How To Start & Stop EC2 Instances On AWS

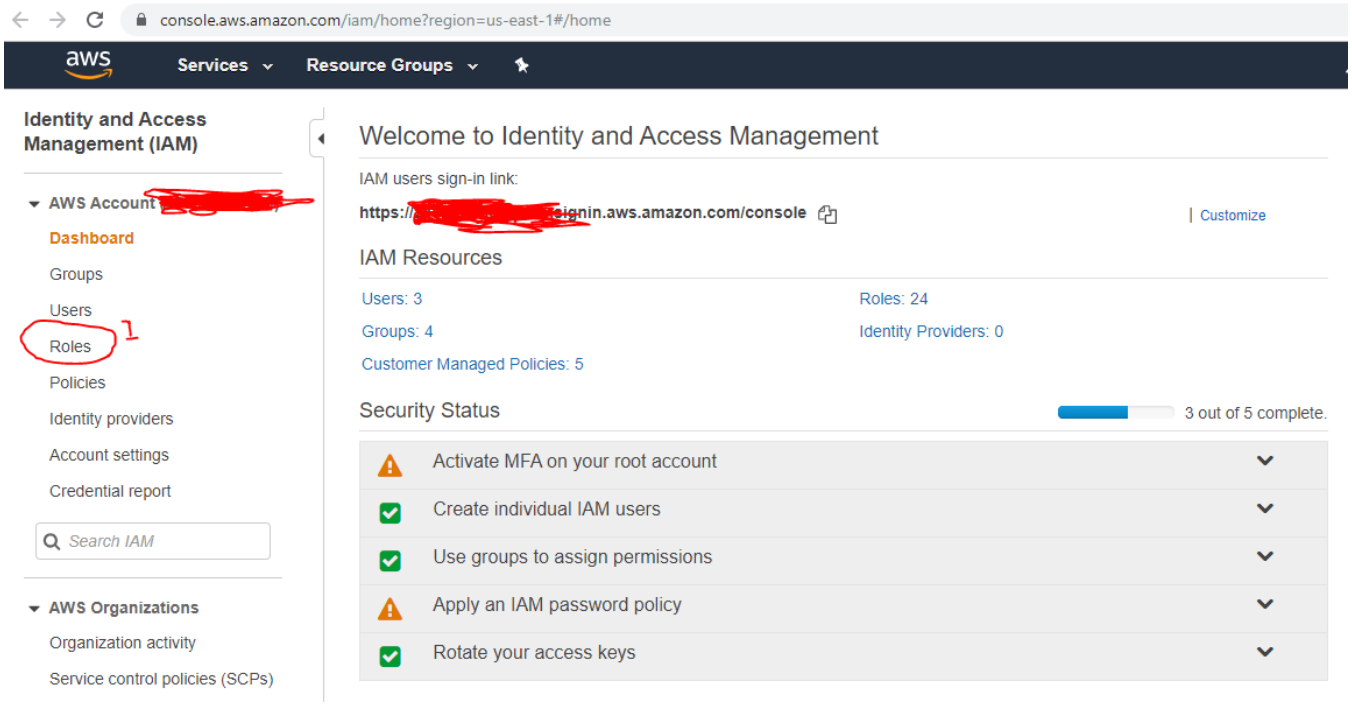
Having your EC2 instances start and stop so that they only run between work hours and not on weekends can be a bit of a puzzle. When you first look around within the AWS EC2 section you won't find any auto start-stop feature. Instead your need to use some of the other features in conjunction with EC2 to accomplish this. At first you'll be introduced to some new technologies that are beyond the scope of this article but don't be shy, it's very easy to get it up and running.

We will use the below technologies :

