

Setting Up Google as Your Mail Service Provider

WHMCS can send emails without any additional configuration using **PHP mail()**. While this works in most cases, other mail providers may give you a better experience and access to additional features.

In addition to the existing SMTP support, WHMCS 8.0 added support for Mailgun, SendGrid, SparkPost, and Google® OAuth with SMTP. WHMCS 8.6 and later also include Microsoft® services. For more information, see:

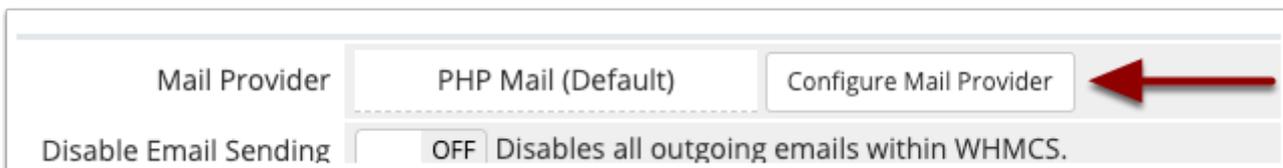
- [Configuring WHMCS to use SMTP for WHMCS 8+](#)
- [Configuring WHMCS to use Other Mail Providers](#)
- [Setting Up Microsoft As Your Mail Service Provider](#)
- [Mail Provider Integrations](#)

Set up Google in WHMCS

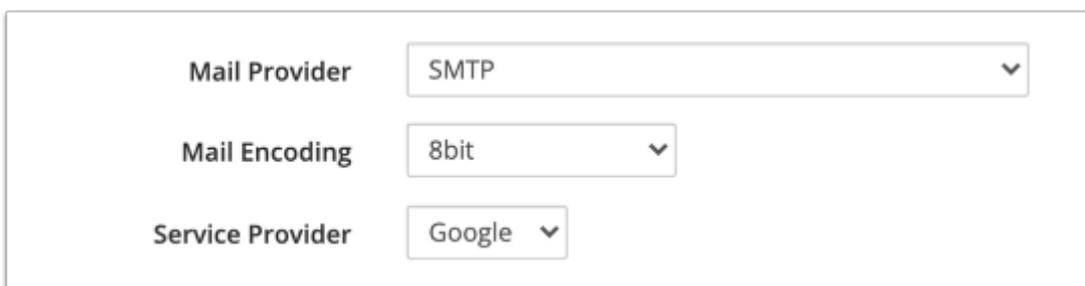
To configure **Google**, you will need to create an app in the Google Cloud console and configure the mail provider in WHMCS.

To configure the mail provider:

1. In the Admin Area, go to the **Mail** tab at **Configuration > System Settings > General Settings**.
2. Click **Configure Mail Provider**.

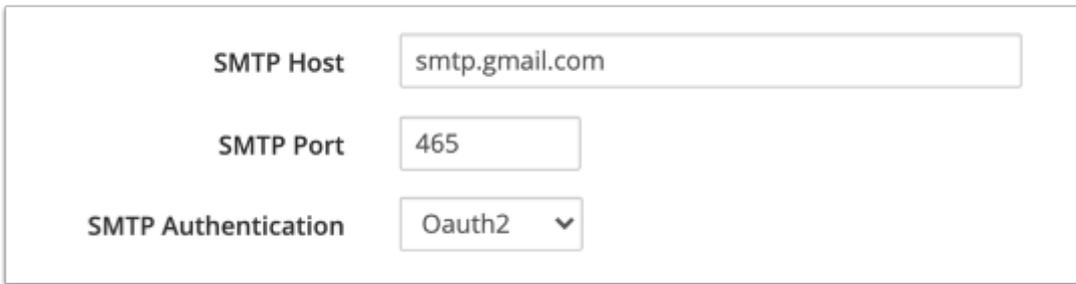


3. Select *SMTP* as your **Mail Provider** and choose a **Mail Encoding**.
4. Select *Google* as your **Service Provider**.



