

# Configuring Google as an OAuth Provider in PHPKB

265 Palwinder Singh February 28, 2023 Documentation

3591 0

Configuring Google as an OAuth Provider in PHPKB Knowledge Base Software is a process that enables users to log in to PHPKB using their Google credentials. This can be a convenient and secure way for users to access the system, as it eliminates the need for separate login credentials and ensures that users are authenticated by a trusted third party.

## How to configure Google as an OAuth provider in PHPKB?

In this tutorial, you will learn how to configure **Google** as **OAuth Provider** with PHPKB OAuth plugin.

### PHPKB OAuth Authentication Plugin

The OAuth 2.0 authentication plugin enables users to log in using their Google, Microsoft, Facebook, or any other account via buttons on the login page of your knowledge base.

Interested to buy this plugin? [Contact Us](#) for Licensing & Pricing.

To configure Google as an OAuth Provider in PHPKB, follow these steps:

## PHP Requirements:

PHP 7.0 or later

CURL extension

JSON extension

OpenSSL extension

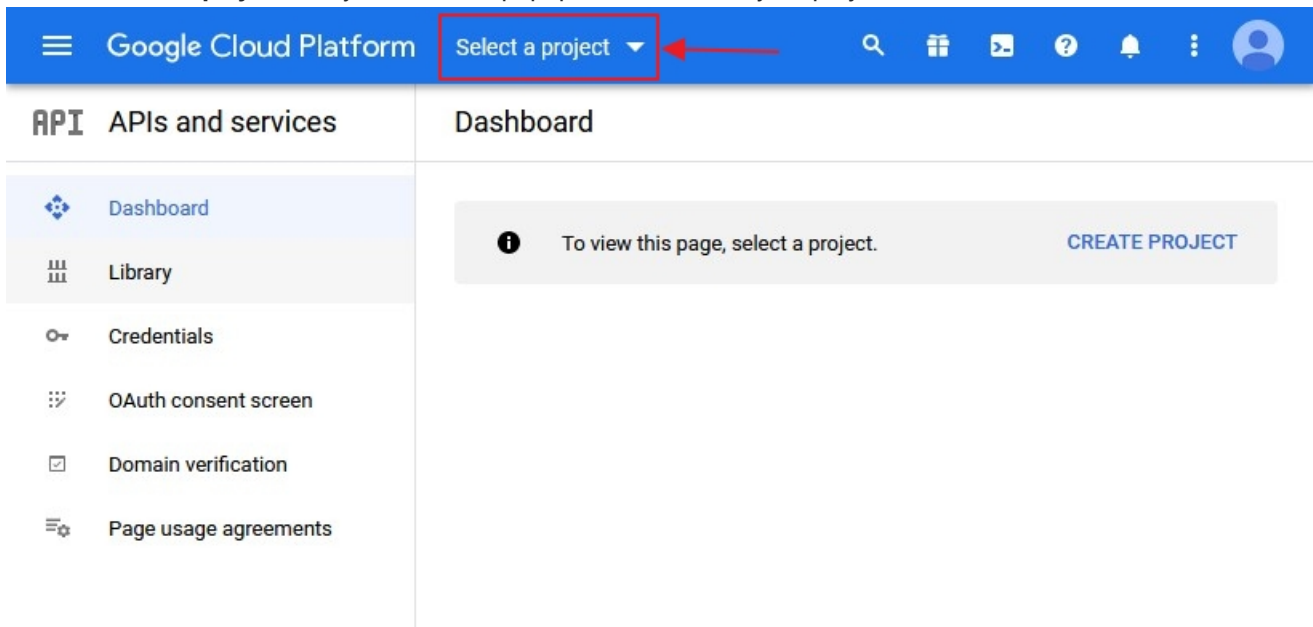
## Download & Installation:

Extract the package (that you received after purchasing this plugin), copy all the files & folders, and paste them at their respective locations under the installation directory of [PHPKB Knowledge Management Software](#) on your server. There is a new folder, called 'add-ons' (applicable to PHPKB v9.0), copy that and put it directly under the root folder (i.e. outside /admin/ folder) of the PHPKB package.

## Step 1: Setup Google as an OAuth Provider


Go to <https://console.developers.google.com> and click on "Login / Signup".


Click on **Select a project**, and you will see a popup with a list of all your projects.



You can click on the **New Project** button to create a new project.

