

# Walkthrough: Collegare la pagina WordPress a MiData tramite OAuth

## Cos'è OAuth?

Hitobito è un provider OAuth 2.0, il che significa che un'applicazione esterna può autenticare gli utenti tramite hitobito (di solito sotto forma di una funzione di "accesso tramite hitobito", simile a quello che succede con Google e Facebook, ecc.). In questa guida imparerai come utilizzare il login MiData per il tuo sito Wordpress.

## Utilizzare il sistema di test

Ogni volta che lavori su un'interfaccia per MiData, dovresti prima provare il tuo progetto sul sistema di test MiData. In questo modo noterai ad esempio se la tua applicazione causa un errore sul sistema o se devi utilizzare un plug-in completamente diverso.

Trovi più informazioni sul sistema test sotto [F.A.Q.](#)

## Configurazione di OAuth App in MiData

Creare una nuova applicazione OAuth in MiData: <https://pbs.puzzle.ch/de/oauth/applications>

<p>Name* <input type="text" value="Pfadi Laupen Test Webseite"/></p> <p>Redirect URIs <input type="text" value="https://pfadilaupen.ch/wp-admin/admin-ajax.php?action=openid-connect-authorize"/></p> <p><small>Ein Eintrag pro Zeile. Für lokale Tests urn:iETF:wg:oauth:2.0:oob verwenden.</small></p> <p>Scopes <input checked="" type="checkbox"/> Lesen deiner E-Mail Adresse (email) <input checked="" type="checkbox"/> Lesen deiner E-Mail Adresse und Name (name) <input type="checkbox"/> Lesen deiner Stammdaten und Rollen (with_roles) <input checked="" type="checkbox"/> Lesen deines OIDC Identity Tokens (openid) <input type="checkbox"/> Lesen aller Personen, Gruppen, Events und Rechnungen auf die du Zugriff hast, via die JSON-Schnittstellen (api)</p> <p>Logo <input type="button" value="Choose File"/> No file chosen</p> <p>Hosts mit API-Zugriff <input type="button" value="Eintrag hinzufügen"/> <small>Falls der Scope "api" aktiviert ist, dürfen Webseiten auf diesen Hosts die Daten aus der JSON-Schnittstelle vom Browser aus abrufen (CORS).</small></p>	<p>A tal fine devono venir fornite le seguenti informazioni</p> <ul style="list-style-type: none"><li>- Nome dell'applicazione</li><li>- Redirect URIs</li><li>- Scopes</li></ul>
---	---

## Istallazione e configurazione del plugin Wordpress

Non appena l'applicazione OAuth è stata creata, Wordpress può essere connesso come client OAuth. Ciò richiede l'installazione e la configurazione di un plug-in aggiuntivo.

	<a href="https://wordpress.org/plugins/daggerhart-openid-connect-generic/">https://wordpress.org/plugins/daggerhart-openid-connect-generic/</a>
	Installare e attivare il plug-in "OpenID Connect Generic Client" tramite Wordpress Plugin Manager
<p><b>Client Settings</b></p> <p>Enter your OpenID Connect identity provider settings.</p> <p><b>Login Type</b> OpenID Connect button on login form Select how the client (login form) should provide login options.</p> <p><b>Client ID</b> W6pi3xDI-QjDOWpn293TJKCS5phKvsl1m5WLSUPTE The ID this client will be recognized as when connecting to the Identity provider server. Example: my-wordpress-client-id</p> <p><b>Client Secret Key</b> 23CWPOFX3aJsDuAhhNrxWtpFt5_uthedHwdK9XL4dFo Arbitrary secret key the server expects from this client. Can be anything, but should be very unique.</p> <p><b>OpenID Scope</b> openid email name Space separated list of scopes this client should access. Example: email profile openid offline_access</p> <p><b>Login Endpoint URL</b> https://pbs.puzzle.ch/oauth/authorize Identify provider authorization endpoint. Example: https://example.com/oauth2/authorize</p> <p><b>Userinfo Endpoint URL</b> https://pbs.puzzle.ch/oauth/userinfo Identify provider User information endpoint. Example: https://example.com/oauth2/userinfo</p> <p><b>Token Validation Endpoint URL</b> https://pbs.puzzle.ch/oauth/token Identify provider token endpoint. Example: https://example.com/oauth2/token</p> <p><b>End Session Endpoint URL</b> https://pbs.puzzle.ch/oauth/logout Identify provider logout endpoint. Example: https://example.com/oauth2/logout</p> <p><b>ACR values</b> Use a specific defined authentication contract from the IDP - optional.</p> <p><b>Identity Key</b> email Where in the user claim array to find the user's identification data. Possible standard values: preferred_username, name, or sub. If you're having trouble, use "sub". Example: preferred_username</p> <p><b>Disable SSL Verify</b> <input type="checkbox"/> Do not require SSL verification during authorization. The OAuth extension uses curl to make the request. By default CURL will generally verify the SSL certificate to see if its valid an issued by an accepted CA. This setting disabled that verification. Not recommended for production sites.</p> <p><b>HTTP Request Timeout</b> 5 Set the timeout for requests made to the IDP. Default value is 5. Example: 30</p>	Effettuare la seguente configurazione del plugin

