
imboclient-php-cli Documentation

Release dev

Christer Edvartsen

December 26, 2015

1	Requirements	3
2	Installation	5
3	Usage	7
3.1	Configuration file	7
3.2	Commands	8

This is a command line wrapper around the official PHP-based client for [Imbo](#) servers.

Requirements

The client requires `PHP >= 5.4`.

Installation

ImboClientCli can be installed using [Composer](#) by requiring `imbo/imboclient-cli` in your `composer.json` file, or by running the following commands:

```
curl -s https://getcomposer.org/installer | php
php composer.phar create-project imbo/imboclient-cli [<dir>] [<version>]
```

Available versions can be located at [packagist](#).

Usage

Once the client is installed you will find an executable called `imboclient` in the `bin` directory where you installed the client.

Below you will find documentation covering most features of the client.

- *Configuration file*
- *Commands*
 - *Activate a server - activate*
 - *Add one or more images - add(add-images)*
 - *Deactivate a server - deactivate*
 - *Delete an image - delete(delete-image)*
 - *Get help - help*
 - *List available commands - list*
 - *List servers in the configuration file - list-servers*
 - *Get the number of images - num-images*
 - *Get the server status - server-status*

3.1 Configuration file

To be able to use the client you will need to create a [YAML](#) configuration file that contains server definitions and key pairs. The client will look for the configuration file in the following paths (in the order specified):

- `./config.yml`
- `~/.imboclient/config.yml`
- `/etc/imboclient/config.yml`

The client will use the first file it finds. A custom path can be specified by using the `--config <path>` command line option.

The client ships with an example configuration file (shown below) that you can use as a base for your own configuration.

```
1 # Example configuration file
2 #
3 # Each server can have its own name (instead of server1, server2 and server3 as stated below).
4 # Remember to update the default server if you change the server names. Servers can be disabled by
5 # Changing the active state to false. The client will refuse to run any operations against inactive
6 # servers.
```

```
7
8 servers:
9     default: server1
10    server1:
11        url: http://imbo1.example.com
12        user: user
13        publicKey: public key
14        privateKey: private key
15        active: true
16
17    server2:
18        url: http://imbo2.example.com
19        user: user
20        publicKey: public key
21        privateKey: private key
22        active: true
23
24    server3:
25        url: http://imbo3.example.com
26        user: user
27        publicKey: public key
28        privateKey: private key
29        active: false
```

The default key should point to the server definition that is to be used per default. If you need to execute a remote command (for instance adding / deleting images) from another server, specify the `--server <server>` option when executing the command. When a server definition is labeled as not active (`active: false`) you can not execute remote commands for that server. These flags can be changed manually by editing the configuration file, or by using the [activate](#) or [deactivate](#) commands in the client.

3.2 Commands

Below you will find all the different commands supported by the client, along with a detailed description of all options and arguments.

- *Activate a server* - `activate`
- *Add one or more images* - `add(add-images)`
- *Deactivate a server* - `deactivate`
- *Delete an image* - `delete(delete-image)`
- *Get help* - `help`
- *List available commands* - `list`
- *List servers in the configuration file* - `list-servers`
- *Get the number of images* - `num-images`
- *Get the server status* - `server-status`

3.2.1 Activate a server - `activate`

3.2.2 Add one or more images - `add` (`add-images`)

3.2.3 Deactivate a server - `deactivate`

3.2.4 Delete an image - `delete` (`delete-image`)

3.2.5 Get help - `help`

3.2.6 List available commands - `list`

3.2.7 List servers in the configuration file - `list-servers`

3.2.8 Get the number of images - `num-images`

3.2.9 Get the server status - `server-status`