Step by step guide on IoT data synchronization using MQTT

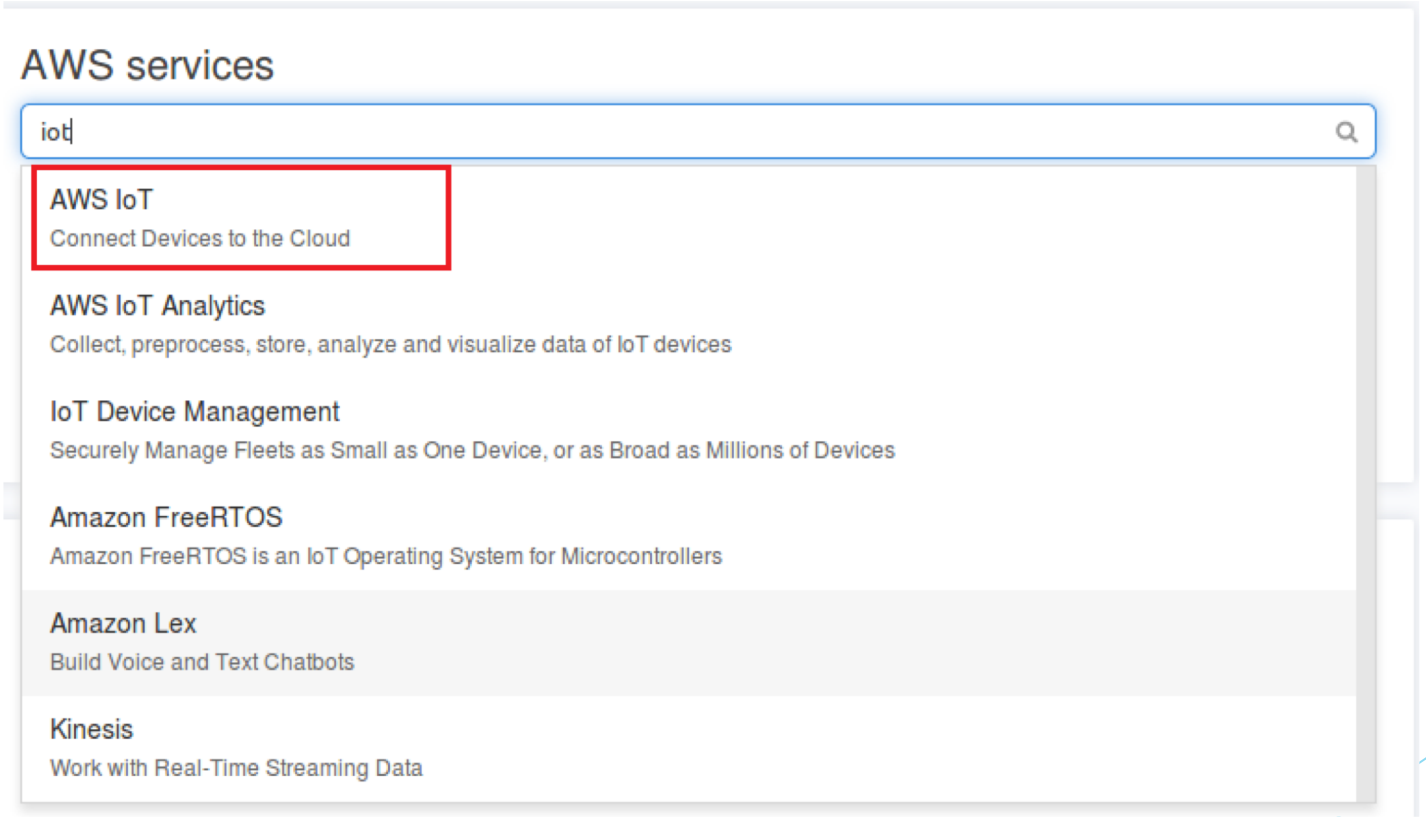
APIIT Education Group
Center of Technology and Innovation
Mustafa Mabrook

Topics & Structure of the report

- Create an IoT thing
- Add a certificate to the created thing
- Add a new policy and attach it to the created certificate
- Subscribe to an IoT topic
- Connect to the thing via a python script
- Publish data to the subscribed topic
- Add an action on receiving an event (Insert to DynamoDB)

