

Getting Started

39 <

Pricing & License

6 <

Tips, Tricks & More

130 <

Extensions & Features

168 <

Themes

8 <

Developer Docs

1 >

API

2 >

Authentication

Endpoints

Authentication

🕒 5 min read

Last Updated: 🕒 2 months ago



GeoDirectory REST API Authentication

The GeoDirectory REST API allows developers to view and manage GeoDirectory data programmatically. While some endpoints, such as those for map marker data and specific post data, are publicly accessible and **do not require authentication**, many other API requests that involve sensitive data or modifications **must be authenticated**.

Standard WordPress Application Passwords (Note)

WordPress provides “Application Passwords” for authenticating external applications with the standard WordPress REST API. You *can* use WordPress Application Passwords to authenticate requests to GeoDirectory API endpoints, as these typically inherit the permissions of the associated WordPress user.

However, **GeoDirectory provides its own dedicated API key authentication system, which is the preferred and recommended method** for interacting with the GeoDirectory REST API. Our system offers specific permission controls (Read, Write, Read/Write) tied directly to the API key, allowing for more granular control over API access.

GeoDirectory API Key Authentication (Preferred Method)

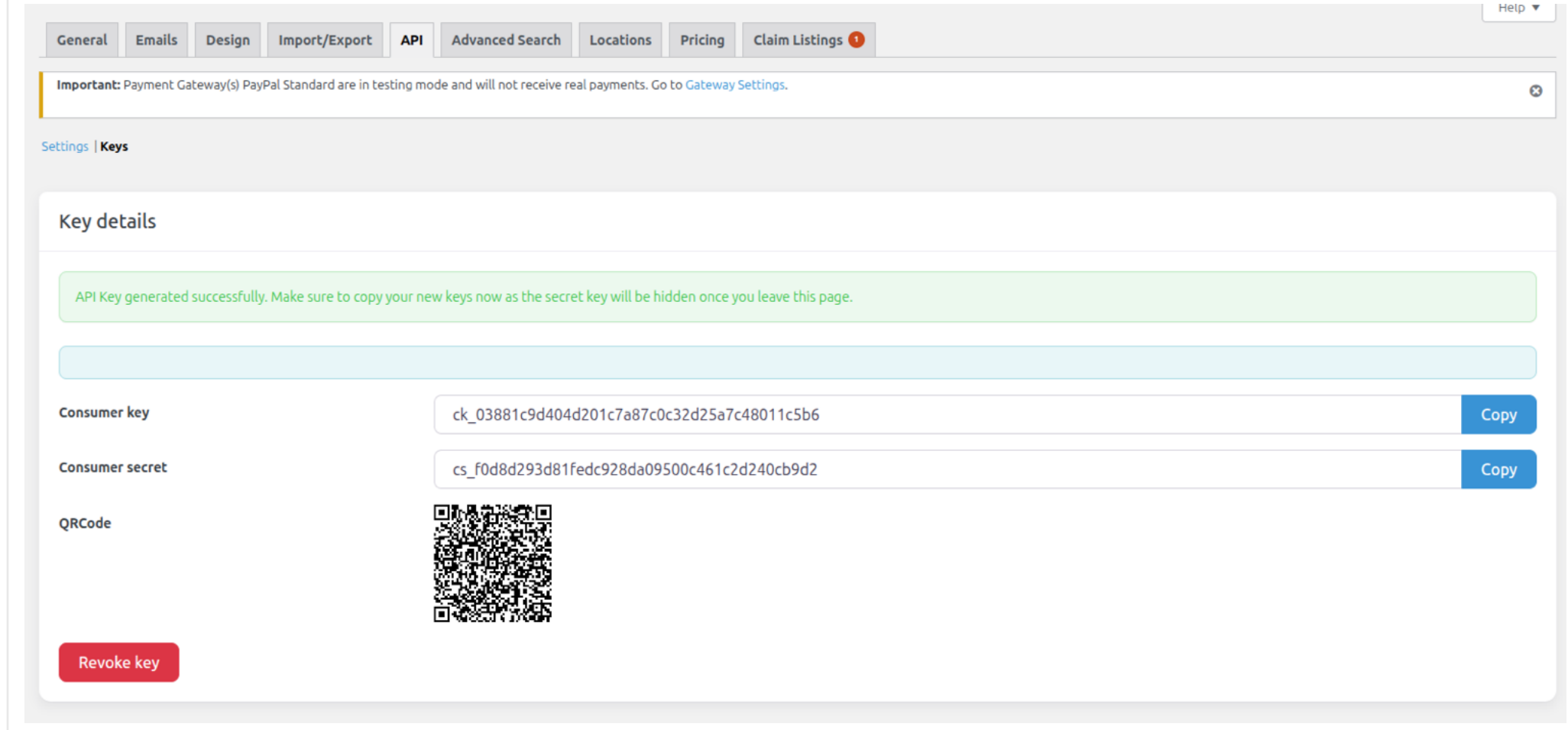
GeoDirectory's API authentication relies on a Consumer Key and Consumer Secret. These keys are generated within your GeoDirectory settings and are tied to a specific WordPress user with defined permissions (Read, Write, or Read/Write). This method is designed for secure connections over SSL (HTTPS) or when operating in a localhost environment.

Generating Your GeoDirectory API Keys

Before you can make authenticated API requests, you need to generate a Consumer Key and Consumer Secret within your WordPress admin area.