- Login Type: Bottone OpenID Connect sul formulario di login
- Client ID: <Prendere il valore da MiData>
- Client Secret Key: <Prendere il valore da MiData>
- ClientID Scope: openid email name
- Login Endpoint URL:
  - o Test: <https://pbs.puzzle.ch/oauth/authorize>
  - o Produzione: <https://db.scout.ch/oauth/authorize>
- Userinfo Endpoint URL:
  - o Test: <https://pbs.puzzle.ch/oauth/userinfo>
  - o Produzione: <https://db.scout.ch/oauth/userinfo>
- Token Validation Endpoint URL:
  - o Test: <https://pbs.puzzle.ch/oauth/token>
  - o Produzione: <https://db.scout.ch/oauth/token>
- End Session Endpoint URL:

**Nickname Key**   
 Where in the user claim array to find the user's nickname. Possible standard values: preferred\_username, name, or sub.  
 Example: preferred\_username

**Email Formatting**   
 String from which the user's email address is built. Specify "{email}" as long as the user claim contains an email claim.  
 Example: {email}

**Display Name Formatting**   
 String from which the user's display name is built.  
 Example: {given\_name} {family\_name}

**Identify with User Name**   
 If checked, the user's identity will be determined by the user name instead of the email address.

**State time limit**   
 State valid time in seconds. Defaults to 180

**Enable Refresh Token**   
 If checked, support refresh tokens used to obtain access tokens from supported IDPs.

**WordPress User Settings**  
 Modify the interaction between OpenID Connect and WordPress users.

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

**Create user if does not exist**   
 If the user identity is not linked to an existing WordPress user, it is created. If this setting is not enabled, and if the user authenticates with an account which is not linked to an existing WordPress user, then the authentication will fail.

**Redirect Back to Origin Page**   
 After a successful OpenID Connect authentication, this will redirect the user back to the page on which they clicked the OpenID Connect login button. This will cause the login process to proceed in a traditional WordPress fashion. For example, users logging in through the default wp-login.php page would end up on the WordPress Dashboard and users logging in through the WooCommerce "My Account" page would end up on their account page.

**Redirect to the login screen when session is expired**   
 When enabled, this will automatically redirect the user back to the WordPress login page if their access token has expired.

**Authorization Settings**  
 Control the authorization mechanics of the site.

**Enforce Privacy**   
 Require users be logged in to see the site.

**Alternate Redirect URI**   
 Provide an alternative redirect route. Useful if your server is causing issues with the default admin-ajax method. You must flush rewrite rules after changing this setting. This can be done by saving the Permalinks settings page.

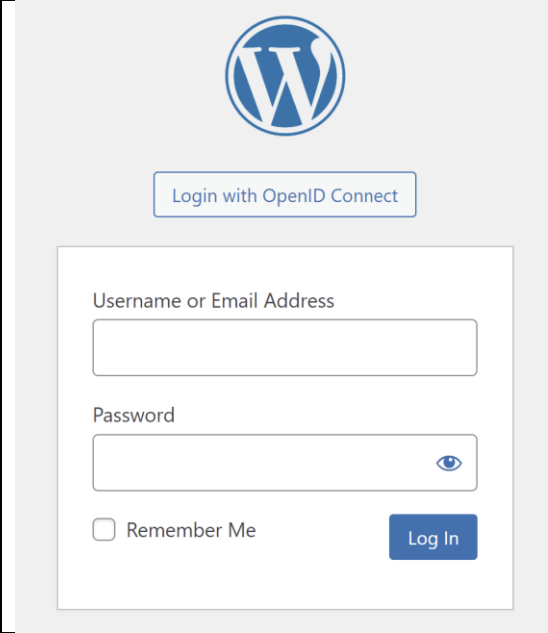

**Log Settings**  
 Log information about login attempts through OpenID Connect Generic.

**Enable Logging**   
 Very simple log messages for debugging purposes.

**Log Limit**   
 Number of items to keep in the log. These logs are stored as an option in the database, so space is limited.

- Test: <https://pbs.puzzle.ch/oauth/logout>
  - Produzione: <https://db.scout.ch/oauth/logout>
- Identity Key: email
  - HTTP Request Timeout: 5
  - Email Formatting: {email}
  - Display Name Formatting: {nickname}
  - Enable Refresh Token: true
  - Link Existing Users: true
  - Create user if does not exist: false, in questo modo gli utilizzatori che vengono autenticati tramite OAuth devono venire previamente creati da Wordpress.
  - Redirect back to Origin Page: false
  - Redirect to the login screen when session expired: true
  - Enforce Privacy: false
  - Alternate Redirect URI: false
  - Enable Logging: false

## Login Experience

	<p>Richiamare la pagina di accesso di Wordpress (ad es. <a href="https://wordpressite.ch/wp-admin/">https://wordpressite.ch/wp-admin/</a>).</p> <p>Fare clic su "Login with OpenID Connect".</p>
	<p>Accedi utilizzando un account MiData</p>

## Passare al sistema produttivo

Non appena sei sicuro che la tua applicazione sta facendo quello che dovrebbe fare, puoi richiedere l'accesso al MiData «giusto».

### Richiedere OAuth API Key

Tramite il seguente formulario puoi richiedere sul MiData produttivo (<https://db.scout.ch/>) un'applicazione OAuth Applikation:

<https://forms.office.com/Pages/ResponsePage.aspx?id=iq6Fcs2Xq0m9ordFTZ0Fa8gnQG-i3p9KkbcKGL9nFhtUMEpMQkYwMzQxNUVEWEIxRTNWTdHPMDVEMS4u>