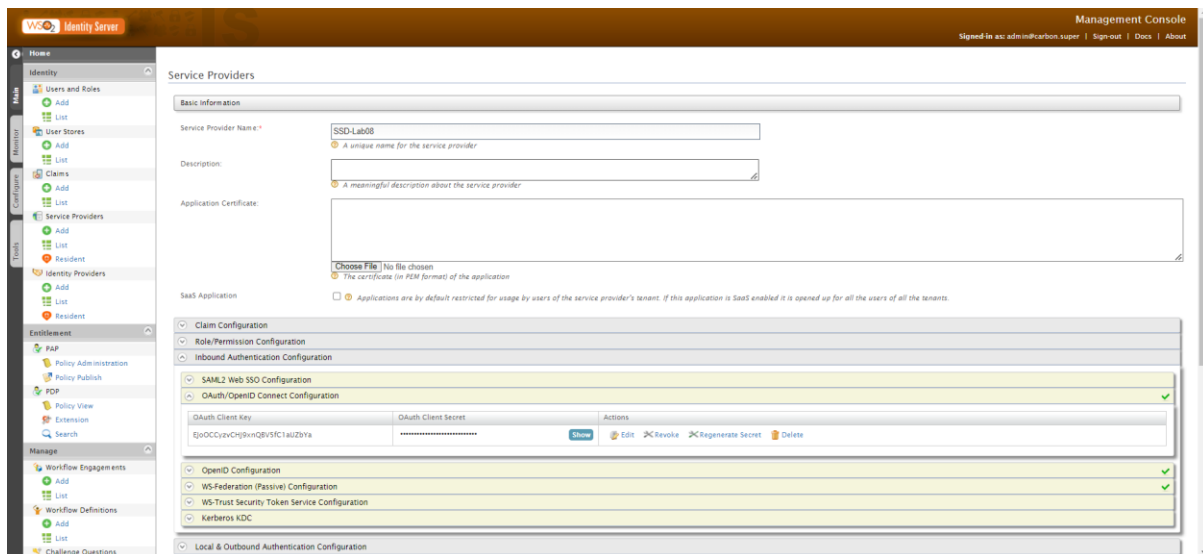


1)



Client ID - EJoOCCyzvCHJ9xnQBV5fC1aUZbYa

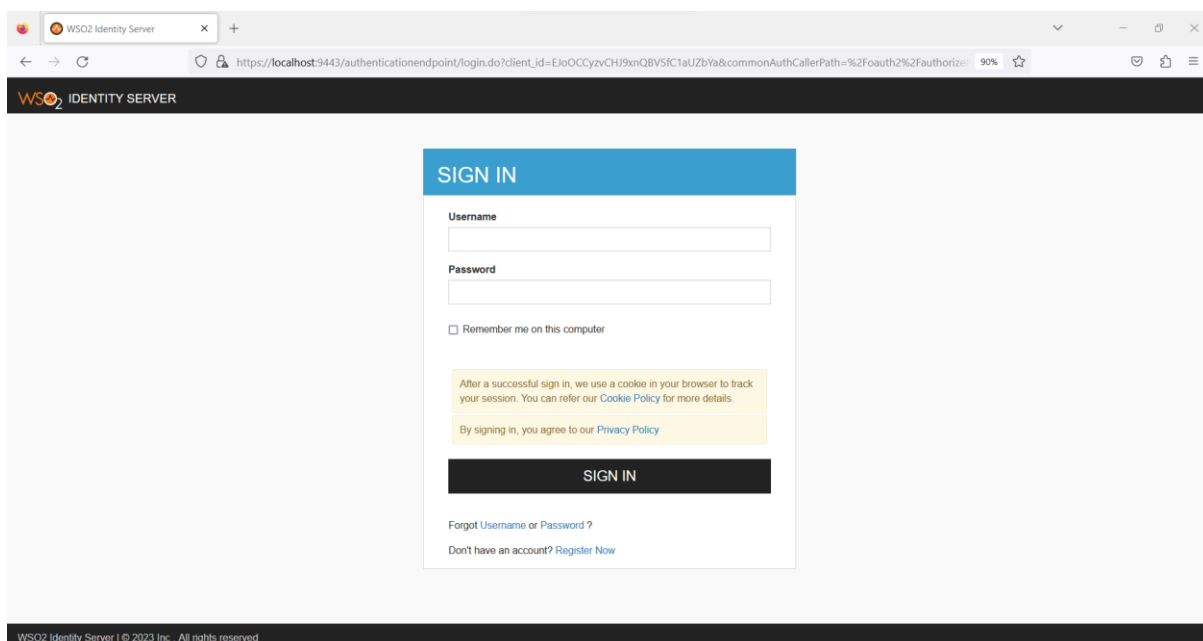
Client Secret - 4p1_ck8MsUAK0WTZxo6BJBbWMJEa

