

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:

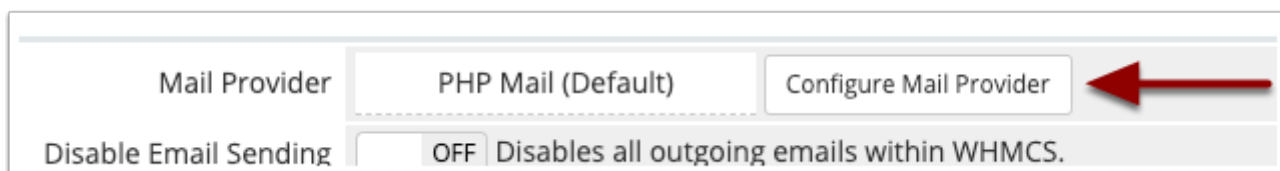
- [Configuring WHMCS to use SMTP for WHMCS 8+](#)
- [Configuring WHMCS to use Other Mail Providers](#)
- [Setting Up Microsoft As Your Mail Service Provider](#)
- [Mail Provider Integrations](#)

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**.

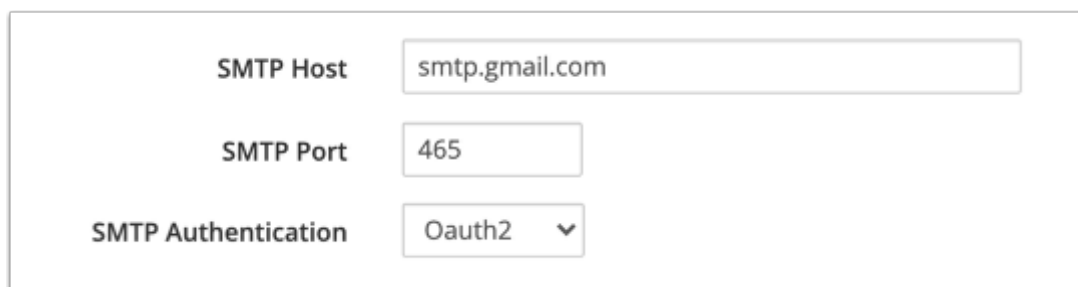


3. Select *SMTP* as your **Mail Provider** and choose a **Mail Encoding**.
4. Select *Google* as your **Service Provider**.

Mail Provider	<input type="text" value="SMTP"/>
Mail Encoding	<input type="text" value="8bit"/>
Service Provider	<input type="text" value="Google"/>

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

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



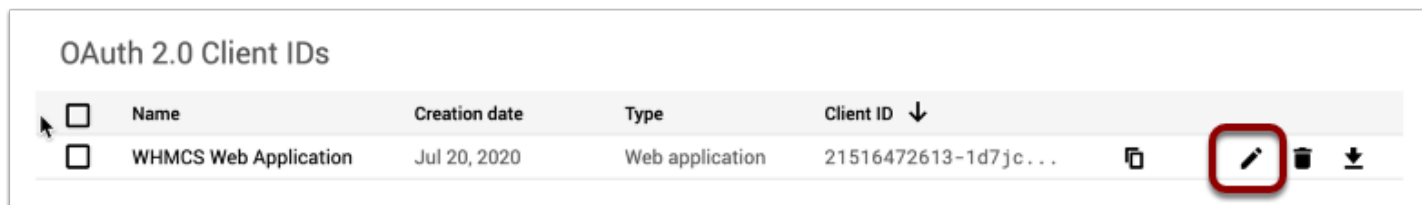
A screenshot of a configuration form for SMTP settings. It contains three fields: 'SMTP Host' with the value 'smtp.gmail.com', 'SMTP Port' with the value '465', and 'SMTP Authentication' with a dropdown menu set to 'Oauth2'.

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.

