

OAuth / OpenID Connect Single Sign-On (SSO) into Joomla using Azure AD

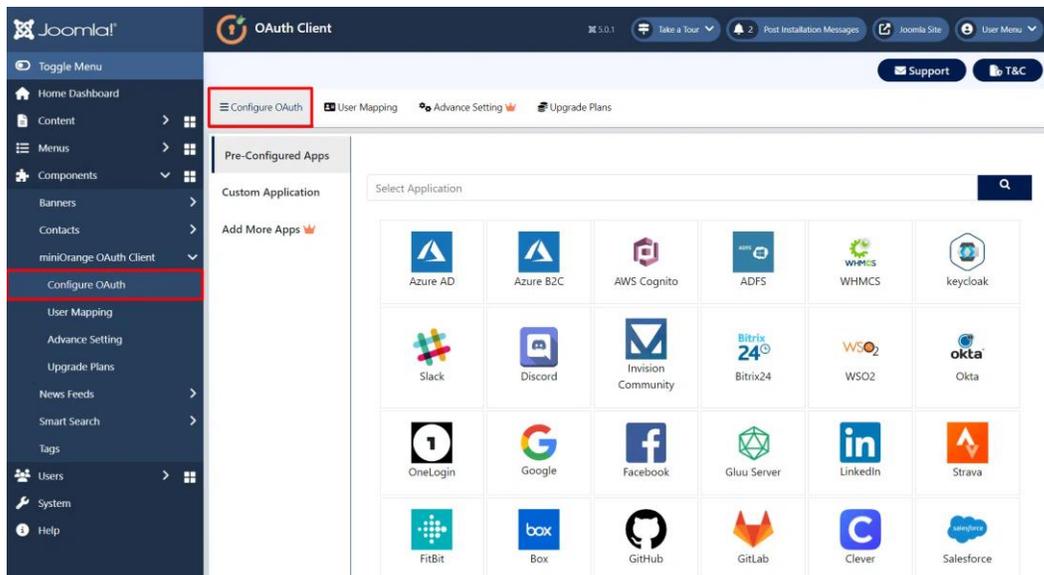
Single Sign-On (SSO) with Microsoft Entra ID (Azure AD) in Joomla uses OAuth Authorization to provide users secure access to the Joomla site. With our Joomla OAuth Single Sign-On (SSO) plugin, Microsoft Entra ID (Azure AD) acts as the OAuth provider, ensuring secure login for Joomla websites.

The integration of Joomla and Microsoft Entra ID (Azure AD) simplifies and secures the login process using OAuth protocol. This solution allows users to access their Joomla sites with Single Sign-On (SSO) using their Microsoft Entra ID (Azure AD) credentials, completely removing the need to store, remember, and reset multiple passwords.

In addition to offering OAuth Single Sign-On (SSO) using Microsoft Entra ID credentials, the plugin also provides advanced SSO features like user profile attribute mapping, role mapping, and Azure multi-tenant login and providing site access based on organization roles. For further insights into the array of features we offer within the Joomla OAuth & OpenID Connect Client plugin, kindly visit our page here. You can follow the below steps to setup Microsoft Entra ID (Azure AD) OAuth SSO with Joomla.