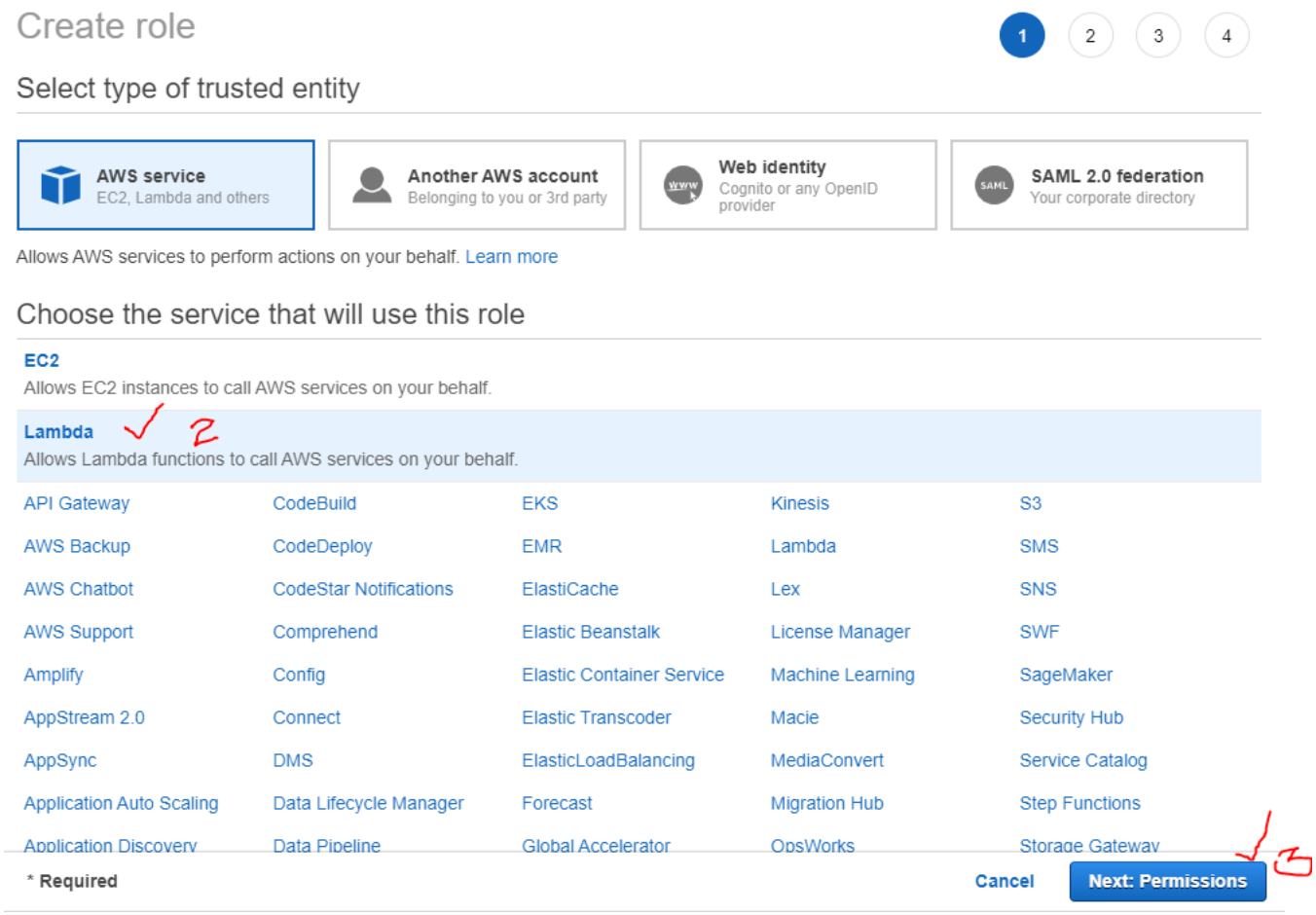
- IAM
- CloudWatch
- Lambda
- EC2

Create IAM Role :

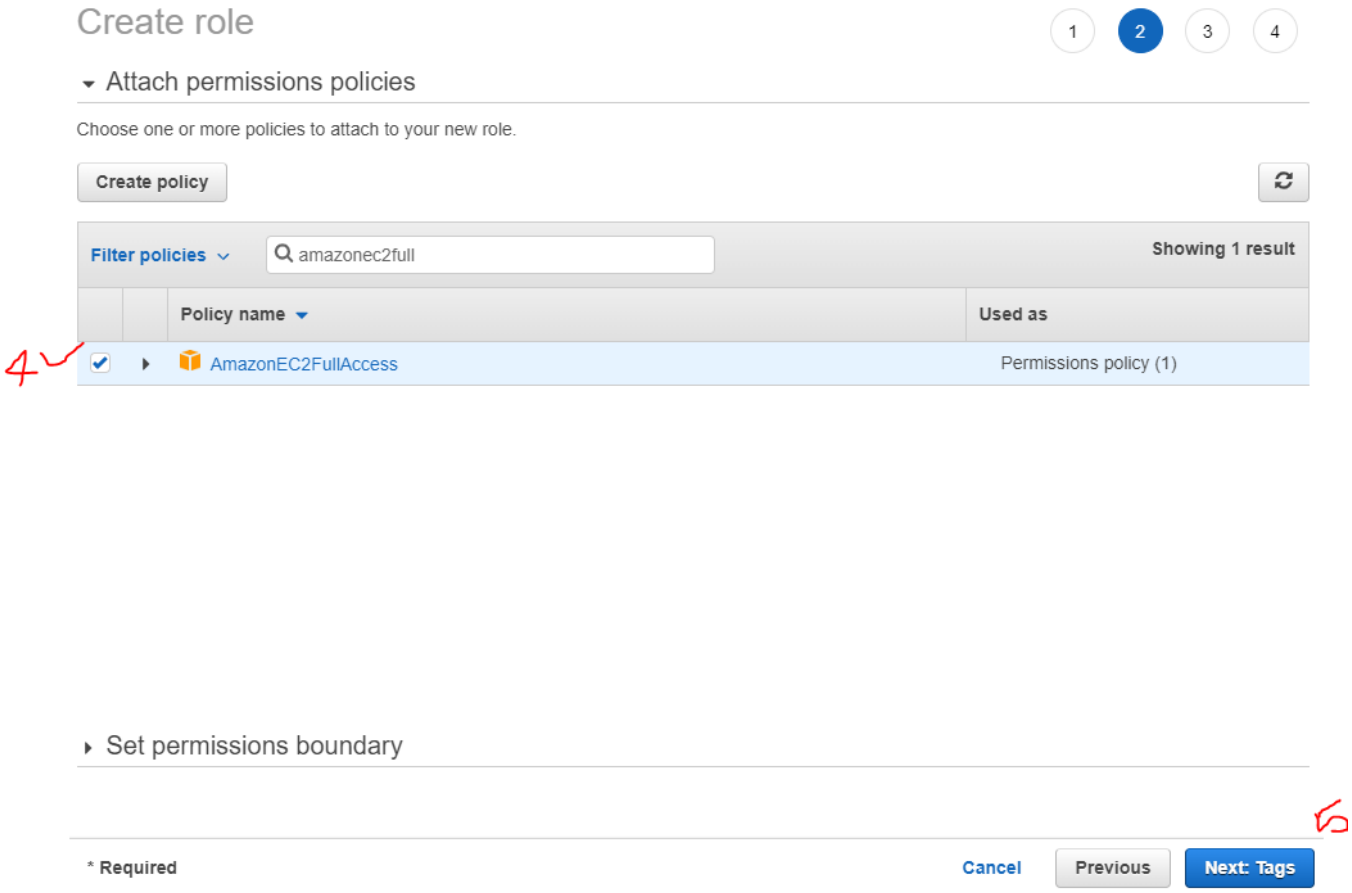
Navigate to AWS IAM service & Roles section, and click on Roles.



