**Assignment 1: Automated Instance Management Using AWS Lambda and Boto3**

**Objective:** In this assignment, you will gain hands-on experience with AWS Lambda and Boto3, Amazon's SDK for Python. You will create a Lambda function that will automatically manage EC2 instances based on their tags.

**Task:** You're tasked to automate the stopping and starting of EC2 instances based on tags. Specifically:

1. Setup:

   - Create two EC2 instances.

   - Tag one of them as `Auto-Stop` and the other as `Auto-Start`.

2. Lambda Function Creation:

   - Set up an AWS Lambda function.

   - Ensure that the Lambda function has the necessary IAM permissions to describe, stop, and start EC2 instances.

3. Coding:

   - Using Boto3 in the Lambda function:

     - Detect all EC2 instances with the `Auto-Stop` tag and stop them.

     - Detect all EC2 instances with the `Auto-Start` tag and start them.

4. Testing:

   - Manually invoke the Lambda function.

   - Confirm that the instance tagged `Auto-Stop` stops and the one tagged `Auto-Start` starts.

**Instructions:**

1. EC2 Setup:

   - Navigate to the EC2 dashboard and create two new t2.micro instances (or any other available free-tier type).

   - Tag the first instance with a key `Action` and value `Auto-Stop`.

   - Tag the second instance with a key `Action` and value `Auto-Start`.

2. Lambda IAM Role:

   - In the IAM dashboard, create a new role for Lambda.

   - Attach the `AmazonEC2FullAccess` policy to this role. (Note: In a real-world scenario, you would want to limit permissions for better security.)

3. Lambda Function:

   - Navigate to the Lambda dashboard and create a new function.

   - Choose Python 3.x as the runtime.

   - Assign the IAM role created in the previous step.

   - Write the Boto3 Python script to:

     1. Initialize a boto3 EC2 client.  
     2. Describe instances with `Auto-Stop` and `Auto-Start` tags.  
     3. Stop the `Auto-Stop` instances and start the `Auto-Start` instances.  
     4. Print instance IDs that were affected for logging purposes.