Steps to Generate API Keys:

- 1 Log in to your WordPress administration panel.
- 2 Navigate to **GeoDirectory > Settings**.
- 3 Click on the **API** tab, then select the **Keys** sub-tab.
- 4 If no keys exist, you will see a “Create an API key” button. Click it. If keys already exist, click the “Add key” button.
- 5 On the “Key details” screen:
 - **Description:** Enter a meaningful description for your API key (e.g., “Mobile App Integration,” “External Data Sync”). This helps you identify the key’s purpose later.
 - **User:** Select the WordPress user this API key will be associated with. The API requests made with this key will effectively be performed “as” this user.
 - **Permissions:** Choose the level of access for this key:
 - **Read:** Allows the key to retrieve data (GET requests).
 - **Write:** Allows the key to create, update, and delete data (POST, PUT, PATCH, DELETE requests).
 - **Read/Write:** Grants full read and write access.
 - Click the **Generate API key** button.
- 6 Once generated, you will see your **Consumer Key** and **Consumer Secret**. **Copy both of these immediately!** For security reasons, the Consumer Secret will be hidden once you leave this page.
 - **Consumer Key (Example):** `ck_03881c9d404d201c7a87c0c32d25a7c48011c5b6`
 - **Consumer Secret (Example):** `cs_f0d8d293d81fedc928da09500c461c2d240cb9d2`



Authenticating API Requests

The GeoDirectory API expects authentication credentials to be sent via **Basic Authentication** over **SSL (HTTPS)**. This is the simplest and most secure method for transmitting your API keys.

To use Basic Authentication:

- **Username:** Use your generated **Consumer Key**.
- **Password:** Use your generated **Consumer Secret**.

These credentials should be sent in the **Authorization** header of your HTTP request. Most HTTP client libraries and tools (like Postman or cURL) have built-in support for Basic Authentication, which will automatically encode these for you.

Example using Postman:

- 1 In your Postman request, go to the “Authorization” tab.
- 2 From the “Type” dropdown, select “Basic Auth”.
- 3 Enter your **Consumer Key** in the “Username” field.
- 4 Enter your **Consumer Secret** in the “Password” field. *Postman will automatically construct the **Authorization** header for you, looking something like: **Authorization: Basic Y2tf...0mNzXy...*** (base64 encoded)

Example using curl (Replace placeholders with your actual keys):

Bash

```
curl --user "ck_YOUR_CONSUMER_KEY:cs_YOUR_CONSUMER_SECRET" "https://yourdomain.com/wp-json/geodir/v2/"
```

Example using wp_remote_get() with authentication

The below example uses authentication to get the GeoDirectory settings groups.

```
// --- Configuration for GET Request ---
$geodir_api_base_url = 'https://yourdomain.com/wp-json/geodir/v2/'; // Replace
$consumer_key       = 'ck_YOUR_CONSUMER_KEY'; // Replace with your actual
$consumer_secret     = 'cs_YOUR_CONSUMER_SECRET'; // Replace with your actual

// Prepare authentication header for Basic Auth
$auth_header = 'Basic ' . base64_encode( $consumer_key . ':' . $consumer_secret );

// --- GET Request to retrieve GeoDirectory Places ---
$get_api_endpoint = $geodir_api_base_url . 'settings';

$get_args = array(
    'method' => 'GET',
    'headers' => array(
        'Authorization' => $auth_header,
        'Content-Type' => 'application/json',
    ),
    'timeout' => 30, // Adjust timeout as needed
    'sslverify' => true, // Always verify SSL for production
);

$response = wp_remote_get( $get_api_endpoint, $get_args );
```

Example using wp_remote_post() with authentication

The below example uses an authenticated POST request to update a GeoDirectory setting, in this case turning on usage_tracking (usually off by default)

```
// --- Configuration for POST Request ---
$geodir_api_base_url = 'https://yourdomain.com/wp-json/geodir/v2/'; // Replace
$consumer_key       = 'ck_YOUR_CONSUMER_KEY'; // Replace with your actual
$consumer_secret     = 'cs_YOUR_CONSUMER_SECRET'; // Replace with your actual

// Prepare authentication header for Basic Auth
$auth_header = 'Basic ' . base64_encode( $consumer_key . ':' . $consumer_secret );

// --- POST Request to update GeoDirectory Settings ---
$post_api_endpoint = $geodir_api_base_url . 'settings';

// Example data to send in the POST request.
// !!! YOU MUST ADJUST THIS TO MATCH THE ACTUAL SETTINGS YOUR API ACCEPTS !!!
$post_data = array(
    'example_setting_key_1' => 'value_for_setting_1',
    'example_setting_key_2' => true, // Boolean example
    // Add all key-value pairs required by your settings endpoint
);

$post_args = array(
    'method' => 'POST',
    'headers' => array(
        'Authorization' => $auth_header,
    ),
    'body' => $post_data, // Pass the array directly, wp_remote_post will handle it
    'timeout' => 30, // Adjust timeout as needed
    'sslverify' => true, // Always verify SSL for production
    'data_format' => 'body', // Important for sending JSON in the body
);

$post_response = wp_remote_post( $post_api_endpoint, $post_args );
```

Important Security Note

- **Always use HTTPS/SSL for your API requests.** Transmitting API keys over insecure HTTP connections can expose your credentials to interception.
- The GeoDirectory API is designed to reject API key authentication attempts over non-SSL connections (excluding localhost).
- Never hardcode your API keys directly into client-side code or public repositories. Store them securely and use environment variables or a secure configuration management system.

On this page

GeoDirectory REST API Authentication

Standard WordPress Application Passwords (Note)

GeoDirectory API Key Authentication (Preferred Method)

Generating Your GeoDirectory API Keys

Authenticating API Requests

Example using wp_remote_get() with authentication

Example using wp_remote_post() with authentication

Important Security Note