.

[Configure ExampleCognitoAzureApp](#)

**Your SAML file**

For later tests, you can start adding Users to your application. In Azure AD select **Enterprise applications** and choose your application. Select **Users and groups**, then **Add user**.

Microsoft Azure

Home > default directory > Enterprise applications - All applications > Categories > Add an application > ExampleCognitoAzureApp - Users and groups

ExampleCognitoAzureApp - Users and groups

Enterprise Application

Getting started

MANAGE

- Properties
- DeploymentPlan
- Owners
- Users and groups
- Single sign-on
- Provisioning
- Application proxy
- Self-service

SECURITY

- Conditional access
- Permissions

ACTIVITY

- Sign-ins
- Audit logs

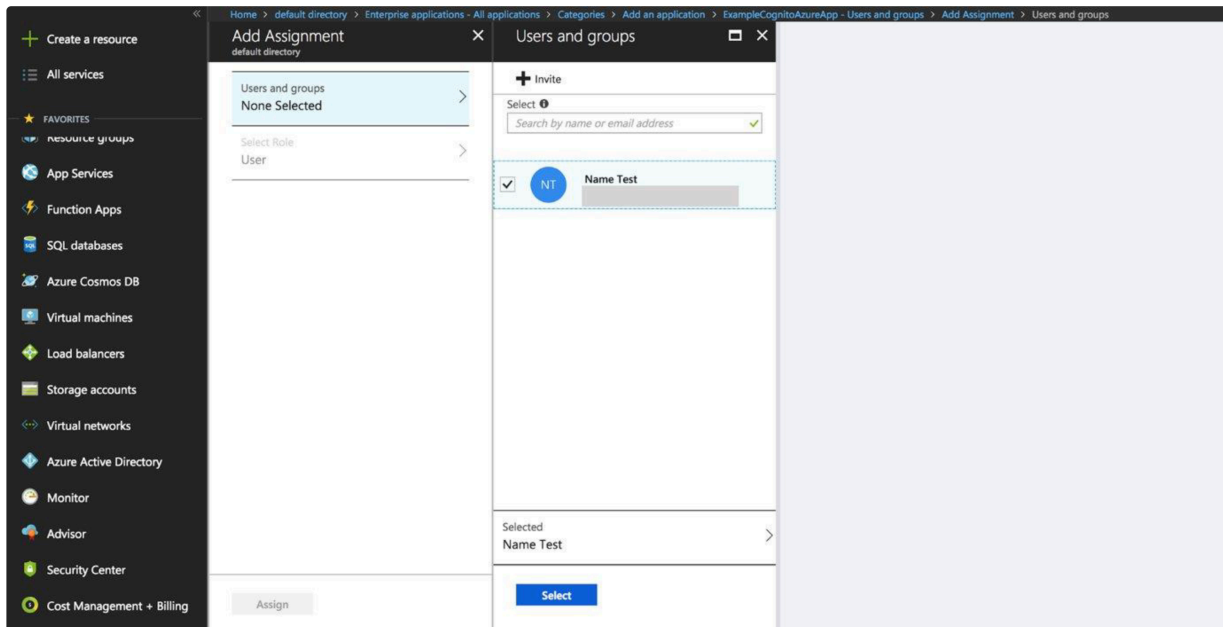
+ Add user Edit Remove Update Credentials

The application will appear on the access panel for assigned users. Set 'visible to users?' to no in properties to prevent this. →

First 100 shown, to search all users & groups, enter a display name.

DISPLAY NAME	OBJECT TYPE	ROLE ASSIGNED
No application assignments found		

Invite new users or select from existing. These users will be able to login with this Azure AD account to your application. When you'll finish adding a user select **Assign**.

