

---

# Trovebox PHP Documentation

*Release 0.0.1*

**Mr. Zen**

July 26, 2014



<b>1</b>	<b>About</b>	<b>1</b>
<b>2</b>	<b>Requirements</b>	<b>3</b>
<b>3</b>	<b>Table of Contents</b>	<b>5</b>
3.1	Quickstart . . . . .	5



### About

---

Trovebox-PHP is a PHP library utilizing Guzzle to interact with the Trovebox API. This documentation serves to provide functional examples on how to use the library.

The best way to get started is to read the *Quickstart Guide*.



---

## Requirements

---

- PHP  $\geq 5.4$
- Composer (to install dependencies)
- PHP Curl Extension *Recommended*.





---

## Table of Contents

---

### 3.1 Quickstart

This page describes how to get started using the Trovebox API quickly.

#### 3.1.1 Installation

Install Trovebox-PHP using [Composer](#)

```
shell$ composer install mrzen/trovebox-php
```

Or add it to your `composer.json` file.

#### 3.1.2 Connecting

To connect to the API you need to provide it with the following data:

- API Endpoint (this will vary depending on your account)
- Consumer Key
- Consumer Secret
- Token
- Token Secret

The keys and secrets can be set up using “Apps” in trovebox.

You can create a client using any of the following options:

- JSON String
- Path to JSON file
- Associative Array
- Setting the `$_SERVER['CONFIG']` to a json string or path to file

The JSON file should look like this:

```
{
    "base_url" : "http://your-account.trovebox.com/",
    "oatuh" : {
        "consumer_key" : "YOUR_CONSUMER_KEY",
```

```
"consumer_secret" : "YOUR_CONSUMER_SECRET",
"token"           : "YOUR_TOKEN",
"token_secret"    : "YOUR_TOKEN_SECRET"
},

"defaults" : { "auth" : "oauth" }
}
```

**Warning:** The defaults section is required for authenticated API operations

### Example

```
$trovebox = new Trovebox\Client($YOUR_CONFIG);
$trovebox->hello();
```

The `Trovebox\Client::hello` method will run the Trovebox “Hello” API call to make sure you’re connected.

### Getting Photos

Now that we’ve connected to the Trovebox API, let’s get some data.

```
// $trovebox is the trovebox client we created in the previous example
$photos = $trovebox->photos(); // Will get the 30 most recent photos
// $photos is an array of \Trovebox\Models\Photo
foreach($photos as $photo) {
    echo $photo . "\n"; // Print out the photo
}
```

For more information, see [Working with photos](#)