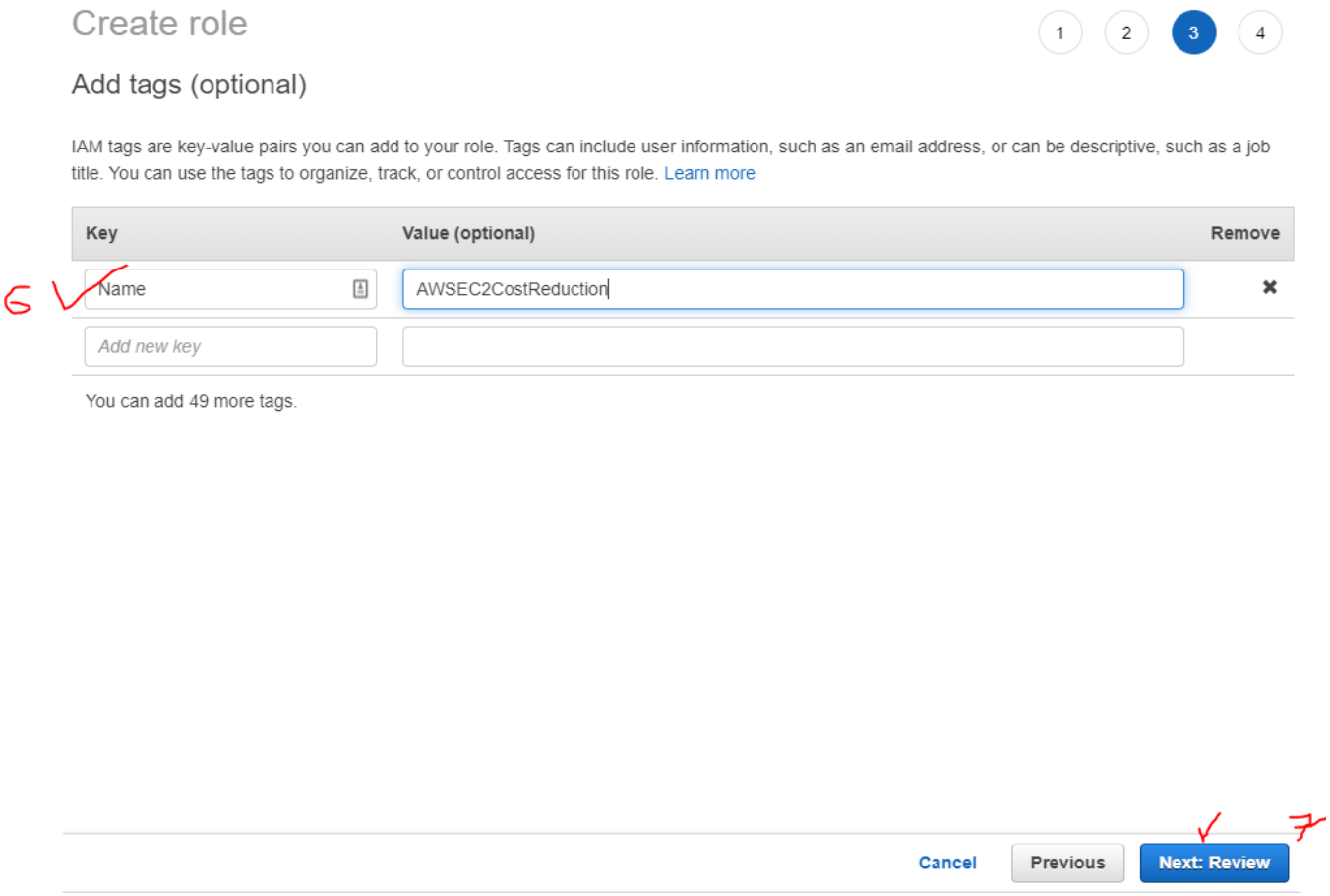
Create a role to allow the Lambda functions to call AWS services.



