PyCon 2012 OAuth Sprint Page

Release 2012

Daniel Greenfeld and friends

Sep 27, 2017

Contents

1 Plan 3

2 Schedule 5
- 2.1 Monday 5
- 2.2 Tuesday through Thursday 5

3 General Coding/Documentation Notes 7

4 Getting your project onto python-guide.org (OAuth 1.0a) 9

We’re going to make Python’s OAuth story better!

- When: March 12 - 15, 2012
- Where: Santa Clara Hotel and around the world
- IRC channel: pycon-oauth
- Twitter hashtag: #pycon-oauth

CHAPTER 1

Plan

1. Get Oauth 1.0a working well. After that's done, work on implementing against select OAuth 2 specifications.
2. Clean up existing OAuth consumers and providers.
3. Document everything.
4. Increase test coverage on everything.
5. Add a page to http://docs.python-guide.org that is the Python OAuth story.
6. Describe a specification for a high level of quality in an OAuth-based project. Those who meet this specification get their project listed on the forthcoming http://docs.python-guide.org page.

CHAPTER 2

Schedule

Monday

OAuthLib and general cleanup

- Idan Gazit and a select group of others work to get OAuthLib to a state where consumer and provider authors can use this new library.

  - **Warning:** I can't put up a wall around Idan, so I'm going to ask that people refrain from distracting him until OAuthlib is ready to go.

- Create OAuthLib documentation and tests. **I'll be putting up rules for pull requests for this shortly**.
- Authors of existing OAuth Consumer and Provider tools can:
  - close out existing bugs.
  - fix documentation.
  - increase test coverage.
  - Create turnkey example projects.
- Describe the formal specification for getting projects into the forthcoming http://docs.python-guide.org OAuth page.
- Describe the rules necessary for registering an example implementations on the forthcoming http://docs.python-guide.org OAuth page.

Tuesday through Thursday

OAuthLib should be ready to go. In which case:

- Authors of existing OAuth Consumer and Provider tools can migrate/port/branch their projects over to OAuthLib.
- Rules forthcoming for getting these listed on the guide page, but we will need Consumer and Provider example implementations against OAuthLib for all Python frameworks:
  - Django
  - Flask
  - Pyramid
  - Web2py
  - Aspen.io
  - Cherrypy
  - Any that I've missed

CHAPTER 3

General Coding/Documentation Notes

It's going to take a little bit of time to get the formal rules up so here's a rough list of what we want:

- **Security is hard**. So write the simplest, cleanest, most documented code you can. Take your time and get it right, not just done.
- Unittest: Even though it can be tricky to test all pieces, get as many tests in as possible. Don't just try to get up your code coverage numbers either, actually test the code.
- Doctest: Again, tricky to do, but this is a great usecase for doctests. Your doctests will serve as working documentation for people examining your code. If your doctests fail for someone that means your doctests.

CHAPTER 4

Getting your project onto python-guide.org (OAuth 1.0a)

Warning: Draft - this will change over the course of the next few days.

The following are the base requirements for geting your OAuth provider or consumer project listed on python-guide.org:

- Easy to implement
- Good code smell. That means:
  - Clean code that has no more complexity than necessary.
  - Comments
- A working tutorials ideally created from doctests. This is the perfect usecase for doctests.
- Unittest coverage > 80%
- Complete and accurate documentation
- Sample implementations included in an examples directory