pypertrail Documentation

Release 1.0.0

Quentin Rousseau

October 06, 2016

Contents

1 pypertrail **3**
- 1.1 Installation . . . 3
- 1.2 Requirements . . . 3
- 1.3 Library . . . 3
- 1.4 CLI . . . 6

2 Documentation **9**

3 History **11**

4 Authors **13**

5 License **15**

6 Installation **17**
- 6.1 Stable release . . . 17
- 6.2 From sources . . . 17

7 Usage **19**

8 Contributing **21**
- 8.1 Types of Contributions . . . 21
- 8.2 Get Started! . . . 22
- 8.3 Pull Request Guidelines . . . 22
- 8.4 Tips . . . 23

9 Indices and tables **25**

Contents:

CHAPTER 1

pypertrail

Python wrapper library and CLI for papertrail API.

1.1 Installation

Install using pip:

```
$ pip install pypertrail
```

1.2 Requirements

- Python 2.6, 2.7, 3.3, 3.4, or 3.5
- A Papertrail account

1.3 Library

1.3.1 Accounts

```
import os
from pypertrail.accounts import Account

accounts = Account(os.environ['PAPERTRAIL_API_TOKEN'])

# List account usage
accounts.list()
```

1.3.2 Archives

```
import os
from pypertrail.archives import Archive

archives = Archive(os.environ['PAPERTRAIL_API_TOKEN'])
```

```
# List archive information
archives.list()
```

1.3.3 Groups

```
import os
from pypertrail.groups import Group

groups = Group(os.environ['PAPERTRAIL_API_TOKEN'])

# List groups
groups.list()

# Show a group
groups.show(group_id)

# Update a group
groups.update(group_id)

# Delete a group
groups.delete(group_id)
```

1.3.4 Saved searches

```
import os
from pypertrail.saved_searches import SavedSearch

saved_searches = SavedSearch(os.environ['PAPERTRAIL_API_TOKEN'])

# List saved_searches
saved_searches.list()

# Show a saved search
saved_searches.show(saved_search_id)

# Create a saved search
payload = {'name':'my_query', 'query':'sshd'}
saved_searches.create(payload)

# Update a saved search
payload = {'search[query]':'another_query'}
saved_searches.update(payload)

# Delete a saved search
saved_searches.delete(saved_search_id)
```

1.3.5 Search

```
import os
from pypertrail.search import Search

search = Search(os.environ['PAPERTRAIL_API_TOKEN'])
```

```
# Search events
search.events("sshd")
```

1.3.6 Systems

```
import os
from pypertrail.systems import System

systems = Search(os.environ['PAPERTRAIL_API_TOKEN'])

# List systems
systems.list()

# Show a system
systems.show(system_id)

# Create a system
payload = {'system[name]':'foo', 'system[hostname]':'bar', 'destination_port':46865}
systems.create(payload)

# Update a system
payload = {'system[name]':'another_name'}
systems.update(system_id, payload)

# Delete a system
systems.delete(system_id)

# Join a group
payload = {'group_id':10}
systems.join_group(system_id, payload)

# Leave a group
payload = {'group_id':10}
systems.leave_group(system_id, payload)
```

1.3.7 Users

```
import os
from pypertrail.users import User

users = User(os.environ['PAPERTRAIL_API_TOKEN'])

# List users
users.list()

# Invite a user
payload = {'email':'contact@quent.in', 'read_only':'true'}
users.invite(payload)

# Delete a user
users.delete(user_id)
```

1.4 CLI

1.4.1 CLI Authentication

Via environment variables:

```
$ export PAPERTRAIL_API_TOKEN=my_token
$ pypertrail users list
```

Via implicit ~/.pypertrail.yml:

```
$ echo "token: my_token" > ~/.pypertrail.yml
$ pypertrail users list
```

Via (–conf/-c) option:

```
$ echo "token: my_token" > /path/to/config
$ pypertrail --conf /path/to/config users list
```