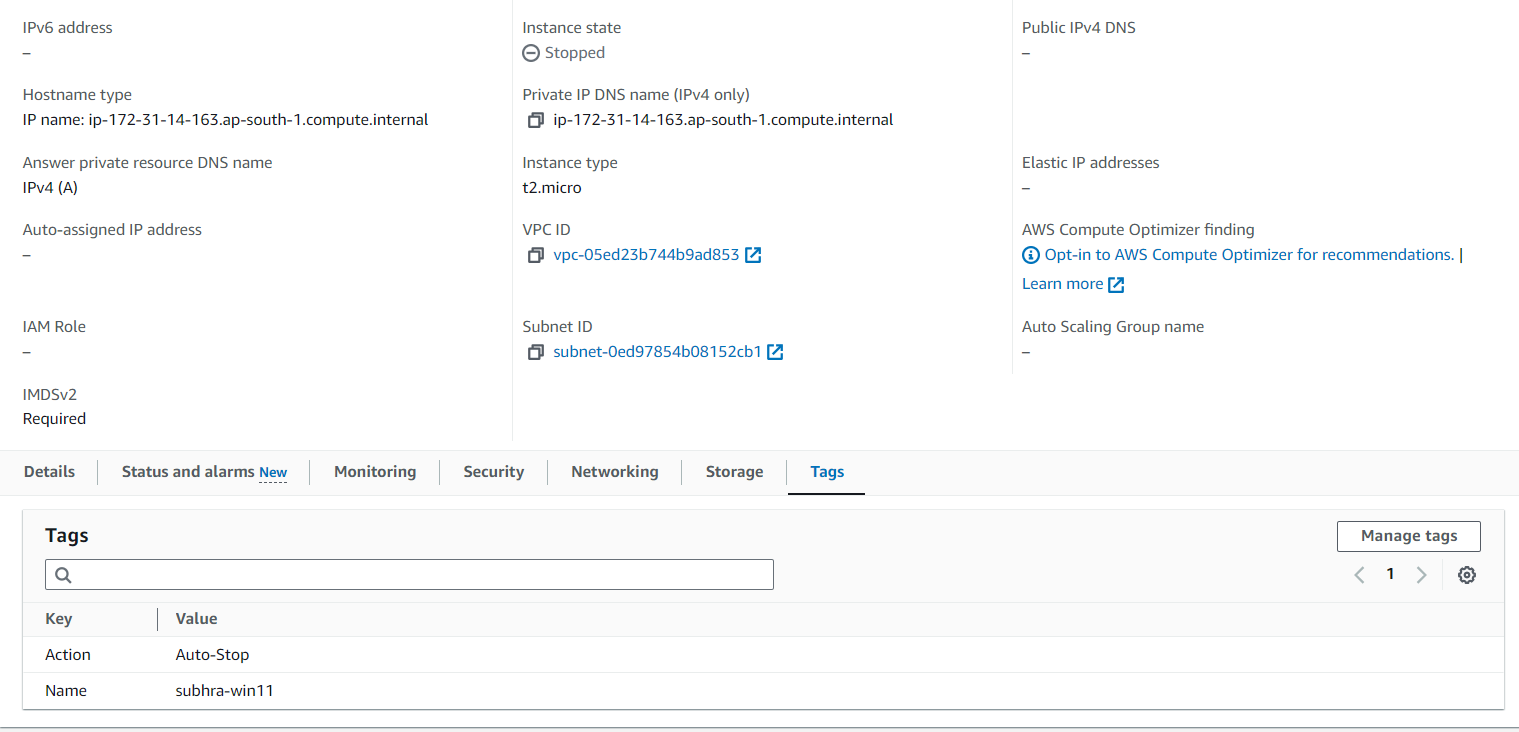
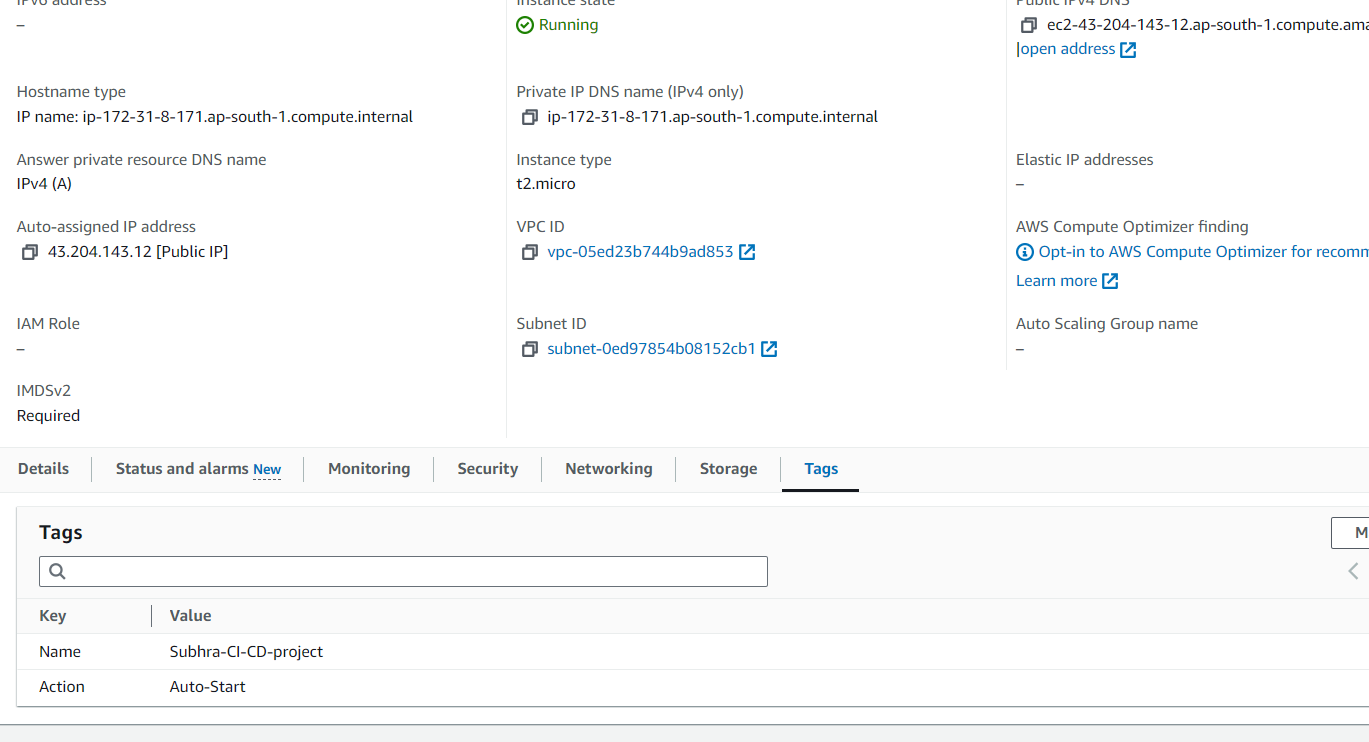
4. Manual Invocation:

   - After saving your function, manually trigger it.