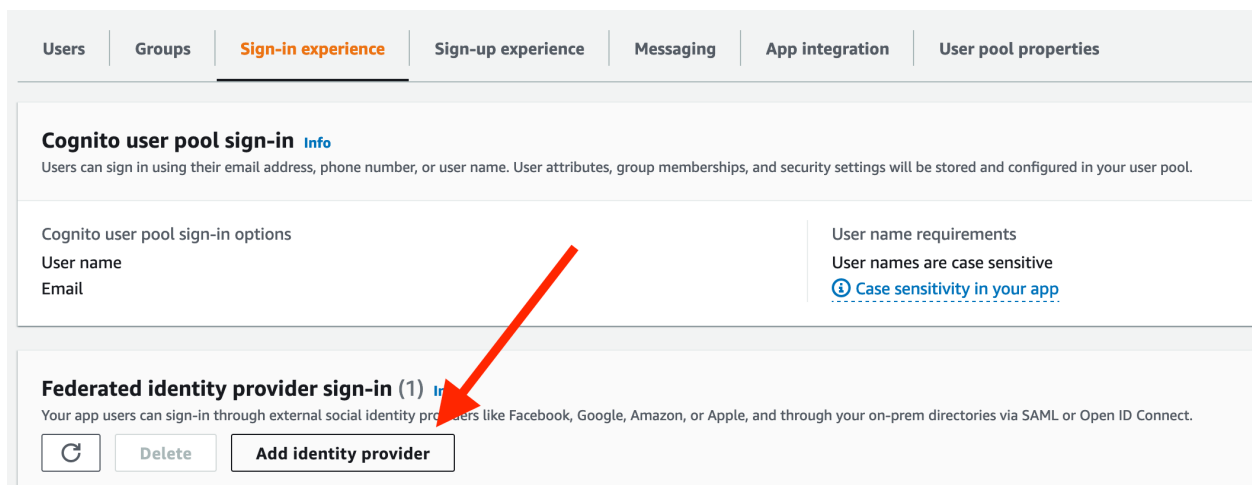


This is all settings in the Azure portal. At the end of this section you should have:

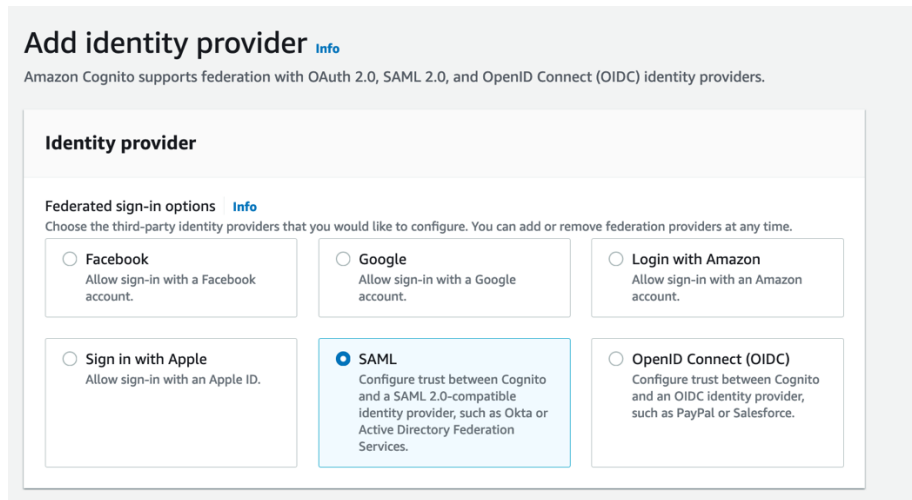
- SAML file with XML format
- user(s) to login

## Configure federation in Amazon Cognito

We will now integrate Azure AD with Cognito. To enable federation from Amazon Cognito side, go to the **Sign-in Experience** tab in the Amazon Cognito console and select **Add Identity Providers**.



