FB Graph API

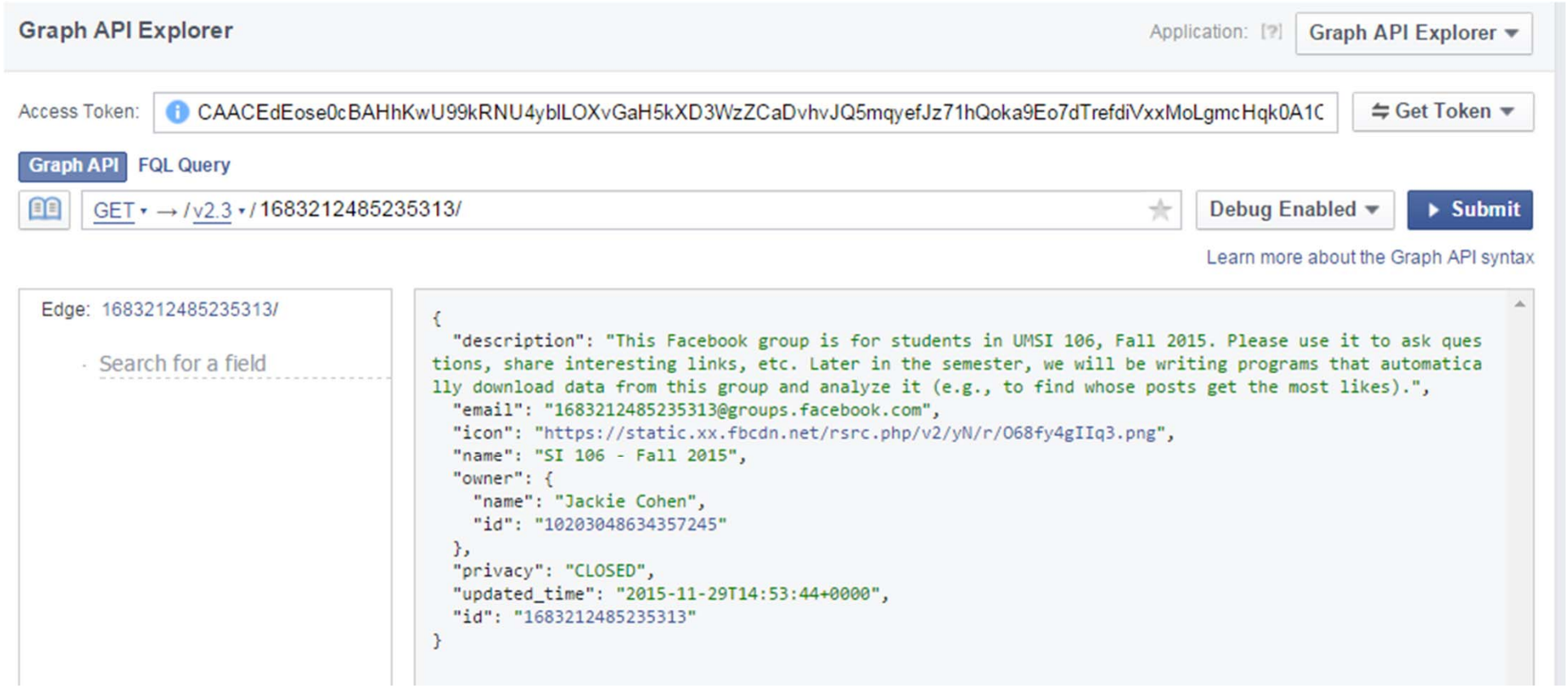
Outline

- The FB Graph API
- Authorization with oauth
- Our workaround for not running a webserver
- The facebook module
- Installing the facebook module

FB: A REST API

- But have to send an “authorization key” as a POST parameter
  - Details of POST parameters not important; we will use a module that takes care of that for us.
- https://graph.facebook.com/?<path>
- Some useful paths
  - /me gets info about current user
  - /me/feed gets their recent posts
  - /me/photos gets their recent photos
  - /245188182322906 gets info about our facebook group
  - 245188182322906/feed gets recent posts to our facebook group