   - Go to the EC2 dashboard and confirm that the instances' states have changed according to their tags

Answer

**1. EC2 Setup**

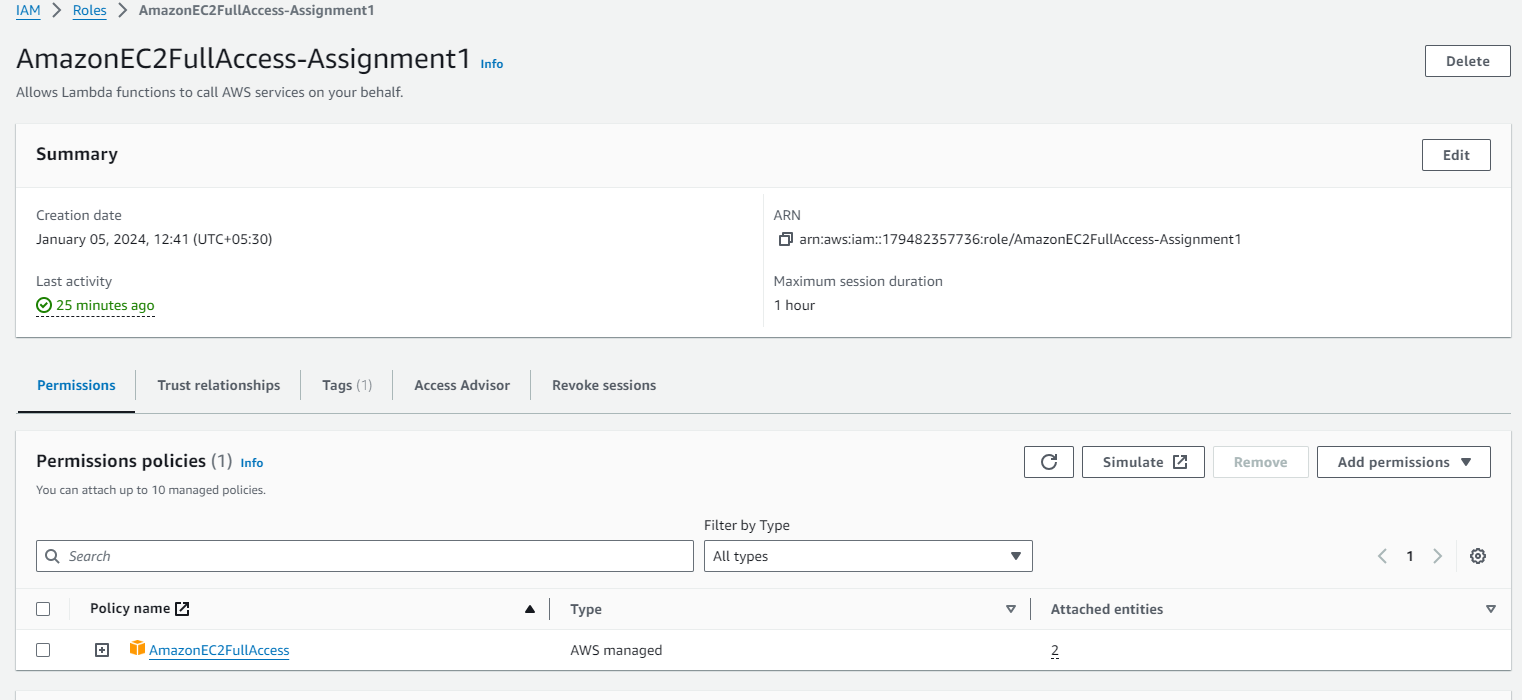
  


Navigate to the EC2 dashboard in the AWS Management Console.

Create two new t2.micro instances (or any other available free-tier type).

Tag the first instance with a key Action and value Auto-Stop.

Tag the second instance with a key Action and value Auto-Start.

