Name : Akash Rathod

Task 1 Report : Automatically Start/Stop EC2 Instances with AWS Lambda and Amazon

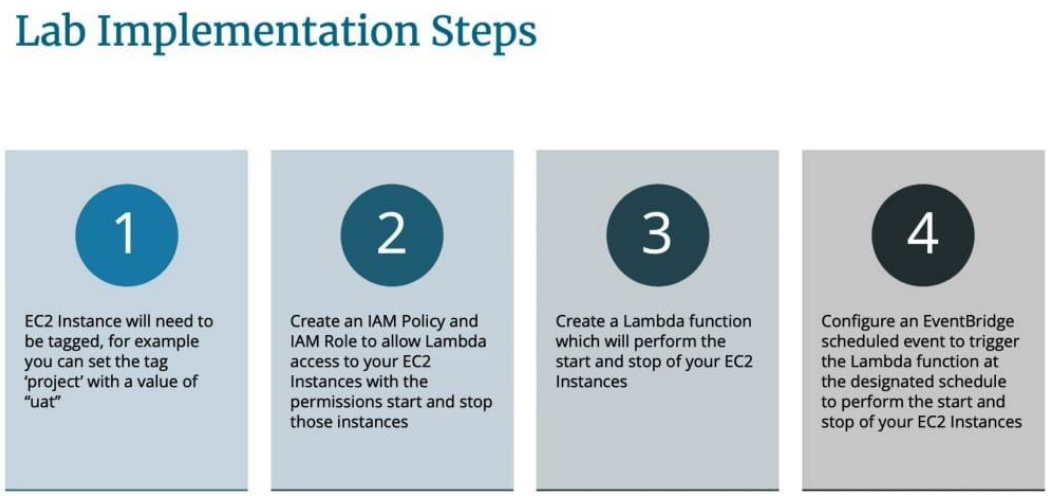
Fundamental

EC2 (Elastic Compute Cloud) is a core component of AWS cloud Computing Platform.

AWS Lambda is a **serverless computing service** offered by AWS.

It enables users to run their code in response to various events, without having to manage servers or infrastructure.

Task is create a automation process to start and shutdown EC2 instance at predefined times using Lambda function



Amazon Web Services (AWS) provides a service called Elastic Compute Cloud (EC2) that allows users to create and run virtual machines in the cloud. EC2 instances can be used for various purposes like hosting websites, running applications, and data processing. In this report, we will discuss how to create an automation process to start and shutdown an EC2 instance at predefined times using a Lambda function.

AWS Lambda: AWS Lambda is a serverless compute service that runs your code in response to events and automatically manages the computing resources for you. Lambda functions can be used to automate tasks, create serverless architectures, and build data processing pipelines. In this project, we will use a Lambda function to start and stop an EC2 instance at predefined times.

EC2 Instance: An EC2 instance is a virtual machine running in the AWS cloud. It can be used to run applications, host websites, and perform various computing tasks. EC2 instances can

be started, stopped, and terminated on demand. In this project, we will start and stop an EC2 instance at predefined times using a Lambda function.

Automation Process: The automation process involves creating a Lambda function that starts and stops an EC2 instance at predefined times. The Lambda function will be triggered by a CloudWatch event, which will be scheduled to run at the predefined times.

Step 1: Create a Lambda Function The first step is to create a Lambda function that starts and stops the EC2 instance. The Lambda function will use the AWS SDK to interact with the EC2 API and start or stop the instance. The Lambda function code will be written in a programming language supported by AWS Lambda like Python, Node.js, Java, or C#.

Step 2: Create an IAM Role The Lambda function needs permission to interact with the EC2 API. Therefore, we need to create an IAM role that has the necessary permissions. The IAM role should have permissions to start and stop EC2 instances.

Step 3: Create a CloudWatch Event The CloudWatch event is used to trigger the Lambda function at predefined times. The CloudWatch event can be scheduled to run at specific times using a cron expression. The cron expression specifies the time and date when the event should be triggered.

Step 4: Test the Automation Process Once the Lambda function, IAM role, and CloudWatch event are created, we can test the automation process. We can manually trigger the CloudWatch event and verify that the Lambda function starts and stops the EC2 instance at the predefined times

---

Reference used:

https://github.com/iaasacademy/aws-how-to-guide

https://youtu.be/VD_rF_tIBOY

platform used :

AWS

- Lambda
- IAM

Special Thanks Sujit sir and Rakesh Sir 😊

Sure, here's an example of how to create an AWS Lambda function that starts and stops an EC2 instance at predefined times:

1. Create a new Lambda function: Go to the AWS Lambda console and create a new function. Choose the language you want to use for your function, and give it a name.
2. Add the required permissions: In the IAM console, create a new role with the `AWSLambdaBasicExecutionRole` policy attached. This will give your Lambda function the necessary permissions to interact with AWS services.
3. Add code to the function: Add code to the function that will start and stop the EC2 instance. You can use the AWS SDK to interact with the EC2 service. Here's an example of Python code to start and stop an instance:

```
python
import boto3

def lambda_handler(event, context):
    # Set up the EC2 client
    ec2 = boto3.client('ec2')
```

```
    # Get the instance ID from the environment variables
    instance_id = os.environ['INSTANCE_ID']

    # Check the current time
    current_time = datetime.datetime.now().time()

    # Check if the instance should be started
    if current_time >= datetime.time(hour=9) and current_time <
datetime.time(hour=18):
        # Start the instance
        ec2.start_instances(InstanceIds=[instance_id])
        print('Started instance: ' + instance_id)
    else:
        # Stop the instance
        ec2.stop_instances(InstanceIds=[instance_id])
        print('Stopped instance: ' + instance_id)
```

This code uses the `boto3` library to interact with the EC2 service. It first gets the instance ID from an environment variable. Then it checks the current time and starts or stops the instance based on the time.

4. Set up a CloudWatch Events rule: Go to the AWS CloudWatch console and create a new rule. Choose the schedule you want to use for starting and stopping the instance. For example, you can set it up to start the instance at 9:00 AM and stop it at 6:00 PM on weekdays.
5. Add a target for the rule: Add the Lambda function you created as a target for the rule. This will ensure that the function is triggered at the scheduled times.
6. Configure environment variables: In the Lambda function configuration, add an environment variable named `INSTANCE_ID` with the ID of the EC2 instance that you want to start and stop.
7. Test the automation process: Test the automation process by manually triggering the CloudWatch Events rule. You can check the logs in the Lambda console to make sure the function is being executed correctly.

With these steps, you can create an automation process to start and stop an EC2 instance at predefined times using a Lambda function. This can help you save costs by only running the instance when it's needed, and also ensure that the instance is not left running when it's not needed.