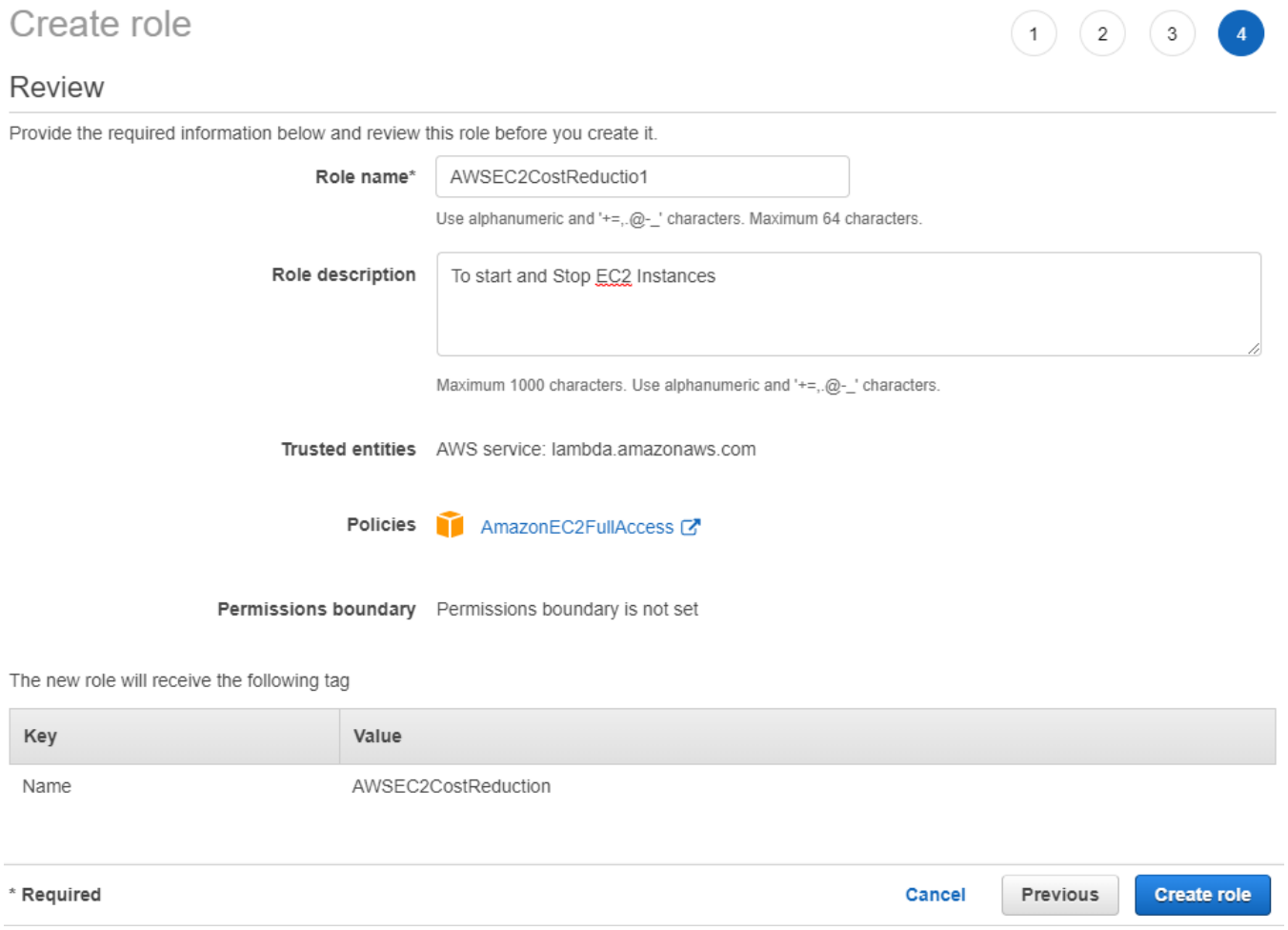
Click on Next button to save the permissions and to attach the roles which are required.



Create a Tag for filter purpose.

