
Flask-Pystmark Documentation

Release 0.1

Steve Leonard

Sep 26, 2019

Contents

1	Installation	3
2	Configuration	5
3	Example	7
4	API	9
5	Pystmark Object	11
6	Message Object	13
	Python Module Index	17
	Index	19

Flask-Pystmark is a Flask extension for the [Postmark API](#) library [Pystmark](#).

The extension contains two objects, `Pystmark` and `Message`. `Pystmark` wraps the [Simple API](#) of the `pystmark` module. Both inject **Flask-Pystmark** configuration variables into the functions or objects they wrap.

See also:

[Pystmark Documentation](#)

Flask-Pystmark supports Python 2.6, 2.7, 3.3 and PyPy.

CHAPTER 1

Installation

```
$ pip install Flask-Pystmark
```


Flask-Pystmark is configured through the standard Flask config API. Any of them can be overridden in calls to the *Pystmark* methods or *Message* construction. These are the available options:

- **PYSTMARK_API_KEY** : Required. Your API key for postmarkapp.com
- **PYSTMARK_HTTPS** : default *True*. Use https for requests to postmarkapp.com
- **PYSTMARK_TEST_API** : default *False*. Use the Postmark test API. *Note: a request is still made to postmarkapp.com, but accesses their sandbox.*
- **PYSTMARK_DEFAULT_SENDER** : default *None*. Default sender email to use for outgoing messages
- **PYSTMARK_DEFAULT_REPLY_TO** : default *None*. Default reply_to email to use for outgoing messages
- **PYSTMARK_DEFAULT_HEADERS** : default *None*. Default headers to apply to outgoing messages. They must be in the format required by Postmark. *Note: these are headers in the email. If you need headers in the request sent to postmarkapp.com, pass them in to the API wrappers as you would in a call to requests.request*
- **PYSTMARK_VERIFY_MESSAGES** : default *False*. Apply sanity checks to all messages when created. Will raise *pystmark.MessageError* if it appears invalid.

CHAPTER 3

Example

```
# app.py
from flask import Flask
from flask_pystmark import Pystmark, Message
from pystmark import ResponseError

app = Flask(__name__)
app.config['PYSTMARK_API_KEY'] = 'your_api_key'
app.config['PYSTMARK_DEFAULT_SENDER'] = 'admin@example.com'
pystmark = Pystmark(app)

@app.route('/')
def send():
    m = Message(to='user@gmail.com', text='Welcome')
    resp = pystmark.send(m)
    try:
        resp.raise_for_status()
    except ResponseError as e:
        return 'Failed to send message. Reason: {}'.format(e)
    else:
        return 'Sent message to {}'.format(resp.message.to)
```


CHAPTER 4

API

Pystmark Object

class flask_pystmark.**Pystmark** (*app=None*)

A wrapper around the Simple API of pystmark.

Refer to <http://pystmark.readthedocs.org/en/latest/api.html#simple-api> for more details.

Parameters **app** – Flask app to initialize with. Defaults to *None*

activate_bounce (*bounce_id, **request_args*)

Activate a deactivated bounce.

Parameters

- **bounce_id** – The bounce’s id. Get the id with *get_bounces ()*.
- ****request_args** – Keyword arguments to pass to *requests.request ()*.

Return type *pystmark.BounceActivateResponse*

get_bounce (*bounce_id, **request_args*)

Get a single bounce.

Parameters

- **bounce_id** – The bounce’s id. Get the id with *get_bounces ()*.
- ****request_args** – Keyword arguments to pass to *requests.request ()*.

Return type *pystmark.BounceResponse*

get_bounce_dump (*bounce_id, **request_args*)

Get the raw email dump for a single bounce.

Parameters

- **bounce_id** – The bounce’s id. Get the id with *get_bounces ()*.
- ****request_args** – Keyword arguments to pass to *requests.request ()*.

Return type *pystmark.BounceDumpResponse*

get_bounce_tags (**request_args)

Get a list of tags for bounces associated with your Postmark server.

Parameters ****request_args** – Keyword arguments to pass to `requests.request()`.

Return type `pystmark.BounceTagsResponse`

get_bounces (**request_args)

Get a paginated list of bounces.

Parameters ****request_args** – Keyword arguments to pass to `requests.request()`.

Return type `pystmark.BouncesResponse`

get_delivery_stats (**request_args)

Get delivery stats for your Postmark account.

Parameters ****request_args** – Keyword arguments to pass to `requests.request()`.

Return type `pystmark.DeliveryStatsResponse`

init_app (app)

Initialize Pystmark with a Flask app

send (message, **request_args)

Send a message.

Parameters

- **message** (*dict* or *Message*) – Message to send.
- ****request_args** – Keyword arguments to pass to `requests.request()`.

Return type `pystmark.SendResponse`

send_batch (messages, **request_args)

Send a batch of messages.

Parameters

- **messages** – Messages to send.
- ****request_args** – Keyword arguments to pass to `requests.request()`.

Return type `pystmark.BatchSendResponse`

Message Object

```
class flask_pystmark.Message (sender=None, to=None, cc=None, bcc=None, subject=None,  
                               tag=None, html=None, text=None, reply_to=None, headers=None,  
                               attachments=None, verify=None, track_opens=None)
```

A container for message(s) to send to the Postmark API. You can populate this message with defaults for initializing an `Interface` from the `pystmark` library. The message will be combined with the final message and verified before transmission.

