MY OAuth DIY

Author: LIANG Yaohua Student no. 201301008
Client demo App : http://www.liangyaohua.com/client/
Authorization Server : http://www.liangyaohua.com/oauth/
Resource Server : http://www.liangyaohua.com/openid/

Test MyOpenID account:
Email: gert.l.mikkelsen@alexandra.dk
Password : 123

Authorization Flow

- A client app must be authorized on the Authorization Server, then get an appID (Access Token), this will be used for authorizing this app to use MyOpenID as an login approach. If the client app haven't been authorized by the Authorization Server, it will not be able to use MyOpenID.

- An authorized app (our demo app) has a appID and a specific url, which should be matched when request to use MyOpenID.

- On the Client App, user could choose using MyOpenID as a login option, this will redirect to MyOpenID login page (if client app isn't authorized, request will be rejected).

- Users are supposed to have a MyOpenID account, if not, they can create one on the signup page (http://www.liangyaohua.com/openid/signup.php/)

- Users enter MyOpenID account information on the login page, if it success, it will redirect to the client app page, and fill in user's information which was retrieved from MyOpenID (Resource Server)

I implemented this flow just like what facebook login does, first, the app need to be authorized, which means get an appID and its url be registered. Then this app will be able to retrieve users' information from this 3rd party resource server.

Threats:

- Code injection : the sql query should be improved to prevent this (I'll think about how to make it later)
- Access Token is easy to get, this could cause Man-in-the-middle attacks

It will be easy to implement with OpenID instead of MyOpenID, I think just need to registered the client app, and add some code into it. It will be better, maybe ☺