Google Cloud Platform

Select from **NO ORGANISATION** 



**NEW PROJECT** 

API APIs and services

- Dashboard
- Library
- Credentials
- OAuth consent screen
- Domain verification
- Page usage agreements

Search projects and folders

RECENT STARRED ALL


Name	ID
 No organisation 	0

CANCEL OPEN



Enter your Project name under the **Project Name** field and click on **Create**.

Google Cloud Platform

### New Project

Project name \*  

Project ID: -322611. It cannot be changed later. [EDIT](#)

Organisation \*   

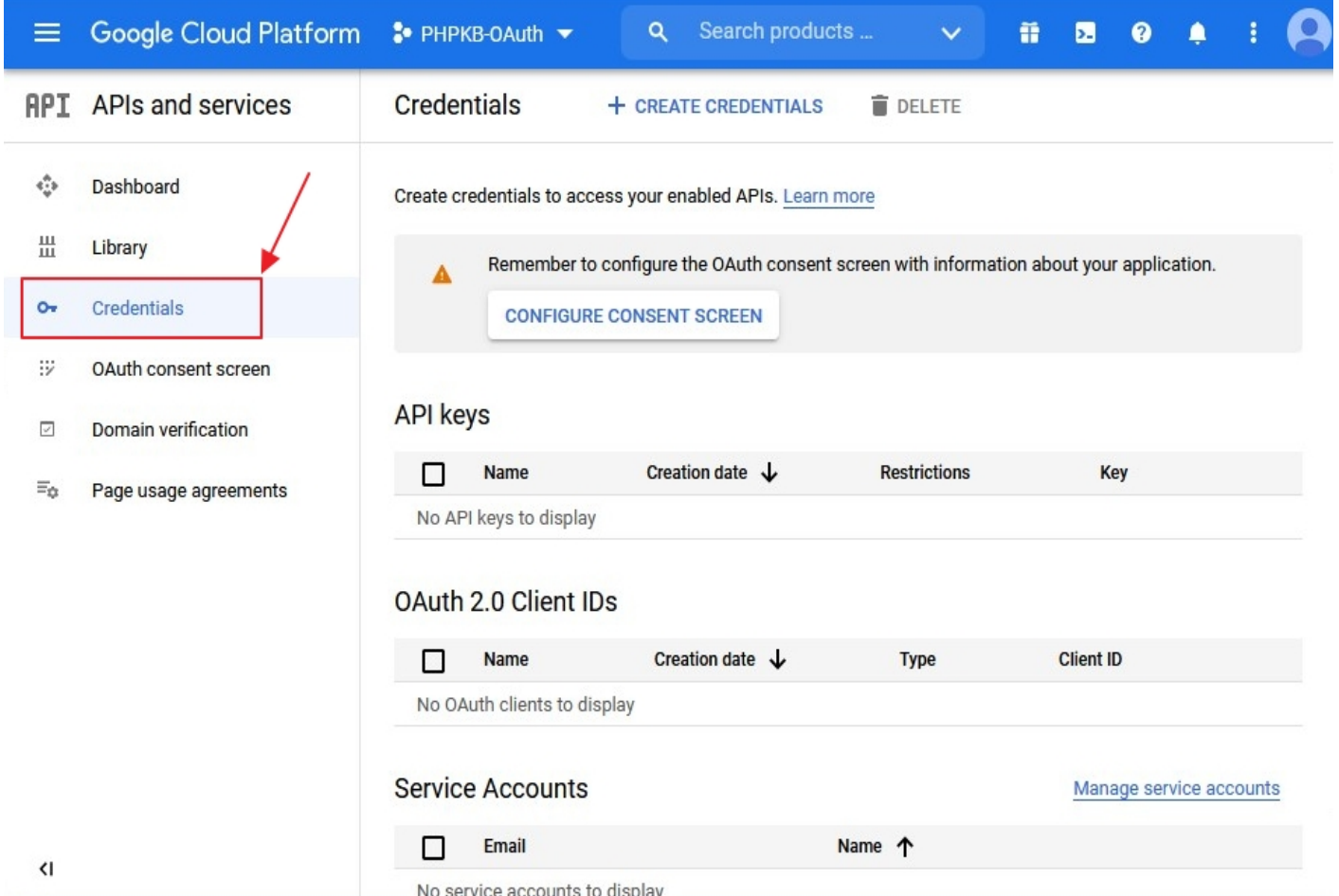
Select an organisation to attach it to a project. This selection can't be changed later.