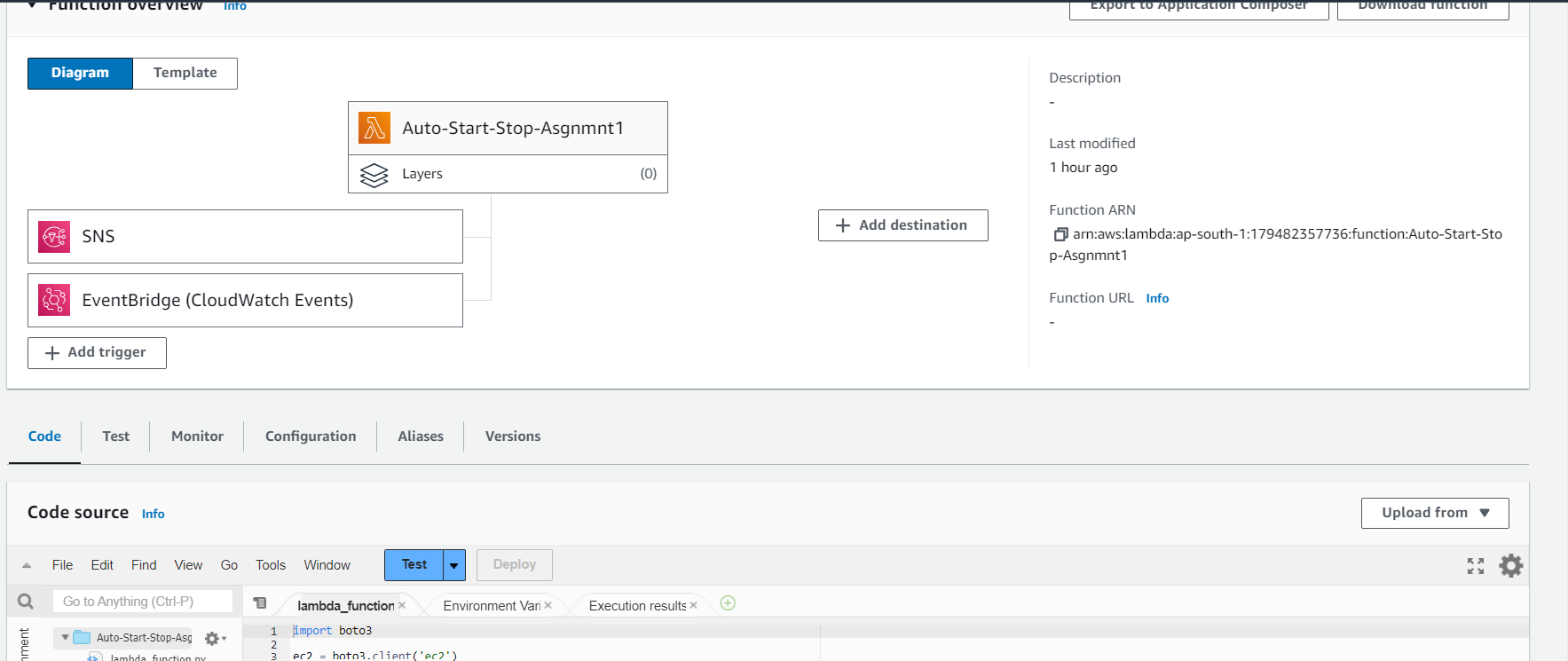
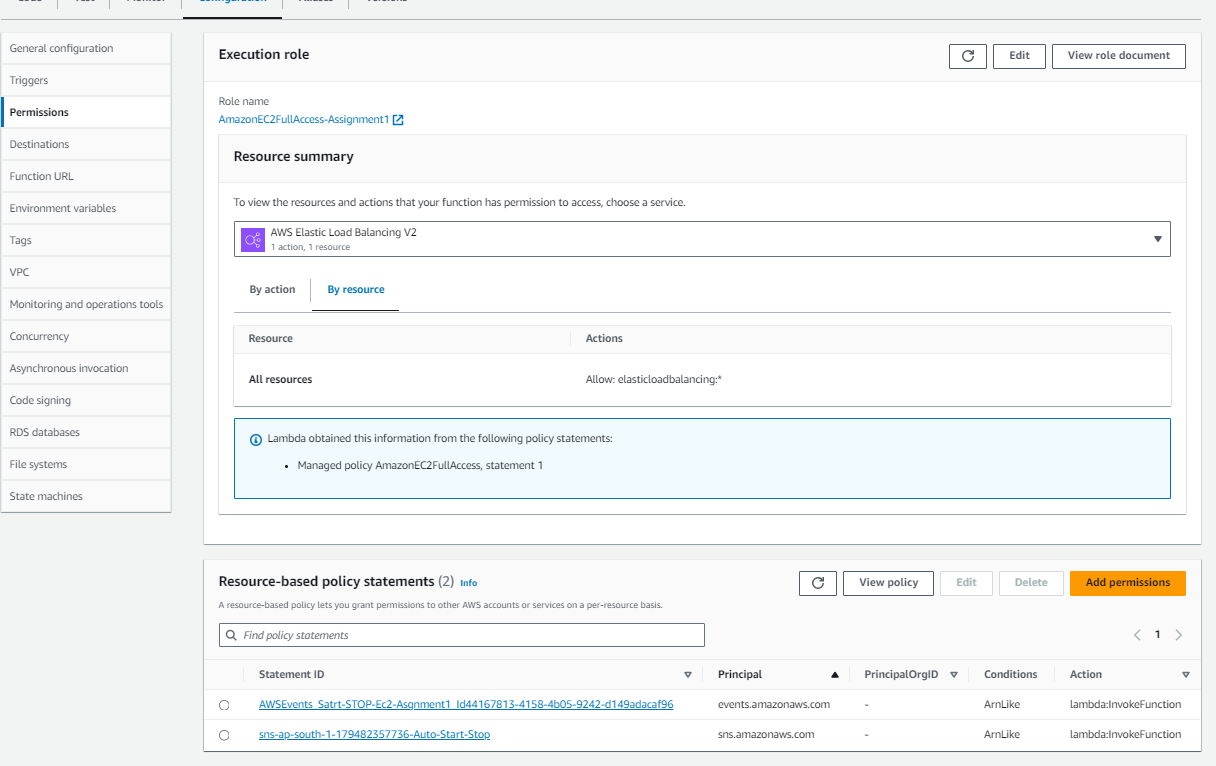
**2. Lambda IAM Role  
**

