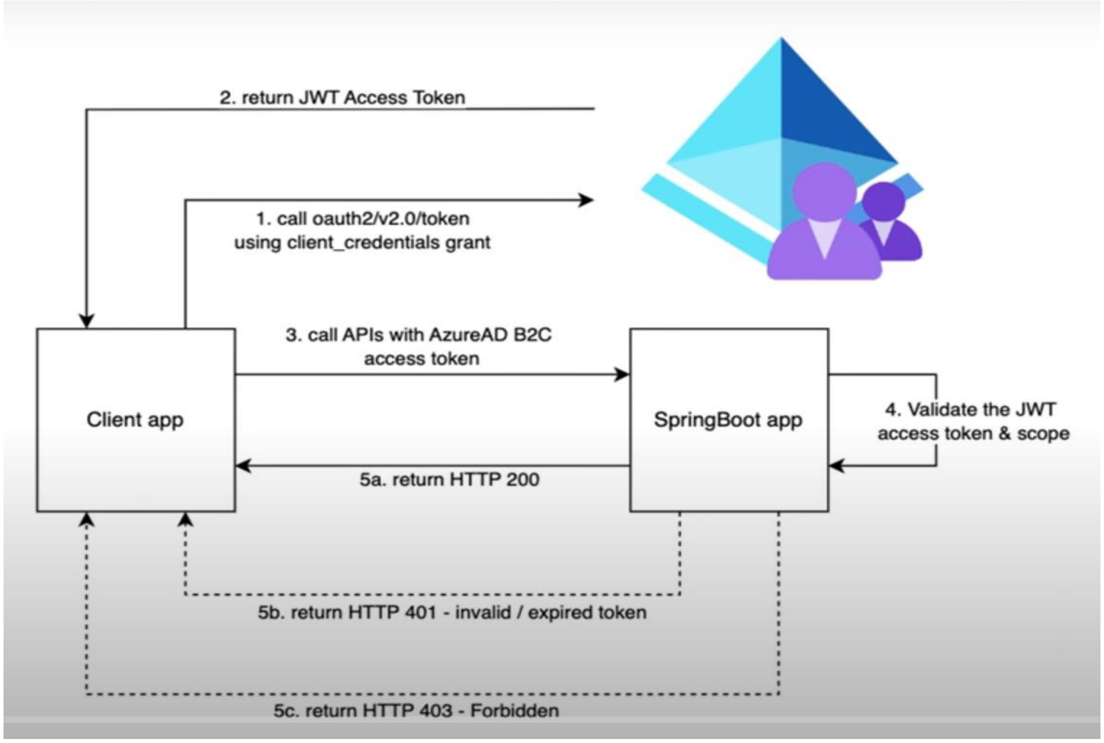
❖ Authenticate the Spring Boot API using Microsoft azure AD.

1. Application Flow –


2. return JWT Access Token
1. call oauth2/v2.0/token using client_credentials grant
3. call APIs with AzureAD B2C access token
Client app
SpringBoot app
4. Validate the JWT access token & scope
5a. return HTTP 200
5b. return HTTP 401 - invalid / expired token
5c. return HTTP 403 - Forbidden

2. Register a Backend Application -

Step 1: Access Azure AD B2C Portal

- Log in to the Azure AD B2C portal.
- Navigate to App registrations.

Step 2: Register the Application

- Click on the + New registration tab.
- Enter a name for the application, e.g., UserApp.
- Click on the Register button to complete the registration.

3. Expose an API

Step 1: Access Expose an API Section

- After registering the app, click on the Expose an API tab.
- Click on Add to expose your API.

Step 2: Set the Application URL

- Copy the Domain Name from the settings section.
- Paste the copied domain name into the Edit Application ID URI section, ensuring it is formatted as a URL, e.g., *https://YCITUAT.com/597e1e79-1494-41c5-9737-d8dec56d4e70..*