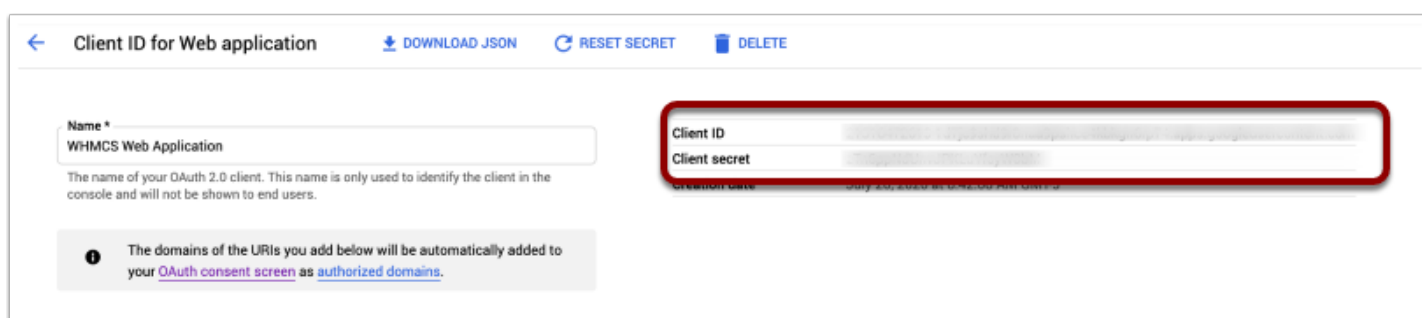
8. Copy-and-paste the **Client ID** and **Client Secret** from the Google Cloud Console into the appropriate boxes in the confirmation message.

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' table in the Google Cloud Console. The table has columns for Name, Creation date, Type, and Client ID. The first row is highlighted and shows 'WHMCS Web Application' with a creation date of 'Jul 20, 2020' and a type of 'Web application'. The Client ID is partially visible as '21516472613-1d7jc...'. To the right of the table, there are icons for copy, edit (highlighted with a red box), 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. The form has a 'Name' field with the value 'WHMCS Web Application' and a 'Client ID' field with a value that is partially visible as '21516472613-1d7jc...'. The 'Client secret' field is also visible. There are buttons for 'DOWNLOAD JSON', 'RESET SECRET', and 'DELETE'. A red box highlights the 'Client ID' and 'Client secret' fields.

Client ID for Web application

DOWNLOAD JSON RESET SECRET DELETE

Name *
WHMCS Web Application

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.

Client ID
21516472613-1d7jc...

Client secret
[REDACTED]

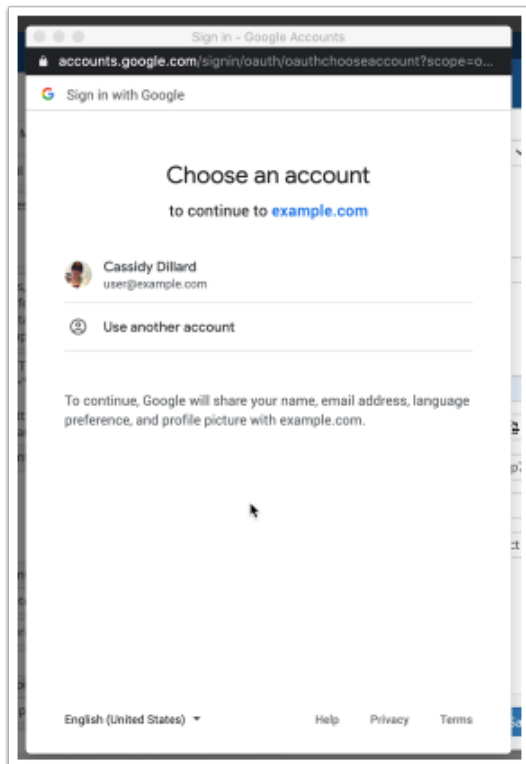
Creation date
July 20, 2020

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.)