Steps to Install Joomla OAuth Client Plugin

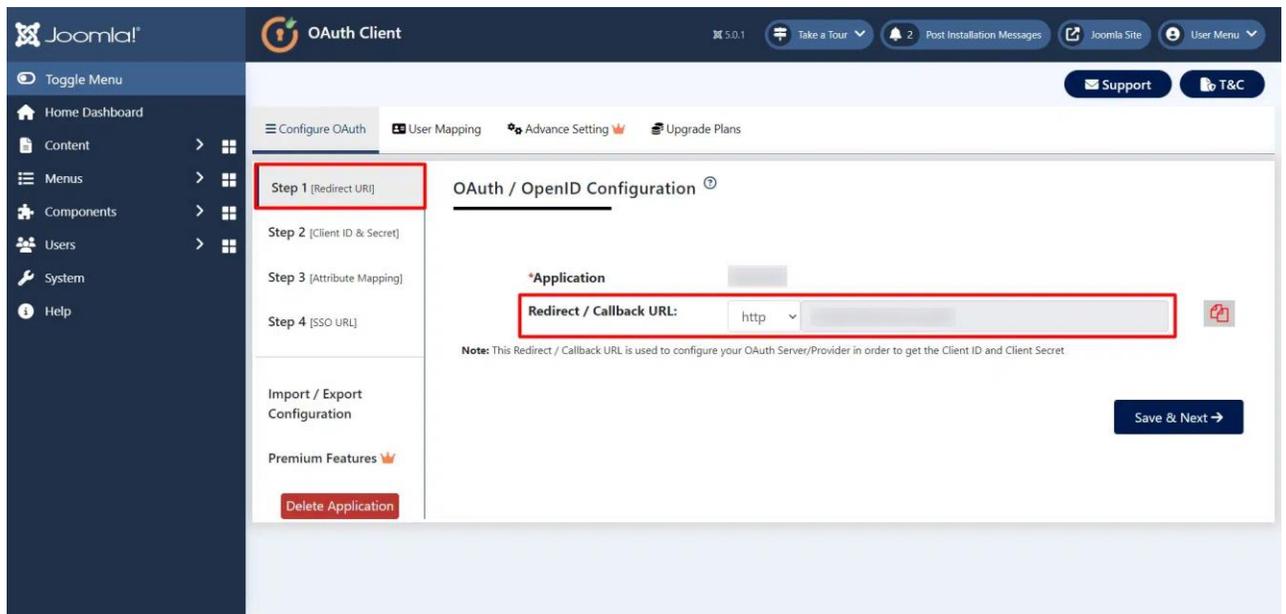
- [Download](#) the zip file for the **miniOrange OAuth Client plugin** for Joomla.
- Login into your Joomla site's **administrator console**.
- From left toggle menu, click on **System**, then under **Install** section click on **Extension**.
- Upload the downloaded zip file to install the **Joomla OAuth Client** plugin.
- Installation of the plugin is successful. Now click on **Start Using miniOrange OAuth Client plugin**.
- Under **Configure OAuth -> Pre-Configured Apps** tab, select your **OAuth Provider**. You can also search for **custom OAuth** or **custom OpenID application** in the search bar, and configure your own custom provider.



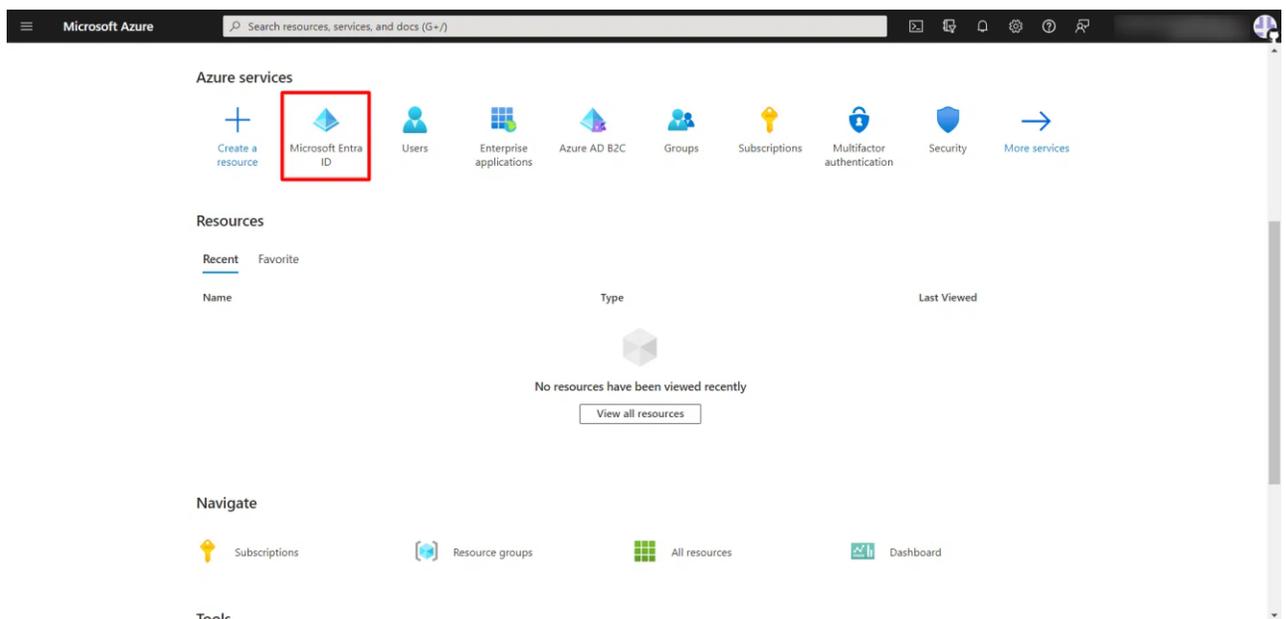
Step 1: Steps to configure Microsoft Entra ID OAuth SSO into Joomla