Facebook's Graph API Explorer



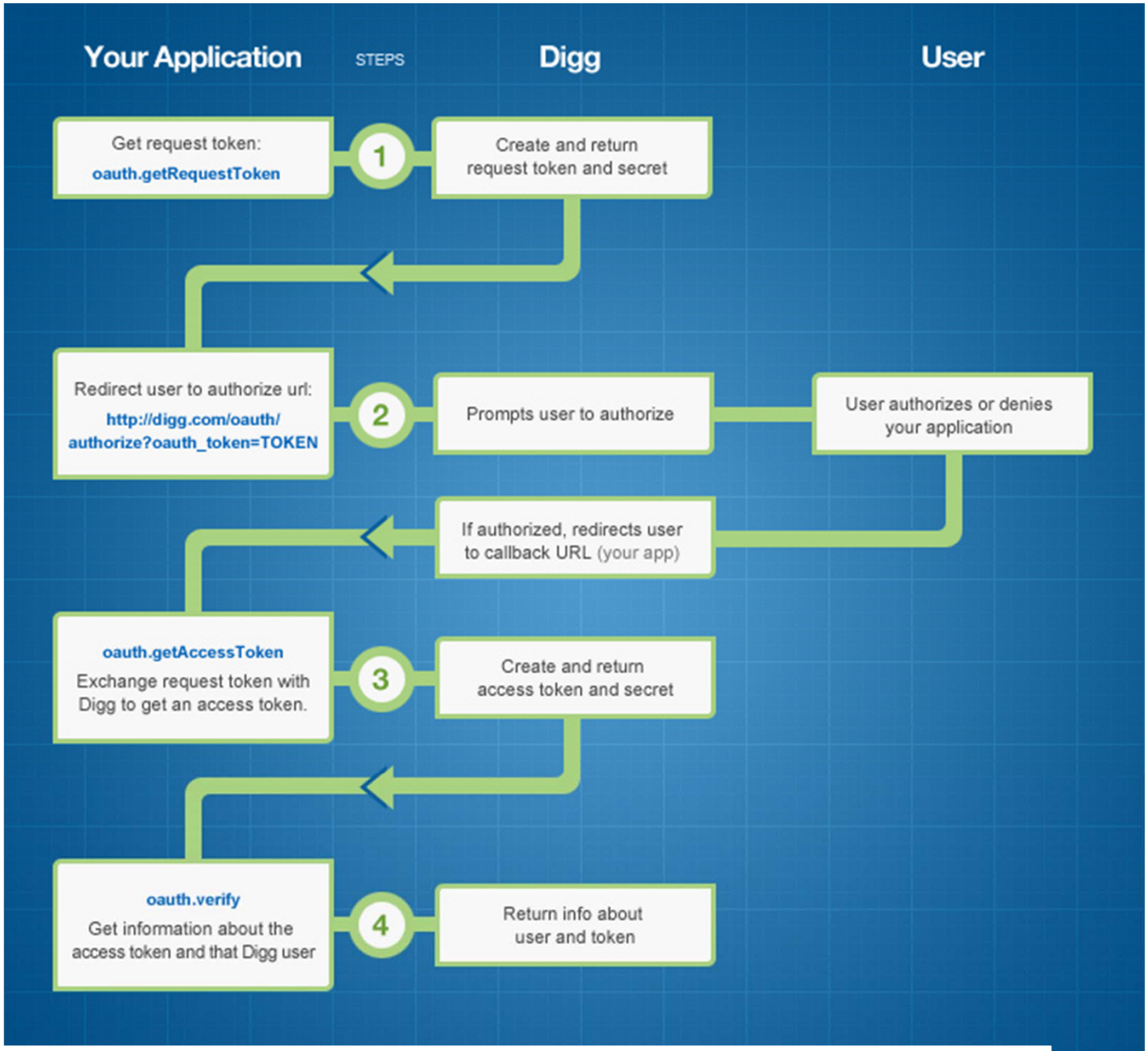
OAUTH

Purpose

- User interacts with site A
  - (or our python program)
- Authentication
  - Prove you own an account on site B
- (limited) Authorization
  - Give site A ability to access things on site B, acting on behalf of user
- Don't reveal your site B password to site A
  - maybe not even username

oAuth Terminology

- User: the person who has to log in
- Consumer (site A)
  - the application that will access data from elsewhere
  - e.g., your code running on your desktop
- Service Provider (site B)
  - The place the user has to log in to
  - e.g., FaceBook, Digg, Twitter



from http://developers.digg.com/authentication;
similarly https://developer.yahoo.com/oauth/guide/oauth-auth-flow.html