Location \*  [BROWSE](#)

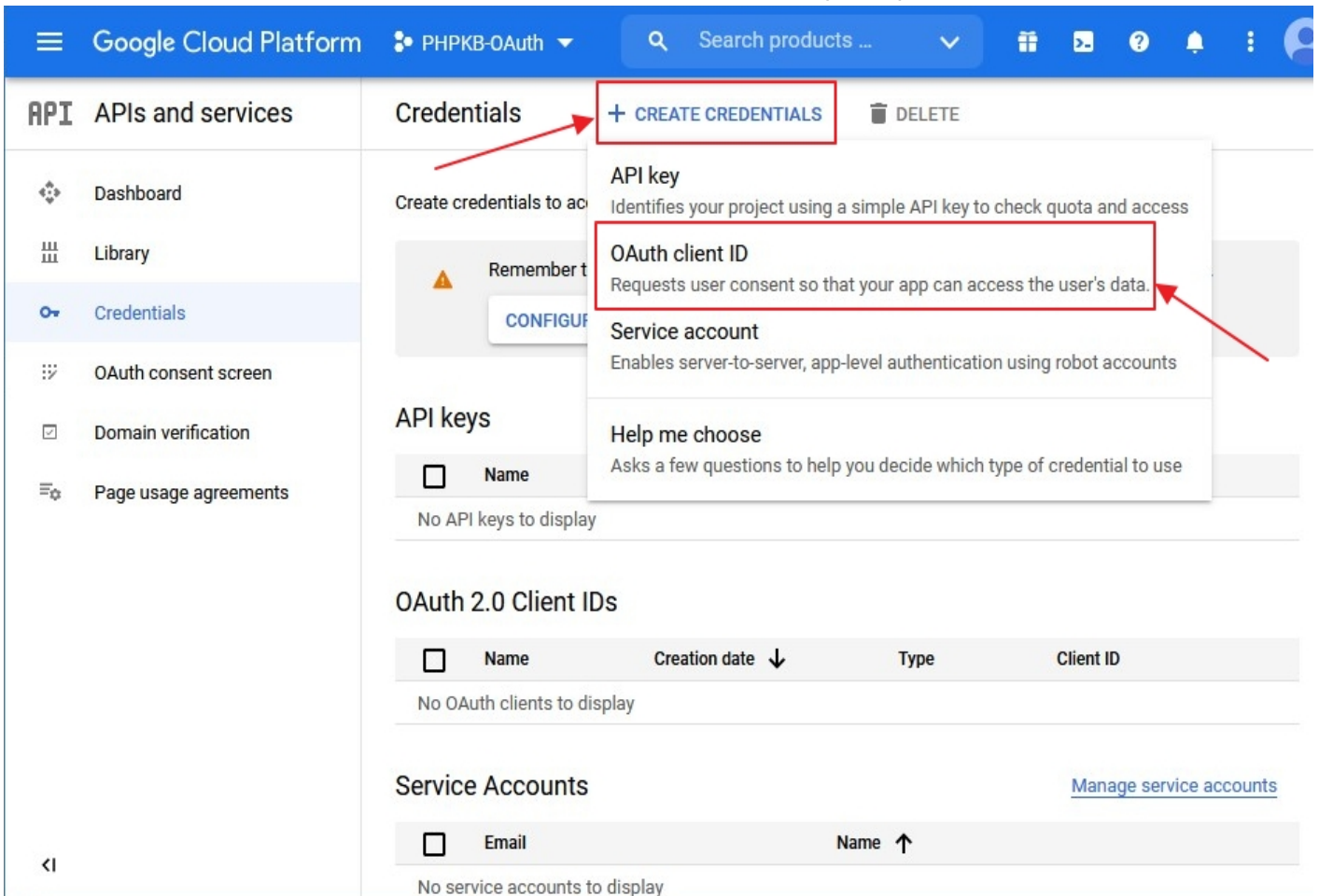
Parent organisation or folder

**CREATE** CANCEL

Go to **Navigation Menu > APIs > Services > Credentials** .



Click on **Create Credentials** button and then select **OAuth Client ID** from the options provided.



In case you are facing some warning saying that in order to create an **OAuth Client ID**, you must set a product name on the consent screen. Click on the **Configure consent screen** button.