1. Configure Callback/Redirect URL

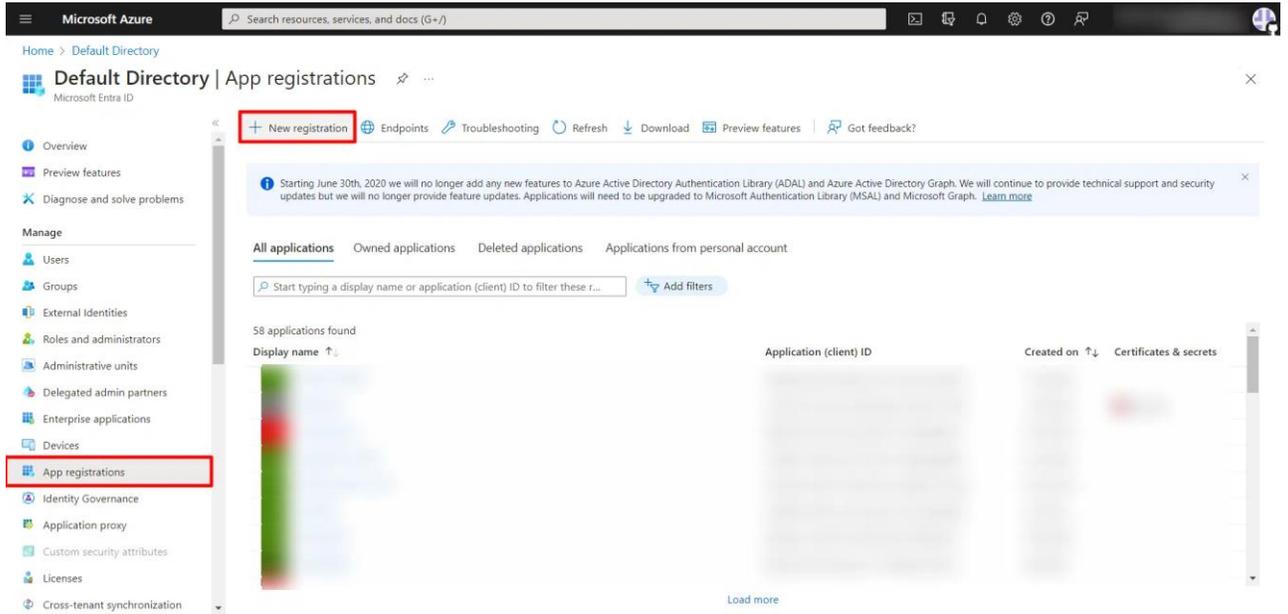
- After selecting your OAuth provider, you will be redirected to the **Step 1 [Redirect URL]** tab. Now copy the **Callback/Redirect URL** which we will use to configure Microsoft Entra ID as OAuth Server, then click on the **Save & Next** button.



- Log into the [Azure portal](#).
- Click on **Microsoft Entra ID** under **Azure services**.



- In the left-hand navigation pane, click the **App registrations**, and click on **New registration**.



- When the Create page appears, enter your application's registration information:

Name: Name of your application.

Application type :

1. Select *"Native"* for client applications that are installed locally on a device. This setting is used for OAuth public native clients
2. Select *"Web app / API"* for client applications and resource/API applications that are installed on a secure server. This setting is used for OAuth confidential web clients and public user-agent-based clients. The same application can also expose both a client and resource/API.

Sign-on URL :

1. For *"Web app / API"* applications, provide the base URL of your app. **eg, https://<domain-name>/mo_login** might be the URL for a web app running on your local machine. Users would use this URL to sign in to a web client application.
2. For *"Native"* applications, provide the URI used by Azure AD to return token responses. Enter a value specific to your application. **eg, https://localhost/joomla**

- Under **Redirect URL**, select **Web** from the dropdown and enter the **Callback URL** copied earlier in the given field. Then, click on the **Register** button to register the new application.

Microsoft Azure

Home > Default Directory | App registrations >

Register an application

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

TestOAuth

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

Accounts in this organizational directory only (Default Directory only - Single tenant)
 Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant)
 Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)
 Personal Microsoft accounts only

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 and it can be changed later, but a value is required for most authentication scenarios.

Web | https://domain_name/oauth/callback

Register an app you're working on here. Integrate gallery apps and other apps from outside your organization by adding from Enterprise applications.

By proceeding, you agree to the Microsoft Platform Policies

Register

2. Configure Client ID and Secret

- Now go to the **Overview** tab of your **registered application**. Here, copy the **Application ID** and the **Directory ID**, this will be your **Client ID** and **Tenant ID** respectively.

