**Authentication via IBM Cloud service App Id and OAUTH**

We are presenting the Ancillary sales admin Tool by adding Authentication to backend and managing user specific data using IBM Cloud service APP Id.

This Cloud service provided by IBM has various features to easily add authentication, secure back ends and APIs, and manager user-specific data for web apps and mobile.

The users who all can access the ASAT tool should be created under IBM App ID first and they are able to login the ASAT tool.

We are securing the Rest API ‘s calls written at the at backend i.e. ASAT-API side using OAUTH and by calling the IBM cloud service App Id API’s to authenticate the user when login to the admin application.

For providing Authentication, we have configured various properties from APP Id like the tenantId, clientId and clientSecret at Backend side which is Admin API project in this case.

When the user login to the ASAT Web application, to Authenticate the user using App Id and OAUTH, we have called below mentioned API’s at backend side.

To get the Token using Basic authentication Scheme, When the Login request coming from ASAT Web application has “Basic” String in the Authorization Header, we are extracting the username and password info for the user by decoding this Basic token. And then using these user’s credentials as request body for the below API. We are encoding the Client Id and Client secret obtaining from App Id to the Authorization header along with basic string using Java 8 Base64 encoder.

1. **Calling /token POST REST API to generate the token when the user’s login to the web application.**

[https://appid-oauth.eu-gb.bluemix.net/oauth/v3/{tenantId}/token](https://appid-oauth.eu-gb.bluemix.net/oauth/v3/%7btenantId%7d/token)

Headers:

Content-Type": MediaType.***APPLICATION\_JSON\_VALUE***

Authorization: “Basic " + encode(clientId + ":" + clientSecret)

Request Body as JSON:

{

"grant\_type":"password",

"username": "username",

"password": "password"

}

Where tenantId, cleintId and ClientSecret are from App Id and we set as properties in our application.properties files.

When calling this API, we get the access\_token, id\_token, token\_type and expires\_in as response body. This token is of type Bearer and active for 1 hour and we make use of this access\_token to call the /introspect API to check the token validity for further request for accessing the other urls on admin page.

1. **We will call the below /introspect API to check if the access\_token is valid or not so that we can pass this access token in subsequent request i.e. for accessing other urls on admin tool.**

[https://appid-oauth.eu-gb.bluemix.net/oauth/v3/{tenantid}/introspect](https://appid-oauth.eu-gb.bluemix.net/oauth/v3/%7btenantid%7d/introspect)

Headers:

Content-Type" : MediaType.***APPLICATION\_JSON\_VALUE***

Authorization: “Basic " + encode(clientId + ":" + clientSecret)

Request Body as JSON:

{

“token”:"access\_token"

}

While calling this API will get the status of the token whether the token is active or has expired.

If the response body returns **“active”: true** we will pass this token access\_token for subsequent requests i.e. when the user will access the other links like Top sales, Out of stock and other Configuration related data.

Else will throw the Exception that “Token has expired”. Please Login again!