

Gmail API Provider Setup

Last Modified on 03/19/2020 6:36 pm EDT

On this page

To authenticate a Gmail connector instance you must create a project with a web application in the Google API console. After you create the project and app, Google provides a **Client ID** and **Client secret** which you will use to authenticate. You'll also need the **Authorized redirect URI** that you configure while creating the app.

When you set up your app, you must also enable the Gmail API.

If you've already set up an app and just need to know how to find your **Client ID** and **Client secret**, see [Locate Credentials for Authentication](#). If you need to create a project and register an app, see [Create an Application](#).

See the latest setup instructions in the [Google API documentation](#).

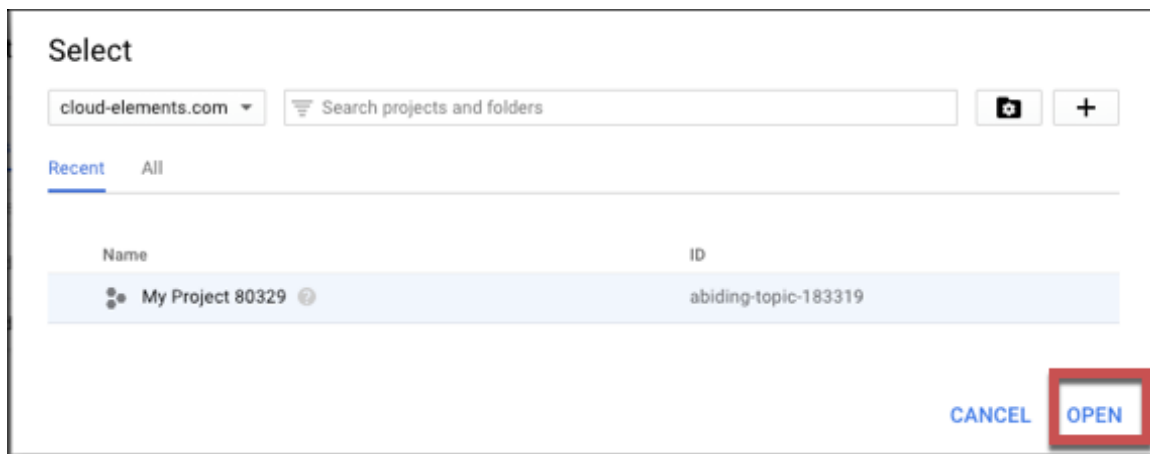
Locate Credentials for Authentication

If you already created a project and application, follow the steps below to locate the **Client ID**, **Client secret**, and **Authorized redirect URI**. If you have not created an app, see [Create an Application](#).

Note: Your app must have the Gmail API enabled. If not, see [Enable APIs for more information](#)