Client ID	<input type="text" value="1234567890123456789012345678901234567890"/>
Client Secret	<input type="text" value="abcdefghijklmnopqrstuvwxyz0123456789"/>
Connection Token	<input type="text" value=""/>
	<input type="button" value="Connect"/>

11. Under **Choose an account**, select the account that you used to create the app.



12. Follow the prompts to approve access for your account.

 If you see a *This app isn't verified.* error, click **Advanced** and then click **Go to** at the bottom of the window.

The system will automatically enter a token in **Connection Token**.

13. To ensure that your configuration works, click **Test Configuration**.

14. Click **Save**. The system will test your configuration again when you save.

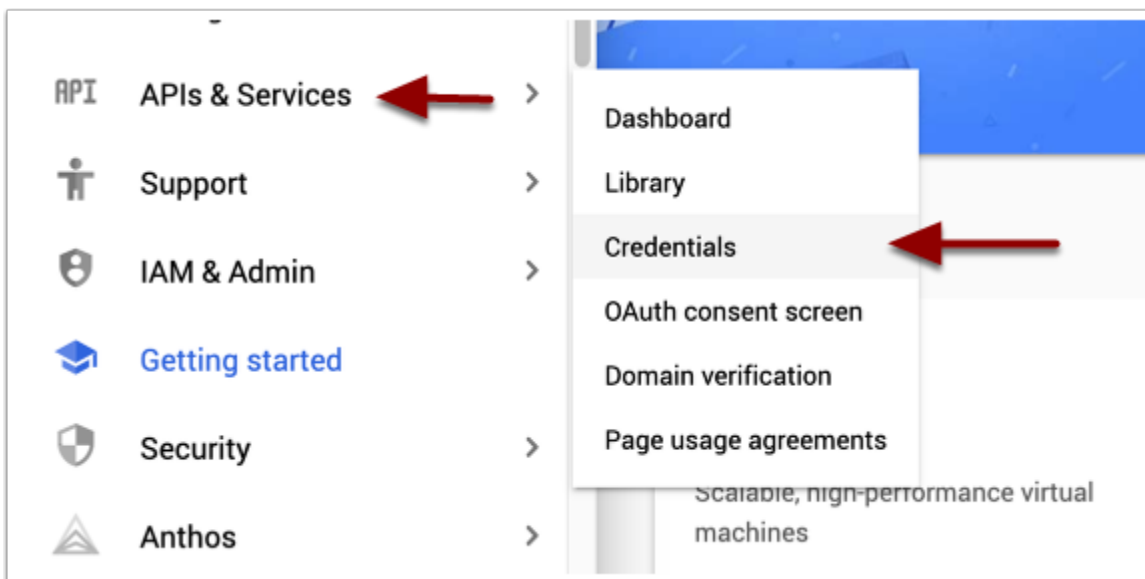
Create your Google application

To use Google as your service provider, you will need to create an app and then create an associated client ID. This will let you connect to Google via WHMCS.

