AEM Mobile: Setting up Google as an Identity Provider

Requirement:

Prerequisite knowledge
- Understanding of AEM Mobile

Required Products
- AEM Mobile
- Google Account

Generating the client ID and secret

To integrate with the Google OAuth service, you will need to generate a pair of client ID and secret with the right configurations.

1. Navigate to the Google developer portal.
2. If you have already logged in, continue to the next step. Otherwise, you will be prompted to sign in with a Google account.
3. From the Google developer portal, select the "Credentials" option from the left rail.
4. Under the "Credentials" tab, select the "Create credentials" button.

5. From the dropdown, select "OAuth client ID"
6. When creating the client ID, select "Web application" as the application type:

   a. Enter the desired name for this application
   b. Enter the authorized redirect URIs, this is also found in the Identity Provider:
      https://es.publish.adobe.com/oauth2
   c. Select the "Create" button
7. When successfully created, the client ID and client secret will be displayed in the next screen.

Adding Google as an Identity Provider

Now that you have the Google OAuth client ID and secret, you can set up Google as an Identity Provider in the AEM Mobile On-Demand Services. Below are the necessary configurations:

1. **Authorization Grant Type** – choose either Authorization Code or Implicit.
2. **Token Endpoint** – enter the following Google OAuth token URL:
   
   https://accounts.google.com/o/oauth2/token

3. **Client Secret** – the Google client secret that was generated previously (see: "Generating the client ID and secret" earlier in this article).
4. **Authorization Endpoint** – enter the following Google OAuth authorization URL with the necessary parameters:
   
   
   b. The authorization URL allows for more optional parameters, for the full list, please refer to the Google documentation: [Using OAuth 2.0 for Client-side Web Applications](https://developers.google.com/accounts/docs/OAuth2WebServer).
5. **Client Identifier** – the Google client ID that was generated previously (see "Generating the client ID and secret" earlier in this article).
6. **Access Token Scope** – enter the desired access that the token will need, such as email, profile.
Granting access based on Google user

After setting up Google as an Identity Provider in AEM Mobile, this will enable users to log in with their Google account. To entitle a Google user to a collection, you will need to set up an entitlement service with the get entitlement request.

1. After a successful login with Google, the AEM Mobile Runtime will send an access token to the entitlement service in the following POST data field:
   - authToken

2. Using the following Google API, you can use the access token to pull the user profile from Google:

3. If you have entered "email" and "profile" in the access token scope previously (see: “Adding Google as an Identity Provider” earlier in this article), then you should see the basic Google user profile and email information.
   - Optionally, you can cache the access token and email so that you can limit the number of requests in step #2 and increase performance. For example, cache the access token and email for up to 24 hours so that the email address is only requested once a day

4. Depending on your entitlement service setup, you can query the entitlement server using the Google user email for the list of entitled product IDs.
Sample Code
The following is an example for an entitlement service written in PHP:

```php
// http://<YOUR_ENTITLEMENT_SERVER>/services/index.php

// one of the many ways of filtering special characters,
// there are also more sophisticated 3rd party libraries that do this
function escapeURLData($dataToEscape) {
    if(get_magic_quotes_gpc()) {
        $dataToEscape = stripslashes($dataToEscape);
    } else {
        $dataToEscape = strip_tags($dataToEscape);
    }
    return $dataToEscape;
}

// attempts to read the access token POST from the AEM Mobile Runtime
$authToken = isset($_REQUEST['authToken']) ? escapeURLData($_REQUEST['authToken']) : null;

// continues only if the access token is provided
if ($authToken) {
    // trade in the access token for user profile using the Google API
    // decode the JSON string in order to access the data
    $googleUser = json_decode($response, true);
    // the Google user email is in the "email" property
    $googleUserEmail = $googleUser['email'];

    // TODO: using the Google user email,
    // query the entitlement server for a list of entitled product IDs
}
?>```
Legal Notice

The contents of this guide is subject to the Terms of Use, is furnished under license and may be used or copied only in accordance with the terms of such license. No part of this guide may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, recording, or otherwise, without the prior written permission of Adobe Systems Incorporated.

Please note that the content in this guide is protected under copyright law even if it is not distributed with software that includes an end user license agreement.

The content of this guide is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Adobe Systems Incorporated. Adobe Systems Incorporated assumes no responsibility or liability for any errors or inaccuracies that may appear in the informational content contained in this guide.

Please remember that existing artwork or images that you may want to include in your project may be protected under copyright law. The unauthorized incorporation of such material into your new work could be a violation of the rights of the copyright owner. Please be sure to obtain any permission required from the copyright owner.