Navigate to the IAM dashboard in the AWS Management Console.

Create a new role for Lambda.

Attach the AmazonEC2FullAccess policy to this role.

**3. Lambda Function**

Navigate to the Lambda dashboard in the AWS Management Console.

Create a new function with Python 3.x as the runtime.

Assign the IAM role created in the previous step.

**4. Coding**

Create a new file named lambda\_function.py and paste the following code:  
  
import boto3

ec2 = boto3.client('ec2')

def lambda\_handler(event, context):

# Describe instances with Auto-Stop and Auto-Start tags

instances\_to\_stop = ec2.describe\_instances(

Filters=[

{'Name': 'tag:Action', 'Values': ['Auto-Stop']}

]

)['Reservations']

instances\_to\_start = ec2.describe\_instances(

Filters=[

{'Name': 'tag:Action', 'Values': ['Auto-Start']}

]

)['Reservations']

# Stop the Auto-Stop instances

for instance in instances\_to\_stop:

for reservation in instance['Instances']:

instance\_id = reservation['InstanceId']

ec2.stop\_instances(InstanceIds=[instance\_id])

print(f'Stopped instance: {instance\_id}')

# Start the Auto-Start instances

for instance in instances\_to\_start:

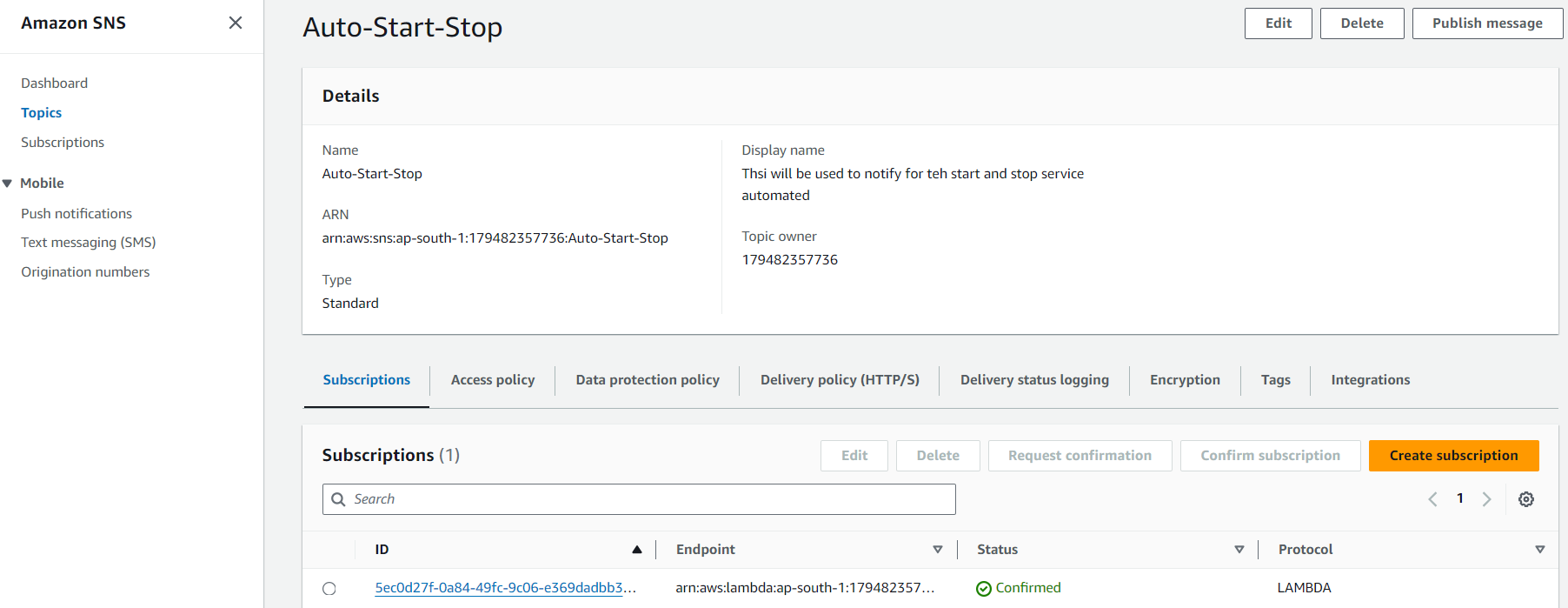
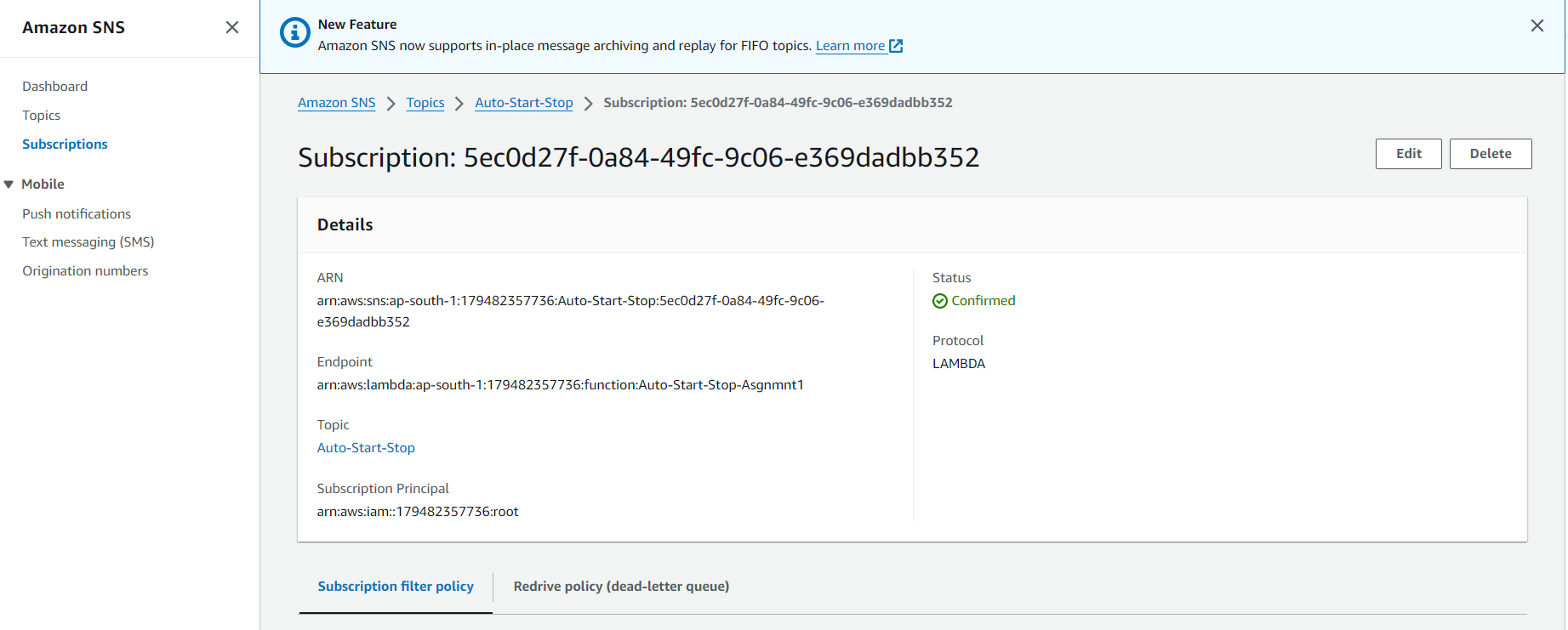
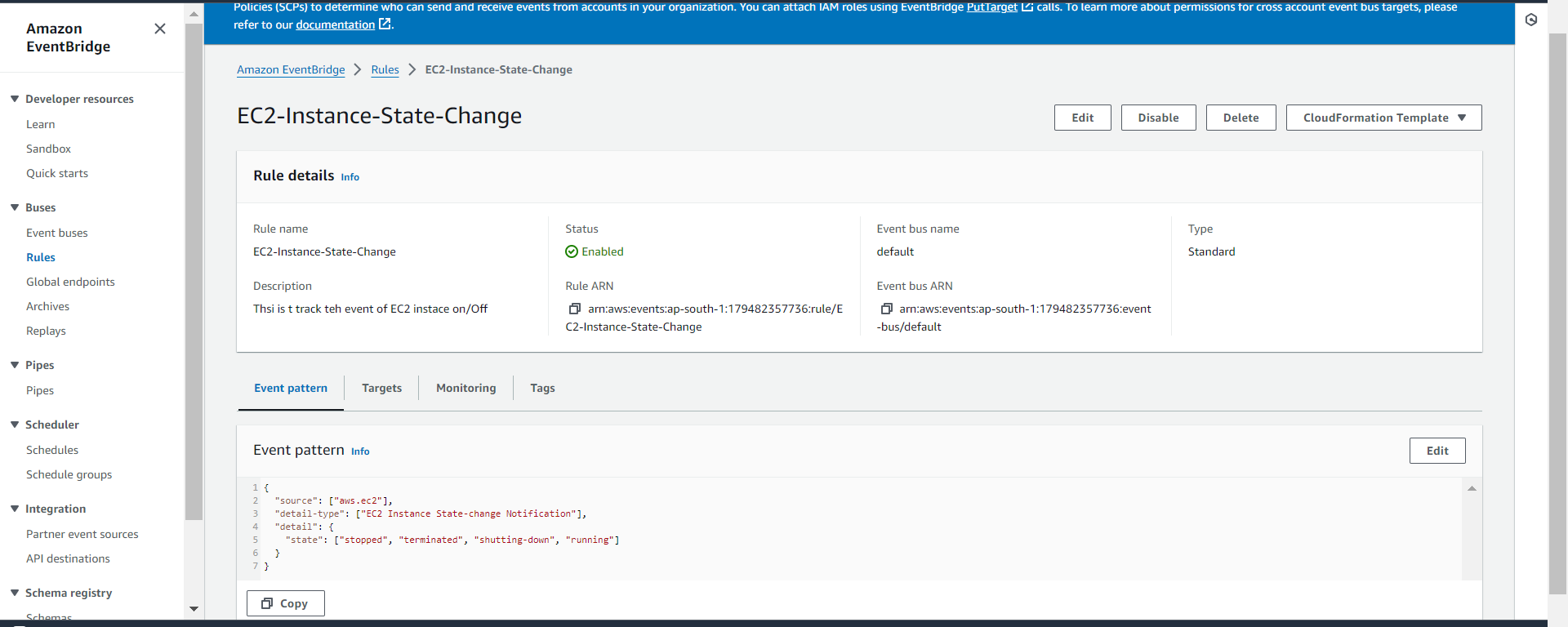
for reservation in instance['Instances']:

