**Set up sign-up and sign-in with an Oracle Cloud account using Azure Active Directory B2C**

***Before you begin***, use the selector above to choose the type of policy you’re configuring. Azure AD B2C offers two methods of defining how users interact with your applications: through predefined [user flows](https://docs.microsoft.com/en-us/azure/active-directory-b2c/user-flow-overview), or through fully configurable [custom policies](https://docs.microsoft.com/en-us/azure/active-directory-b2c/custom-policy-overview). The steps required in this article are different for each method.

## Prerequisites

* [Create a user flow](https://docs.microsoft.com/en-us/azure/active-directory-b2c/tutorial-create-user-flows) to enable users to sign up and sign into your application.
* If you haven't already done so, [register a web application](https://docs.microsoft.com/en-us/azure/active-directory-b2c/tutorial-register-applications), and [enable ID token implicit grant](https://docs.microsoft.com/en-us/azure/active-directory-b2c/tutorial-register-applications#enable-id-token-implicit-grant).

## Create an app in the Oracle IDCS developer dashboard.

To enable sign-in for users with an Oracle IDCS account in Azure Active Directory B2C (Azure AD B2C), you need to create an application in [Oracle IDCS Cloud](https://cloud.oracle.com) . For more information, see [Create Oracle Identity cloud services](https://docs.oracle.com/en-us/iaas/Content/Identity/Concepts/federation.htm). If you don't already have an Oracle account, you can sign up at <https://Cloud.oracle.com/>.

1. Sign in to the [Oracle Cloud Developer Console](https://cloud.oracle.com) with your Oracle cloud account credentials.
2. From the Dashboard, select **Login with Oracle** **Cloud Account**
3. Create an **Oracle Identity Cloud Service application** to define the OAuth resource and OAuth client relationships. This Oracle Identity Cloud Service application can be configured as an OAuth resource server for accessing data resources using OAuth. Please provide below information.

Graphical user interface, text, application, email

Description automatically generated

1. Click on Oracle Identity Cloud Service URL and login with Oracle Cloud Account.

Graphical user interface, application

Description automatically generated

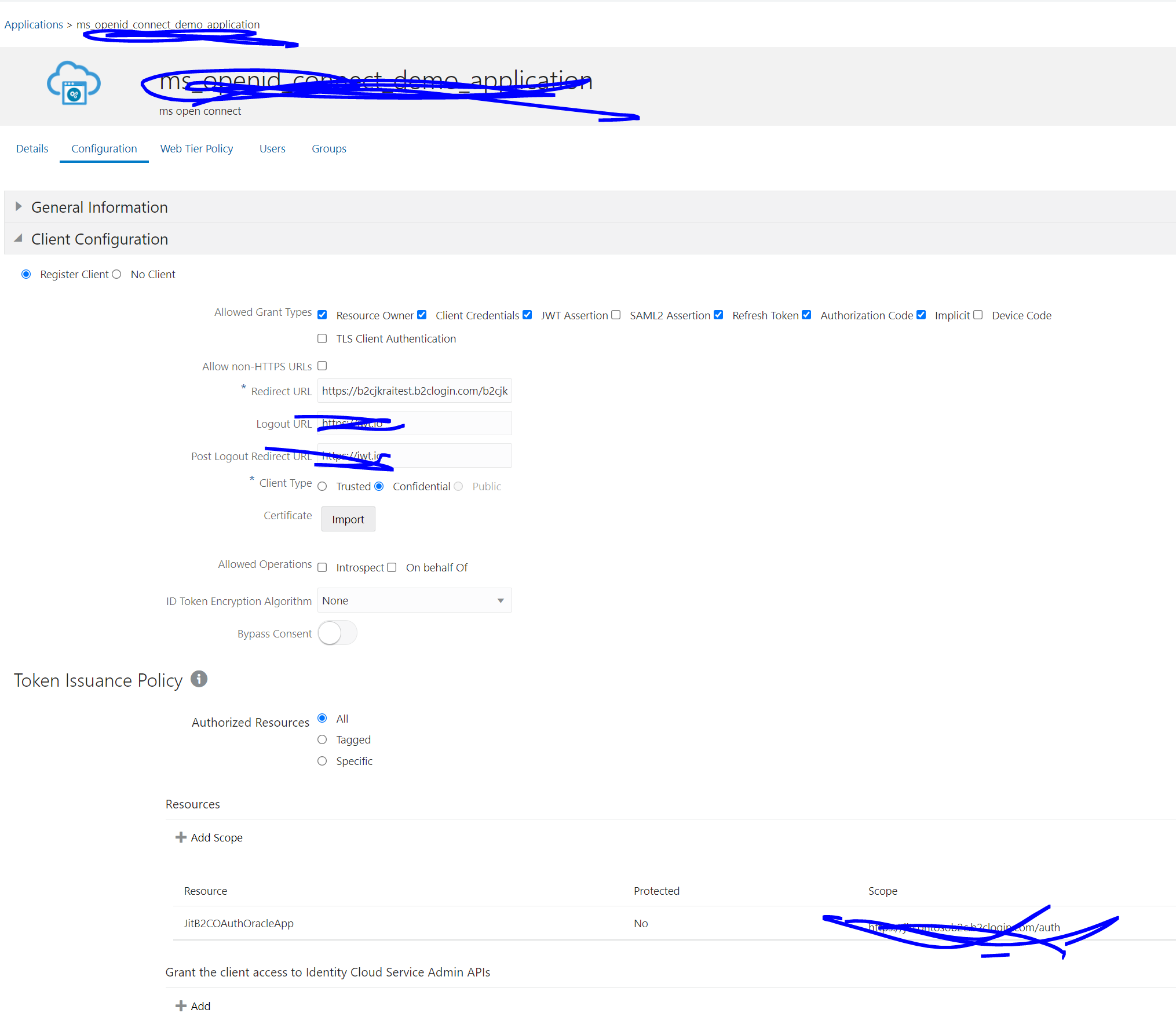
1. After Successful Login, Create Application.
2. In the Oracle Identity Cloud Service console, select the **Application tab**, click Add, and then in the Add Application window, select Trusted Application.