## Select SAML



Under Metadata document, select the **metadata xml file that you got** at the end of the previous section. Next add **Provider Name - "ADTest"** for example

The screenshot shows the 'Set up SAML federation with this user pool' configuration screen. It includes several sections: 'Provider name' with an 'Info' link and a text input field containing 'ADTest'; 'Identifiers - optional' with an 'Info' link and a text area for entering identifiers; 'Sign-out flow' with an 'Info' link and an unchecked checkbox for 'Add sign-out flow'; 'Metadata document source' with an 'Info' link and two radio button options: 'Upload metadata document' (which is selected) and 'Enter metadata document endpoint URL'; 'Metadata document' with an 'Info' link and a 'Choose file' button; and 'Map attributes between your SAML provider and your user pool' with an 'Info' link, a note about required attributes, and an 'Add another attribute' button.

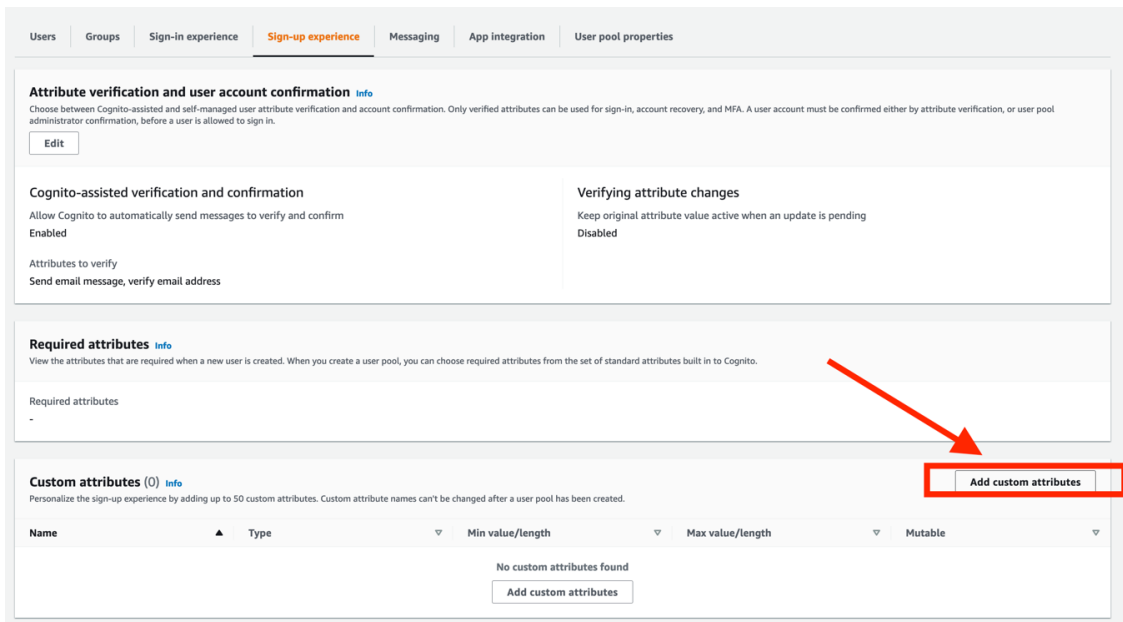


# Attribute Mapping - mapping identity provider attributes to user pool attributes

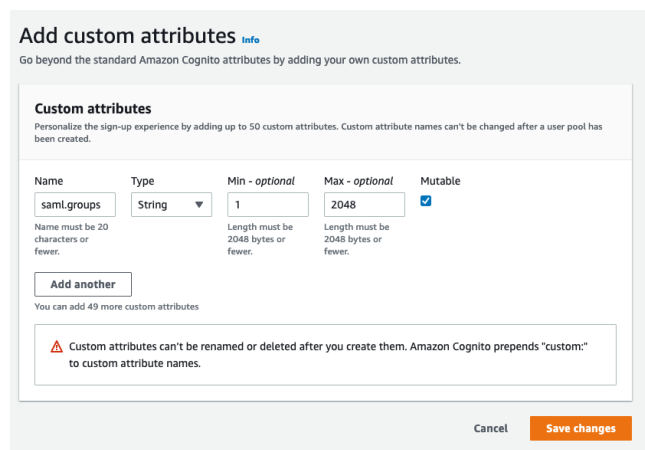
In order to collect the right user information from federated users, you need to map user attributes from external identity providers to the corresponding attributes for Cognito User Pools.