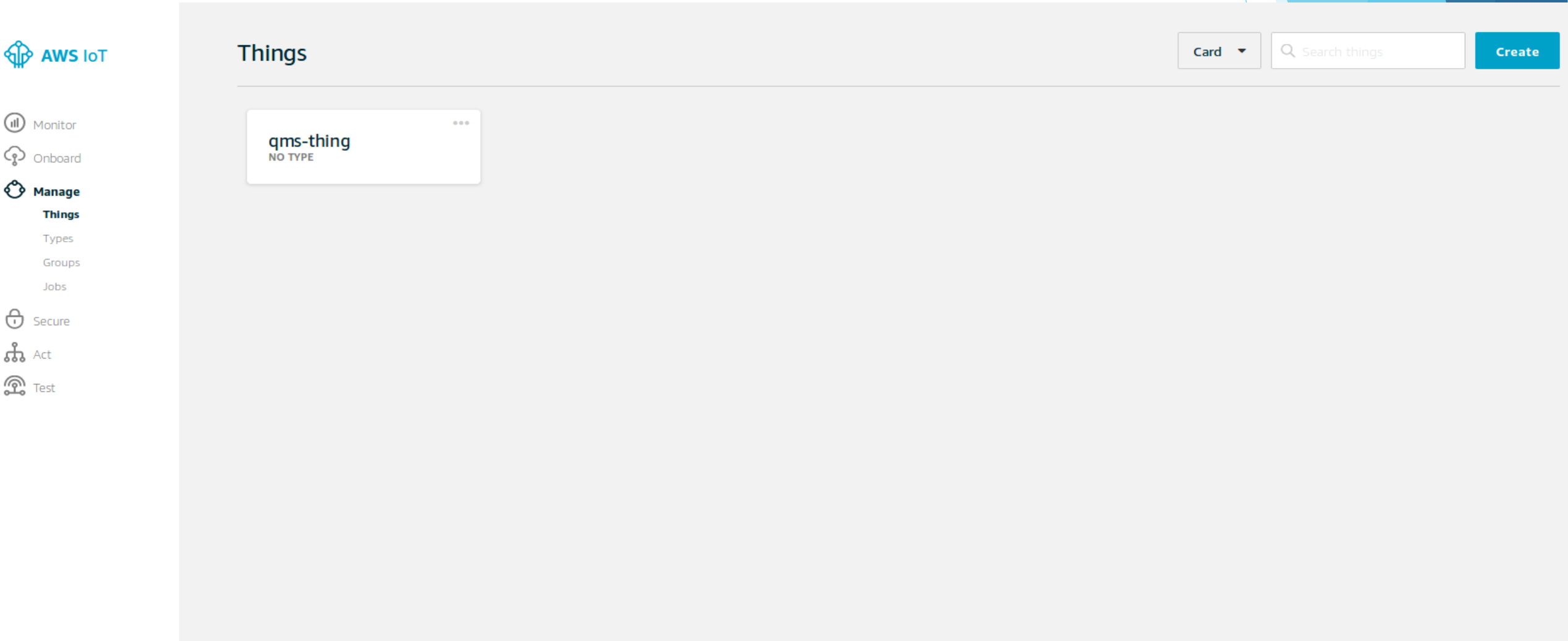
Creating an IoT Thing

1. Navigate to the AWS IoT from AWS console



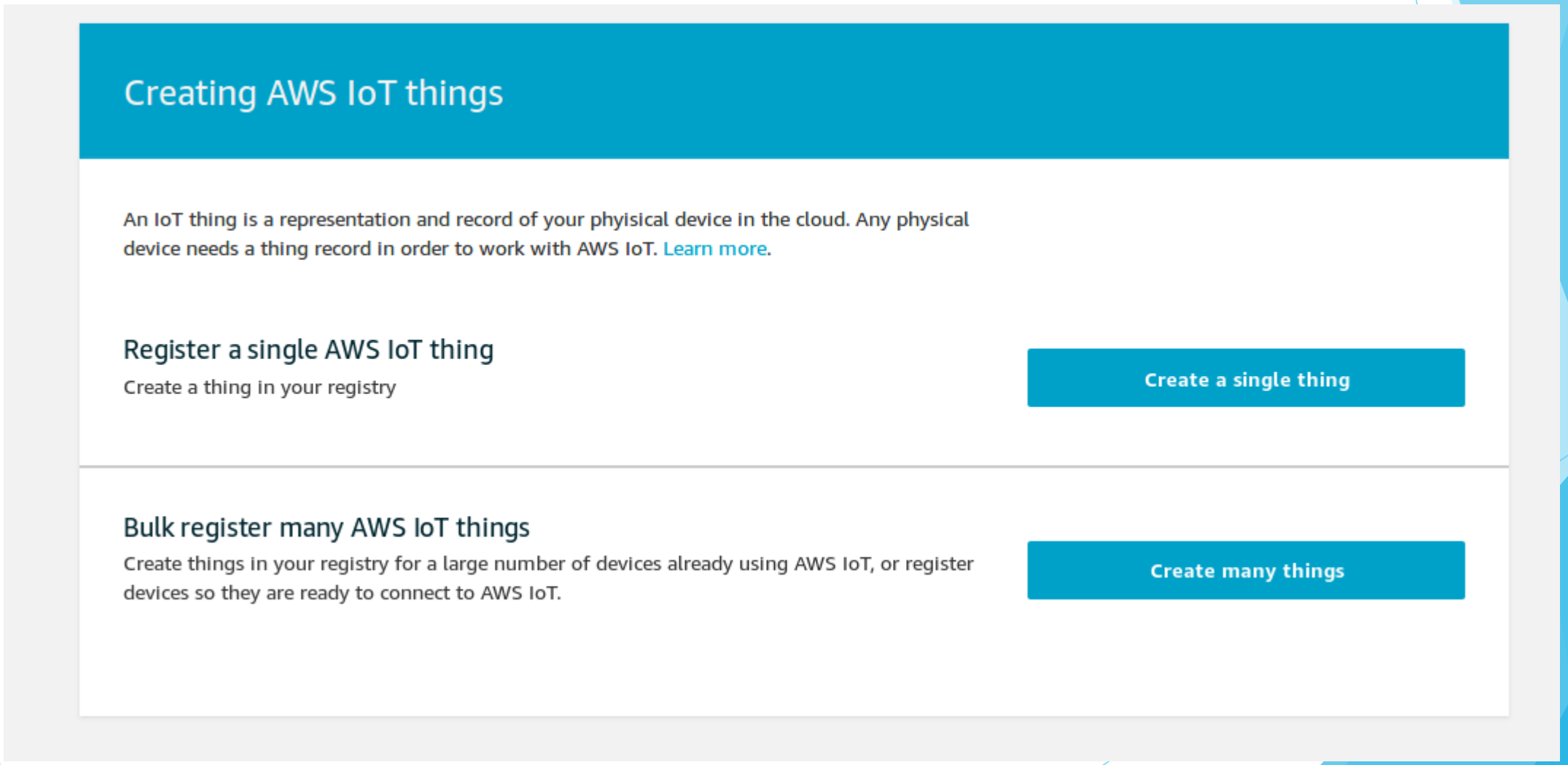
Creating an IoT Thing

2. Navigate to Manage Things option and then click on create button



