PyAC Documentation

Release 0.6.0

juga

Apr 01, 2018

Contents:

1 Documentation **3**

2 Installation **5**

3 Download **7**

4 Bugs and features **9**

5 Current status **11**

6 License **13**

7 Acknowledgments **15**

7.1 Installation 15

7.2 Running 15

7.3 Diagrams 17

7.4 PyAC Python API Reference 17

7.5 TODO 18

7.6 Indices and tables 18

Python Module Index **19**

Autocrypt Level 1 implemention using PGPy . Initially part of py-autocrypt, which uses GnuPG.

CHAPTER 1

Documentation

A more extensive online documentation is available in Read the docs. The documentation source is in this repository.

CHAPTER 2

Installation

See Installation and Running

CHAPTER 3

Download

You can download this project in either zip or tar formats.

You can also clone the project with Git by running:

```
git clone https://github.com/juga0/pyac
```

CHAPTER 4

Bugs and features

If you wish to signal a bug or report a feature request, please fill-in an issue on the pyac issue tracker.

CHAPTER 5

Current status

WIP, still not recommended for end users. Testers welcomed.

See TODO

CHAPTER 6

License

pyac is copyright 2017, 2018 by juga (juga at riseup dot net) and is licensed by the terms of the MIT license.

CHAPTER 7

Acknowledgments

Autocrypt

7.1 Installation

7.1.1 Installation for developers

It is recommended to install `pyac` in a python virtual environment.

Check https://virtualenv.pypa.io/en/latest/installation.html. In Debian:

```
sudo apt install python-virtualenv
```

Create a virtual environment:

```
mkdir ~/.virtualenvs
virtualenv ~/.virtualenvs/pyacenv
source ~/.virtualenvs/pyacenv/bin/activate
```

Get the sources:

```
git clone https://github.com/juga0/pyac
```

Install it:

```
pip install -e .
```

7.2 Running

An updated command line usage description can be obtained with:

autocrypt -h

At the time of writing the output is:

usage: autocrypt [-h] [–version] [-d] [-m PGPHOME] [-l] [-n NEWACCOUNT] [-r NEWPEER] [-a] [-g] [-u] [-p PASSPHRASE] [-c] [-f FROMH] [-t TO] [-s SUBJECT] [-b BODY] [-e PE] [-i INPUT] [-o OUTPUT]

optional arguments:

-h, --help show this help message and exit

--version show program's version number and exit

-d, --debug Set logging level to debug

-m PGPHOME, --pgphome PGPHOME Path to Autocrypt home, ~/.pyac by default

-l, --list List account and peers

-n NEWACCOUNT, --newaccount NEWACCOUNT Email address for the new account. It will also generate new OpenPGP keys.

-r NEWPEER, --newpeer NEWPEER Email address for the new peer.

-a, --genac Generate Autocrypt Email. Use -f, -t, -s, -b, or the defaults will be use

-g, --genag Generate Autocrypt Gossip Email

-u, --genas Generate Autocrypt Setup Email

-p PASSPHRASE, --passphrase PASSPHRASE Passphrase to generate an Autocrypt Setup Email