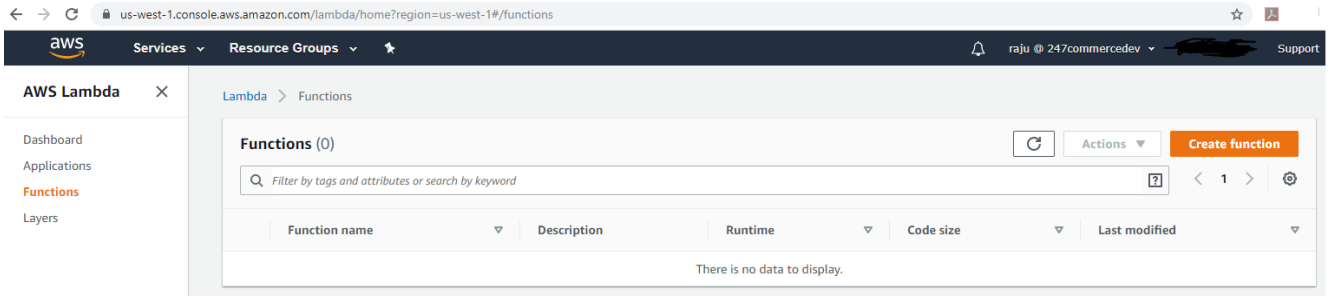


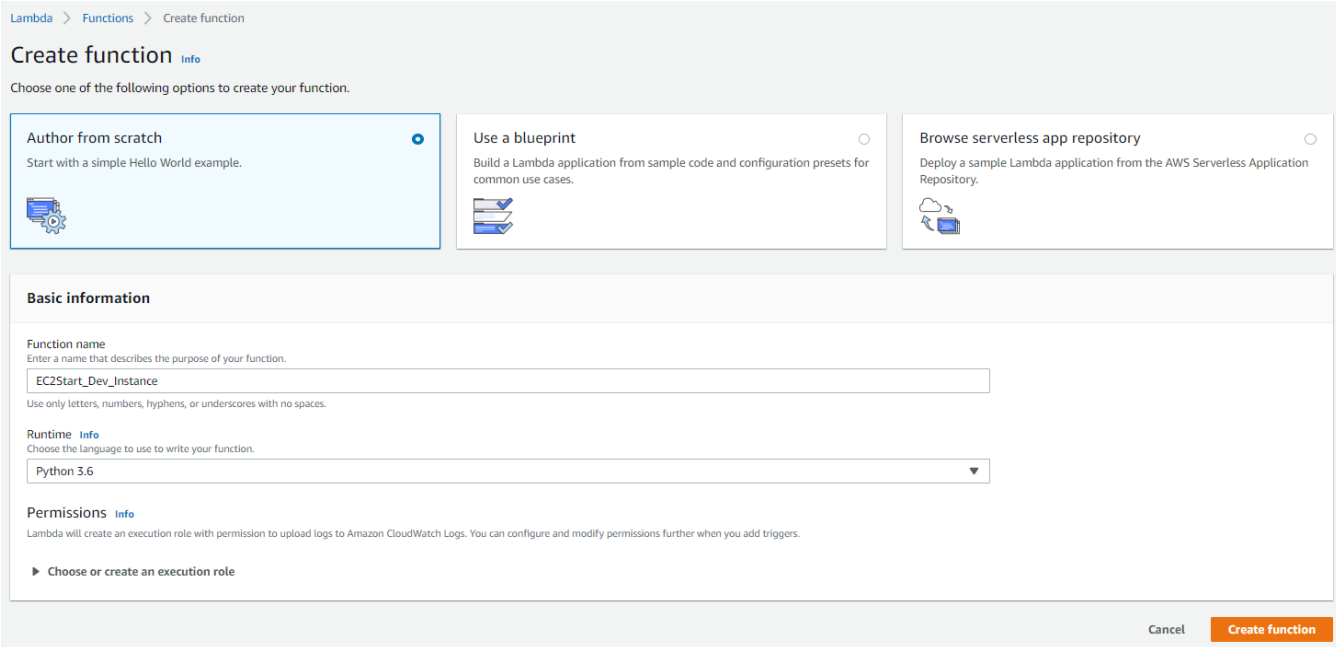
Review the role and save it by specifying the Role name & Role description.



Create Lambda Function :

To create a lambda function select the Lambda service and navigate to Functions, click on 'Create function' button to create it.