Step 0: User Connects to Consumer

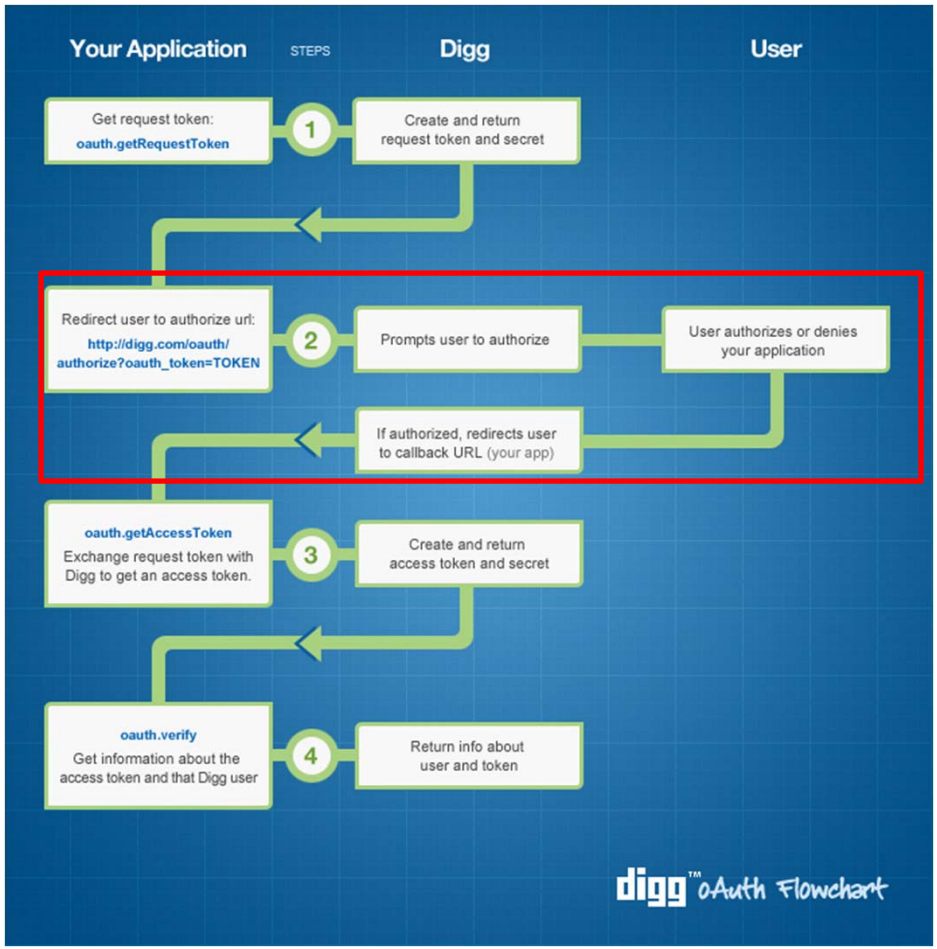
- Consumer decides it needs user to authenticate to Service Provider

