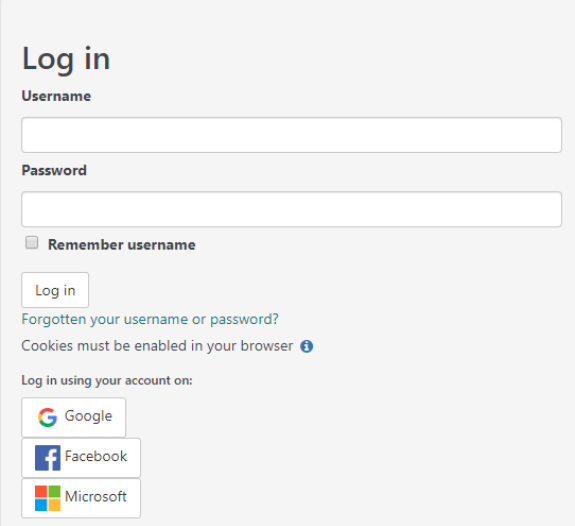
OAuth2

The OAuth 2 plugin allows users to login using an existing account for another service, for example using an existing Microsoft, Google, or Facebook account.

You will need to enable and configure OAuth 2 in two places on Totara (both accessed from the **Site administration menu**):

1. Enable OAuth 2 authentication method in *Plugins* > *Authentication* > *Manage authentication*
2. Configure OAuth 2 services under *Server* > *OAuth 2*

Additionally services will need to be set-up and configured on that services site (for example in the Google developer console).



The screenshot above shows a Totara Learn login box with a number of OAuth2 service enabled including Google, Facebook, and Microsoft accounts.

Enable authentication method

Before you can use OAuth 2 as an authentication method it will need to be enabled (as instructed on the Authentication page).

1. Go to *Plugins* > *Authentication* > *Manage Authentication* from the **Site administration menu**.
2. Click the show icon ( ) alongside **OAuth 2** to enable it (the eye will be open once the authentication method is enabled).

Currently OAuth 2 identification is based on the user's email address. This means that when two users in the system have the same email they can be incorrectly logged in. To avoid issues with OAuth 2 logins it is recommended that you ensure the **Allow accounts with same email** settings is disabled under the list of **Common settings** on the *Plugins* > *Authentication* > *Manage authentication* page.

Lock fields

By clicking on **Settings** alongside the **OAuth 2** authentication method or by going to *Plugins* > *Authentication* > *OAuth 2* you can configure whether certain user data fields should be locked.

This is useful for sites where the user data is maintained by the administrators, either by manually editing user records or uploading using the **Upload users** facility. If you are locking fields that are required by Totara, make sure that you provide that data when creating user accounts or the accounts will be unusable. Consider setting the lock mode to **Unlocked if empty** to avoid this problem.

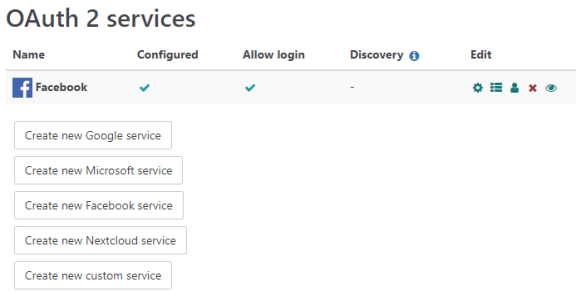
Each user field can be set to either **Unlocked**, **Unlocked if empty**, or **Locked**. Remember to click **Save changes** when you are done.

Create new OAuth 2 service

Once you have enabled the OAuth 2 authentication method you can now set up services to use as a login method. First of all you will need to go to that service and set up authentication on that end. This usually works by going to that services developer console, creating a new app, and then copying the ID and secret. Instructions for some specific services can be found below.

Once you have set up the services in Totara Learn do the following:

1. Go to *Server > OAuth 2 services* from the **Administration menu**.
2. Click **Create a new service** - choosing the right one for the service you are setting up.
3. Configure the settings.
4. Click **Save changes**.