instance\_id = reservation['InstanceId']

ec2.start\_instances(InstanceIds=[instance\_id])

print(f'Started instance: {instance\_id}')  
  
  
  
  
**6. Automating the Lambda Function Execution**

To automate the execution of the Lambda function, you can use Amazon Simple Notification Service (SNS):

Create a new SNS topic.

Subscribe an SNS topic to the Lambda function.

Create a CloudWatch Events rule to trigger the SNS topic at a specific time.

**5. Testing**

To test the Lambda function, follow these steps:

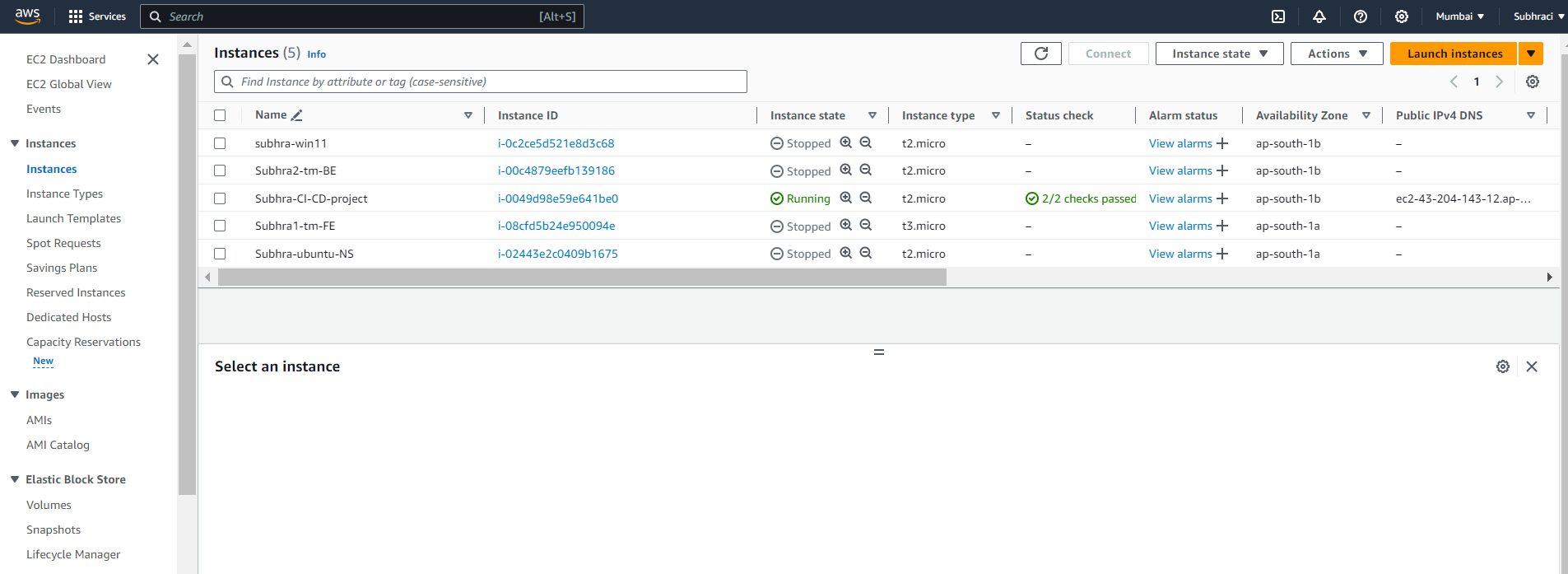
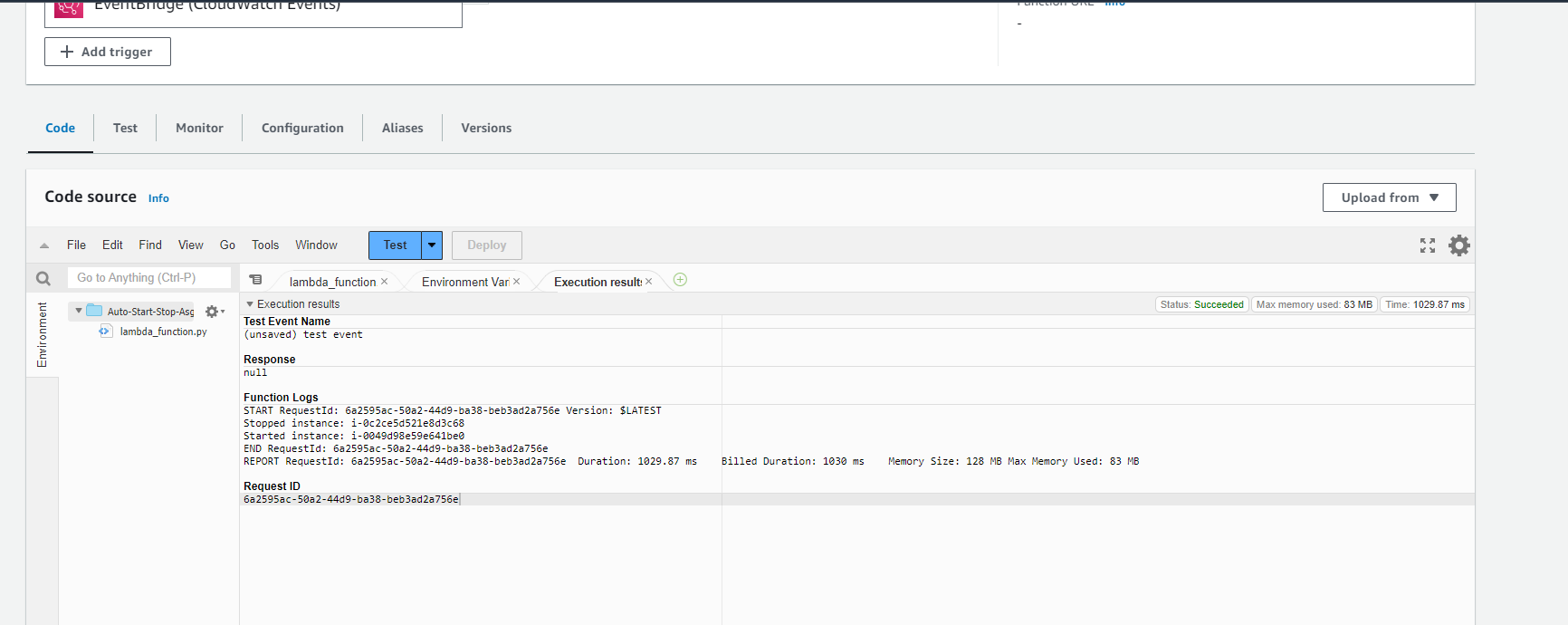
In the Lambda dashboard, click on the "Test" button.

Create a new test event with an arbitrary name and content.

Click on the "Test" button to execute the function.

Check the CloudWatch logs to see the output of the function.

Confirm that the instance tagged Auto-Stop has stopped and the one tagged Auto-Start has started.

  
  
Succesfull