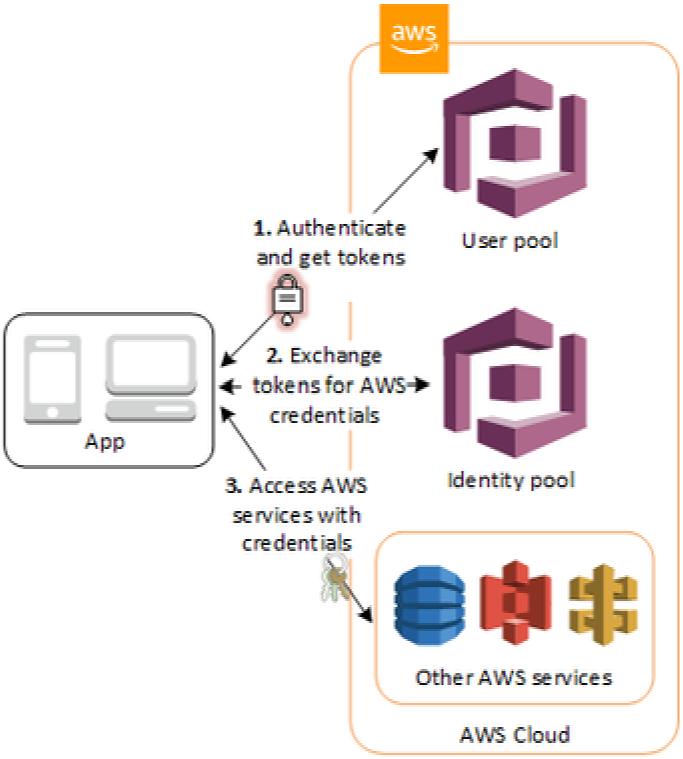
STRATEGY - API validation through Amazon Cognito integration for GUI entrypoints

STRATEGY - API validation through Amazon Cognito integration for GUI entry points


aws
User pool
1. Authenticate and get tokens
2. Exchange tokens for AWS credentials
Identity pool
App
3. Access AWS services with credentials
Other AWS services
AWS Cloud

Implementation Steps:

1.Creating API for HTTP requests (Lambda Function)

2.Creating Infrastructure as code serverless configuration for API GATEWAY authentication that is

Our API gateway uses Cognito as our authentication and authorization layer (To be defined in serverless.yml).

3.Cognito Requirements for serverless:

- Creating a Cognito user pool.
- Creating a Cognito user pool client.
- Subscribing Cognito with API Gateway.

4. Create a Cognito Identity Pool: We use the Cognito Identity Pool as a way to control which AWS resources our logged-in users will have access to. And also tie in our Cognito User Pool as our authentication provider.

- Create a Cognito Identity Pool.
- Creating Identity pool roles.
- Integrating IAM roles with Cognito Identity Pool and create policies.

Aws Cognito Domain Configuration for Hosted UI

Overview: We need to create our domain name, call back URL and sign-out URL in Cognito and Cognito provides us Hosted UI which can be customized for better user experience. A domain name is mainly required for accessing the Hosted UI which is provided by Cognito. Registered users can sign in or users can signup using the Hosted UI where a verification code will be sent to their registered email or mobile that depends on multi-factor authentication we configure. Once the users sign in they will be redirected to Call Back URL where they have id token, access token, type (Bearer), refresh token to update the id token and access token and token expiry time by default it is valid for 2 hours.

URL for accessing Hosted UI:

https://<your_domain>/login?response_type=token&client_id=<your_app_client_id>&redirect_uri=<your_callback_url>

****<your domain>**: Domain name that is set in Cognito user pool.

****<your app client id>**: App Client Id that is generated in Cognito user pool.

****<your callback URL>**: Call back URL that is set in Cognito user pool.

Call Back URL that contains ID token, Access token, Refresh token, token expiry, type:

https://<your_callback_url>/#id_token=123456789tokens123456789&expires_in=3600&token_type=Bearer

Our next process is to store the id token in our local browser session and whenever any request is triggered to our API, we have to use the token from the session and authenticate users.

The Further process needs to be implemented in Front End using React framework.