Microsoft Azure

Home > Default Directory | App registrations >

TestOAuth

Search

Delete Endpoints Preview features

Overview

Quickstart

Integration assistant

Manage

Branding & properties

Authentication

Certificates & secrets

Token configuration

API permissions

Expose an API

App roles

Owners

Roles and administrators

Manifest

Support + Troubleshooting

Troubleshooting

New support request

Essentials

Display name : TestOAuth

Application (client) ID : [Redacted]

Object ID : [Redacted]

Directory (tenant) ID : [Redacted]

Supported account types : My organization only

Client credentials : Add a certificate or secret

Redirect URIs : 1.web, 0.spa, 0.public client

Application ID URI : Add an Application ID URI

Managed application in L... : TestOAuth

Get Started Documentation

Microsoft identity platform Code samples Help and Support

Authentication scenarios Microsoft Graph Glossary

Authentication libraries

- Go to **Certificates and Secrets** from the left navigation pane and click on **New Client Secret**. Enter description and expiration time and click on **Add** option.

TestOAuth - Certificates & secrets

Search (Ctrl+/) <<

- Overview
- Quickstart
- Manage
 - Branding
 - Authentication
 - Certificates & secrets**
 - Token configuration (preview)
 - API permissions
 - Expose an API
 - Owners
 - Roles and administrators (Previ...
 - Manifest
- Support + Troubleshooting
 - Troubleshooting
 - New support request

Add a client secret

Description
Secret Key

Expires
 In 1 year
 In 2 years
 Never

Add Cancel

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as a...

+ New client secret

Description	Expires	Value
No client secrets have been created for this application.		

- Copy **value**. This will be your **Client Secret**.

Home > Default Directory | App registrations > AzureCustomer

AzureCustomer | Certificates & secrets

Search << Got feedback?

- Overview
- Quickstart
- Integration assistant
- Manage
 - Branding & properties
 - Authentication
 - Certificates & secrets**
 - Token configuration
 - API permissions
 - Expose an API
 - App roles
 - Owners
 - Roles and administrators
 - Manifest

Got a second to give us some feedback? →

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Application registration certificates, secrets and federated credentials can be found in the tabs below.

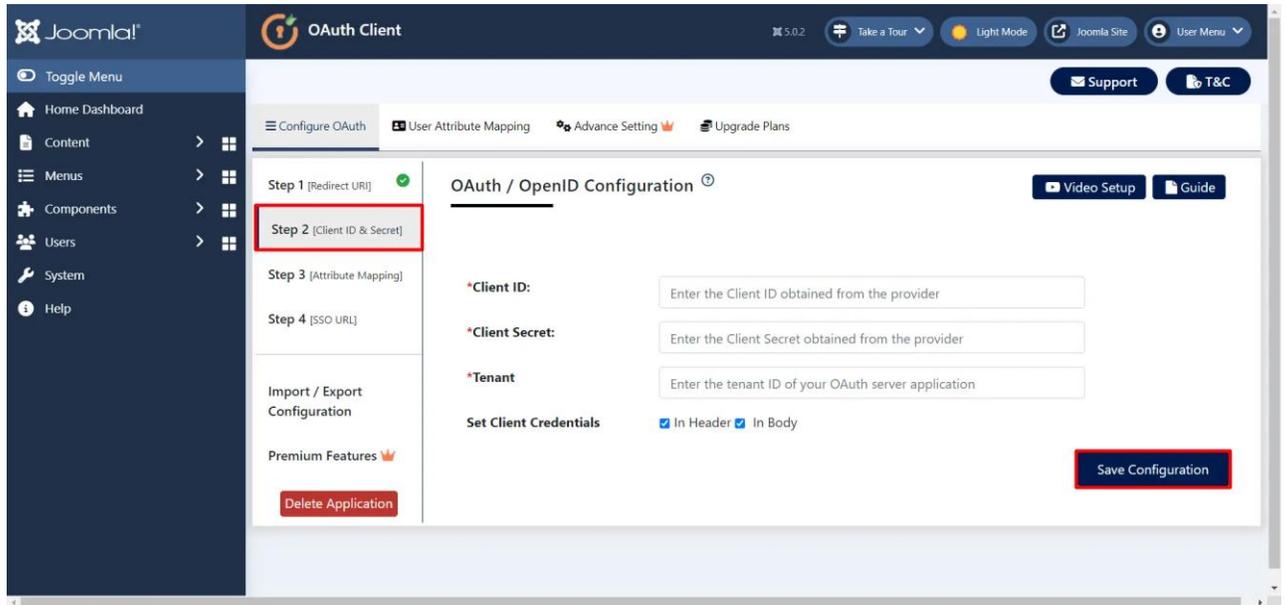
Certificates (0) **Client secrets (1)** Federated credentials (0)