Google Cloud Platform PHPKB-OAuth Search products and resources

API APIs and services < Create OAuth client ID

Dashboard  
Library  
Credentials  
OAuth consent screen  
Domain verification  
Page usage agreements

A client ID is used to identify a single app to Google's OAuth servers. If your app runs on multiple platforms, each will need its own client ID. See [Setting up OAuth 2.0](#) for more information.

To create an OAuth client ID, you must first set a product name on the consent screen.

CONFIGURE CONSENT SCREEN



Google Cloud Platform PHPKB-OAuth Search products and resources

API APIs and services OAuth consent screen

Dashboard  
Library  
Credentials  
OAuth consent screen  
Domain verification  
Page usage agreements

Choose how you want to configure and register your app, including your target users. You can only associate one app with your project.

User Type

Internal ⓘ

Only available to users within your organisation. You will not need to submit your app for verification. [Learn more](#)

External ⓘ

Available to any test user with a Google Account. Your app will start in testing mode and will only be available to users that you add to the list of test users. Once your app is ready to push to production, you may need to verify your app. [Learn more](#)

CREATE

[Let us know what you think](#) about our OAuth experience

Enter your product name under the **Product Name Shown** to Users field. Click on the **SAVE** button to save your settings.

Google Cloud Platform PHPKB-OAuth Search products and resources

API APIs and services

- Dashboard
- Library
- Credentials
- OAuth consent screen**
- Domain verification
- Page usage agreements

### Edit app registration

Application home page  
Provide users a link to your home page

Application privacy policy link  
Provide users a link to your public privacy policy

Application Terms of Service link  
Provide users a link to your public Terms of Service

#### Authorised domains

When a domain is used on the consent screen or in an OAuth client's configuration, it must be pre-registered here. If your app needs to go through verification, please go to the [Google Search Console](#) to check if your domains are authorised. [Learn more](#) about the authorised domain limit.

phpkb.com

+ ADD DOMAIN

#### Developer contact information

Email addresses \*  
These email addresses are for Google to notify you about any changes to your project.

SAVE AND CONTINUE CANCEL

Now for configuring scopes, click on **Add or Remove the Scopes** button.

Google Cloud Platform PHPKB-OAuth Search products and resources

API APIs and services

- Dashboard
- Library
- Credentials
- OAuth consent screen**
- Domain verification
- Page usage agreements


### Edit app registration

OAuth consent screen — 2 **Scopes** — 3 Summary

Scopes express the permissions that you request users to authorise for your app and allow your project to access specific types of private user data from their Google Account. [Learn more](#)

ADD OR REMOVE SCOPES

Now, select the **Scopes** to allow your project to access specific types of private user data from their Google Account and click on **Save and Continue** button.



Go to the **Credentials** tab and click on **Create Credentials** button. Select **Web Application** from the dropdown list to create a new application.

## API APIs and services

## ← Create OAuth client ID

- Dashboard
- Library
- Credentials
- OAuth consent screen
- Domain verification
- Page usage agreements

A client ID is used to identify a single app to Google's OAuth servers. If your app runs on multiple platforms, each will need its own client ID. See [Setting up OAuth 2.0](#) for more information.

Application type \*

Web application

Android

Chrome app

iOS

TVs and Limited Input devices

Desktop app

Universal Windows Platform (UWP)

Enter the name you want for your Client ID under the name field and enter the **Redirect / Callback URI** from PHPKB OAuth / OpenID-Connect plugin under the Redirect URL field.

- Dashboard
- Library
- Credentials**
- OAuth consent screen
- Domain verification
- Page usage agreements

A client ID is used to identify a single app to Google's OAuth servers. If your app runs on multiple platforms, each will need its own client ID. See [Setting up OAuth 2.0](#) for more information.

Application type \*  
Web application

[Learn more](#) about OAuth client types

Name \*  
PHPKB-OAuth-CL

The name of your OAuth 2.0 client. This name is only used to identify the client in the console and will not be shown to end users.

The domains of the URIs you add below will be automatically added to your [OAuth consent screen](#) as [authorised domains](#).

Authorised JavaScript origins ?

For use with requests from a browser

+ ADD URI

Authorised redirect URIs ?