Step 1: Consumer Connects to Service Provider

- Please create and return oauth_token and oauth_secret

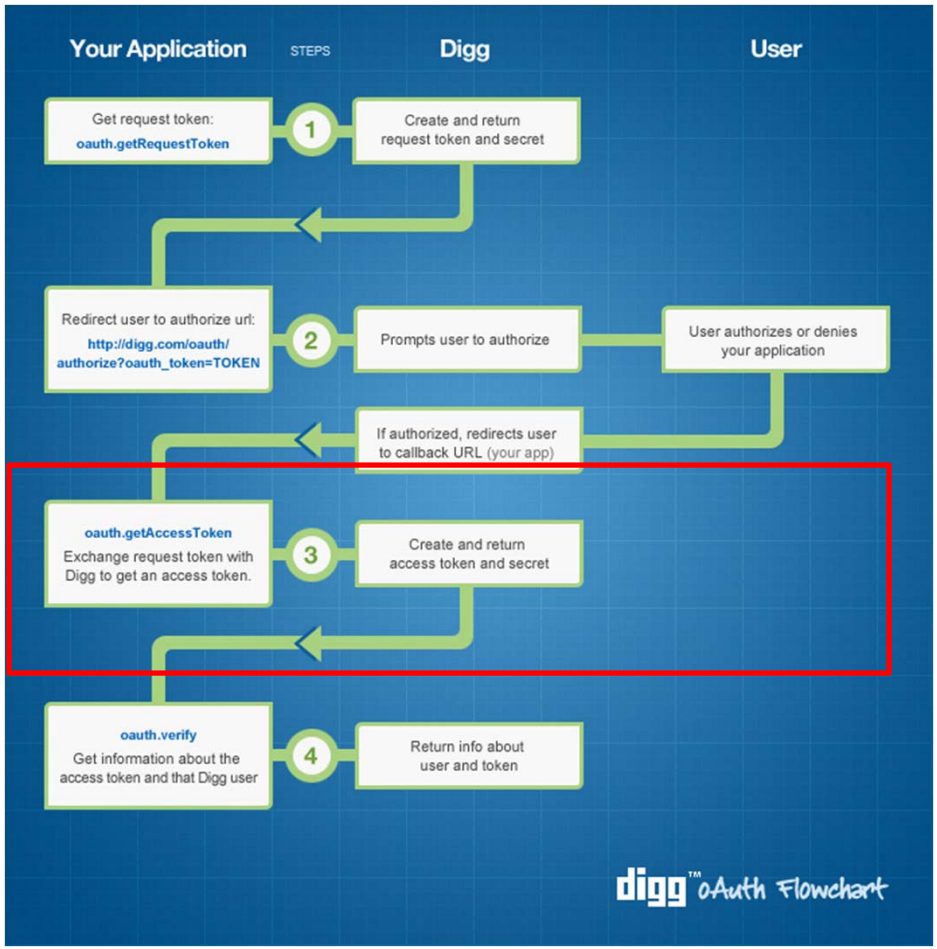
Step 2: User Authenticates to Service Provider

- Redirect User to Digg URL
  - Encode within that Digg URL a URL at your site to redirect to after user authenticates
- When Digg redirects user to your site
  - authorization code also sent as query parameter