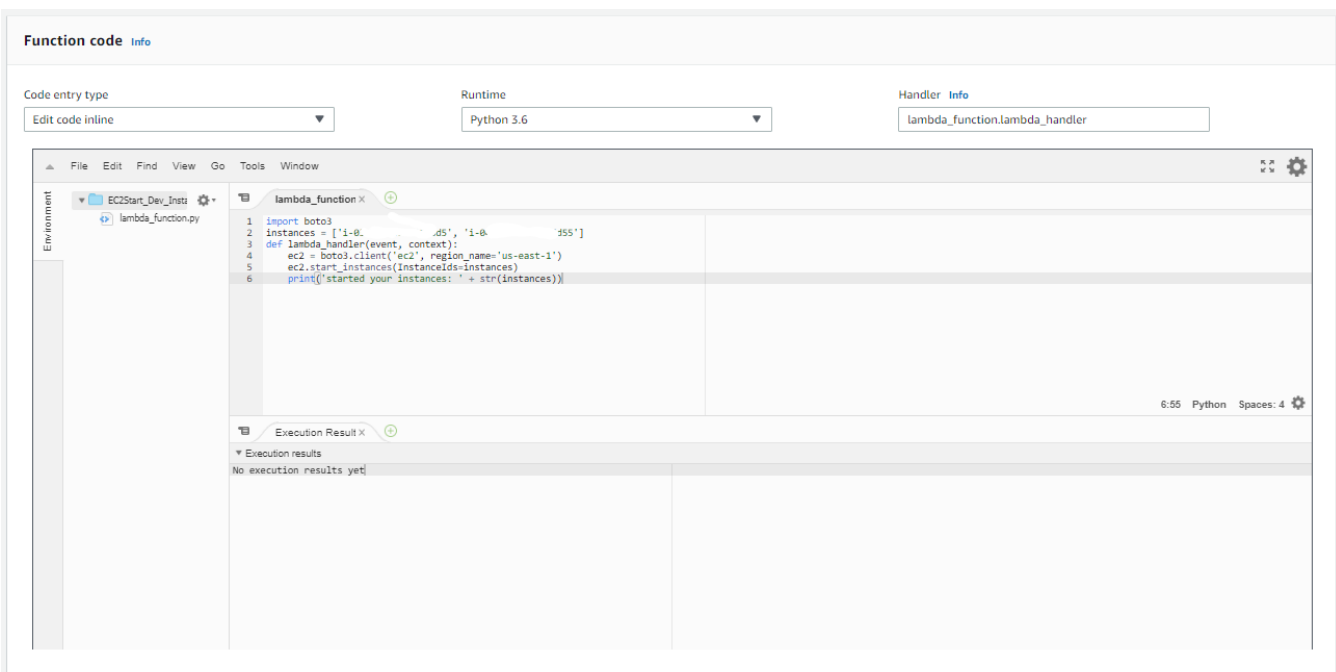
Paste the code to start instances, for stop function replace the start word with stop in the code and repeat the same steps.

For Start :

```
import boto3
instances = ['i-XXXXXX1', 'i-XXXXXX2']
def lambda_handler(event, context):
ec2 = boto3.client('ec2', region_name='us-east-1')
ec2.start_instances(InstanceIds=instances)
print('started your instances: ' + str(instances))
```

For Stop :

```
import boto3
instances = ['i-XXXXXX1', 'i-XXXXXX2']
def lambda_handler(event, context):
ec2 = boto3.client('ec2', region_name='us-east-1')
ec2.stop_instances(InstanceIds=instances)
print('started your instances: ' + str(instances))
```



Environment variables

You can define environment variables as key-value pairs that are accessible from your function code. These are useful to store configuration settings without the need to change function code. Learn more

Key	Value	Remove

▶ Encryption configuration

Tags

You can use tags to group and filter your functions. A tag consists of a case-sensitive key-value pair. Learn more

Name	AWSEC2CostReduction	Remove
Key	Value	Remove

Execution role

Choose a role that defines the permissions of your function. To create a custom role, go to the IAM console.

Use an existing role

Existing role
Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs.

AWSEC2CostReduction

View the AWSEC2CostReduction role on the IAM console.

Basic settings

Description
To Start EC2 Dev Instances

Memory (MB) Info
Your function is allocated CPU proportional to the memory configured.
128 MB

Timeout Info
0 min 3 sec

Network

Virtual Private Cloud (VPC) Info
Choose a VPC for your function to access.
No VPC

AWS X-Ray Info

Enable active tracing to record timing and error information for a subset of invocations.
☐ Active tracing

View traces in X-Ray

Concurrency

Unreserved account concurrency **1000**

◉ Use unreserved account concurrency
○ Reserve concurrency

Error handling

DLQ resource Info
Choose the AWS service to send the event payload to after maximum retries are exceeded.
None

Auditing and compliance

AWS CloudTrail can log this function's invocations for operational and risk auditing, governance, and compliance. Get started on the CloudTrail console.