Settings

Setting	Description	Notes
Name	The name of the issuer service (e.g. Google, Facebook, etc.) this may be displayed on the login page.	-
Client ID	The unique ID provided by the issuer.	-
Client secret	A unique password or secret generated by the issuer.	-
Authenticate token requests via HTTP headers	Utilise the HTTP basic authentication scheme when sending client ID and password with a refresh token request. Recommended by the OAuth 2 standard, but may not be available with some issuers.	-
Scopes included in a login request	Some systems require additional scopes for a login request in order to read the user's basic profile. The standard scopes for an OpenID Connect compliant system are "openid profile email".	-
Scopes included in a login request for offline access	Each OAuth system defines a different way to request offline access. E.g. Microsoft requires an additional scope "offline_access".	-
Additional parameters included in a login request	Some systems require additional parameters for a login request in order to read the user's basic profile.	
Additional parameters included in a login request for offline access	Each OAuth system defines a different way to request offline access. E.g. Google requires the additional parameters: "access_type=offline&prompt=consent". These parameters should be in URL query parameter format.	-
Service base URL	Base URL used to access the service.	-
Login domains	If set, this setting is a comma separated list of domains that logins will be restricted to when using this provider.	-
Logo URL	This is usually the logo used by the issuer, and it may be displayed on the login page.	-
Show on login page	If the OAuth 2 authentication plugin is enabled, this login issuer will be listed on the login page to allow users to log in with accounts from this issuer.	-
Require email verification	Require that all users verify their email address before they can log in with OAuth. This applies to newly created accounts as part of the login process, or when an existing Totara account is connected to an OAuth login via matching email addresses.	-

Edit

After a service has been set up you can edit it via the **Edit** column from *Server* > *OAuth 2 services* via the **Administration menu**.

- Edit () allows you to adjust the settings
- Configure endpoints () allows you to edit, delete, or add endpoint URLs
- Configure user field mappings () allows you to edit, delete, or create mappings between user data fields on the issue site and your Totara site to ensure the correct information is brought across
- Delete () allows you to remove that service
- Disable () or Enable () - disabling a service means that it can no longer be used but all of the configuration information is kept in the system for future use

The open eye icon () means a service is enabled, therefore clicking it disables the service. Whereas a closed eye icon () means the service is disabled, therefore clicking it enables the service.

Login via Microsoft account

If you wish to enable Microsoft account login then you will need to enable the OAuth 2 plugin on your Totara site and go to the Microsoft developer console to configure authentication.

1. Go to the Microsoft developer console.
2. Click **Add an app** and give it a name e.g. 'Totara Learn'.
3. Click **Create application**.
4. Under **Platform** click **Add Platform** and select **Web**.
5. Untick the **Allow Implicit Flow** setting.
6. Add your site's URL appended with **/admin/oauth2callback.php** to the **Redirect URLs** section e.g. https://totaralearn.com/admin/oauth2callback.php
7. Ensure the **User.Read** permission is available under **Microsoft Graph Permissions** and if it is not then add it.
8. Configure the options in the **Profile** section as these will appear on the consent screen.
9. Click **Save**.
10. Under **Application secrets** click **Generate New Password** and make sure to carefully copy the password shown as it will only appear once.
11. Take a note of the **Application ID**.
12. In Totara Learn go to *Server* > *OAuth 2 services* from the **Administration menu**.
13. Click **Create a new Microsoft service**.
14. Enter the password generate in the Microsoft developer console as the **Secret** and the application ID as the **Client ID**.
15. Click **Save changes**.

You can see more instructions from Microsoft on their website.

Login via Google account

If you wish to enable Google account login then you will need to enable the OAuth 2 plugin on your Totara site and go to the Google developer console to configure authentication.

1. Go to the Google developer console.
2. Create a new project using either the **Select a project** dropdown at the top or the **Create** button.
3. Give the project a name e.g. 'Totara Learn login'.
4. Click **Create**.
5. Go to **Credentials** from the left hand menu.
6. Select the **OAuth consent screen** section and complete the settings.
7. Click **Save**.
8. Click on the **Credentials** tab and then choose **OAuth client ID** from the **Create credentials** dropdown.
9. Choose the **Web application** option and set the **Authorized redirect URIs** as your site's URL appended with **/admin/oauth2callback.php** e.g. https://totaralearn.com/admin/oauth2callback.php
10. Click **Create**.
11. Take a note of the client ID and secret generated.
12. In Totara Learn go to *Server* > *OAuth 2 services* from the **Administration menu**.
13. Click **Create a new Google service**.
14. Enter the **Secret** and **Client ID** given in the Google developer console.
15. Click **Save changes**.

You can see more about Google and OAuth 2 on their website.

Login via Facebook

If you wish to enable Facebook login then you will need to enable the OAuth 2 plugin on your Totara site and also go to the Facebook developer portal to configure authentication via their login system. The basic process is:

1. Create a Facebook app via Facebook for developers. This will need to have a **Display name** and **Contact email**.
2. In the **Product** select **Facebook Login**.
3. Choose the **Web** option and configure the settings.
4. Make a note of the **App ID** and **App Secret**.
5. In Totara Learn go to *Server* > *OAuth 2 services* from the **Administration menu**.
6. Click **Create a new Facebook service**.
7. Enter the **Secret** (the **App Secret**) and **Client ID** (the **App ID**) given in Facebook.
8. Click **Save changes**.

You can find details on how to configure Facebook login in their help documentation.