For use with requests from a web server

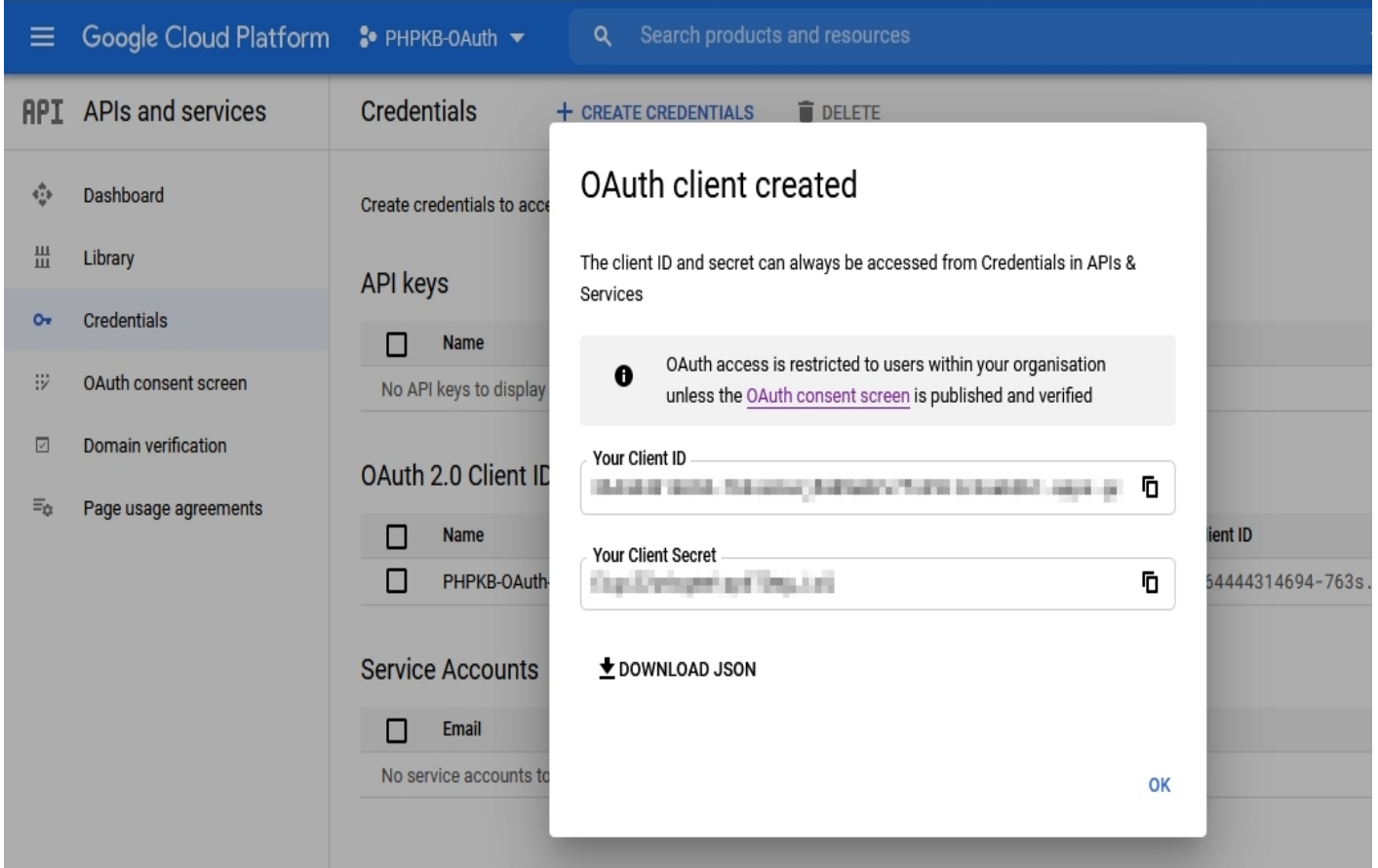
URIs \*

+ ADD URI

CREATE CANCEL

Click on the SAVE button to save your configurations. Now, you have successfully completed your Google App OAuth Server side configurations.






### Google Endpoints and Scope:

<b>Scope</b>	openid email
<b>Authorize Endpoint</b>	https://accounts.google.com/o/oauth2/auth
<b>Access Token Endpoint</b>	https://www.googleapis.com/oauth2/v4/token
<b>Get User Info Endpoint</b>	https://www.googleapis.com/oauth2/v1/userinfo
<b>Grant Type</b>	Authorization Code

## Step 2: Setup PHPKB as OAuth Client

Login to **Administrator Control Panel** as **Superuser** and go to **Tools > OAuth / OpenID-Connect**.  
Let's configure the **Basic Configuration**.

