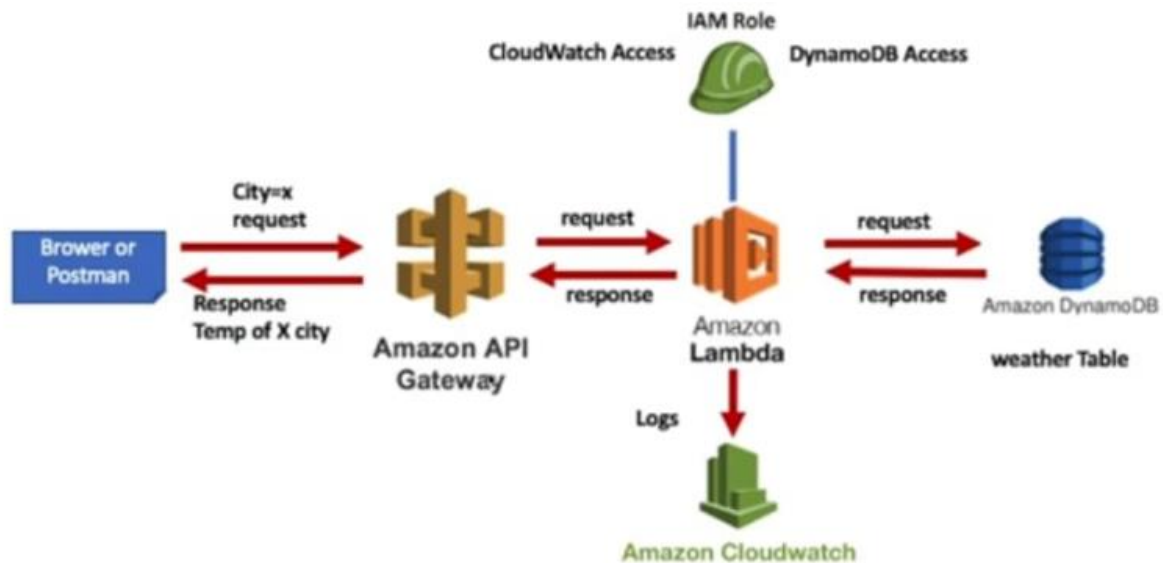


Lab 6



Lab Scenario

Introduction:

API Gateway provides tools for creating and documenting web **APIs** that route HTTP requests to **Lambda** functions. You can secure access to your **API** with authentication and authorization controls. Your **APIs** can serve traffic over the internet or can be accessible only within your VPC.

Please visit to know detailed about AWS API Gateway:

<https://docs.aws.amazon.com/apigateway/latest/developerguide/welcome.html>

Lab Objective: Using AWS API Gateway and connect it with AWS DynamoDB and AWS Lambda function. Using API Gateway, you can call AWS Lambda function while something asked through the web browser and the HTTP request hits to DynamoDB table/database.

The target scenario is given above in the attached figures.

Activities:

Task 0: Create the IAM Role of Full DynamoDB and CloudWatch service using IAM service.

Task 1: Create Lambda function and provide the created role as in task 0.

Task 2: Write the code in the Lambda function as given in sample code.

Task 3: Go to DynamoDB database service of AWS and Create a table named as weather and key name as city.

Task 4: Create the items in the weather table or you can fill automatically.

Example:

city	temp
Delhi	18
Mumbai	25
London	5
Italy	3

Task 5: Create an Amazon API Gateway by selecting REST Architecture. API name you can choose as per your choice. (Endpoint type should be regional)

Task 6: Then Create API resource (any relevant name you can give)

Task 7: Create a GET method and select Lambda function as integration type and **tick the Use Lambda Proxy Integration** then provide the created Lambda function name.

Task 8: Now deploy the API.

Task 9: Then Test the deployed API by using invoke url of API.

Task 10: Take the snapshots of all performed tasks and create a doc/pdf of your enrolment number_lab06 (Ex: E18CSE022_Lab06) and upload the file on LMS.

Sample Code:

```
/*  
import json  
import boto3  
  
dynamodb = boto3.resource('dynamodb')  
table = dynamodb.Table('weather')  
  
def lambda_handler(event, context):  
    city = event ['queryStringParameters'] ['city']  
    print("Name : ", city)  
    response = table.get_item(Key={'city' : city})
```

```
print(response)
print(response["Item"])
return {
    'statusCode': 200,
    'body': json.dumps(response['items'])
}
*/
```

Web Url:

https://docs.aws.amazon.com/code-samples/latest/catalog/code-catalog-python-example_code-apigateway.html

Video url:

https://www.youtube.com/watch?v=Lg61ynrYCPw&ab_channel=BinodSumanAcademy