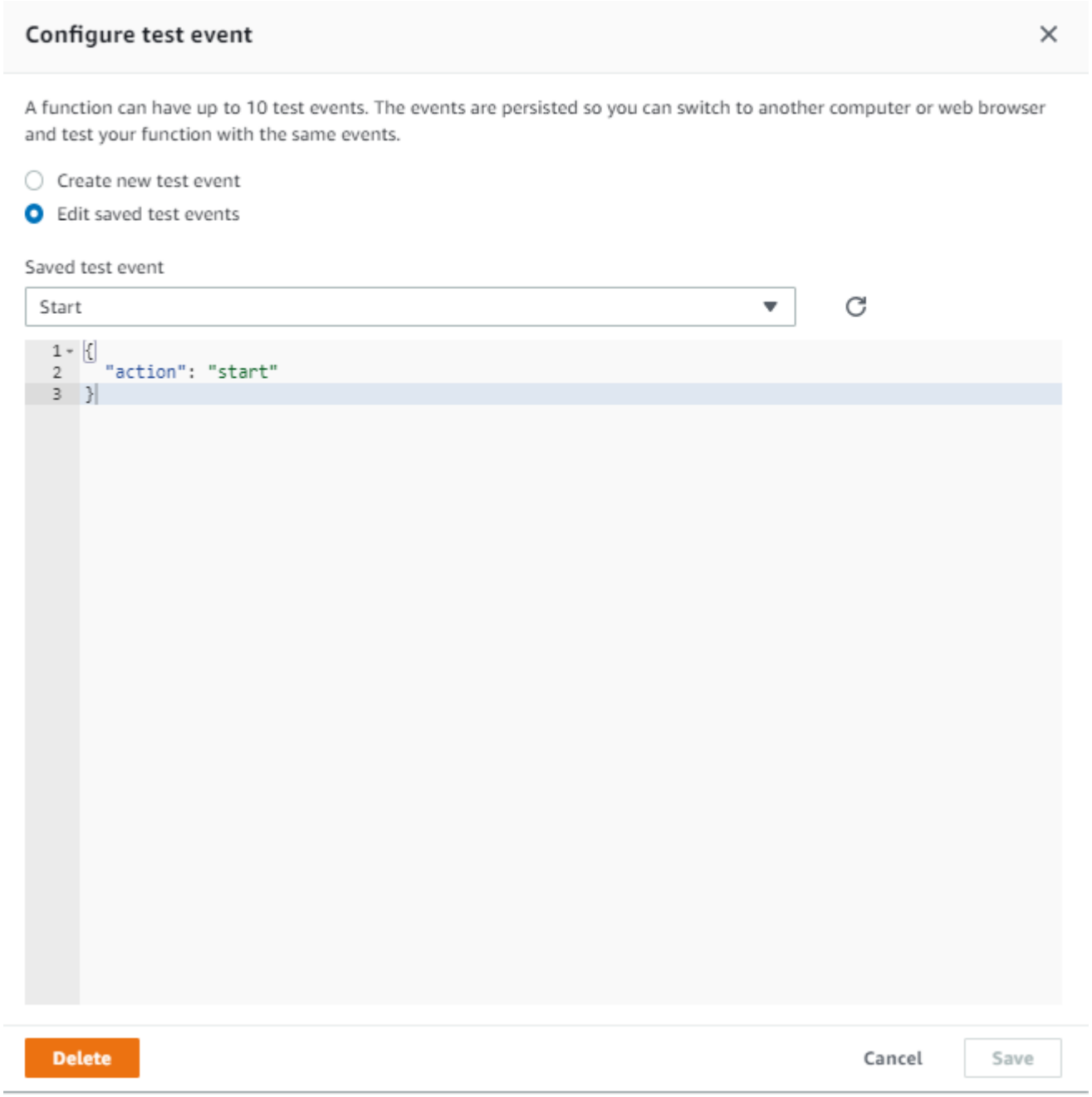
✓ Successfully updated the function **EC2Start_Dev_Instance.**