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

+ New client secret

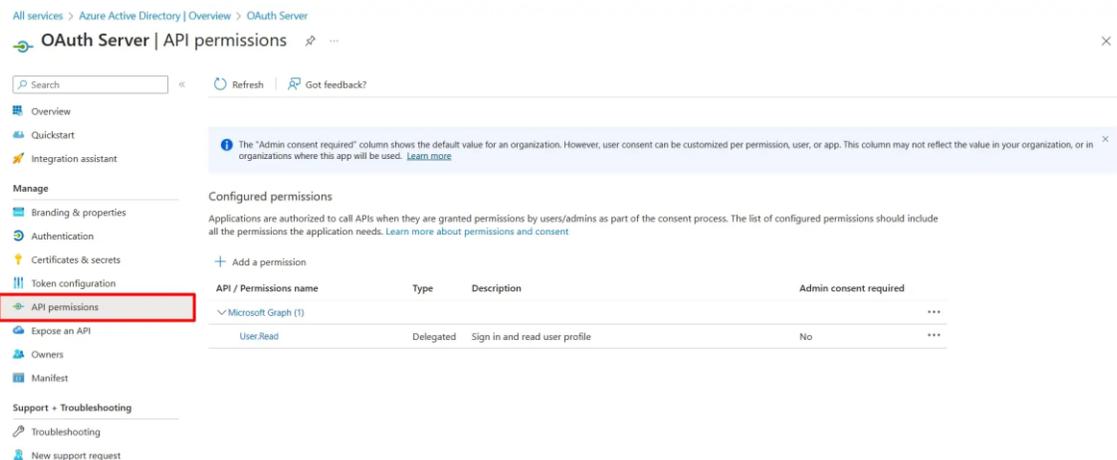
Description	Expires	Value	Secret ID
Test	7/18/2023	[Redacted]	[Redacted]

- Go to the **Step 2 [Client ID & Secret]** tab of the Joomla OAuth Client plugin, here paste the **Client ID**, **Client Secret** and **Tenant**. Click on the **Save Configuration** button.



2.1. Scope & Endpoints

- If you want to enable scopes, you can follow the following steps:
- Go to **Application** -> Select the application where you want to enable scopes. Now, Go to the **API Permissions** tab.



- Click on the **Add permission** button, and then **Microsoft Graph API -> Delegated Permissions** and select **openid, Profile scope** and click on the **Add Permissions** button.

Request API permissions

< All APIs

Microsoft Graph
<https://graph.microsoft.com/> [Docs](#)

What type of permissions does your application require?

Delegated permissions
Your application needs to access the API as the signed-in user.

Application permissions
Your application runs as a background service or daemon without a signed-in user.

Select permissions expand all

i The "Admin consent required" column shows the default value for an organization. However, user consent can be customized per permission, user, or app. This column may not reflect the value in your organization, or in organizations where this app will be used. [Learn more](#)

Permission	Admin consent required
<input checked="" type="checkbox"/> OpenId permissions (3) <input checked="" type="checkbox"/> email <small> ⓘ</small> View users' email address	No
<input type="checkbox"/> offline_access <small> ⓘ</small> Maintain access to data you have given it access to	No
<input checked="" type="checkbox"/> openid <small> ⓘ</small> Sign users in	No
<input checked="" type="checkbox"/> profile <small> ⓘ</small> View users' basic profile	No

- Click on the **Grant admin consent for default directory** for Demo button.

Configured permissions

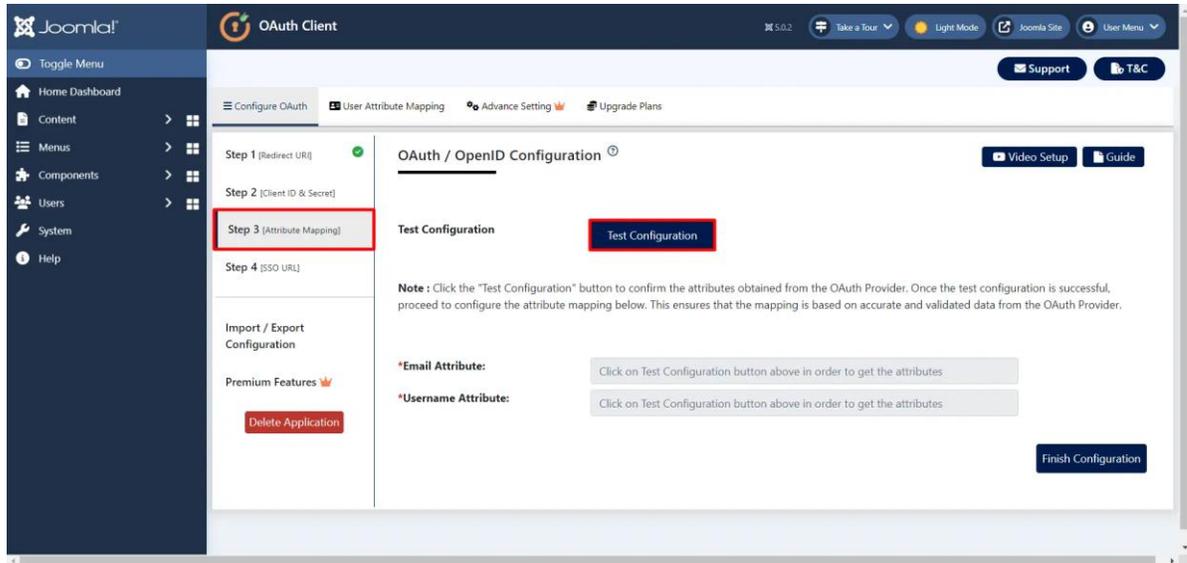
Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. [Learn more about permissions and consent](#)