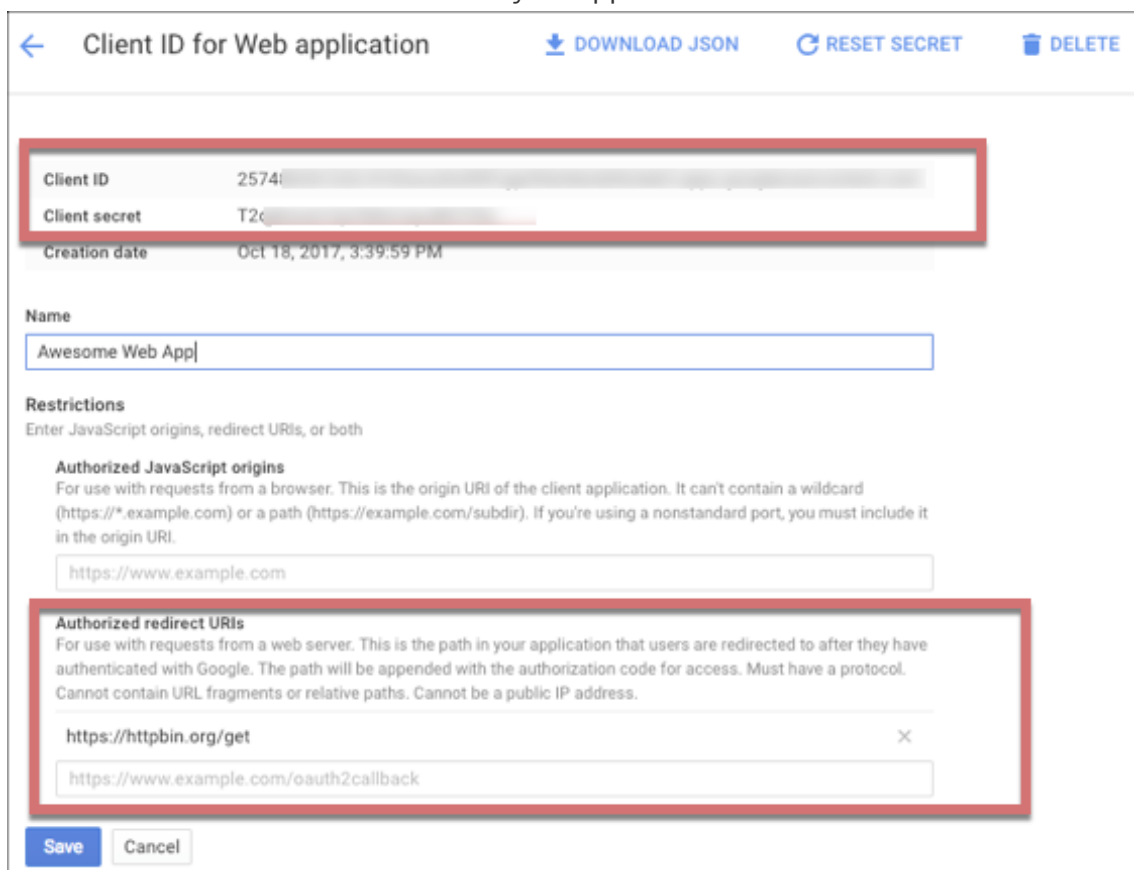
To find your OAuth 2.0 credentials:

1. Log in to your account at [Google](#).
2. Click **Select a project**, choose your project from the list, and then click **Open**.



Google displays your apps and associated **Client ID**.

3. Click the pencil icon to see the **Client ID**, **Client secret**, and **Authorized redirect URI**.
4. Record the **Client ID** and **Client secret**.
5. Record the **Authorized redirect URI** for your app.



Create an Application

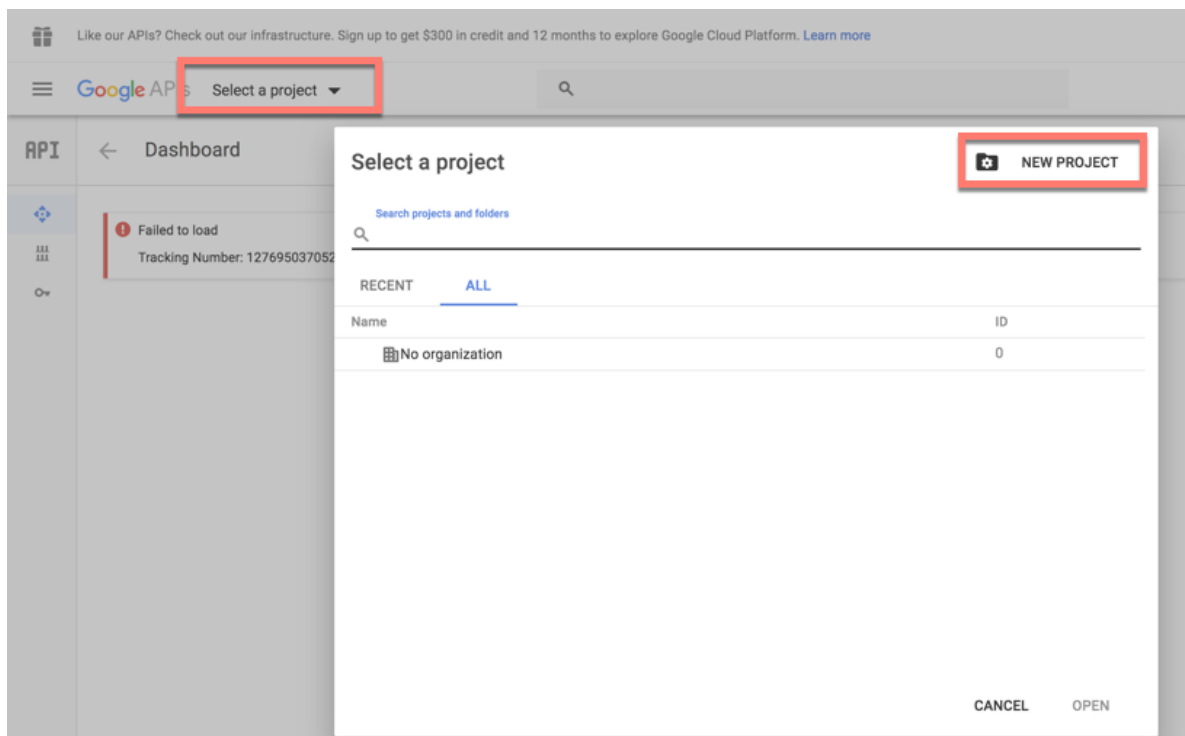
If you have not already created a project and application, you need one to authenticate with Google. Creating an application is a multi-step process:

1. [Create a project](#)
2. [Enable APIs](#)
3. [Create a web application](#)

Create a Project

To create a project:

1. Log in to your account at [Google](#).
2. Click **Select a project**, and then click **New Project**.



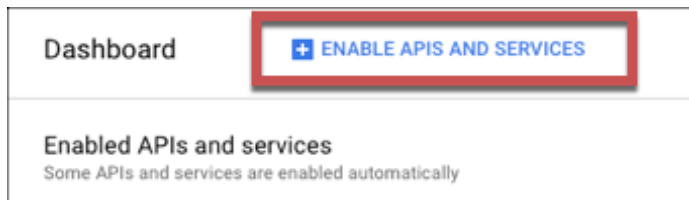
3. Accept the terms of service, and then complete the required information.
4. Click **Create**.

Enable APIs

To make all of the requests available in the Gmail connector, you must enable the GMail API.

To enable APIs:

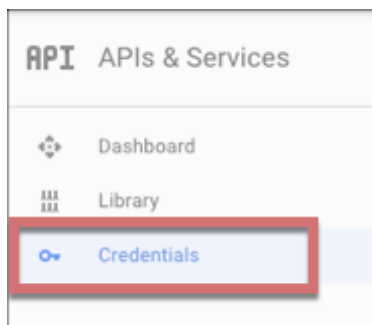
1. Select the project that you just created
2. Click **Enable APIs and Services**.



3. Search for and enable the Gmail API.

Create a Web Application

1. Click **Credentials** on the left menu.



2. Click the **OAuth consent screen** tab.
3. Enter a Product Name and add any optional information, and then click **Save**.

Credentials

Credentials **OAuth consent screen** Domain verification

Email address [?]

Product name shown to users [?]


Awesome Docs App

Homepage URL (Optional)

https://cloud-elements.com

Product logo URL (Optional) [?]

http://www.example.com/logo.png

 This is how your logo will look to end users
Max size: 120x120 px

Privacy policy URL

Optional until you deploy your app

https:// or http://

Terms of service URL (Optional)