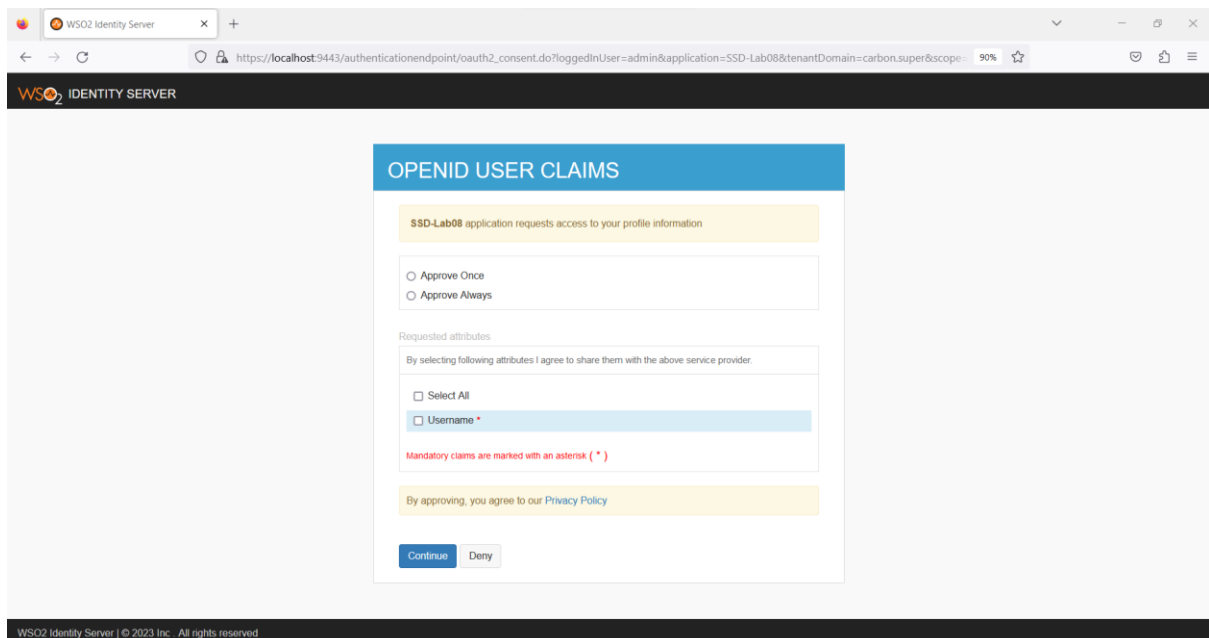
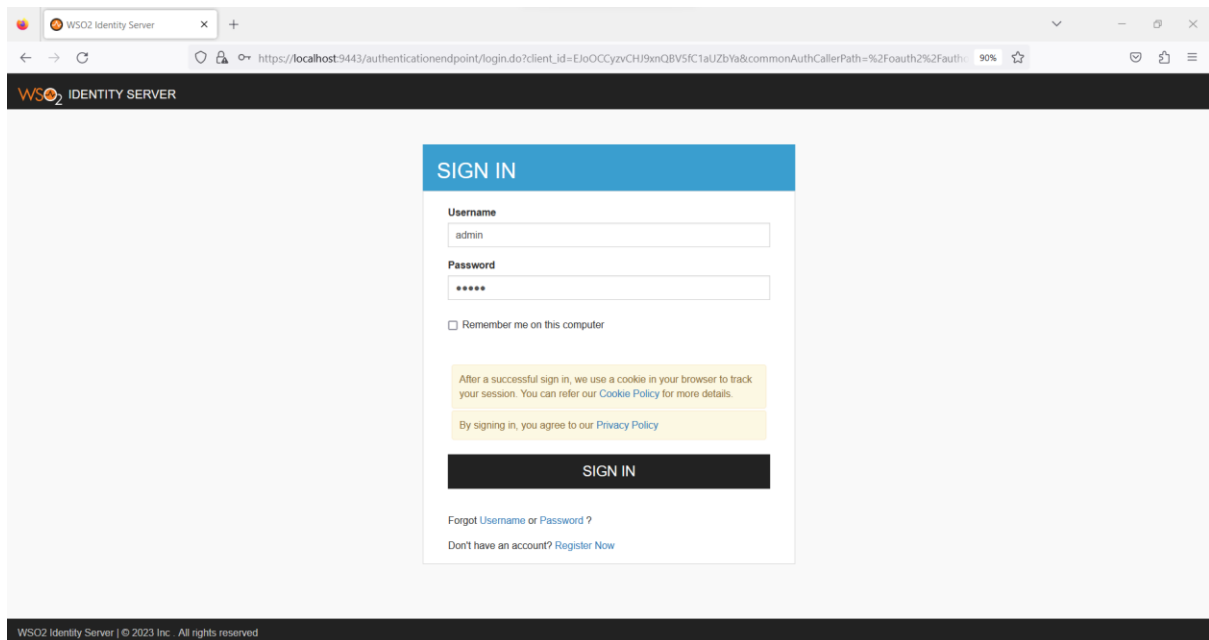
Redirect URL - https://localhost:8080/callback