5. Enter `smtp.gmail.com` for the **SMTP Host** and `465` for the **SMTP Port**.

6. Select *Oauth2* for **SMTP Authentication**.



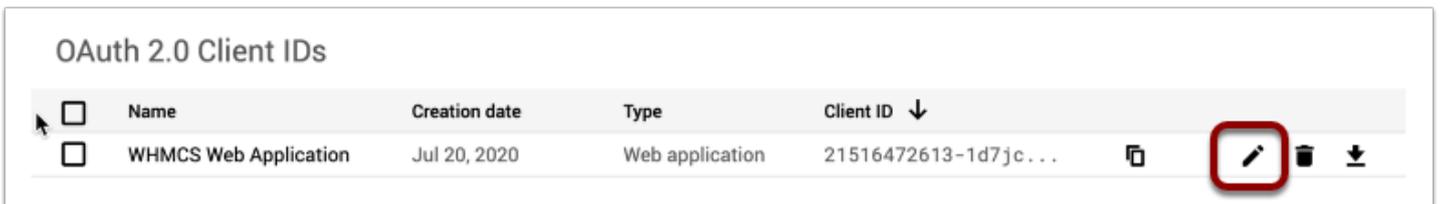
A screenshot of a configuration form with three rows. The first row has 'SMTP Host' on the left and a text input field containing 'smtp.gmail.com'. The second row has 'SMTP Port' on the left and a text input field containing '465'. The third row has 'SMTP Authentication' on the left and a dropdown menu with 'Oauth2' selected and a downward arrow.

7. For **SMTP Username**, enter the Gmail™ address that you will be using in your application.

 For steps to generate the Client ID and Client Secret in Google, see the [Create your Google Application](#) section below.

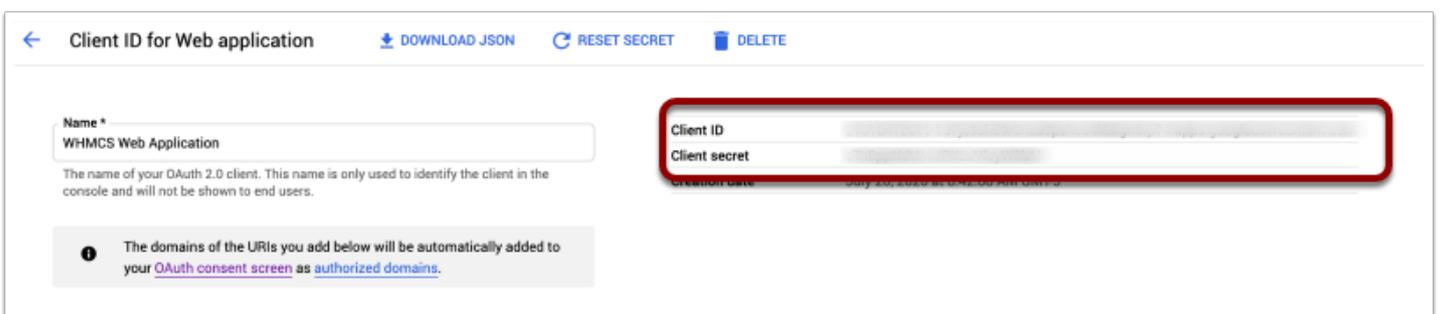
8. Copy-and-paste the **Client ID** and **Client Secret** from the Google Cloud Console into the appropriate boxes in the confirmation message.

9. You can also edit them by going to **Credentials** and clicking the edit icon for the appropriate **OAuth 2.0 Client IDs** row:



A screenshot of the 'OAuth 2.0 Client IDs' section in the Google Cloud Console. It shows a table with the following columns: Name, Creation date, Type, and Client ID. There is a checkbox in the first column. The table contains one row with the following data: Name: WHMCS Web Application, Creation date: Jul 20, 2020, Type: Web application, Client ID: 21516472613-1d7jc... To the right of the table are icons for copy, edit (circled in red), delete, and download.