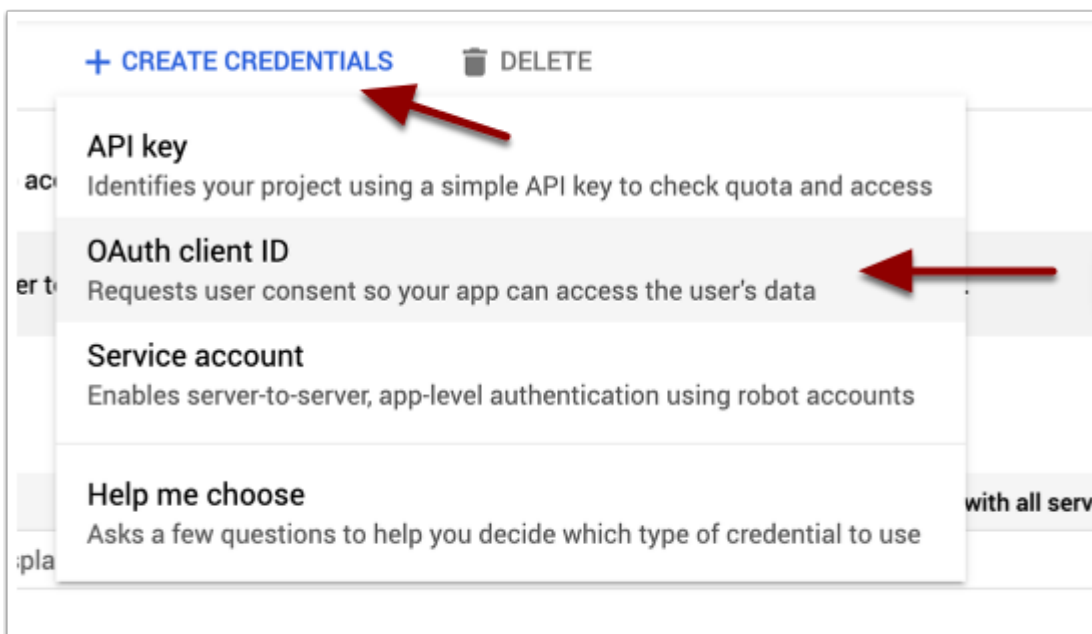
💡 If you have already used Google Cloud's console, you may not need to perform some of these steps, or interfaces may not look like the screenshots below.

First, create the app:

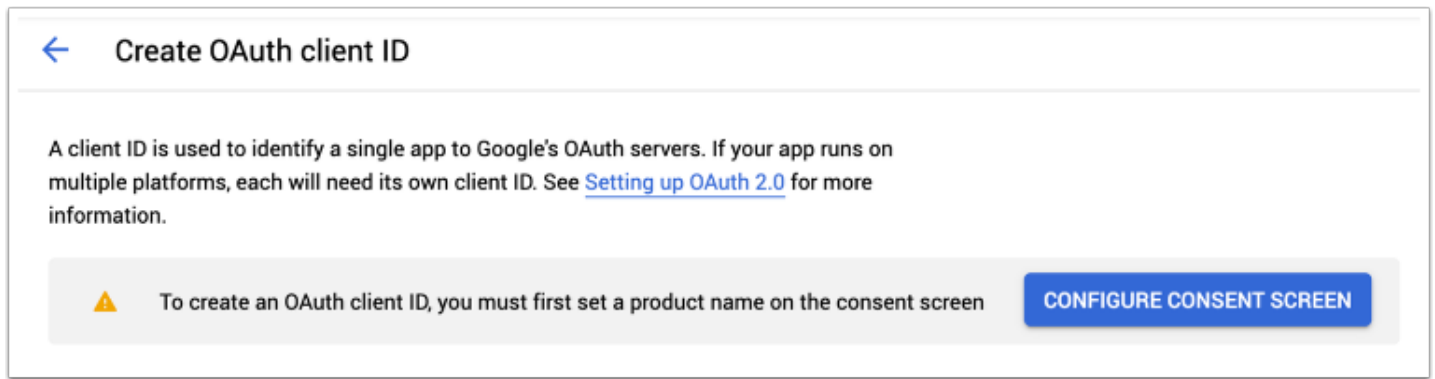
1. Log in to [the Google Cloud console](#). If you haven't before, select your country and agree to Google's *Terms of Service*.
2. Go to **APIs and Services > Credentials**.