Refer to <http://pystmark.readthedocs.org/en/latest/api.html#message-object> for more details.

Parameters

- **sender** – Email address of the sender. Defaults to `PYSTMARK_DEFAULT_SENDER` if defined.
- **to** – Destination email address.
- **cc** – A list of cc'd email addresses.
- **bcc** – A list of bcc'd email address.
- **subject** – The message subject.
- **tag** – Tag your emails with this.
- **html** – HTML body content.
- **text** – Text body content.
- **reply_to** – Email address to reply to. Defaults to `PYSTMARK_DEFAULT_REPLY_TO`, if defined.
- **headers** (A list of *dict*, each with the keys 'Name' and 'Value'.) – Additional headers to include with the email. If you do not have the headers formatted for the Postmark API, use `Message.add_header()`. Defaults to `PYSTMARK_DEFAULT_HEADERS`, if defined.
- **attachments** (A list of *dict*, each with the keys 'Name', 'Content' and 'ContentType'.) – Attachments to include with the email. If you do not have the attachments formatted for the Postmark API, use `Message.attach_file()` or `Message.attach_binary()`.

- **verify** – Verify the message when initialized. Defaults to PYST-MARK_VERIFY_MESSAGES if provided, otherwise *False*.

add_header (*name, value*)

Attach an email header to send with the message.

Parameters

- **name** – The name of the header value.
- **value** – The header value.

attach_binary (*data, filename, content_type=None, content_id=None*)

Attach a file to the message given raw binary data.

Parameters

- **data** – Raw data to attach to the message.
- **filename** – Name of the file for the data.
- **content_type** – mimetype of the data. It will be guessed from the filename if not provided.
- **content_id** – ContentID URL of the attachment. A RFC 2392- compliant URL for the attachment that allows it to be referenced from inside the body of the message. Must start with 'cid:'

attach_file (*filename, content_type=None, content_id=None*)

Attach a file to the message given a filename.

Parameters

- **filename** – Name of the file to attach.
- **content_type** – mimetype of the data. It will be guessed from the filename if not provided.
- **content_id** – ContentID URL of the attachment. A RFC 2392- compliant URL for the attachment that allows it to be referenced from inside the body of the message. Must start with 'cid:'

bcc

A comma delimited string of receivers for the message 'Bcc' field.

cc

A comma delimited string of receivers for the message 'Cc' field.

data ()

Returns data formatted for a POST request to the Postmark send API.

Return type *dict*

json ()

Return json-encoded string of message data.

Return type *str*

load_from (*other, **kwargs*)

Create a *Message* by merging *other* with *self*. Values from *other* will be copied to *self* if the value was not set on *self* and is set on *other*. :param other: The *Message* to copy defaults from. :type other: *Message*
:param kwargs: Additional keyword arguments to construct

Message with.

Return type *Message*

classmethod `load_message` (*message*, ***kwargs*)

Create a *Message* from a message data *dict*.

Parameters

- **message** – A *dict* of message data.
- **kwargs** – Additional keyword arguments to construct *Message* with.

Return type *Message*

recipients

A list of all recipients for this message.

to

A comma delimited string of receivers for the message 'To' field.

verify ()

Verifies the message data based on rules and restrictions defined in the Postmark API docs. There can be no more than 20 recipients in total. NOTE: This does not check that your attachments total less than 10MB, you must do that yourself.

f

flask_pystmark, ??

A

`activate_bounce()` (*flask_pystmark.Pystmark method*), 11
`add_header()` (*flask_pystmark.Message method*), 14
`attach_binary()` (*flask_pystmark.Message method*), 14
`attach_file()` (*flask_pystmark.Message method*), 14

B

`bcc` (*flask_pystmark.Message attribute*), 14

C

`cc` (*flask_pystmark.Message attribute*), 14

D

`data()` (*flask_pystmark.Message method*), 14

F

`flask_pystmark` (*module*), 1

G

`get_bounce()` (*flask_pystmark.Pystmark method*), 11
`get_bounce_dump()` (*flask_pystmark.Pystmark method*), 11
`get_bounce_tags()` (*flask_pystmark.Pystmark method*), 11
`get_bounces()` (*flask_pystmark.Pystmark method*), 12
`get_delivery_stats()` (*flask_pystmark.Pystmark method*), 12

I

`init_app()` (*flask_pystmark.Pystmark method*), 12

J

`json()` (*flask_pystmark.Message method*), 14

L

`load_from()` (*flask_pystmark.Message method*), 14
`load_message()` (*flask_pystmark.Message class method*), 15

M

`Message` (*class in flask_pystmark*), 13

P

`Pystmark` (*class in flask_pystmark*), 11

R

`recipients` (*flask_pystmark.Message attribute*), 15

S

`send()` (*flask_pystmark.Pystmark method*), 12
`send_batch()` (*flask_pystmark.Pystmark method*), 12

T

`to` (*flask_pystmark.Message attribute*), 15

V

`verify()` (*flask_pystmark.Message method*), 15