+ Add a permission ✔ Grant admin consent for Default Directory

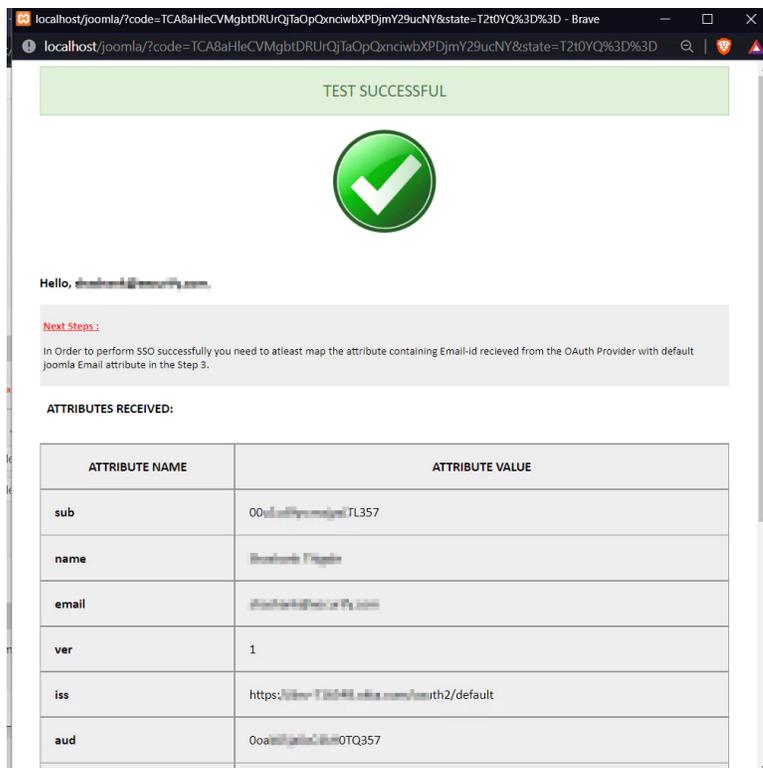
API / Permissions name	Type	Description	Admin consent requ...	Status
Microsoft Graph (9) ...				
openid	Delegated	Sign users in	No	✔ Granted for Default Dire... ...
profile	Delegated	View users' basic profile	No	✔ Granted for Default Dire... ...

3. Configure Attribute Mapping

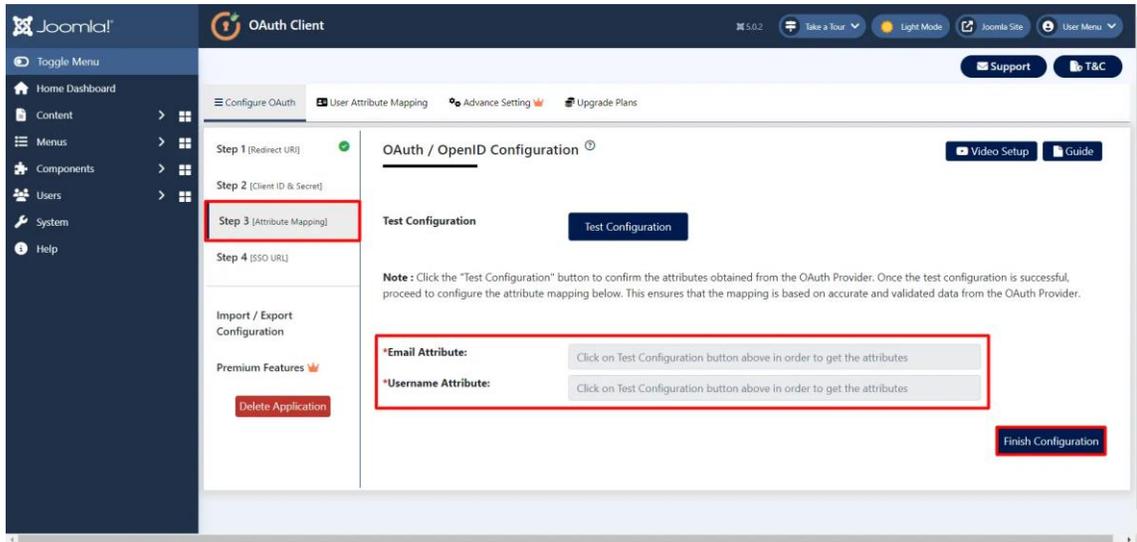
- User Attribute Mapping is mandatory for enabling users to successfully login into Joomla. We will be setting up user profile attributes for Joomla using below settings.
- Go to **Step 3 [Attribute Mapping]** tab and click on **Test Configuration** button.