<input type="checkbox"/>	Name	Creation date	Type	Client ID ↓				
<input type="checkbox"/>	WHMCS Web Application	Jul 20, 2020	Web application	21516472613-1d7jc...				



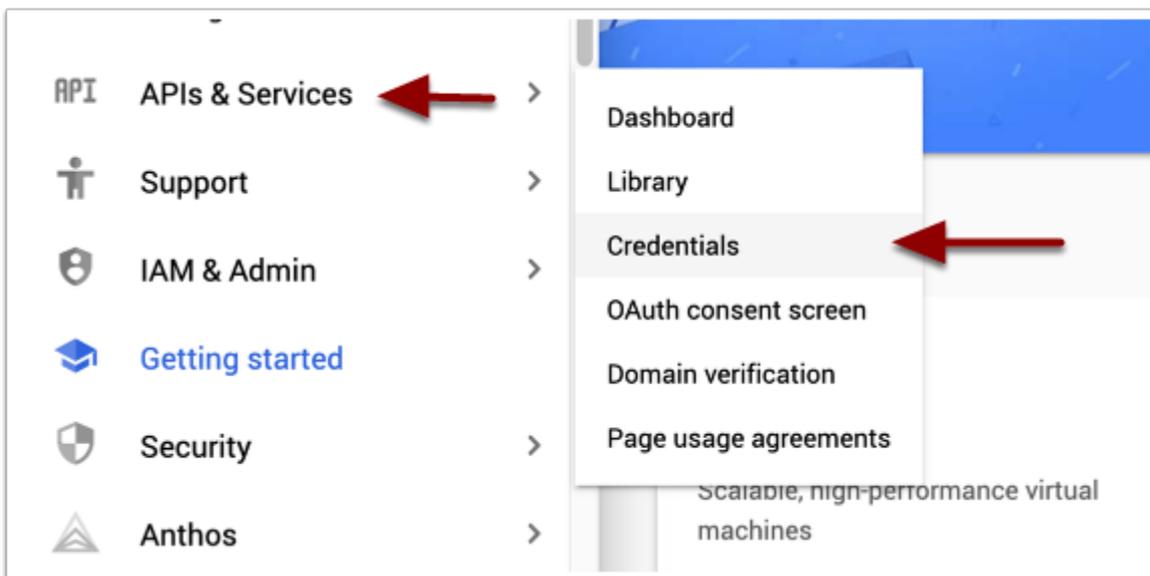
A screenshot of the 'Client ID for Web application' form in the Google Cloud Console. At the top, there are navigation links: a back arrow, 'Client ID for Web application', 'DOWNLOAD JSON', 'RESET SECRET', and 'DELETE'. The form has a 'Name *' field containing 'WHMCS Web Application' and a text area below it with the text: '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.' To the right, there are fields for 'Client ID' and 'Client secret', both of which are circled in red. Below the form is a note: 'The domains of the URIs you add below will be automatically added to your OAuth consent screen as authorized domains.'