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



4. Click **Configure Consent Screen**.



← Create OAuth client ID

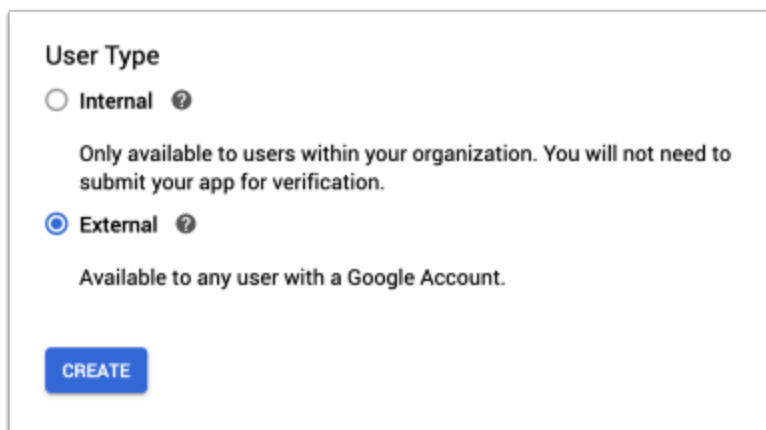
A client ID is used to identify a single app to Google's OAuth servers. If your app runs on multiple platforms, each will need its own client ID. See [Setting up OAuth 2.0](#) for more information.

⚠ To create an OAuth client ID, you must first set a product name on the consent screen

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.



User Type

☐ Internal ?

Only available to users within your organization. You will not need to submit your app for verification.

☒ External ?

Available to any user with a Google Account.

CREATE

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.

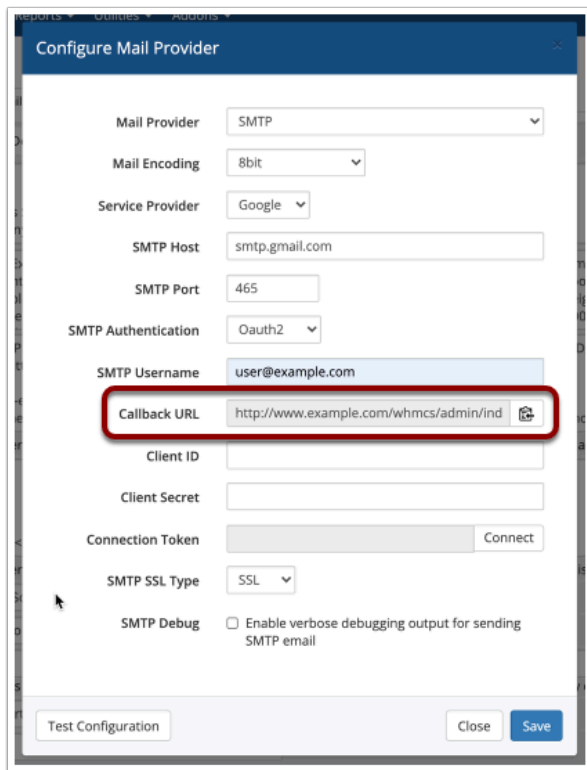
Application type *
Web application
▼

[Learn more](#) about OAuth client types

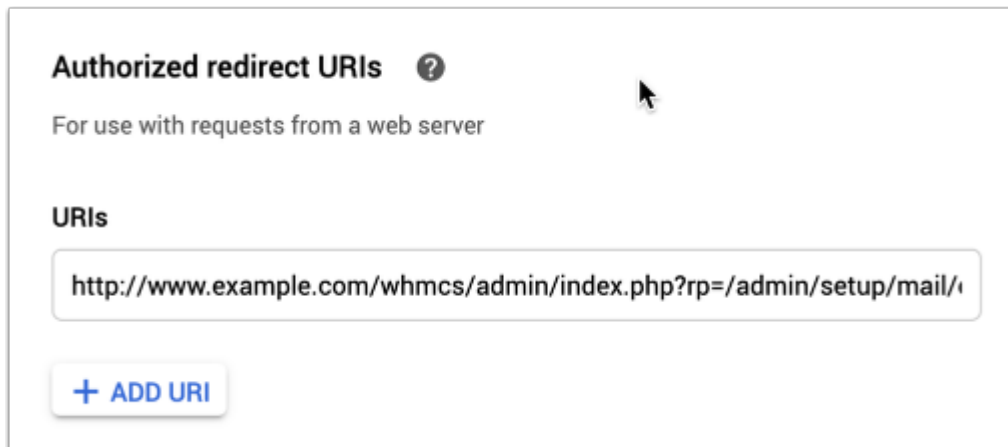
4. Enter a name for your application.