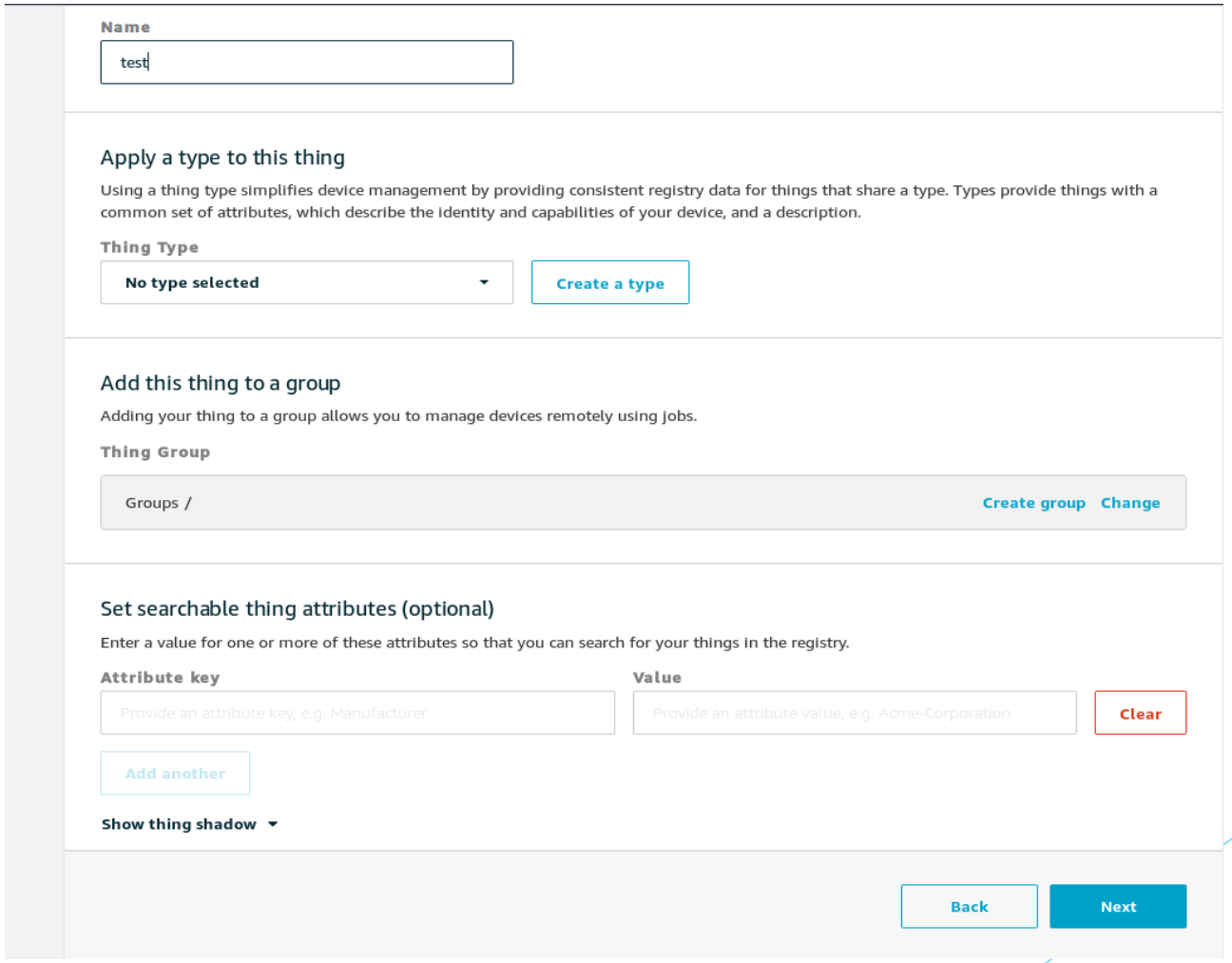
Creating an IoT Thing

3. Choose “Create a single thing” option



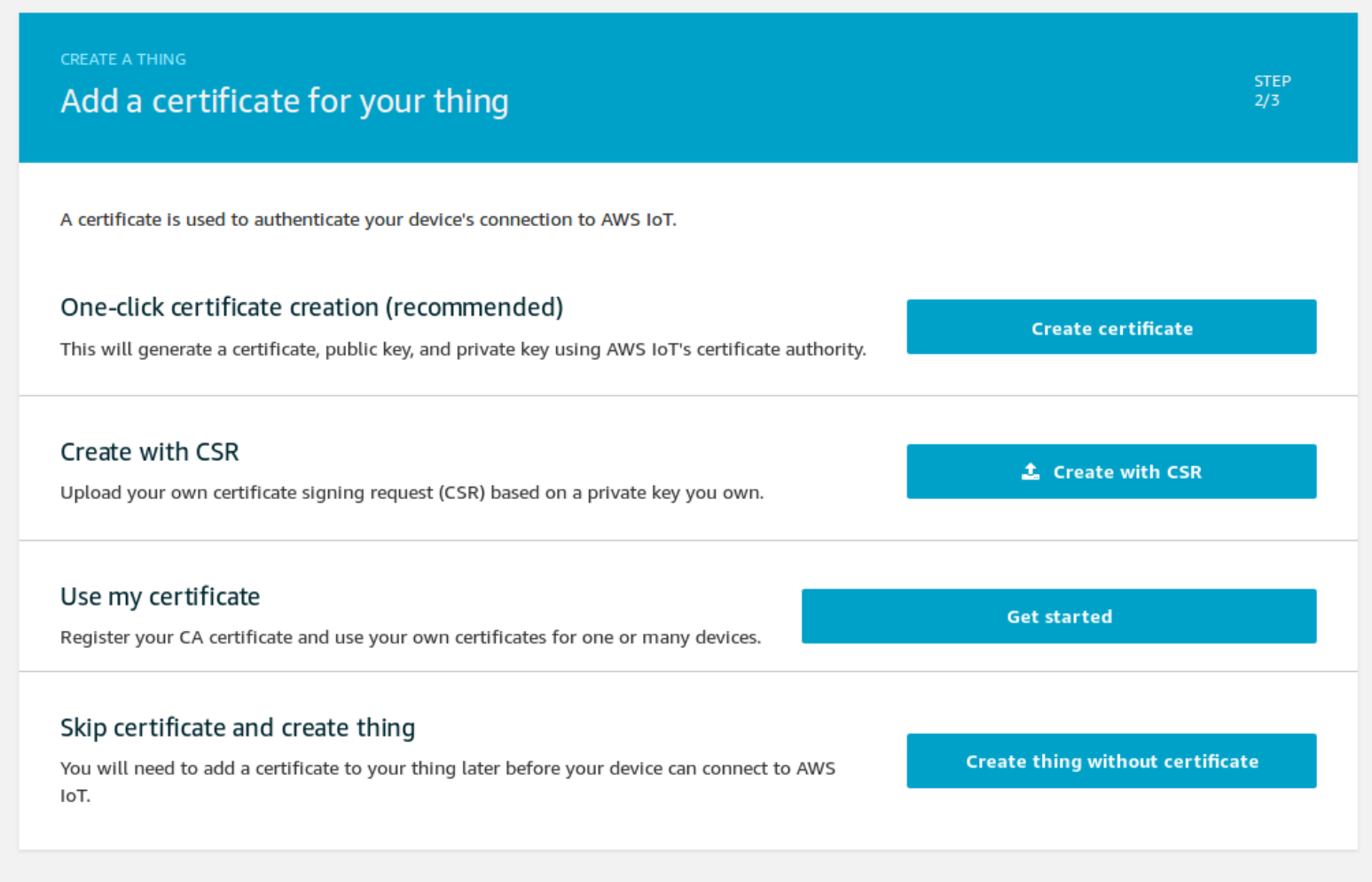
Creating an IoT Thing

4. Key in the thing name, choose the