10. Next to **Connection Token**, click **Connect**. (**Connection Token** will be empty.)

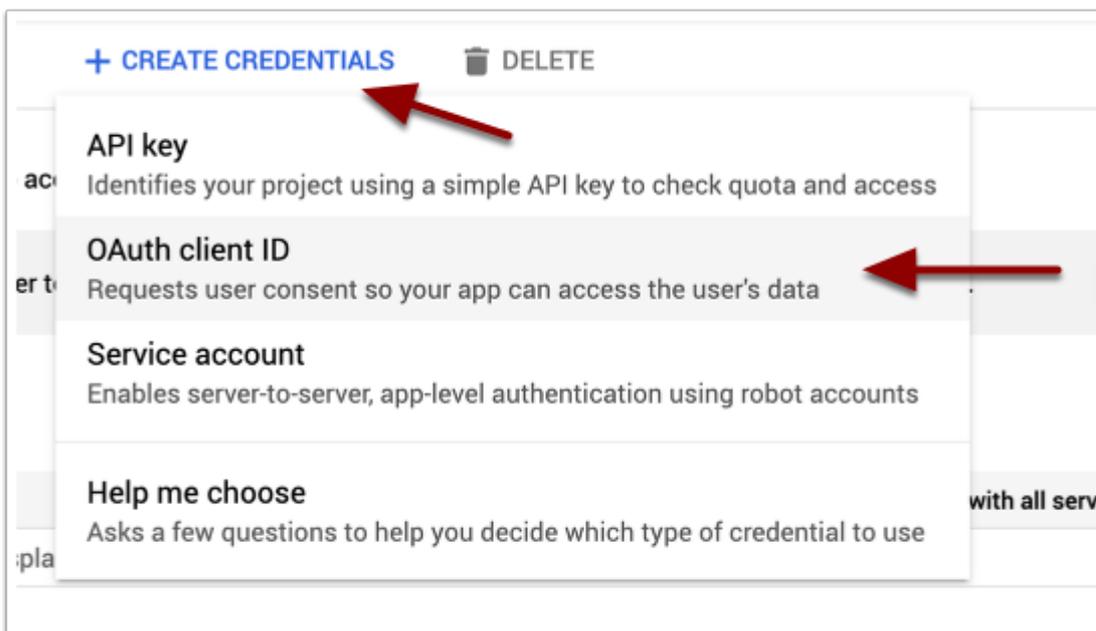
💡 If you have already used Google Cloud's console, you may not need to perform some of these steps, or interfaces may not look like the screenshots below.

First, create the app:

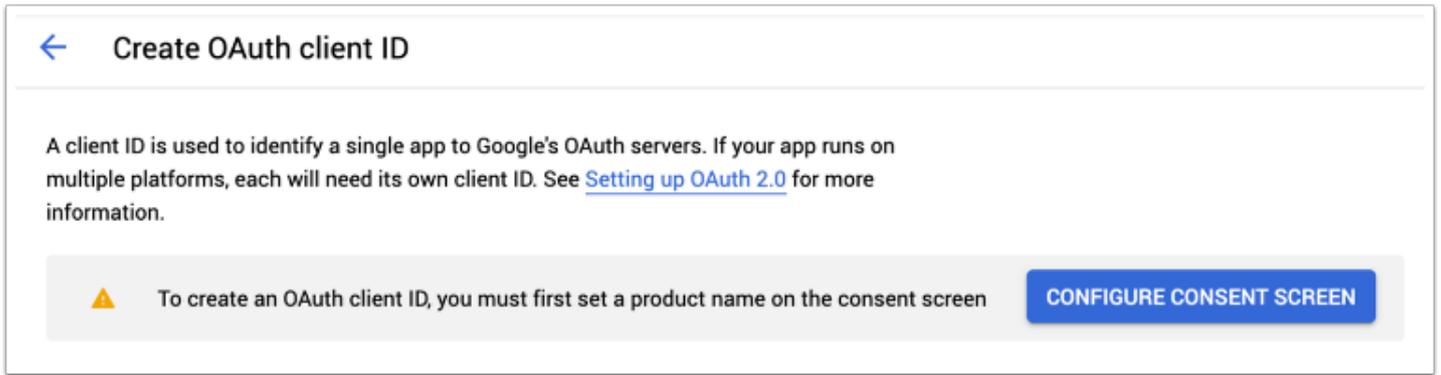
1. Log in to [the Google Cloud console](#). If you haven't before, select your country and agree to Google's *Terms of Service*.
2. Go to **APIs and Services > Credentials**.