- You will be able to see the attributes in the Test Configuration output as follows.

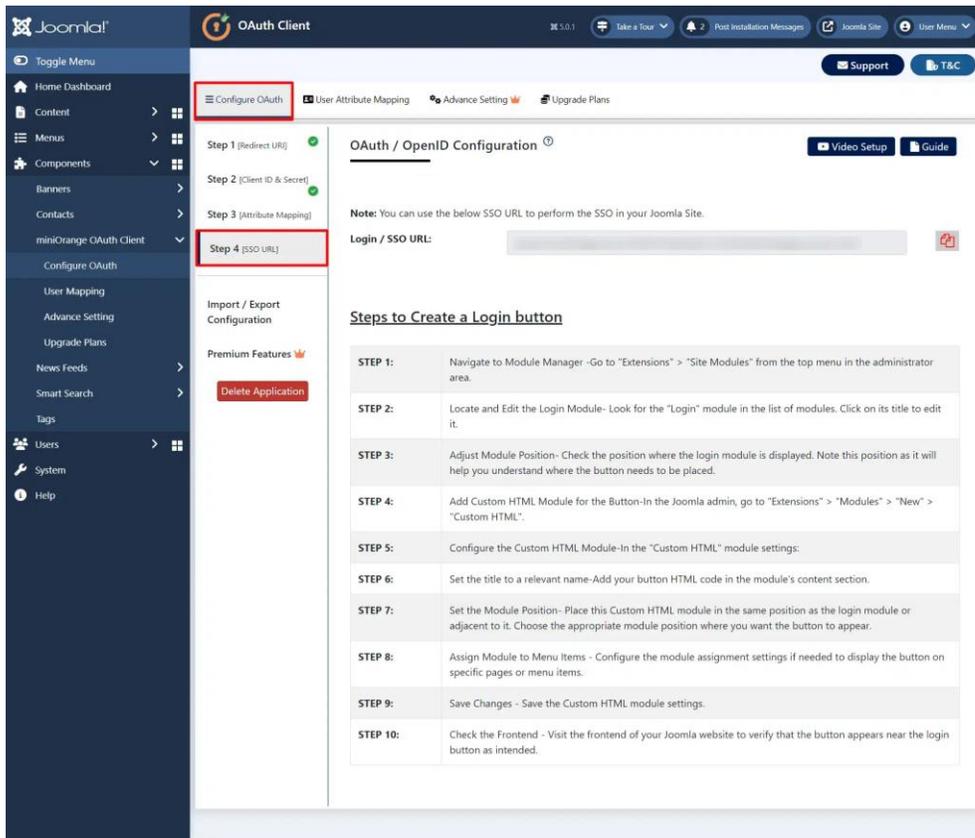


- Now go to the **Step 3 [Attribute Mapping]** tab and Select the attribute name for **Email and Username** from dropdown. Then click on **Finish Configuration** button.