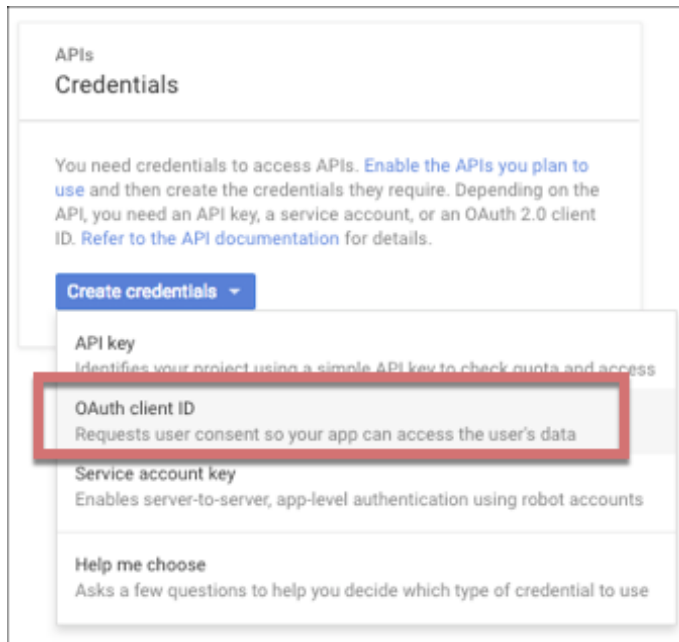
https:// or http://

Save Cancel

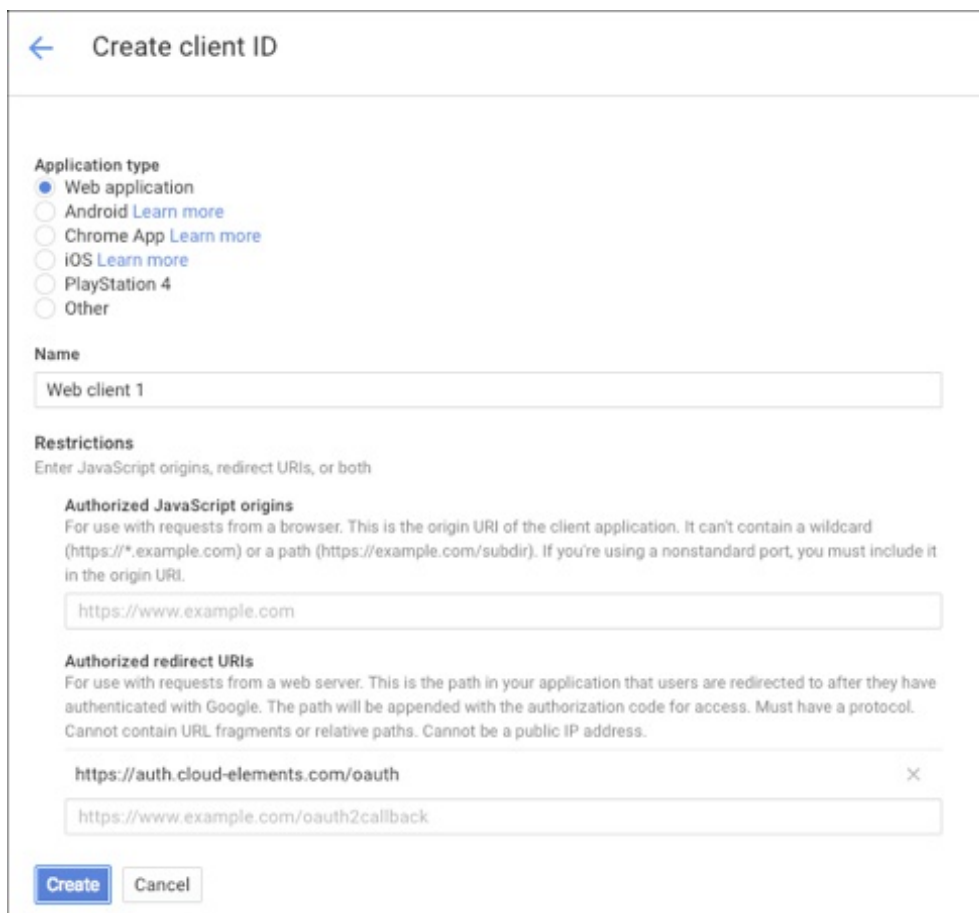
The consent screen will be shown to users whenever you request access to their private data using your client ID. It will be shown for all applications registered in this project.

You must provide an email address and product name for OAuth to work.

4. Click **Create Credentials**, and then select **OAuth Client ID**.



5. Select **Web application** as the Application type.
6. Enter a **Name** and the **Authorized redirect URI** for your app. Record this as the OAuth Callback URL that you will need to authenticate.



7. Click **Create**.

8. Record the **Client ID** and **Client secret** to use when you authenticate.

← Client ID for Web application [DOWNLOAD JSON](#) [RESET SECRET](#) [DELETE](#)

Client ID	2574: [REDACTED]
Client secret	T2 [REDACTED]
Creation date	Oct 18, 2017, 3:39:59 PM

Name

Restrictions
Enter JavaScript origins, redirect URIs, or both

Authorized JavaScript origins
For use with requests from a browser. This is the origin URI of the client application. It can't contain a wildcard (https://*.example.com) or a path (https://example.com/subdir). If you're using a nonstandard port, you must include it in the origin URI.

Authorized redirect URIs
For use with requests from a web server. This is the path in your application that users are redirected to after they have authenticated with Google. The path will be appended with the authorization code for access. Must have a protocol. Cannot contain URL fragments or relative paths. Cannot be a public IP address.

https://httpbin.org/get ×

[Save](#) [Cancel](#)