## Step 1: Add "saml.groups" custom attribute in you Amazon Cognito User Pool

We first need to create a new custom attribute to capture **Groups** from Azure AD. Under **Sign-up experience**, select **Add custom attributes**



Name the new custom attributes "saml.groups" with minimum length 1 and maximum length 2048 (should set to highest as this is immutable later on), check mutable and save.



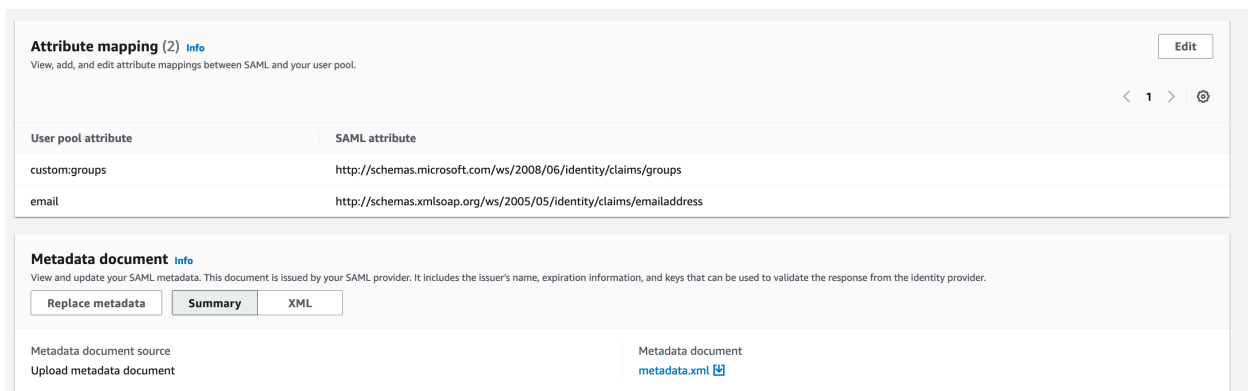
## Step 2: Map identity provider attributes to user pool attributes

Under the **Sign-In** tab in the Amazon Cognito console, select the ADTests identity provider and go to the **Attribute Mapping** section.

You will now define the mapping between attributes of the user pool and those from the SAML response received from the customer side.

Select your Identity Provider Name, and map the following attributes:<http://schemas.microsoft.com/ws/2008/06/identity/claims/groups>

SAML Attribute	User pool attribute
<a href="http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress">http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress</a>	email
<a href="http://schemas.microsoft.com/ws/2008/06/identity/claims/groups">http://schemas.microsoft.com/ws/2008/06/identity/claims/groups</a>	custom:groups



**Attribute mapping (2)** [Info](#) Edit

View, add, and edit attribute mappings between SAML and your user pool.

User pool attribute	SAML attribute
custom:groups	<a href="http://schemas.microsoft.com/ws/2008/06/identity/claims/groups">http://schemas.microsoft.com/ws/2008/06/identity/claims/groups</a>
email	<a href="http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress">http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress</a>

**Metadata document** [Info](#)

View and update your SAML metadata. This document is issued by your SAML provider. It includes the issuer's name, expiration information, and keys that can be used to validate the response from the identity provider.