2)

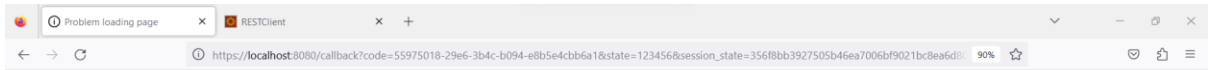
Request to get the access token -

https://localhost:9443/oauth2/authorize?response_type=code&client_id=EJoOCCyzvCHJ9xnQBV5fC1aUZbYa&scope=openid&state=123456&redirect_uri=https%3A%2F%2Flocalhost%3A8080%2Fcallback





Response - https://localhost:8080/callback?code=55975018-29e6-3b4c-b094-e8b5e4cbb6a1&state=123456&session_state=356f8bb3927505b46ea7006bf9021bc8ea6d80b78811317b88f6222072e305c6.-u3_BdAfaRlciM9VlxMvyw



Unable to connect

An error occurred during a connection to localhost:8080.

- The site could be temporarily unavailable or too busy. Try again in a few moments.
- If you are unable to load any pages, check your computer's network connection.
- If your computer or network is protected by a firewall or proxy, make sure that Firefox is permitted to access the web.

Try Again

Request to get the Open ID token –

POST request - <https://localhost:9443/oauth2/token>

Headers –

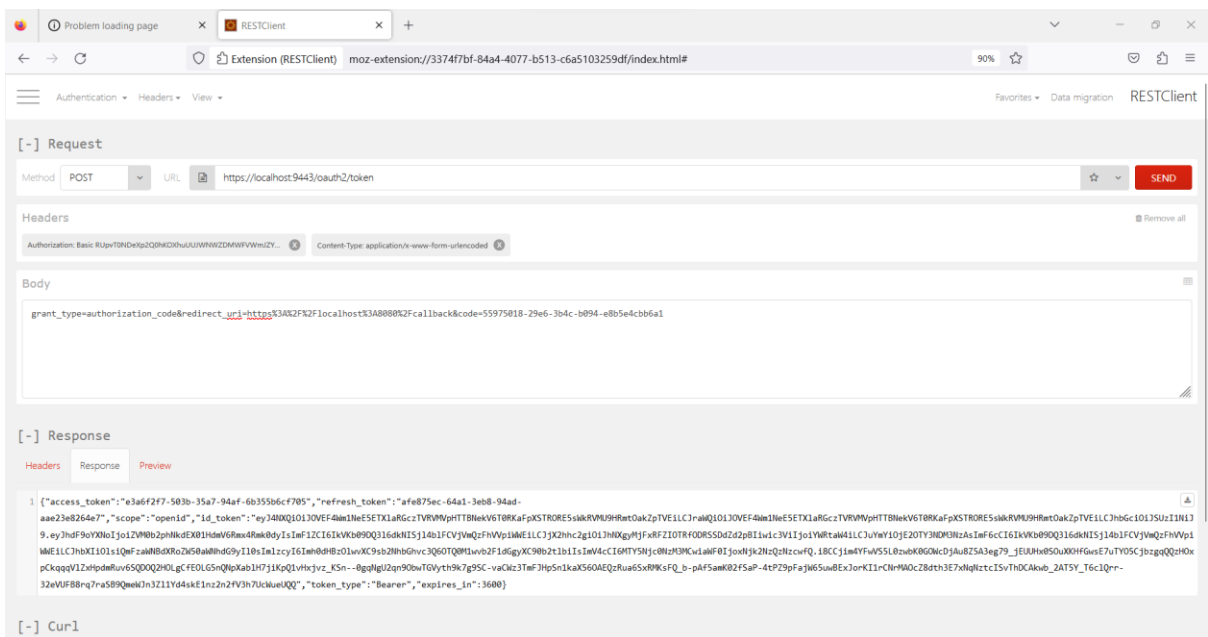
Authorization -> Basic

RUpvT0NDeXp2Q0hKOXhuUUJWNWZDMWFVWmJZYTo0cDFfy2s4TXNVQWswV1RaeG82QkpCYldNs
kVh

Content-Type -> application/x-www-form-urlencoded

Body -

