JAM CHALLENGE - SignUp not permitted ?!

Introduction:

This challenge demonstrates how to analyze and troubleshoot Amazon Cognito and Lambda related issue. In this challenge you will learn how to fix Amazon Cognito configurations and write Lambda code to make API calls to test the configurations. Completing this challenge will enable you to configure and troubleshoot Amazon Cognito User pools to add user sign-up, sign-in and access control to web applications.

Topics covered

By solving and walking through this challenge, you will be able to:

- Rectify Amazon Cognito configuration as per the use case
- Write AWS Lambda code to make AWS API calls

Technical knowledge prerequisites

To successfully complete the challenge, you should be familiar with AWS Lambda and Amazon Cognito.

Services you should use

AWS Lambda, Amazon Cognito

Description:

You are the developer in company GreenTech and you've been appointed to fix an issue in which new users are not allowed to sign up on their own. Rather it gives the error: "SignUp is not permitted for this user pool".

In the DEV environment, Amazon Cognito User Pool has been created and AWS Lambda is calling Cognito's SignUp API.

You are required to fix this error on high priority so that new customers are not impacted when they try to sign up themselves.

Category Difficulty:

100 – Easy

Task 1: Fix the missing piece

You are working with customer ABC and they're using Amazon Cognito Sign-in and Sign-up features for integrating their backend application for Authentication and Authorization purposes.

Somehow, when new users are trying to Sign-up they are facing challenge of getting an error of “SignUp is not permitted for this user pool”.

You are assigned to resolve this challenge.

You can see this error by triggering the pre-existing AWS Lambda function “LambdaJam” which will call Amazon Cognito's “sign_up” API.

You are assigned to resolve the issue of “SignUp is not permitted for this user pool”

For testing purpose, you may use AWS Lambda function “LambdaJam”

Clues:

What do I check?

Please check the Amazon Cognito User Pool configuration

Solution

For New Amazon Cognito Console:

1. Go to the Amazon Cognito User Pool console
2. Navigate to tab "Sign-up experience"
3. Edit "Self-service sign-up" -> Enable self-registration
4. Save

For Old Amazon Cognito Console:

1. Go to the Amazon Cognito User Pool console
2. Navigate to "General Settings" -> "Policies"
3. Under the heading "Do you want to allow users to sign themselves up?", choose "Allow users to sign themselves up"
4. Save

Task 2: Validate the Solution

Now, that you have configured the Amazon Cognito user pool configuration for Sign-up, you would like to test it by adding new users using AWS Lambda function.

AWS Lambda function is already created with name “LambdaJam”. You need to add the missing code in the lambda function and run it. Ensure that it successfully creates an AWS Cognito User using “sign_up” API call.

NOTE: Boilerplate code has already been provided along with comments to help you out. You may refer this boto3 documentation for more information

Clues:

What shall I do?

You need to provide values for

- ClientId
- Username
- Password
- UserAttributes.Name
- UserAttributes.Value

You may refer `Output Properties` in the left navigation page on JAM console to get the value for `ClientId` using `UserPoolClientId`

Once done, deploy the Lambda and run it.

Solution

How to fetch ClientId:

You may refer `output_properties` file to get the value for `ClientId`, or

Using New Amazon Cognito Console:

1. Go to the Amazon Cognito User Pool console
2. Navigate to tab "App Integration"
3. Under "App client list", copy the value for "Client ID"

Using Old Amazon Cognito Console:

1. Go to the Amazon Cognito User Pool console

2. Navigate to "General Settings" -> "App clients"
3. Copy the value under "App client id"

Edit Lambda Function:

```
client = boto3.client('cognito-idp');
            response = client.sign_up(
              ClientId='Use above fetched Client ID',          # The ID of
the client associated with the user pool.
              Username='user123',          # The user name of the user you
wish to register.
              Password='anyP@s$w0rd!',          # The password of the user
you wish to register.
              UserAttributes=[
                {
                  'Name': 'email',       #The name of the attribute.
                  'Value': 'anyemail@amazon.com'       #The value of the
attribute.
                }
              ]
            )
```

Deploy and Run the Lambda function