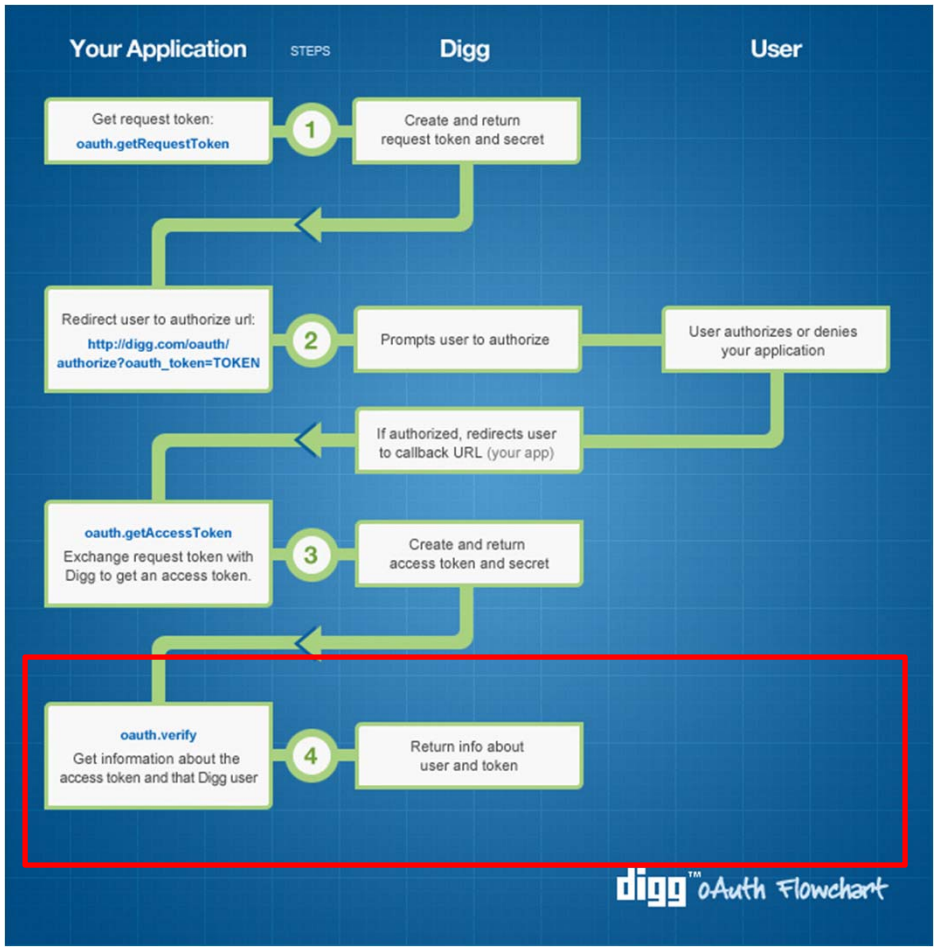
Step 3: Consumer gets a token

- Consumer sends auth code to Service Provider
  - With some info about the Consumer (application)
- Get back an access token



Step 4: use access token

- Consumer sends access token along with requests for data
  - Digg keeps track of access tokens it has issued
  - Digg decides permissions based on who the access token was for



The Problem For Us

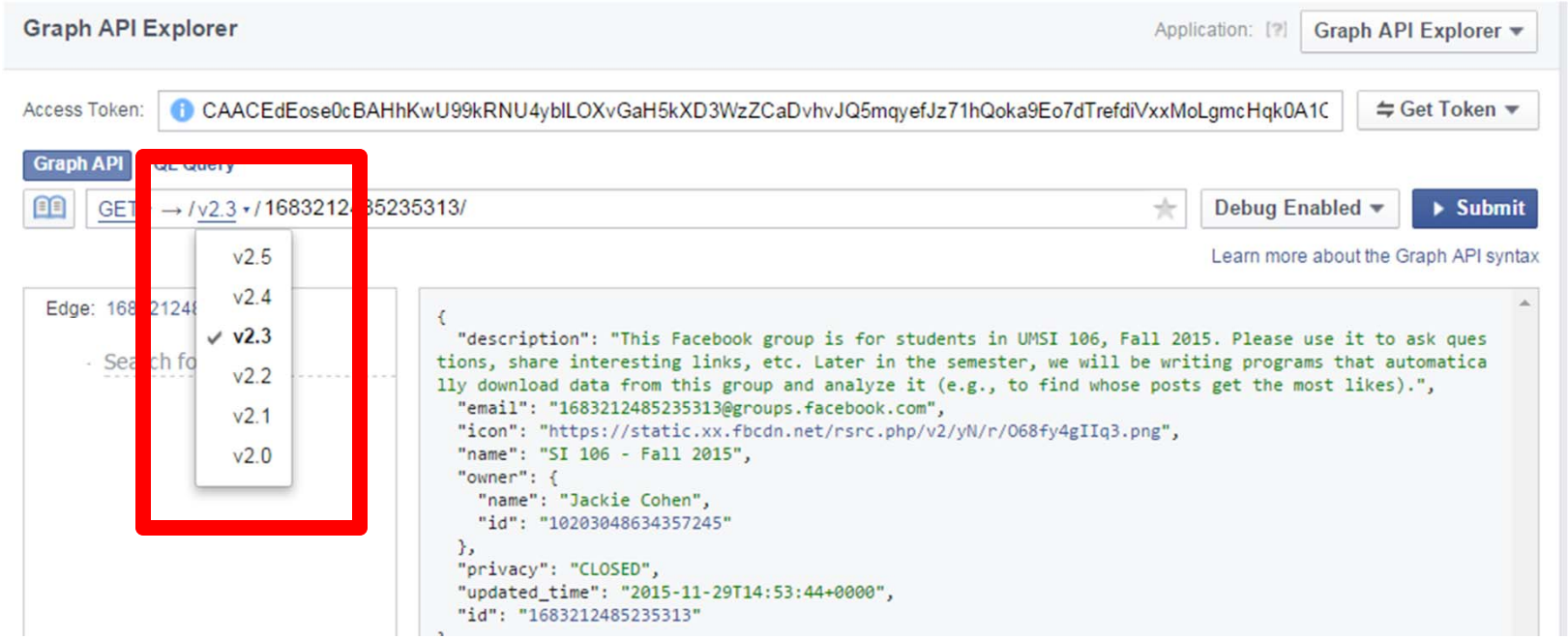
- We are not running a web server
- The step where service redirects user to our site, passing a token, won't work for us