grant_type=authorization_code&redirect_uri=https%3A%2F%2Flocalhost%3A8080%2Fcallback&code=55975018-29e6-3b4c-b094-e8b5e4cbb6a1



Response –

```
{
  "access_token": "e3a6f2f7-503b-35a7-94af-6b355b6cf705",
  "refresh_token": "afe875ec-64a1-3eb8-9ad-aae23e8264e7",
  "scope": "openid",
  "id_token": "eyJ4NXQiOiJOVEF4Wm1NeE5ETXlaRGczTVRVMVpHTTBNEkV6T0RkaFpXSTRORE5sWkrVMU9HRmtOakZpTVEiLCJraWQzOiJOVEF4Wm1NeE5ETXlaRGczTVRV\nMVpHTTBNEkV6T0RkaFpXSTRORE5sWkrVMU9HRmtOakZpTVEiLCJhbGciOiJSUzI1NiJ9.eyJhdF9oYXNo\nljoizVM0b2phNkdEX01HdmV6Rmx4Rmk0dylslmF1ZCI6IkVKb09DQ3I6dkNISjI4blFCVjVmQzFhVVpiW\nWEiLCJjX2hhc2giOiJhNXgyMjFhZjZlOTRfODRSSDd2d2pBliwic3ViljoieYWRTaW4iLCJuYmYiOiJ2OTY3ND\nWM1NeE5ETXlaRGczTVRVMVpHTTBNEkV6T0RkaFpXSTRORE5sWkrVMU9HRmtOakZpTVEiLCJhbGciOiJSUzI1NiJ9\nCl6MTY5Njc0NzM3MCwiaWF0IjoxNjk2NzQzNzcwfwQ.i8CCjim4YfWVS5L0zwbKOGOWCdjAu8Z5A3eg79\n_jEUUHx0SOuXKHfGwsE7uTYO5CjbzqgQQzHOxpCkqqqVLzXhpdmRuv6SQDOQ2HOLgCfEOLG5nQNpX\nablH7jiKpQ1vHXjxz_KSn--0ggNgU2qn9ObwTGvyth9k7g9SC-\nvaCWz3TmfJHpSn1kaX56OAEQzRua6SxRMKsFQ_b-pAf5amK02fSaP-\n4tPZ9pfajW65uwBExJorKI1rCNrMAOCz8dth3E7xNqNztclSvThDCAkwB_2AT5Y_T6clQrr-\n32evUFb8rq7raSB9QmeWJn3ZI1Yd4skE1nz2nfV3h7UcWueUQQ",
  "token_type": "Bearer",
  "expires_in": 3600
}
```

3)

