Authenticating Using WeChat OAuth With the Ayla User Service

05/18/2017

Introduction

Ayla's Mobile SDK for Android release 5.6 has support for WeChat OAuth. WeChat login for third-party applications operates similarly to the OAuth 2.0 standard. A known limitation is that only a single application is supported for WeChat OAuth.

Before implementing WeChat login in your app, you'll need to register for a developer account on their developer portal, have your app approved, and obtain the corresponding App ID and App Secret. Login functionality will be available after your application has been approved.

Note: Currently, only the original login method is allowed for WeChat login via mobile application. Therefore users must install the WeChat App in their Androi device.

The following changes are needed for integrating WeChat OAuth in your app.

1. Configure on Ayla Dashboard

1. Sign in to the Ayla Dashboard with `OEM:Admin` or higher access account

2. Navigate to `OEM Profile|OAuth Credentials`, and create a Wechat OAuth credential. Fill the Client ID and Client Secret fields with Wechat App ID and App secret you see in Wechat developer portal.

2. Wechat SDK

Ayla sdk build file already has Remote WeChat library dependency included. Notice the line

compile 'com.tencent.mm.opensdk:wechat-sdk-android-without-mta:+' in build.gradle for aylasdk library.

Note: Since this is included in the sdk no need to include in the app build file.

3. Integrating

The following changes are needed for integrating WeChat oAuth in your app

1. Create a wxapi directory in your package and add a WXEntryActivity class inherited from Activity.
For example, if the app package is com.aylanetworks.aura add a new package called wxapi and a new activity called as WXEntryActivity under that package

2. You can directly use the file WXEntryActivity distributed with Aura code or in case you want to create your own class do the following.

Have your Activity Implement the IWXAPIEventHandler interface. In the onCreate method of that Activity do the following

```
public void onCreate(Bundle savedInstanceState){
    super.onCreate(savedInstanceState);

    api = WXAPIFactory.createWXAPI(this,
Constants.WECHAT_APP_ID, false);
    api.handleIntent(getIntent(), this);

    finish();
}
```

Requests sent by WeChat will be called back by onReq. Responses from WeChat will be called back by onResp.

The overridden method onResp has the token needed by our SDK method AylaWeChatAuthProvider.activityDidAuthenticate

Make Sure in the onResp method the following code is added
```
public void onResp(BaseResp resp) {
   switch (resp.errCode) {
    case BaseResp.ErrCode.ERR_OK:
     switch (resp.getType()) {
       case ConstantsAPI.COMMAND_SENDAUTH:
```

```
        SendAuth.Resp
sendResp = (SendAuth.Resp) resp;
            AylaWeChatAuthProvider.activityDidAuthenticate(sendResp.code);
            break;
        }
}
```

3. Now add an exported parameter for it (set to True) in the manifest file:
```
 <activity
        android:name=".wxapi.WXEntryActivity"
        android:exported="true">
```

4. Next, call Wechat sign-in API (usually from your sign in page). Here is the sample code from Aura:

```
AylaOAuthProvider aylaOAuthProvider = new
AylaWeChatAuthProvider(WECHAT_APP_ID);
AylaNetworks.sharedInstance().getLoginManager().signIn
(aylaOAuthProvider,
        MainActivity.SESSION_NAME,
        new Response.Listener<AylaAuthorization>() {
        @Override
        public void onResponse(AylaAuthorization response) {
                // Cache the authorization
                finish();
        }
        },
        new ErrorListener() {
        @Override
         public void onErrorResponse(AylaError error) {
                //Log this error or Show a Toast message
    }
});
```