Lambda > Functions > EC2Start_Dev_Instance

ARN - arn:aws:lambda:us-east-1:[redacted].function:EC2Start_Dev_Instance

EC2Start_Dev_Instance

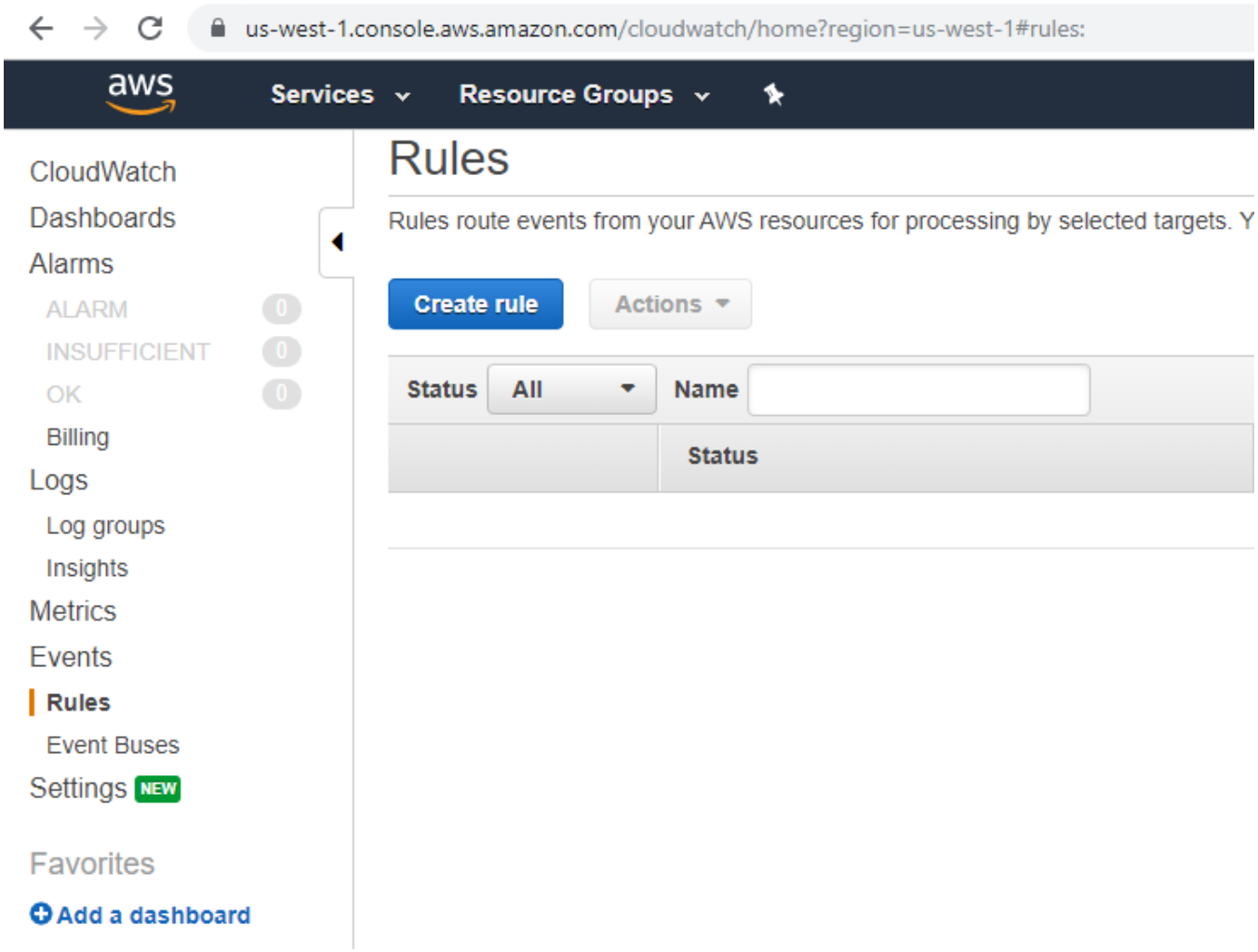
Throttle | Qualifiers | Actions | Start | Test | Save



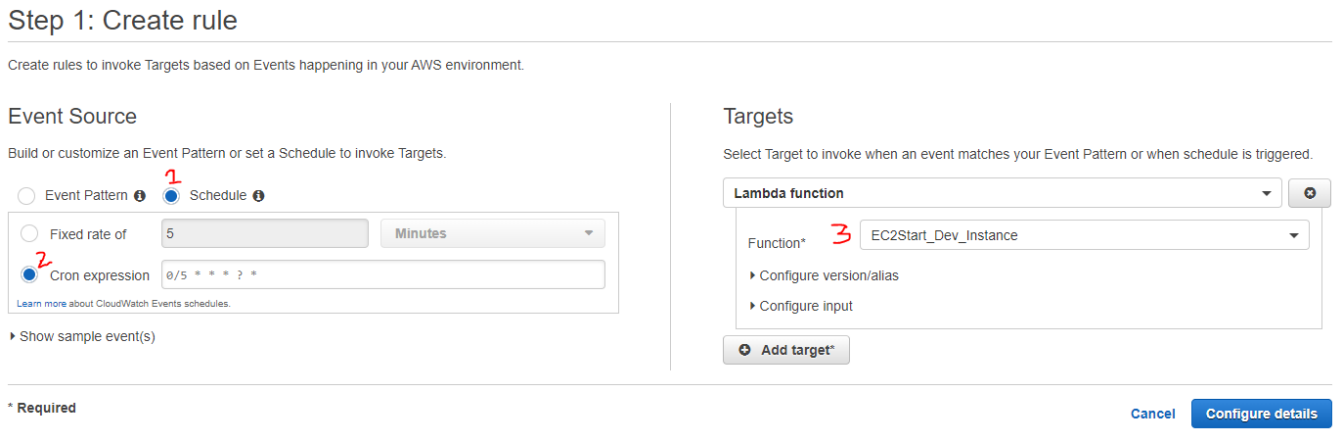
For stop function, create the event with stop name.

Create CloudWatch Rule :

To create the CloudWatch rule, select the CloudWatch service and navigate to Rules section, and click on Create rule button.



Select Cron expression under Schedule option, specify the time at what time you need to start/stop the instances. In the target section select lambda function respective to the CloudWatch rule.



Step 2: Configure rule details

Rule definition

Name* EC2Start_Dev_Instance

Description To Start EC2 Dev Instances

State ☐ Enabled

CloudWatch Events will add necessary permissions for target(s) so they can be invoked when this rule is triggered.

*** Required**

Cancel | Back | Update rule

Specify the rule name and description and update the rule.
Repeat the same process for Stop rule also.