Encoded	PASTE A TOKEN HERE
eyJ4NXQwOiJOVEF4Wm1NeE5ETXlaRGczTVRVMPvHTTBnekV6T0RkaFpXSTRORE5SkkRVMU9HRmtOakZpTVEiLCJraWQiOiJOVEF4Wm1NeE5ETXlaRGczTVRVMPvHTTBnekV6T0RkaFpXSTRORE5SkkRVMU9HRmtOakZpTVEiLCJhbGciOiJSUzI1NiJ9.eyJhdF9oYXNoIjoiaGVhbnB2bHkndEX0tHdmV6Rmx4Rmk0dyIsImFiICI6IklkbG90dDQlZkdNISIj14b1FCVjVmOzFhVWVpiWEiLCJj2xhc2giOiJhNXgyMjFfZFZIOtRfODRSSDdZd2pBIiwic3ViIjoiaWRtaW4iLCJuYmYyIjoE2OTY3NDM3NzAsImF6cCI6IklkbG90dDQlZkdNISIj14b1FCVjVmOzFhVWVpiWEiLCJhbGciOiJIeSIjOmFzaWNnbXRocW50aWNhdG9yIl0sIm1zc2cyIj6ImhbHBzOlwvXC9sb2Nhbg9vc3Q6OTQ0MTwvb2F1dGgyXC90b2t1biIsImV4CiC6MTY5Njc0bzMzMwIawWF0IjojaXNk2NzQzNzwcfQ.i8CCjim4YFWvs5L0zwbK0GOWcDJAu8Z5A3eg79_jEUUHx0SOUXKHfGwsE7uTY05CjbzgqQQzHOxpckqqqVLzxHpdmRu v6SQDQ2HOLgCfEOLG5nQnpXabIH7jiKp01vHXjvz_KSn--8ggNgU2qn90bwTGvYyth9k7g9SC-vaCWz3TmFJhpSn1kaX560AEQzRua6SxRMKSfq_b-pAf5amK02fSaP-4tP29pfajW65uwBEExJorKI1rCNrMAOCz8dth3E7xNqNZtcISvThDCAkwB_2AT5Y_T6clQrr-32eVUFB8qr7rasB9QmeWJn3Zl1Yd4skE1nz2n2fV3h7UcWueUQQ	
HEADER: ALGORITHM & TOKEN TYPE	
{ "x5t": "NTAxZmMxNDMyZDg3MTU1ZGM0MzEzODJhZW14NDN1ZDU1OGFkNjF1MQ" ,"kid": "NTAxZmMxNDMyZDg3MTU1ZGM0MzEzODJhZW14NDN1ZDU1OGFkNjF1MQ" ,"alg": "RS256" }	
PAYLOAD: DATA	
{ "at_hash": "eS4oja6GD_MGoveF1xFi4w", "aud": "EJo0CcYzvCHJ9xnQBVSfc1aUZbYa", "c_hash": "a5x221qDVH94_84RH7YwJa", "sub": "admin", "nbf": 1696743770, "azp": "EJo0CcYzvCHJ9xnQBVSfc1aUZbYa", "amr": ["BasicAuthenticator"], "iss": "https://localhost:9443/oauth2/token", "exp": 1696747370, "iat": 1696743770 }	
VERIFY SIGNATURE	
RSASHA256(base64urlEncode(header) + "." + base64urlEncode(payload), <div>Public Key in SPKI, PKCS #1, X.509 Certificate, or JWK string format.</div> <div>Private Key in PKCS #8, PKCS #1, or JWK string format. The key never leaves your browser.</div>)	

OpenID tokens consist of 3 segments separated by two dots.

- 1) Header – Declaring the type as JWT, Hash or keyed hash algorithm
- 2) Payload – useful user-specific information (issuer, subject, expiry date etc)
- 3) Signature- encoded header and payload separated by a dot and encrypted with a hash algorithm afterwards

As shown in above figure, x5t is the signature of the entire id token. When auth server has to deal with multiple id tokens, this x5t attribute is used to uniquely identify and search for each token. Algorithm is mentioned as RS256 which is asymmetric encryption. In the payload subject is the admin, because wso2 admin login was used for getting the token and the issuer is the token endpoint of the identity server. Signature is also available in an encrypted format.