pySafe Documentation

Release dev

May 06, 2019

Contents

python interface for the C FFI API of the SAFE Network (www.maidsafe.net)

contributors: rid-dim, duncankushnir

The design goal of pySafe is to provide a full python interface to SAFE network, via the SAFE binaries.

The package (will) include an importable and installable module, as well as utilities to generate the bindings and hopefully eventually some example applications beyond ‘hello world’.

Designed to be minimally dependent; only cffi is not included in the base python install. We do however use features that currently require python 3.6+ . With enough interest, we could probably back port it, but this will depend on our chosen asynchronous implementation.

Currently in development, so it is quite raw.

- On linux, running upg_local.sh will pip update your local environment to the current pySafe state (any python prog can import pySafe, sym-linked to the development directory):
- sh ./upg_local.sh
- On Windows: upg_local.bat (note that at present you will have to compile your own binaries. Windows support is a ways off yet)

CHAPTER 1

example apps

we're written some very basic examples to show how pySafe can already be used to interact with the SAFE Network

some bacis usage examples can be seen here:

- simple chat example
- uploading some mutable data to safe and generating an xor link for it
- uploading some immutable data to safe

CHAPTER 2

yet missing examples

we'd like to add some other examples (but didn't manage to do so yet - so if you want to chime in and like our ideas (or just have your own idea of an example app) feel free to add it and just open a pull request!

some ideas you might want to test/implement to support this library

- thankscoin implementation in python
- SAFE dns name resolution in python
- alternative dns system (petname system) for safe in python
- alternative authenticator implementation (launcher style - local rpc/rest server that exposes the API functions through a simpler to contact interface than the c api provided by maidsafe
- simple file-synchronization app
- looking into https://github.com/home-assistant/home-assistant and (if possible) make it run with safe
- alternative messenger app interacting with Maidsafes messenger demo app

CHAPTER 3

Tests

Automatic tests are run through travis-ci and appveyor on windows, osx and linux

To add tests the test files that are meant to be executed need to be added to the tests directory and need to start with **test_*** or end with ***_test**

tests to be executed

the test functions that are meant to be executed must be named **test_*** and need to include an **assert** at the end

```
# super simple first test
def test_answers():
    assert inc(4) == 5
```

CHAPTER 4

Documentation

feel free to just improve this docs and open a pull request on github.

CHAPTER 5

Contributing

If you want to help us and work together with us on this project you can just join our telegram channel or get in touch with us in the dev forum or you just open a pull request / contact us on github :)

The more people get involved, the easier it is to utilize safe and the more powerful this library gets the better for freedom of humanity =) Together we work on creating a peoples internet where everybody wins and access to knowledge and freedom of speech are no privileges but universal rights of every human.