thing group if any and then click next



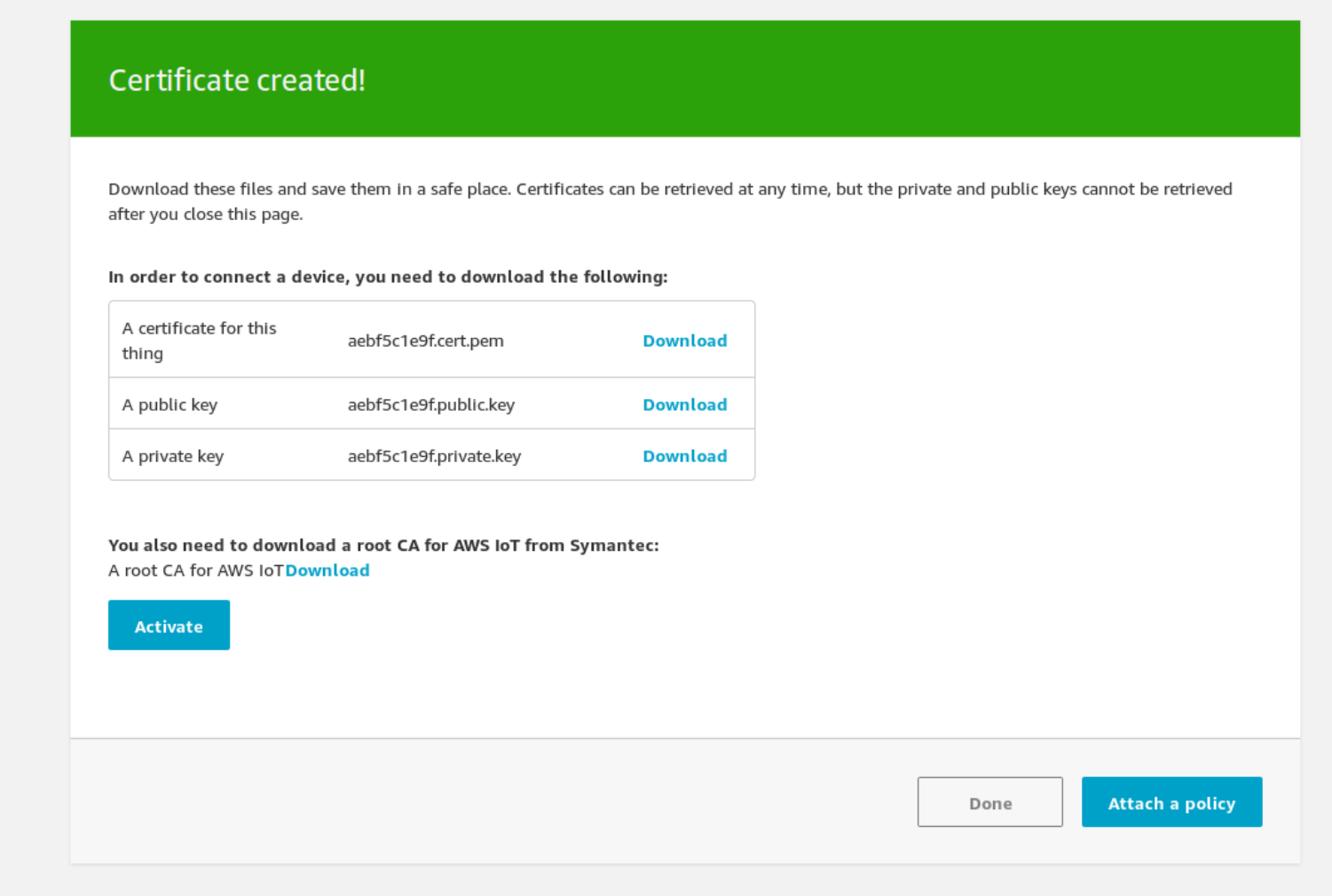
Add a certificate to the created thing

1. Choose “Create certificate” option



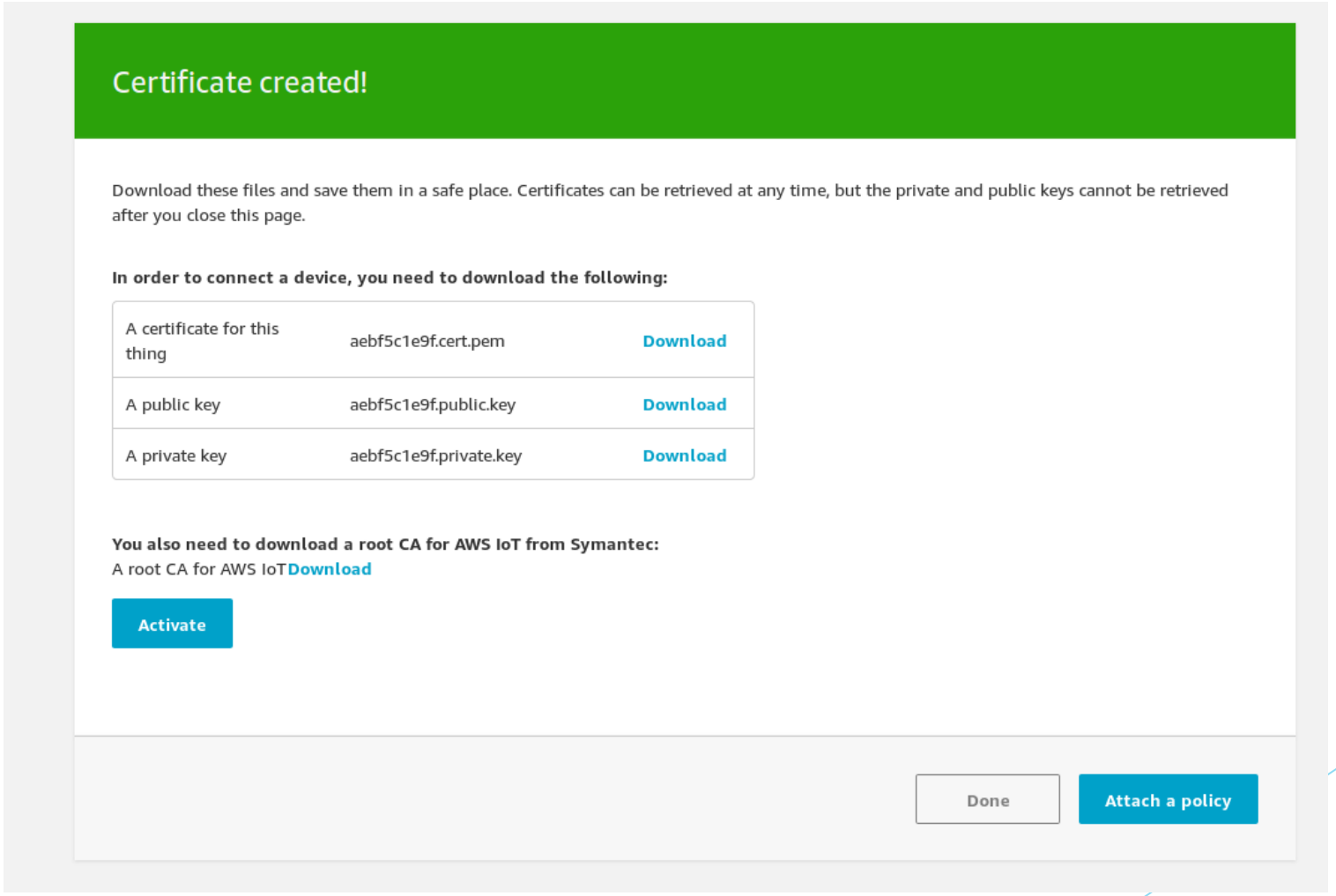
Add a certificate to the created thing