3. Click **Create Credentials** and select **OAuth client ID**.

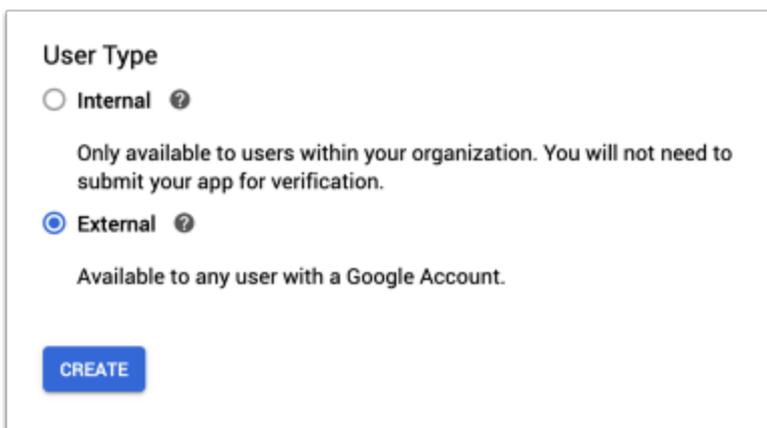


4. Click **Configure Consent Screen**.



5. Select *External* for **User Type**.

Selecting this allows anyone to use the generated client ID after a verification process. However, when creating a client ID that is only for WHMCS, you do not need verification.



6. Click **Create**.

7. Enter a new **App name**.

8. Select a **User support email** address.

9. Click **Add Domain** and enter the domain for your WHMCS installation.

Authorized 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 authorized. [Learn more](#)  about the authorized domain limit.

Authorized domain 1 *

example.com

[+ ADD DOMAIN](#)

Developer contact information

Email addresses *

user@example.com 

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

[SAVE AND CONTINUE](#)

CANCEL

10. Click **Save and Continue**.

11. Click **Add or Remove Scopes** and add the following scopes:

- *userinfo.email*
- *userinfo.profile*
- *openid*

[ADD OR REMOVE SCOPES](#)

Your non-sensitive scopes

API ↑	Scope	User-facing description	
	.../auth /userinfo .email	See your primary Google Account email address	
	.../auth /userinfo .profile	See your personal info, including any personal info you've made publicly available	
	openid	Associate you with your personal info on Google	

12. Click **Save and Continue**.

13. For **Test users**, click **Save and Continue** without making any changes.

After you create the app, you can create the client ID:

1. In the left sidebar, click **Credentials**.
2. Click **Create Credentials** and select **OAuth client ID** again.
3. For the **Application Type**, select **Web Application**.

← **Create OAuth client ID**

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

4. Enter a name for your application.

5. Under **Authorized redirect URIs**, click **Add URI**.

6. Enter the **Callback URL** that displays in WHMCS.

The screenshot shows the 'Configure Mail Provider' interface. The 'Mail Provider' is set to 'SMTP', 'Mail Encoding' to '8bit', and 'Service Provider' to 'Google'. The 'SMTP Host' is 'smtp.gmail.com' and the 'SMTP Port' is '465'. 'SMTP Authentication' is set to 'OAuth2' and the 'SMTP Username' is 'user@example.com'. The 'Callback URL' field is highlighted with a red box and contains 'http://www.example.com/whmcs/admin/ind'. Below this are fields for 'Client ID', 'Client Secret', and 'Connection Token' (with a 'Connect' button). 'SMTP SSL Type' is set to 'SSL' and there is an unchecked checkbox for 'SMTP Debug'. At the bottom are 'Test Configuration', 'Close', and 'Save' buttons.

The screenshot shows the 'Authorized redirect URIs' page. It has a title 'Authorized redirect URIs' with a help icon and a subtitle 'For use with requests from a web server'. Under the heading 'URIs', there is a text input field containing 'http://www.example.com/whmcs/admin/index.php?rp=/admin/setup/mail/'. Below the list is a '+ ADD URI' button.

7. Click **Create**. A confirmation message will appear, with the **Client ID** and **Client Secret** to use in the steps above.