 **Tip:** For easier configuration, there are helpful notes added under the fields/options where it is needed.

## BASIC CONFIGURATION

Configure basic settings, like Enable OAuth, Redirect URL, App Name, Client ID, Client Secret, Scope, Authorize Endpoint, Access Token Endpoint, etc.

**Enable OAuth?**  Yes  
Whether you want to enable & show OAuth on Login page to users or not.

**Issuer URL \***   
The initial endpoint that is contacted by the relying party to begin a flow (must be a valid Base URL for Discovery configuration and PHPKB will fetch Authorization URL, Issuer URL, etc. using this Base URL). It must start with HTTPS for production and it may start with HTTP on localhost.  
For example: If the Discovery Document URL is https://login.example.com/well-known/openid-configuration then just enter https://login.example.com

**Redirect URL**  add-ons/oauth/index.php   
This is where user will be redirected after a successful authentication. You should add this URL in your OAuth Server as OAuth Redirect or Callback URL.

**App Name (Optional)**   
You can enter the App Name, like Google. Default: OAuth

**Display Name (Optional)**   
Please enter what you want to show on Login button. Default: Login with <App Name>

**Client ID \***   
A publicly exposed string that is used by the service API to identify the application. It is also used to build authorization.

**Client Secret \***   
Secret is used to authenticate the identity of the application to the service API when the application requests to access a user account. It must be kept private between the application and the API.

**Scope**   
The scopes that are associated with access tokens determine what resources are available when they are used to access OpenID connect protected endpoints. For example: openid profile name email phone  
Note: This is fetched automatically if the Discovery configuration is found.

**Authorize Endpoint \***

**Access Token Endpoint \***

**Get User Info Endpoint \***

**Grant Type**

Now configure the **Advanced Settings** section.

## ADVANCED SETTINGS

You can configure more options, like Create user if not exists, Keep existing users, Update user data, Default group(s) assignment, and Default Role.

Create user if not exists?  Yes

Auto-provisioning. If user does not exist, PHPKB will create a new user with the data provided by the Identity Provider.

Auto-linking existing users?  Yes

If a PHPKB account already exists with the same identity as a newly-authenticated user over OpenID Connect, login as that user instead of generating an error.

Update User Data?  Yes

Auto-update. PHPKB will update the account details of user with the data provided by the server.

Match PHPKB account by

email

Select what field will be used in order to find the user account. If "email", the plugin will prevent the user from changing his/her email address in My Profile page.

Default Group(s) Assignment

Choose default group(s) assignment

The default group(s) that will be assigned to Member users when they would log in to PHPKB via OAuth.

Then configure the **Attribute Mapping** section.

 **Tip:** To create users as Member users, you can set the 'Default Role' as 'Member', otherwise, change it accordingly.

## ATTRIBUTE MAPPING

Sometimes the names of the attributes sent by the identity provider do not match the names used by PHPKB for the user accounts. In this section you can set the mapping between provider fields and PHPKB fields. Note: This mapping may also be set at identity provider's side (if supported).

Username \*

email

Email \*

email

First Name

name

Last Name

Role


The attribute that contains the role of the user, for example 'memberOf'.

Default Role

Member

Default role assignment to all users.

The next is the **Role Mapping** section where you can map the roles returned by IdP with the roles that are available in PHPKB.

 If your IdP does not return any roles then you can skip this section.

## ROLE MAPPING

The Identity Provider can use its own roles. In this section, you can set the mapping between IdP and PHPKB roles. Accepts comma separated values. Example: admin,owner,superuser

Member


Writer

Writer-Trusted

Editor

Superuser

Then you can set **Role Precedence** for different roles.

 If your IdP does not return any roles then you can skip this section.

## ROLE PRECEDENCE

In some cases, the IdP returns more than one role. In this section, you can set the precedence of the different roles. The smallest integer will be the role chosen.

Member

Writer

Writer-Trusted

Editor

Superuser

Finally, set up **Security Settings** accordingly (if needed).

## SECURITY SETTINGS

Configure Proxy and **cert** (certificate) path.

Configure Proxy?

Configure a proxy. For example: http://my.proxy.com:80

Configure Cert Path?

Configure a cert. For example: /path/to/my.cert

Test the configuration by going to the **Login** page (either in the Public area or Admin area).

That's all! By following these steps, you can configure Google as an OAuth Provider in PHPKB, providing users with a convenient and secure way to access the system.

Online URL: <https://www.phpkb.com/kb/article/configuring-google-as-an-oauth-provider-in-phpkb-265.html>