2. Download the created certificate, the public key, the private key, and root CA



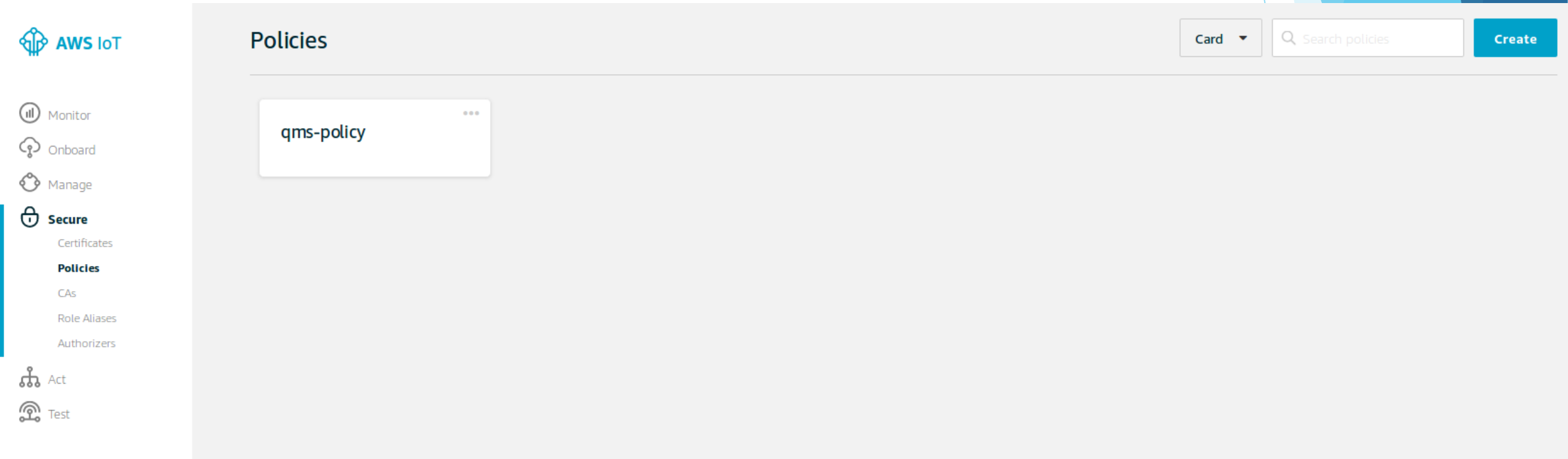
Add a certificate to the created thing

3. Activate the root CA and then click Done