4. Setup Login/SSO URL

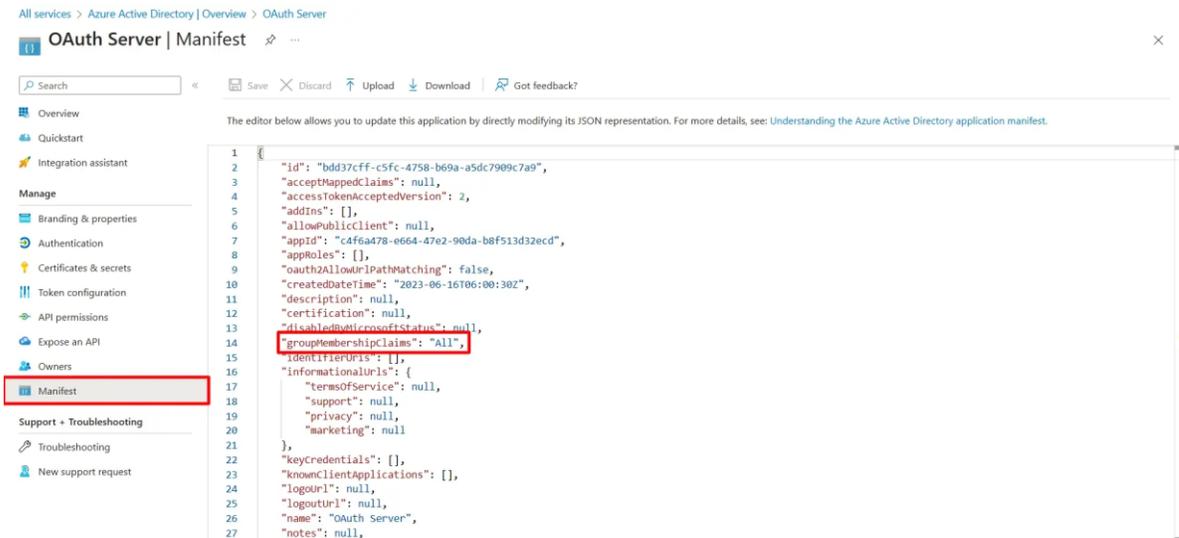
- Now go to **Step 4 [SSO URL]** tab, here copy the **Login/SSO URL** and add it to your Site by following the given steps.



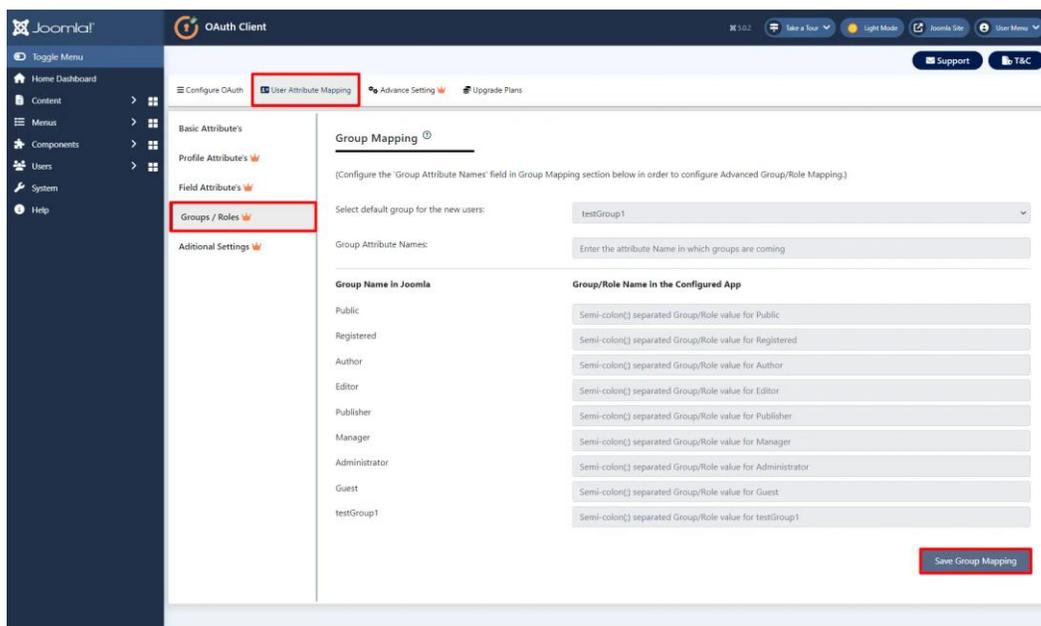
- Now logout and go to your Joomla site's pages where you have added this link. You will see a login link where you placed that button. Click on this button to perform SSO.

5. Configure Group/Role Mapping

- Go to **Manifest** tab and find **groupMembershipClaims** and changes it's value to "All" and click on the **save** button.



- Now go to the **User Attribute Mapping** tab in **Joomla OAuth Client plugin**, and click on **Group/Roles** tab from the left hand bar. Here, you can configure the '**Group Attribute Names**' field in order to configure Advanced Group/Role Mapping. Then click on the **Save Group Mapping** button.



In this guide, you have successfully configured Joomla Azure AD Single Sign-On (SSO) by configuring Azure AD as OAuth Provider and Joomla as OAuth Client using our Joomla OAuth Client plugin. This solution ensures that you are ready to roll out secure access to your Joomla site using Azure AD login credentials within minutes.