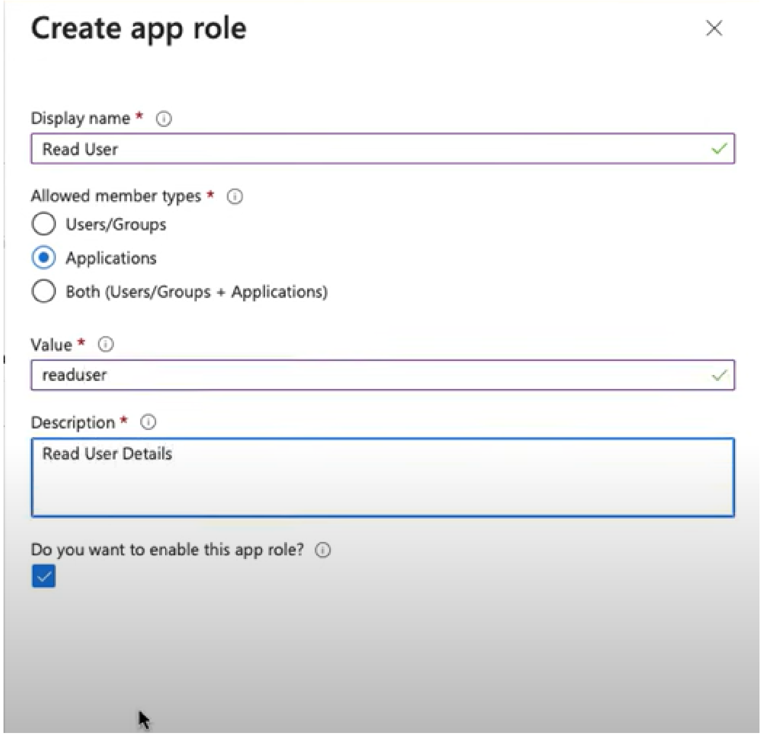
4. Assign App Roles -

Step 1: Create App Roles

- Navigate to the App roles section.
- Click on Create app role and fill in the required details.

Step 2: Verify Roles in the Manifest File

- After adding the roles, go to the Manifest file.
- Verify that the newly created roles are listed correctly.



Create app role

Display name *

Read User

Allowed member types *

○ Users/Groups

● Applications

○ Both (Users/Groups + Applications)

Value *

readuser

Description *

Read User Details

Do you want to enable this app role?

☑

5. Register Client/Frontend Application -

Step 1: Register the Client Application

- Repeat the same steps from Section 2 to register your client application.

Step 2: Modify the Manifest File

- In the Manifest file of the client application, change the value of "accessTokenAcceptedVersion" to 2.

6. Configure API Permissions -

Step 1: Grant API Permissions to Client Application

- Go back to the App registrations section and search for your client application.

- In the API permissions tab, add the necessary permissions that the client application needs to access your Spring Boot API.

7. Obtain an Access Token -

Step 1: Access Token Endpoint

- To obtain an access token, use the predefined Azure AD endpoint: Ex.*https://YCITUAT.b2clogin.com/YCITUAT.onmicrosoft.com/B2X_1_authflow/oauth2/v2.0/token*

  - Domain Name: YCITUAT.b2clogin.com
  - User Flow: B2X_1_authflow

Step 2: Prepare the Request

- Content-Type: application/x-www-form-urlencoded
- Method: POST
- Grant Type: client_credentials
- Client ID: Your client ID from the client application.
- Client Secret: Your client secret from the client application.
- Scope: The exposed API URL with .default suffix.
- Ex - *https://YCITUAT.com/597e1e79-1494-41c5-9737-d8dec56d4e70/.default*

Step 3: Send the Request

- Send the POST request to obtain the access token.

8. Use the Access Token -

Step 1: Authenticate API Requests

- Use the received access token to authenticate API requests by passing it in the Bearer section of the Authorization header in your Spring Boot API.

❖ *Spring Boot Application Requirements –*

a. Spring Boot Security dependency.
b. Audience (In yml file under spring security. Copy it from decoding the access the token.)
c. Issuer url (In yml file under spring security. Copy it from decoding the access the token.)
d. If required add extra layer of Jwt token security in API.

❖ *How to Create User Flow –*

1. Access External Identities Settings

Step 1: Navigate to Microsoft Entra ID

- Log in to the Microsoft Entra ID portal.
- Go to the External Identities section.

Step 2: Configure External Collaboration Settings

- Within External Identities, click on External collaboration settings.
- Set the following details:
    1. Guest User Access Restrictions:
        - Select Guest users have limited access to properties and memberships of directory objects.
    2. Guest Invite Restrictions:
        - Set to Only users assigned to specific admin roles can invite guest users.
    3. External User Leave Settings:
        - Set to Yes.

2. Enable and Create a User Flow -

Step 1: Enable User Flow Creation

- After configuring the external collaboration settings, the option to create a user flow will be enabled.

Step 2: Create a User Flow

- Click on the enabled Create user flow option.
- Follow the on-screen instructions to set up the user flow according to your requirements.

Step 3: Use the User Flow in Token Creation API

- Once the user flow is created, it can be used in the token creation API to authenticate users according to the defined flow.