Our Workarounds

- For FB
  - Use the FB developer tool to get the token manually
  - Paste it into your code (or terminal window)
- For Twitter (Wednesday)
  - Open a web browser for the login; copy and paste token from the browser
- For your project
  - Read the API documentation for the particular service
  - If oAuth required, do something similar to what we've done for FB or Twitter

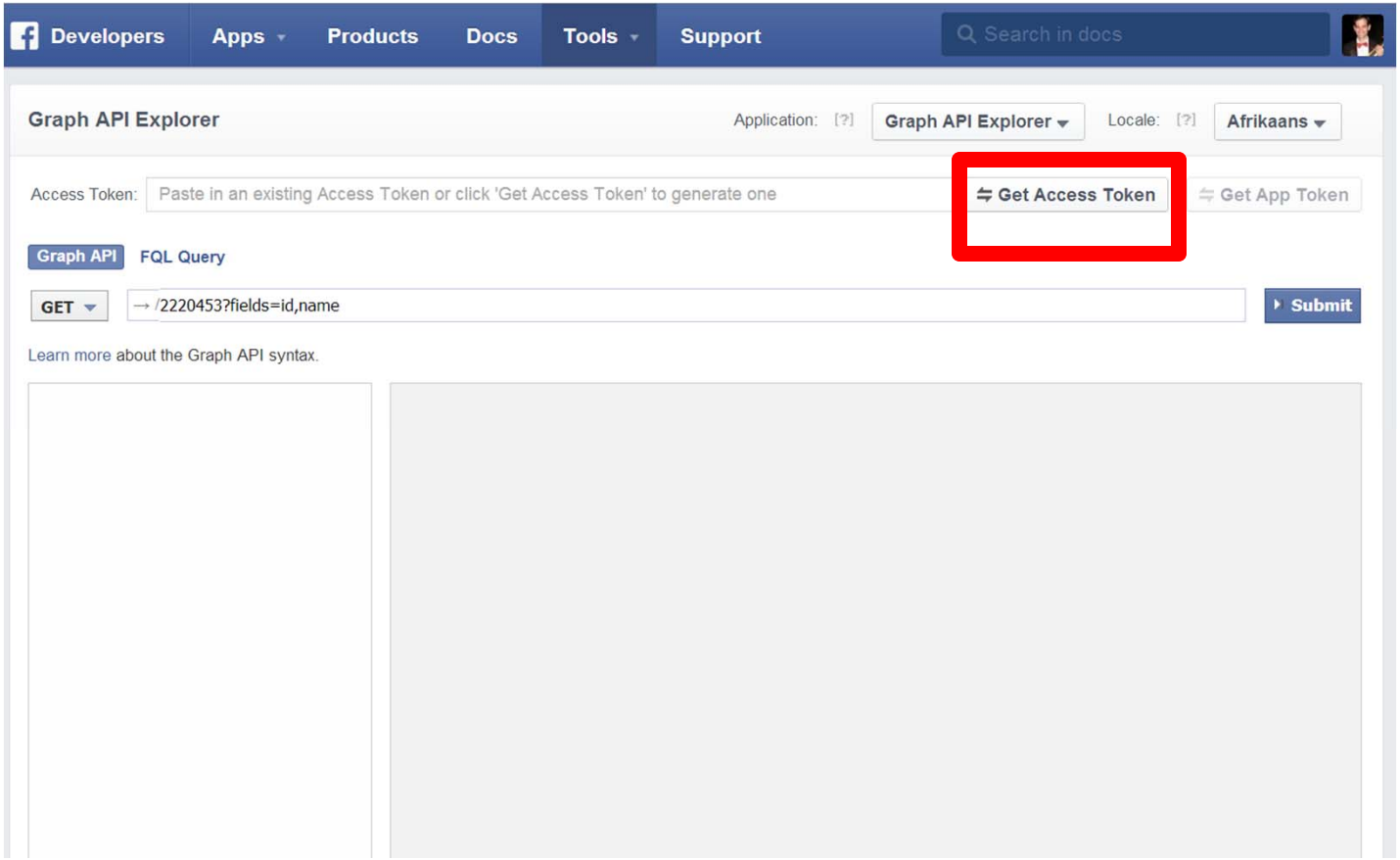
Getting the user_groups permission

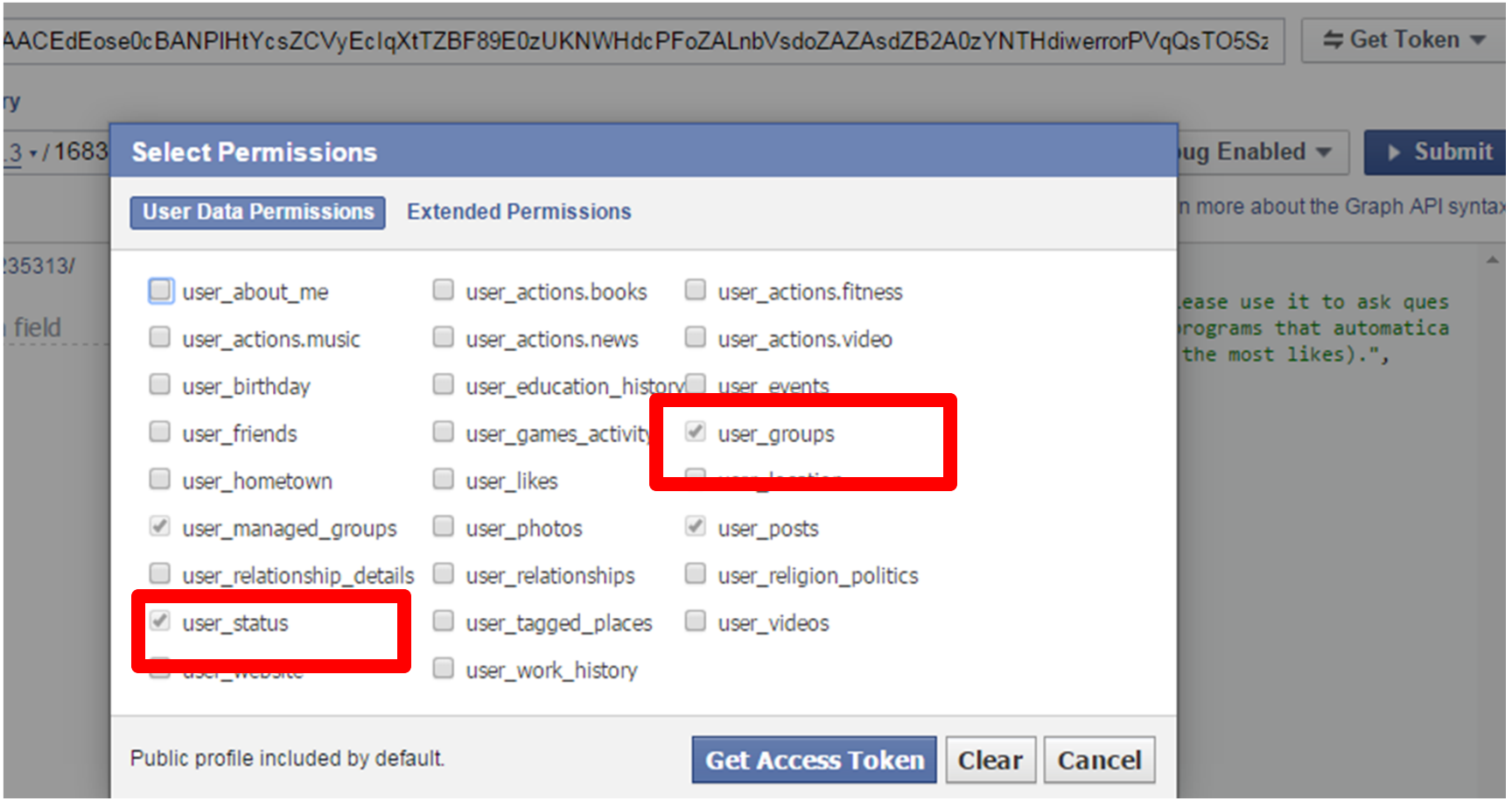
- Now deprecated; but available in version 2.3 of the FB API