Via (–token/-t) option:

```
$ pypertrail --token my_token users list
```

1.4.2 Subcommands

1.4.3 Accounts

Examples:

```
$ pypertrail --pretty accounts list
```

1.4.4 Archives

Examples:

```
$ pypertrail --pretty archives list
```

1.4.5 Saved searches

Examples:

```
$ pypertrail saved_searches create --payload '{"search[name]":"foo", "search[query]":"bar"}'
$ pypertrail saved_searches delete 1
$ pypertrail saved_searches list
$ pypertrail saved_searches show 1
$ pypertrail saved_searches update 1 --payload '{"search[query]":"another_query"}'
```

1.4.6 Groups

Examples:

```
$ pypertrail groups delete 1
$ pypertrail groups list
$ pypertrail groups show 1
$ pypertrail groups update 1 --payload '{"group[name]":"another_name"}'
```

1.4.7 Search

Examples:

```
$ pypertrail search events
$ pypertrail search events --follow
$ pypertrail search events --follow --color program
$ pypertrail search events --follow --query sshd
$ pypertrail search events --follow --json
```

1.4.8 Systems

Examples:

```
$ pypertrail systems create --payload '{"system[name]":"foo", "system[hostname]":"bar", "destination_
$ pypertrail systems delete 1
$ pypertrail systems join_group 1 --payload '{"group_id":10}'
$ pypertrail systems leave_group 1 --payload '{"group_id":10}'
$ pypertrail systems list
$ pypertrail systems show 1
$ pypertrail systems update 1 --payload '{"system[name]":"another_name"}'
```

1.4.9 Users

Examples:

```
$ pypertrail users delete 1
$ pypertrail users invite --payload '{"email":"contact@quent.in", "read_only":true}'
$ pypertrail users list
```

CHAPTER 2

Documentation

- https://pypertrail.readthedocs.io
- http://help.papertrailapp.com/kb/how-it-works/http-api

CHAPTER 3

History

View the changelog

CHAPTER 4

Authors

- Quentin Rousseau

CHAPTER 5

License

CHAPTER 6

Installation

6.1 Stable release

To install pypertrail, run this command in your terminal:

```
$ pip install pypertrail
```

This is the preferred method to install pypertrail, as it will always install the most recent stable release.

If you don't have pip installed, this Python installation guide can guide you through the process.

6.2 From sources

The sources for pypertrail can be downloaded from the Github repo.

You can either clone the public repository:

```
$ git clone git://github.com/kwent/pypertrail
```

Or download the tarball:

```
$ curl  -OL https://github.com/kwent/pypertrail/tarball/master
```

Once you have a copy of the source, you can install it with:

```
$ python setup.py install
```

CHAPTER 7

Usage

To use pypertrail in a project:

```
import pypertrail
```

CHAPTER 8

Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

You can contribute in many ways:

8.1 Types of Contributions

8.1.1 Report Bugs

Report bugs at https://github.com/kwent/pypertrail/issues.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

8.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with "bug" and "help wanted" is open to whoever wants to implement it.

8.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with "enhancement" and "help wanted" is open to whoever wants to implement it.

8.1.4 Write Documentation

pypertrail could always use more documentation, whether as part of the official pypertrail docs, in docstrings, or even on the web in blog posts, articles, and such.

8.1.5 Submit Feedback

The best way to send feedback is to file an issue at https://github.com/kwent/pypertrail/issues.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

8.2 Get Started!

Ready to contribute? Here's how to set up *pypertrail* for local development.

1. Fork the *pypertrail* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/pypertrail.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv pypertrail
$ cd pypertrail/
$ python setup.py develop
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

   Now you can make your changes locally.

5. When you're done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 pypertrail tests
$ python setup.py test or py.test
$ tox
```

   To get flake8 and tox, just pip install them into your virtualenv.

6. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

8.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.

2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 2.6, 2.7, 3.3, 3.4 and 3.5, and for PyPy. Check https://travis-ci.org/kwent/pypertrail/pull_requests and make sure that the tests pass for all supported Python versions.

8.4 Tips

To run a subset of tests:

```
$ python -m unittest tests.test_pypertrail
```

CHAPTER 9

Indices and tables

- genindex
- modindex
- search