Graphical user interface

Description automatically generated with medium confidence

1. On the Details page of the Add Trusted Application wizard, give the new application a name. If you want, set other values such as description and tags. Click Next.
2. On the Client page, select Configure this application as a client now.
3. Additional options appear on the page. Set the following values:
4. Allowed Grant Types: Client Credentials and JWT Assertion.
5. Redirect URL: Enter a Redirect URL for the user to be sent after authentication, such as the Oracle SaaS application home page.

(https://<tenantname>.b2clogin.com/<tenantName>.onmicrosoft.com/oauth2/authresp)



1. Client Type: Select the Confidential option.
2. On the Resources page, select Configure this application as a resource server now.
3. Additional options appear on the page.
4. Set the following values:
5. Primary Audience: <your Oracle Applications base address>.
6. Allowed Scopes: Click Add and create a scope with the value /. Check the Requires Consent box.
7. Click Next, and on the Authorization page, click Finish to save the application.
8. An “Application Added” notification is shown. **Make a copy of the Client ID and Client Secret:** you’ll need to provide them to Oracle Support later.
9. If you need them later, **the Client ID and Client Secret** also appear on the Configuration tab in the Details section for the application.

Graphical user interface, application, Teams

Description automatically generated

1. With the app created and saved, select the Configuration tab, and expand the Client Configuration section. In the Accessing APIs from Other Applications section, under Allowed Scopes, click Add. In the Add Scope dialog, add one or more Resources. You can check the box for a resource to add all its scopes or click the right arrow for a given resource to select individual scopes. You can check the box for the whole app to add all resources. Click the Add button. Click Save to save your changes.
2. As you add scopes, they are listed by application and allowed scope in the Allowed Scopes area. You can select a scope in this area and click the Remove button to remove it.
3. To activate the application, from the Oracle Identity Cloud Service console, select Applications, and select the application. Click the Activate button to the right of the application name.

Graphical user interface, application, Teams

Description automatically generated

## Create Oracle identity provider.

1. In your Azure AD B2C tenant, select new OpenID Connect Provider using **Identity Provider under Manage menu.**
2. Provide the name (say OracleIDCS) , IDCS metadata URL, client id and client secret with required information.
3. Select **Save**.

A screenshot of a computer

Description automatically generated with medium confidence

## **Add Oracle identity provider to a user flow.**

At this point, the Oracle identity provider has been set up, but it's not yet available in any of the sign-in pages. To add the Oracle identity provider to a user flow:

1. In your Azure AD B2C tenant, select **User flows**.
2. Click the user flow that you want to add the Open Connect identity provider as you created.
3. Under the **Social identity providers**, select **OracleIDCS** .
4. Select **Save**.
5. To test your policy, select **Run user flow**.
6. For **Application**, select the web application named testapp1 that you previously registered. The **Reply URL** should show https://jwt.ms.
7. Select the **Run user flow** button.
8. From the sign-up or sign-in page, select **Oracle** to sign in with Oracle account.

If the sign-in process is successful, your browser is redirected to https://jwt.ms, which displays the contents of the token returned by Azure AD B2C.