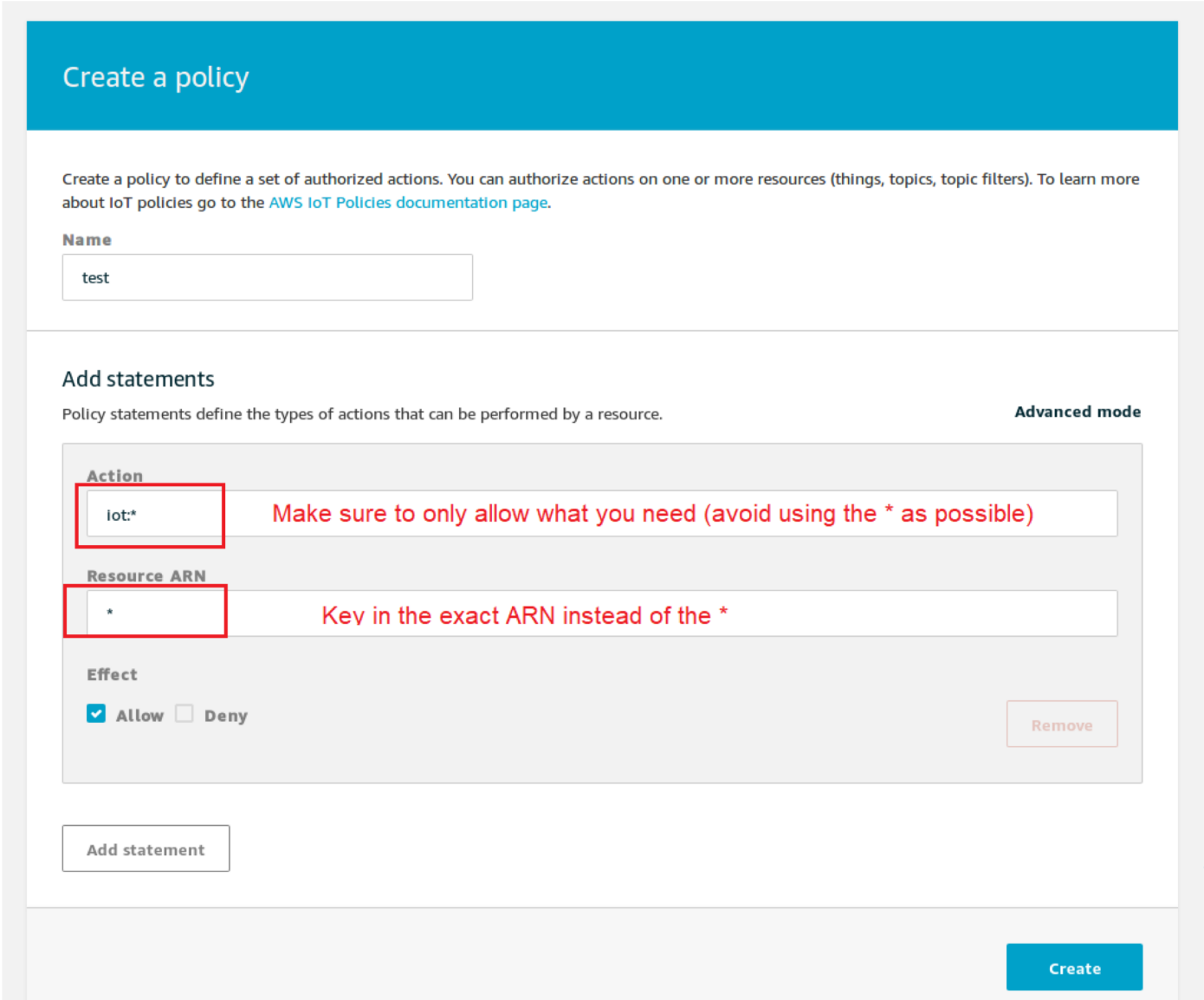
Add a new policy

1. Navigate to “Secure” menu option and then select “Policies”
2. Click on the “Create” button to create a new policy



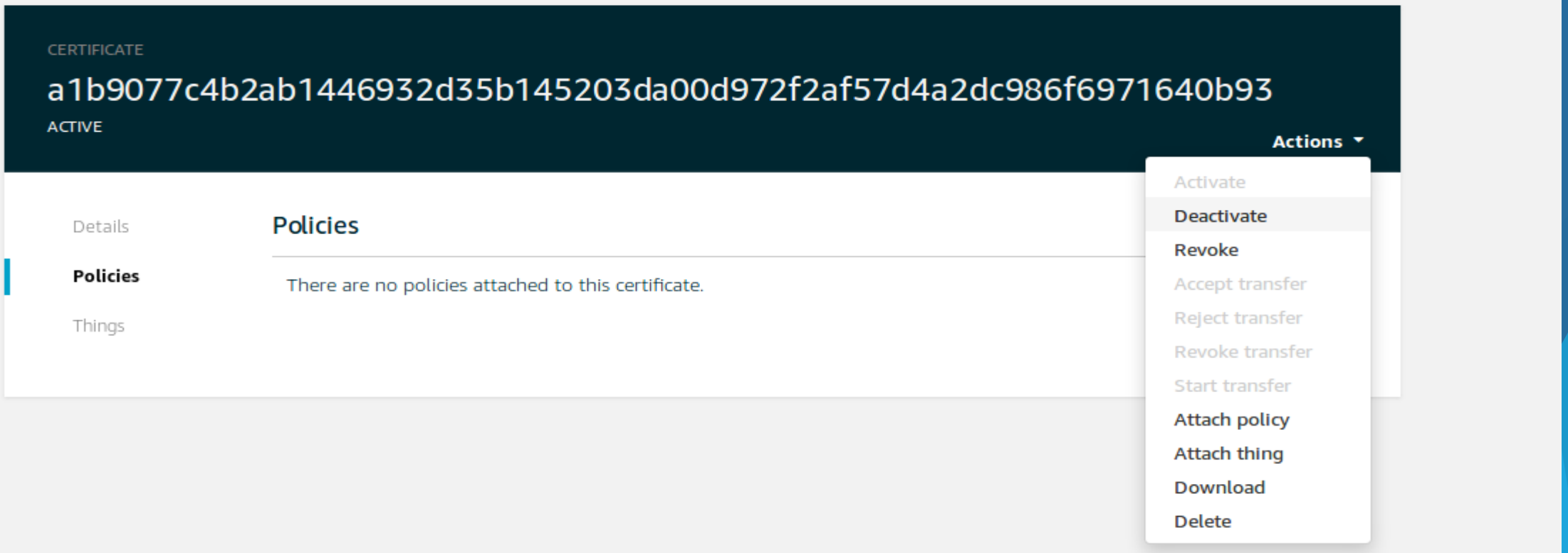