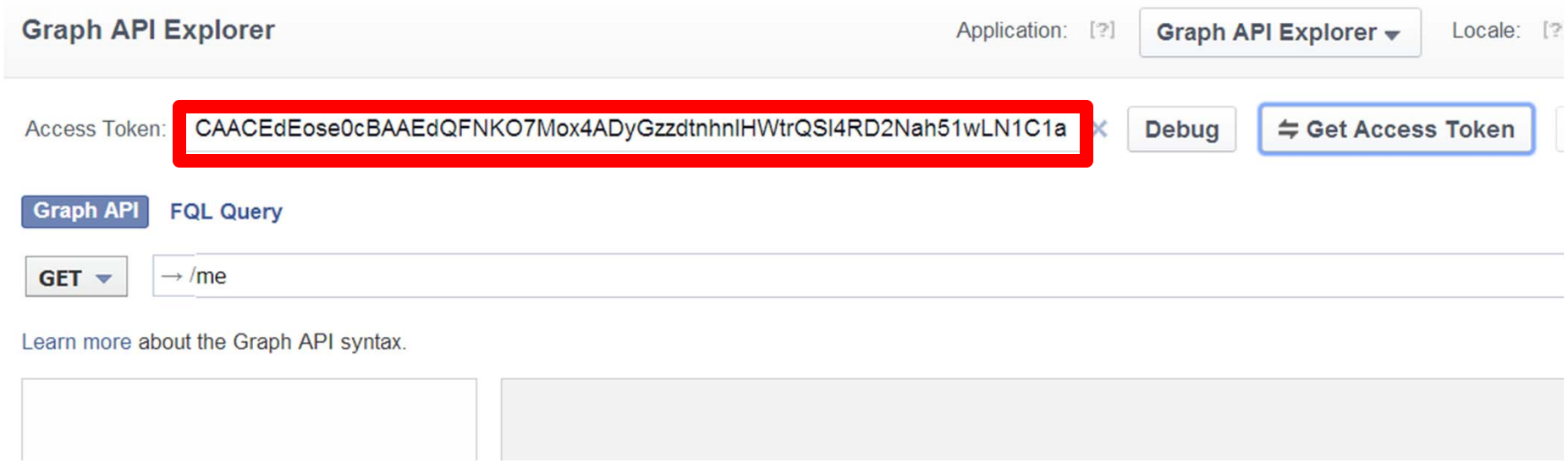
Getting the Token

- https://developers.facebook.com/tools/explorer




AACEdEose0cBANPlHtYcsZCVyEcIqXtTZBF89E0zUKNWHdcPFoZALnbVsdoZAZAsdZB2A0zYNTHdiwerrorPVqQsTO5Sz
Get Token
Select Permissions
User Data Permissions
Extended Permissions
user_about_me
user_actions.books
user_actions.fitness
user_actions.music
user_actions.news
user_actions.video
user_birthday
user_education_history
user_events
user_friends
user_games_activity
user_groups
user_hometown
user_likes
user_managed_groups
user_photos
user_posts
user_relationship_details
user_relationships
user_religion_politics
user_status
user_tagged_places
user_videos
user_work_history
Public profile included by default.
Get Access Token
Clear
Cancel
Submit
more about the Graph API

Cut and paste the access token



See fboauth.py for sample code

The facebook-sdk module

- Defines class GraphAPI
- __init__ takes an auth key
- Method get_object() takes a string
  - String is in format that FB defines
    - “/me” gets info about current user
    - “/me/feed” gets their recent posts
  - fimit parameter says how many items to retrieve
  - fields parameter says which fields of the items to retrieve
  - get_object method does a lot of stuff behind the scenes
    - Calls requests.get
    - Passes the auth_key in the way FB wants
    - Gets the response
    - Calls json.loads on it
    - Returns a dictionary

But You Have To Get Your Computer Prepped to Use It

- Install the facebook-sdk module
- Install pip [already done]
- Use pip to install the requests module [already done]

Use pip to Install requests (should have done this already)

- On Mac
  - pip install requests
- On Windows
  - /c/Python27/Scripts/pip install requests
- Note: if you get permission errors during installation, try running the install commands with a prefix of sudo (super user do)
    - E.g., sudo pip install requests

Installing the FB module

- pip install facebook-sdk
- On Windows
  - /c/Python27/Scripts/pip install facebook-sdk
- Note: if you get permission errors during installation, try running the install commands with a prefix of sudo (super user do)
    - E.g., sudo pip install faceook-sdk

Now fboauth.py Should Run!

- Note that the get_object() method does a lot of things that we've done manually before
  - Calls requests.get()
  - Reads the response as text
  - Calls json.loads()
  - What you get back is a python dictionary

Exercises

- Print out the text of all the messages returned in the feed
- Print out the names of everyone who left a comment on any of the posts