Purpose Of Guide

This guide details the step-by-step procedures required to enable a cloud service client to retrieve tenants registered on the cloud service by using a restful API. This restful API is part of the cloud service and uses the Open ID Connect protocol for authentication. It also requires the use of Bearer tokens and returns responses in JSON.
The steps below explain all the instructions to be followed to authorize a client to access this restful API endpoint.

The most commonly used approaches for authorizing a client are the authorization code flow and implicit flow. This guide uses the authorization code flow.

Prerequisites

1. A client already registered on the cloud service with the following information:
- Client name: EcmaApp
- Client ID: a.unique.client.id.string
- Client secret: a.unique.client.secret.string
- Redirect URI: https://ecma-app.com/app/callback
- Authentication method: client_secret_basic
- Scope value: list_tenant

2. The cloud service discovery endpoint that returns information about the open-id configuration and the required endpoints.
3. The restful API endpoint for retrieving the list of organizations registered on the cloud service.
4. An app like Powershell/Terminal on mac/windows or postman for making cURL calls

Step 1: Querying service discovery endpoint

Send a GET request call to the cloud service discovery endpoint to get the open-id configuration information and the endpoints required for the remaining steps.

Sample curl
curl -I --request GET
'https://cloud-ecma.herokuapp.com/.well-known/openid-configuration'

The response returned should look like this:


```
→  ~ {
   "authorization_endpoint":"https://cloud-ecma.herokuapp.com/api/v2/idp/authorize",
   "subject_types_supported":[
      "public",
      "pairwise"
   ],
   "jwks_uri":"https://cloud-ecma.herokuapp.com/api/v2/idp/keys",
   "response_types_supported":[
      "code"
   ],
   "token_endpoint":"https://cloud-ecma.herokuapp.com/api/v2/idp/token",
   "id_token_signing_alg_values_supported":[
      "HS256"
   ],
   "issuer":"https://cloud-ecma.herokuapp.com/"
}
```


Step 2: Authenticating the client

Send an authentication request to the authorization server to authorize the client.
The endpoint of the authorization server can be retrieved from the json response from step one using the **authorization_endpoint** field name**.** The response type value is also found in that json response with field name **response_types_supported**. The other request param values should be available to you after the client is registered on the service.

Sample curl

curl -I --request GET

'https://cloud-ecma.herokuapp.com/api/v2/idp/authorize?response_type=code&scope=openid%20list_tenants&client_id=a.unique.client.id.string&redirect_uri=https%3A%2F%2Fecma-app.com%2Fapp%2Fcallback'

The authorization endpoint accepts GET or POST method calls with the format `application/x-www-form-urlencoded`. Using the GET method requires the client to add the request params to the query component of the URL. Using the POST method requires form serialization. We use the GET method in this example.

Upon successful authentication, the response below is returned.
The code value returned in the Location field should be used in the next step.

```
➜  ~ HTTP/1.1 302 Found
Server: Cowboy
Connection: keep-alive
Location: https://ecma-app.com/app/callback\?code\=f1a8edc63933ff54
Vary: Accept
Content-Type: text/plain; charset=utf-8
Content-Length: 77
Date: Fri, 10 Jun 2022 18:50:29 GMT
Via: 1.1 vegur
```

Step 3: Fetching tokens

Send a token request to the authorization server to authenticate the client and obtain an ID Token and an Access Token.This is a POST method call.

The token request is made up of the following:

- The authorization code obtained after authenticating the client in step 2
- The grant type as specified by Section 4.1.3 of **OAuth 2.0** [RFC6749]
- The redirect uri as setup when registering the client
- A basic auth header with a base64 encoded value of the client id and client secret
- A content type header of value x-www-form-urlencoded

The token endpoint can be retrieved from the json response from step one using the **token_endpoint** fieldname**.**

Sample Curl

```
curl --request POST 'https://cloud-ecma.herokuapp.com/api/v2/idp/token' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--header 'Authorization: Basic YS51bmlxdWUuY2xpZW50LmlkLnN0cmluZzphLnVuaXF1ZS5jbGllbnQuc2VjcmV0LnN0cmluZw==' \
--data-urlencode 'grant_type=authorization_code' \
--data-urlencode 'code=b2f6f849d20c4a5c' \
--data-urlencode 'redirect_uri=https://ecma-app.com/app/callback'
```

A successful response returned is a json string as shown in the image below.

The access token and id token values should be validated as per the rules in Section 3.1.3.7 and Section 3.1.3.8 of the openid-connect-core documentation and also the validation rules of RFC 6749 (especially Sections 5.1 and 10.12). This validation is done to confirm that contents of the id token can be trusted. Once the validation is done we proceed to our last step.

Step 4 : Calling list tenant API

This is the last step in this process. At this point, the client has been authorized and has the token details required to access the list tenant API via an API call. This API call is a GET method call and uses the following values from the token response in the third step.

- token type: The token type in the response is Bearer. This means the GET call will need to use the authorization header of type Bearer.
- access_token: The value to be passed as the bearer token value in the authorization header of the GET request. This token is meant to be read and validated by the API.

Sample Curl

curl --request GET 'https://cloud-ecma.herokuapp.com/api/v2/tenants' \

--header'Authorization:Bearer

eb2c988a631394a58a4fe42131413e9660964543c3b83de5ee94babdc41c06c5'

The GET request returns a successful response as shown in the image below:

```
➜  ~ {
  "items": [
    {
      "id": "f313ecf6-9256-4afd-9d47-72e032ee81d0",
      "name": "The Qwerty Tenant",
      "enabled": true
    },
    {
      "id": "1e4d7438-0ebe-11e7-b131-c7b5bde6feed",
      "name": "The FooBar Tenant",
      "enabled": false
    },
    {
      "id": "5d92a310-0ee7-11e7-95e6-5f64824358de",
      "name": "The Another Tenant",
      "enabled": true
    }
  ]
}
```

Resources For Further Reading

- https://openid.net/specs/openid-connect-core-1_0.html