Add a new policy

3. Key in the policy name and specify the action and the resource ARN
4. Choose the appropriate effect and click create



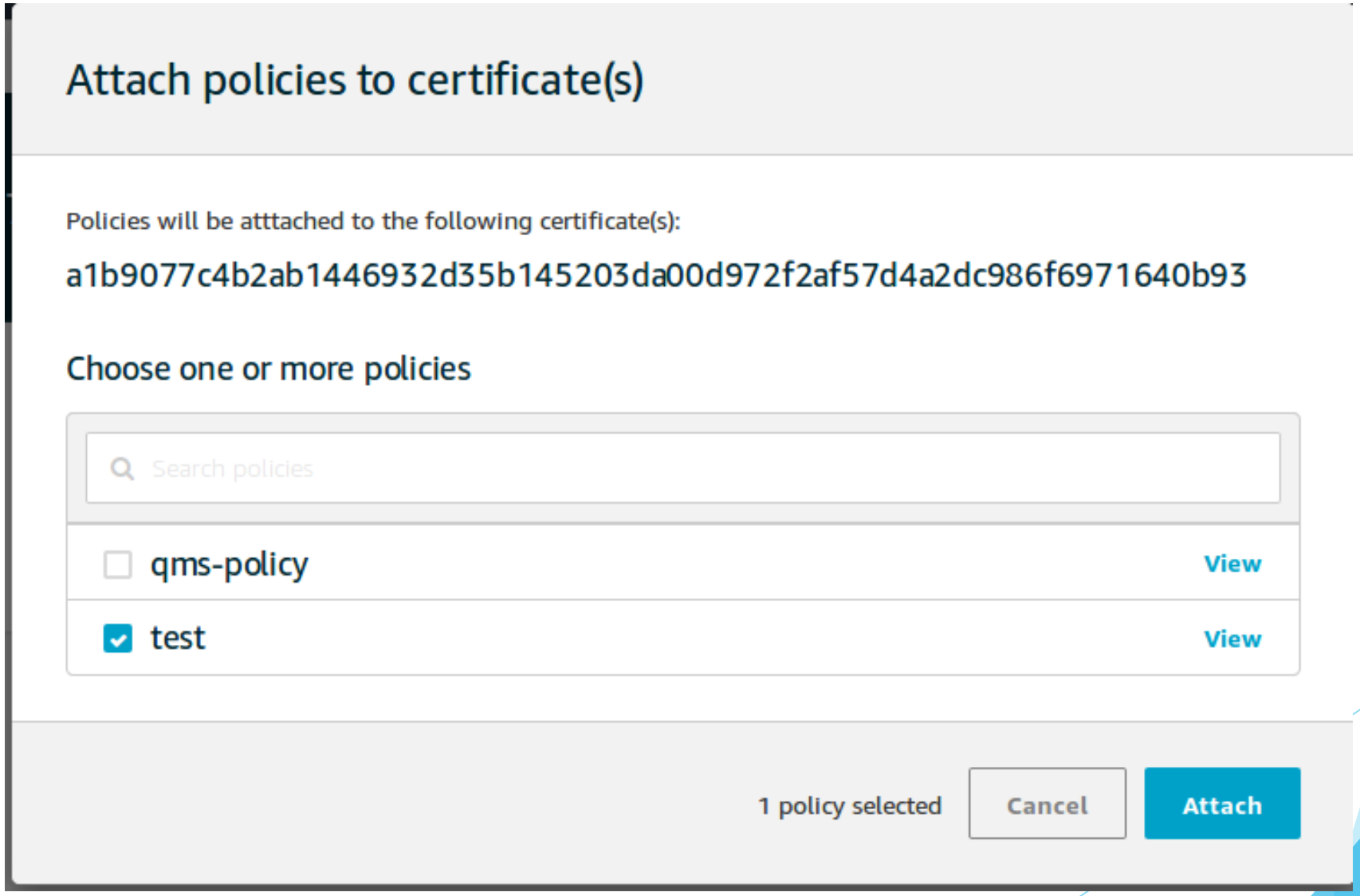
Attach the created policy to the certificate