[Replace metadata](#) [Summary](#) [XML](#)

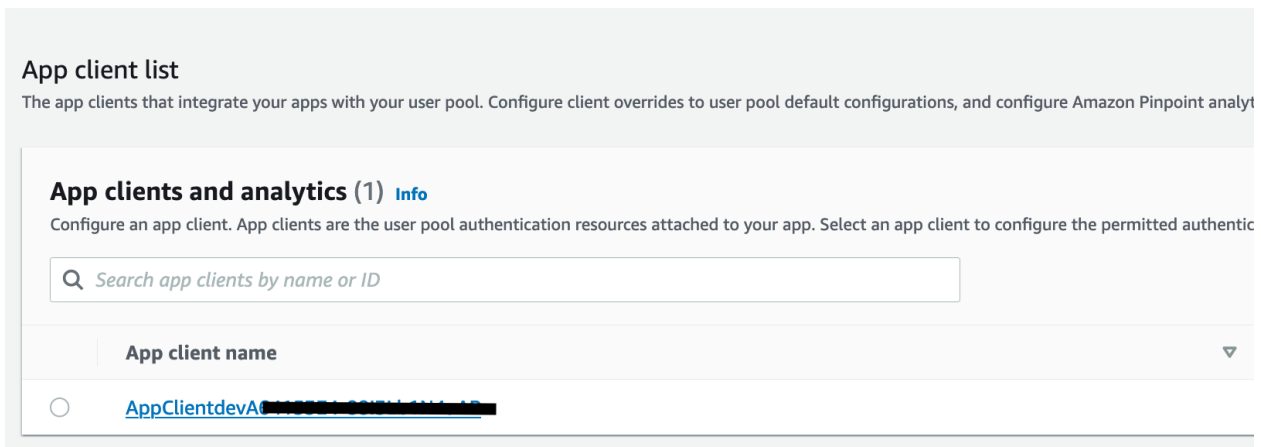
Metadata document source [Upload metadata document](#) Metadata document [metadata.xml](#)

You can map other attributes if required. You will find each available **SAML Attribute** in the XML file.

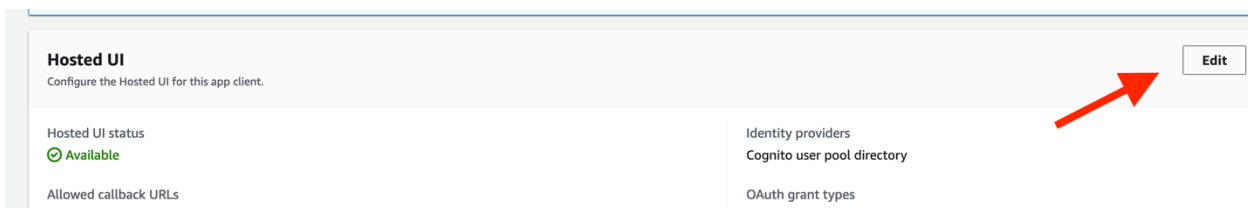
## **Register the identity Provider in you App Client Settings**

During its deployment, data.all already creates an app client in your Cognito User Pool. Each of your app clients can use different identity providers and OAuth 2.0 settings. You must enable at least one identity provider for each app client. By default, the app client created by data.all uses Cognito User Pool to authenticate. This has to be changed to Azure AD federation.

Go to the **Application Integration** tab in the Amazon Cognito console and in the **App client settings** section, click in the hyperlink of the app client.



Under **Hosted UI**, click on Edit.



In the Identity providers section, **uncheck Cognito User Pool** and **check the Identity Provider Name** (ADTest in our example) you have created in this guide.

That's all settings which you should do in AWS console and Azure portal. You can now test your set-up.

## Testing your setup

You can easily test your setup when opening data.all. On the login page, you should see the name of Identity Provider you created. Click on it to login.

If you do not already have an active session opened in you browser, this redirects you to an authentication portal. use your usual credentials to connect to Data.all.

Inside data.all, create a new organization to check that attribute mapping worked as expected. This is what you need to verify:

- **Email address** : displayed when clicking your initial at the top right of the screen
- **Groups** : When creating an Organization in data.all, you can indicate a group. Check that you can select any group you are part of in your Azure AD environment.