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

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

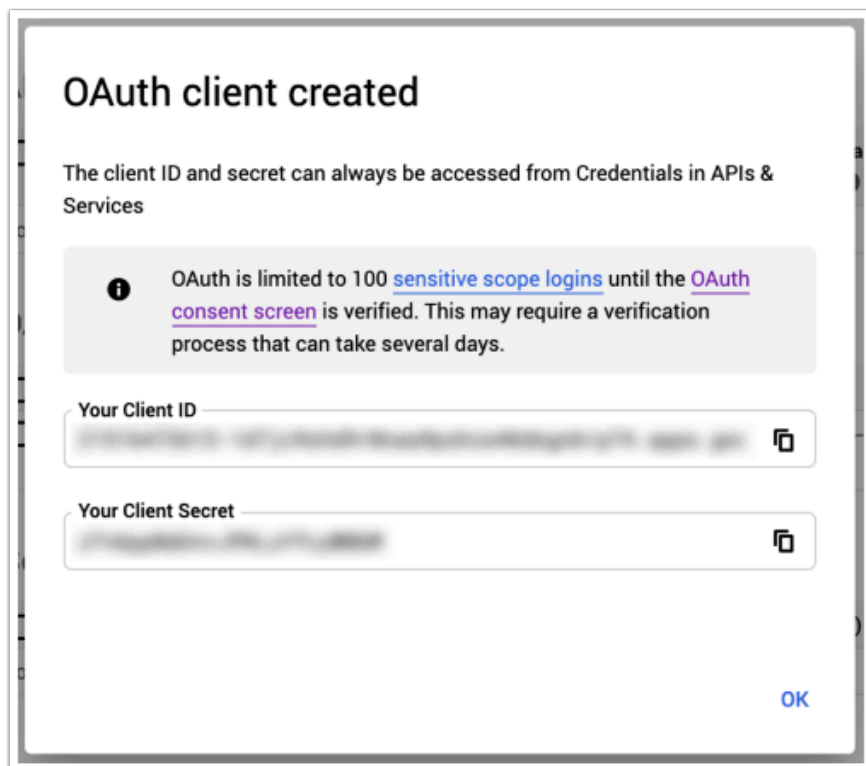


The screenshot shows the 'Configure Mail Provider' dialog box. The 'Mail Provider' is set to 'SMTP'. The 'Mail Encoding' is '8bit'. The 'Service Provider' is 'Google'. The 'SMTP Host' is 'smtp.gmail.com'. The 'SMTP Port' is '465'. The 'SMTP Authentication' is 'OAuth2'. The 'SMTP Username' is 'user@example.com'. The 'Callback URL' is 'http://www.example.com/whmcs/admin/ind', which is highlighted with a red rectangle. Below this are fields for 'Client ID', 'Client Secret', and 'Connection Token'. There is a 'Connect' button next to the 'Connection Token' field. The 'SMTP SSL Type' is 'SSL'. There is a checkbox for 'SMTP Debug' with the label 'Enable verbose debugging output for sending SMTP email'. At the bottom are buttons for 'Test Configuration', 'Close', and 'Save'.



The screenshot shows the 'Authorized redirect URIs' page. The title is 'Authorized redirect URIs' with a question mark icon. Below the title is the text 'For use with requests from a web server'. Under the 'URIs' section, there is a single URI: 'http://www.example.com/whmcs/admin/index.php?rp=/admin/setup/mail/'. At the bottom is a button labeled '+ ADD URI'.

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



You can now continue configuring SMTP with OAuth 2.0 (see the section above).