-c, --genasc Generate Autocrypt Setup Code

-f FROMH, --fromh FROMH Email sender address and OpenPGP UID

-t TO, --to TO Email recipient addresses separatade by comma

-s SUBJECT, --subject SUBJECT Subject for the Autocrypt Email

-b BODY, --body BODY Body for the Autocrypt Email

-e PE, --pe PE prefer-encrypt for the Autocrypt Email

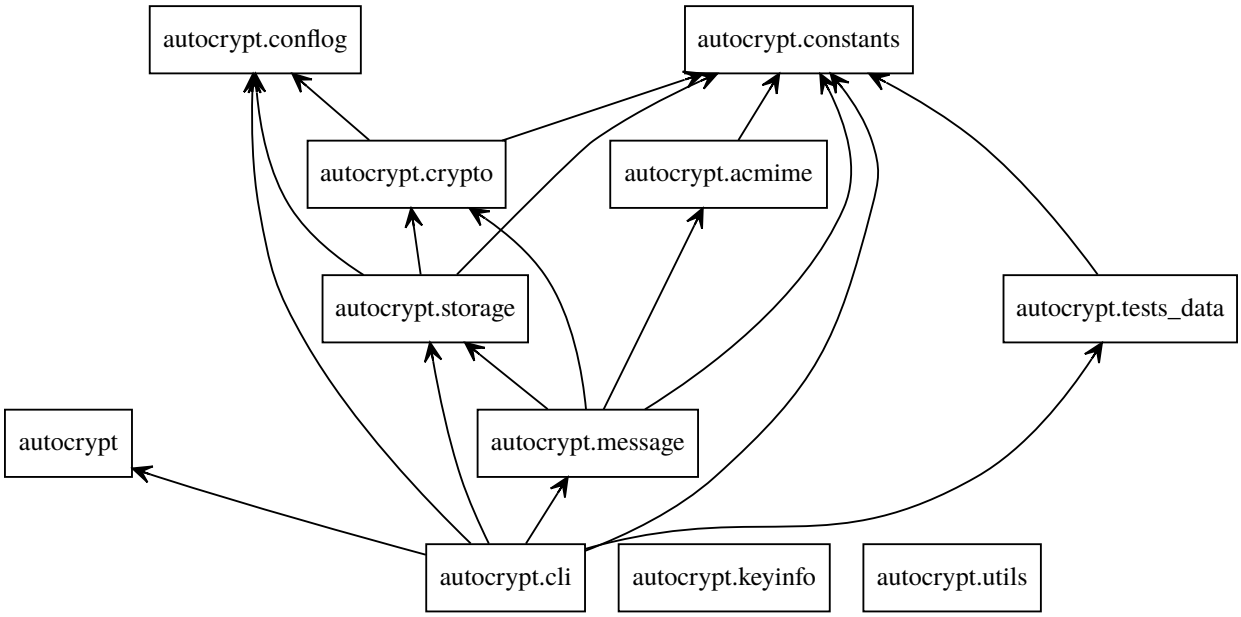
-i INPUT, --input INPUT Path to the Email to parse, by default: /home/user/_my/code/mailencrypt-related/pyac/tests/data/example-simple-autocrypt- pyac.eml

-o OUTPUT, --output OUTPUT Path to store the Autocrypt Email, by default: /tmp/output.eml

An useful argument when reporting bugs is `-d`.

7.3 Diagrams

7.3.1 Package diagram



7.4 PyAC Python API Reference

7.4.1 autocrypt package

autocrypt.acmime module

class autocrypt.acmime.**MIMEApplicationACSetupPayload**(*_data*, *_subtype='autocrypt-setup'*, *_encoder=<function encode_noop>*, ***_params*)

Bases: email.mime.application.MIMEApplication

Class for generating application/autocrypt-setup MIME documents.

class autocrypt.acmime.**MIMEMultipartACSetup**(*_data=None*, *_subtype='mixed'*, *boundary=None*, ***_params*)

Bases: email.mime.multipart.MIMEMultipart

Base class for MIME multipart/mixed including application/autocrypt-setup.

class autocrypt.acmime.**MIMETextACSetupDescription**(*_data='This message contains all information to transfer your Autocryptnsettings along with your secret key securely from your originalndevice.nnTo set up your new device for Autocrypt, please follow theninstuctions that should be presented by your new device.nnYou can keep this message and use it as a backup for your secretnkey. If you want to do this, you should write down the Setup Codenand store it securely.n'*, *_subtype='plain'*)

Bases: email.mime.text.MIMEText

Class for generating text/plain MIME documents.

autocrypt.conflog module

Logging configuration.

autocrypt.constants module

autocrypt.test_data module

.

autocrypt.crypto module

autocrypt.message module

7.5 TODO

- [ ] Implement recommendations

7.6 Indices and tables

- genindex
- modindex
- search

Python Module Index

a

autocrypt.acmime, 17
autocrypt.conflog, 18
autocrypt.constants, 18
autocrypt.tests_data, 18

Index

A

autocrypt.acmime (module), 17
autocrypt.conflog (module), 18
autocrypt.constants (module), 18
autocrypt.tests_data (module), 18

M

MIMEApplicationACSetupPayload (class in autocrypt.acmime), 17
MIMEMultipartACSetup (class in autocrypt.acmime), 17
MIMETextACSetupDescription (class in autocrypt.acmime), 17