

# How to set up the Auth Plugins

## How to Get the Oauth plugins Activated that are provided:

If you want to get the auth plugins in the auth folder please look at the list to copy the HTML code to Paste into the login.php file in the Main Folder

LIST:

- [Discord](#)
- [Facebook](#)
- [Google](#)
- [Github](#)
- [Steamauth](#)
- [Twitter](#)

## Authentication SSO Setup

What is a SSO?

SSO stands for Single Sign-On. It is a mechanism that allows users to authenticate themselves once and gain access to multiple systems or applications without the need to re-authenticate for each individual system. In other words, with SSO, users only need to log in once to access multiple applications or services.

Here's how SSO typically works:

1. **User Authentication:** When a user attempts to access an application or system, they are prompted to authenticate themselves by providing their credentials (e.g., username and password).
2. **Authentication Request:** The application or system that the user is trying to access sends an authentication request to the SSO system.
3. **SSO System Verification:** The SSO system verifies the user's credentials by checking against its user directory or database. If the credentials are valid, the SSO system generates a token.
4. **Token Generation:** The SSO system generates a token, which contains information about the user's authentication status and privileges.

5. Token Response: The SSO system sends the token back to the requesting application or system.
6. Token Validation: The application or system validates the received token with the SSO system to ensure its authenticity and integrity.
7. Access Granted: If the token is valid, the application or system grants the user access without requiring them to enter their credentials again.

The benefits of SSO include improved user experience, increased productivity, reduced password fatigue, and simplified user management for administrators. With SSO, users can seamlessly navigate between different applications or systems without the hassle of remembering multiple usernames and passwords. SSOs can be implemented for Employees (B2E) or Administrators (BTA) and Customers (B2C)

TABLE of Supported SSO authentication protocols

- OAuth 1
- OAuth 2
- Open ID

Now for the guide

Adding Authentication to a Login Page Using Config.php

To implement authentication in a login page using the Config.php file, follow the steps below:

Step 1: Configure the OAuth client credentials for each SSO provider

In the Config.php file, locate the following lines and replace the empty values with the OAuth client credentials for each SSO provider:

```
// The OAuth client ID associated with your API Provider Developer account.
define('provider_oauth_client_id', 'YOUR_CLIENT_ID');
// The OAuth client secret associated with your API Provider Developer account.
define('provider_oauth_client_secret', 'YOUR_CLIENT_SECRET');
// The URL to the Provider OAuth file.
```

```
define('provider_oauth_redirect_uri', 'YOUR_REDIRECT_URI');
```

Replace 'YOUR\_CLIENT\_ID', 'YOUR\_CLIENT\_SECRET', and 'YOUR\_REDIRECT\_URI' with the actual credentials provided by each SSO provider.

For example, to configure the SSO providers you mentioned, the lines would look like this:

Discord

```
define('discord_oauth_client_id', 'YOUR_DISCORD_CLIENT_ID');  
define('discord_oauth_client_secret', 'YOUR_DISCORD_CLIENT_SECRET');  
define('discord_oauth_redirect_uri', 'YOUR_DISCORD_REDIRECT_URI');
```

Facebook

```
define('facebook_oauth_client_id', 'YOUR_FACEBOOK_CLIENT_ID');  
define('facebook_oauth_client_secret', 'YOUR_FACEBOOK_CLIENT_SECRET');  
define('facebook_oauth_redirect_uri', 'YOUR_FACEBOOK_REDIRECT_URI');
```

Google

```
define('google_oauth_client_id', 'YOUR_GOOGLE_CLIENT_ID');  
define('google_oauth_client_secret', 'YOUR_GOOGLE_CLIENT_SECRET');  
define('google_oauth_redirect_uri', 'YOUR_GOOGLE_REDIRECT_URI');
```

Github

```
define('github_oauth_client_id', 'YOUR_GITHUB_CLIENT_ID');  
define('github_oauth_client_secret', 'YOUR_GITHUB_CLIENT_SECRET');  
define('github_oauth_redirect_uri', 'YOUR_GITHUB_REDIRECT_URI');
```

Twitter

```
define('twitter_oauth_client_id', 'YOUR_TWITTER_CLIENT_ID');  
define('twitter_oauth_client_secret', 'YOUR_TWITTER_CLIENT_SECRET');  
define('twitter_oauth_redirect_uri', 'YOUR_TWITTER_REDIRECT_URI');
```

Microsoft

```
define('microsoft_oauth_client_id', 'YOUR_MICROSOFT_CLIENT_ID');  
define('microsoft_oauth_client_secret', 'YOUR_MICROSOFT_CLIENT_SECRET');  
define('microsoft_oauth_redirect_uri', 'YOUR_MICROSOFT_REDIRECT_URI');
```

STEAM (Open ID)

```
define('steamauth_openid_key', 'YOUR_STEAM_OPENID_KEY');  
define('steamauth_openid_redirect_uri', 'YOUR_STEAM_OPENID_KEY');
```

Remember to obtain the OAuth or OpenID client credentials for each SSO provider by creating an application or project on their respective developer platforms.

Step 2: Implement the authentication flow in login.php

In your login.php file, add the following HTML code below the submit button to display the authentication options:

```
//Discord  
<a href="/auth/discord-oauth.php">Login with Discord</a>  
  
//Facebook  
<a href="/auth/facebook-oauth.php">Login with Facebook</a>  
  
//Google  
<a href="/auth/google-oauth.php">Login with Google</a>  
  
//Github  
<a href="/auth/github-oauth.php">Login with Github</a>  
  
//Twitter  
<a href="/auth/twitter-oauth.php">Login with Twitter</a>  
  
//Microsoft  
<a href="/auth/microsoft-oauth.php">Login with Microsoft</a>  
  
//Steam  
<a href="/auth/steamauth.php">Login with Steam</a>
```

This code will create links for each SSO provider, allowing users to choose their preferred method of authentication.

Make sure to replace `"/auth/provider-oauth.php"` in each link with the appropriate endpoint for each SSO provider. For example, if the Discord authentication endpoint is `"/auth/discord-oauth.php"`, the link should be `<a href="/auth/discord-oauth.php">Login with Discord</a>`.

Step 3: Implement the authentication logic in your application

Remember to handle any necessary error cases, such as invalid credentials or revoked access.

That's it! You have now set up authentication in a login page using the `Config.php` file and implemented the necessary code in your `login.php` file to display the authentication options.

IF you have OpenID I have not really coded a OpenID Application for anything but **steam** before so your best bet is to ask a Developer that Does Open ID for any other platform because I have never done it for any other provider.