AWS API Gateway and Cognito

COMS 6998 - Cloud Computing and Big Data

Goutham Reddy Kotapalle (gk2547)

API Gateway

- Service for creating, publishing, maintaining, monitoring, and securing REST and WebSocket APIs at any scale.

- API Gateway handles all the tasks involved in accepting and processing up to hundreds of thousands of concurrent API calls, including traffic management, CORS support, authorization and access control, throttling, monitoring, and API version management.

- Create APIs that adhere to REST protocol. Allows methods such as GET, PUT, POST, DELETE, ANY on the resources.

Steps to set up API Gateway

1. Go to the API Gateway service on console.
2. Create API -> Select New API in Create New API -> Enter the API name. Keep endpoint as region. Click on Create API.
3. Go to Actions, and create a new resource. Create the methods allowed on the new resource.
4. Deploy and test the API.
5. Detailed steps can be found here - link

Demo

- Let's create a service which accepts GET and POST requests.

Assignment Project Exam Help

- In the POST request, we send a JSON request payload as following,
    - {"username": "John"} https://powcoder.com

- The output of this REST POST call will be "Hello John."

Add WeChat powcoder

- Note that if we make a GET request instead, we redirect to Google.com and the response body will be the html content of https://www.google.com

See in Action!

- Can be tested using a frontend, or a client such as Postman, or through cURL commands

- On Unix based machines, open the terminal and type in the following,
  - curl -X POST 'https://{api-id}.execute-api.{region}.amazonaws.com/test/helloworld' -d '{ "name": "John" }'
    - {"statusCode": 200, "body" : "\"Welcome to AWS!John.\""}
  - curl -X GET 'https://{api-id}.execute-api.{region}.amazonaws.com/Test/welcome'
    - "statusCode": 200, "body": "\"Welcome to AWS!\""}

# Cognito

- Amazon Cognito provides authentication, authorization, and user management for your web and mobile apps. Your users can sign in directly with a username and password, or through a third party such as Facebook, Amazon, Google, or Apple.

- The two main components of Amazon Cognito are **user pools** and **identity pools.** User pools are user directories that provide sign-up and sign-in options for your app users. Identity pools enable you to grant your users access to other AWS services. You can use identity pools and user pools separately or together.

Cognito

Assignment Project Exam Help

https://powcoder.com

Add WeChat powcoder
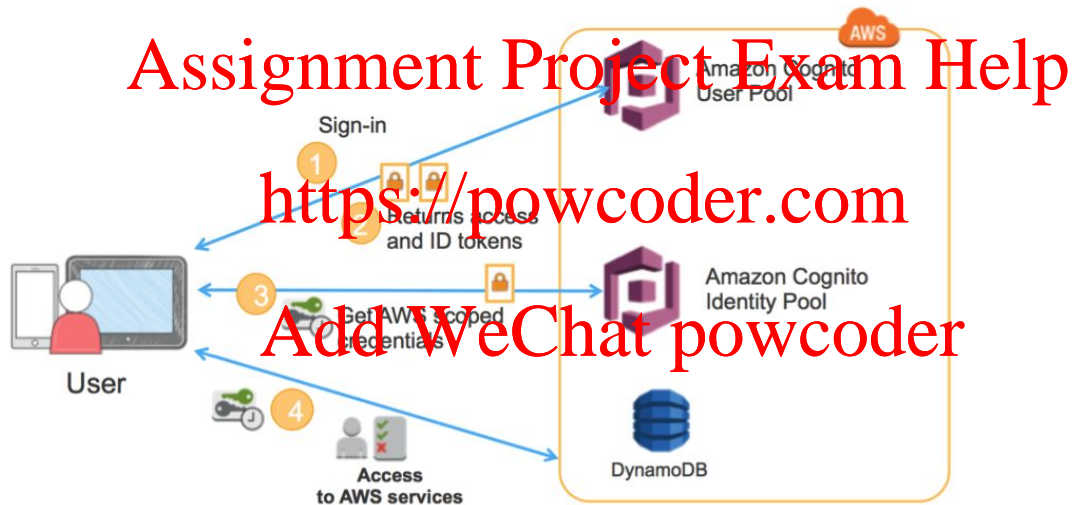
Demo

- Sign in into our app using Google + – **Authentication**

- **Once authenticated, fetch the id-token from the response. Authorize** your app user to access AWS S3 or any other application service using the temporary credentials as follows -

Assignment Project Exam Help

https://powcoder.com

Add WeChat powcoder

```
function signinCallback(authResult) {
  if (authResult['status']['signed_in']) {
    // Add the Google access token to the Amazon Cognito credentials login
map.
    AWS.config.credentials = new AWS.CognitoIdentityCredentials({
        IdentityPoolId: 'IDENTITY_POOL_ID',
        Logins: {
            'accounts.google.com': authResult['id_token']
        }
    });

    // Obtain AWS credentials
    AWS.config.credentials.get( function(){
        // Access AWS resources here.
    });
  }
}
```