1. Navigate to “Secure” menu option and then select “Certificates”
2. Select the created certificate and navigate to “Policies”
3. From the “Actions” option, select the “Attach policy” option



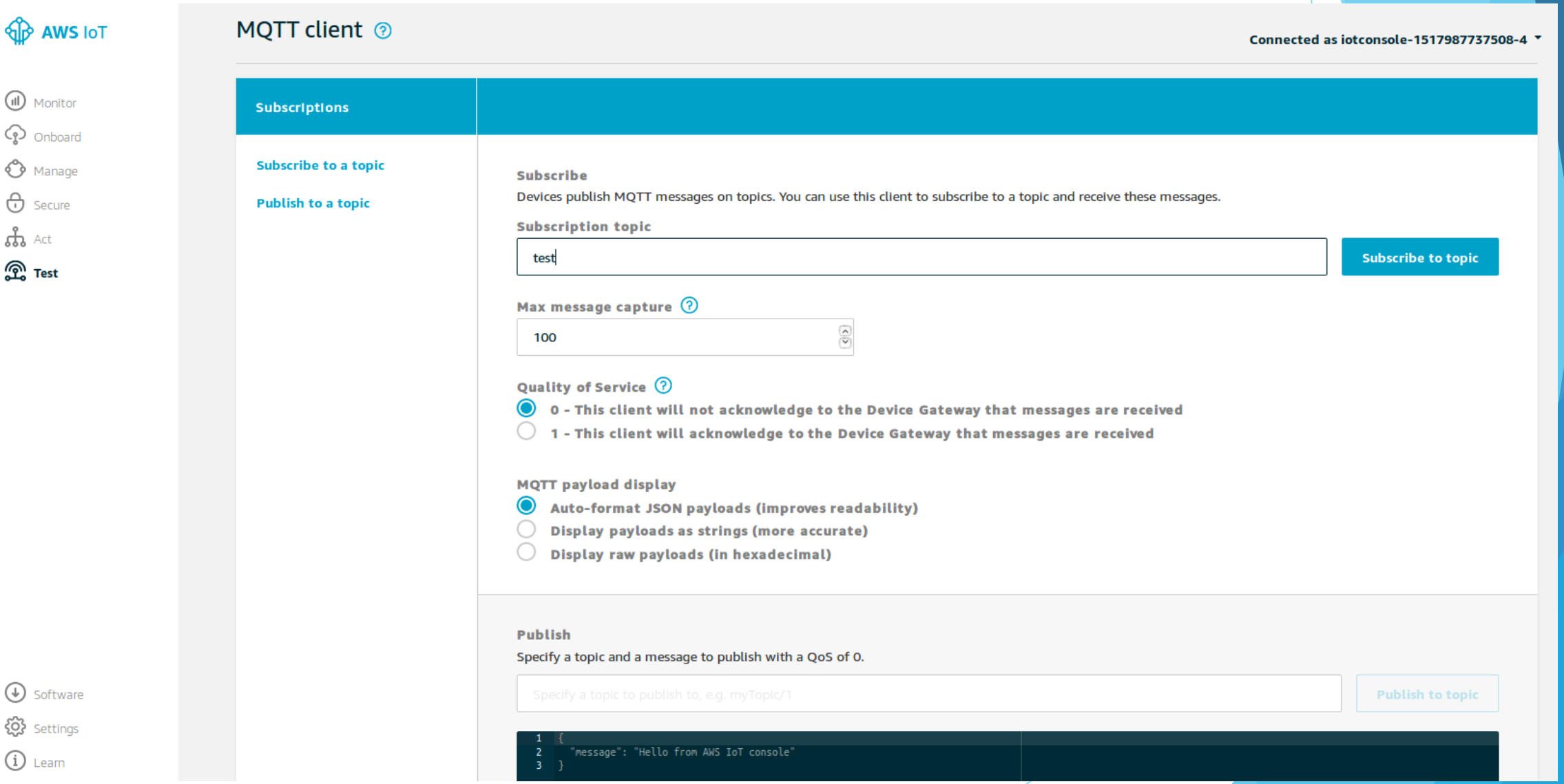
Attach the created policy to the certificate

4. Select the created policy which is “test” in this case and click “Attach”



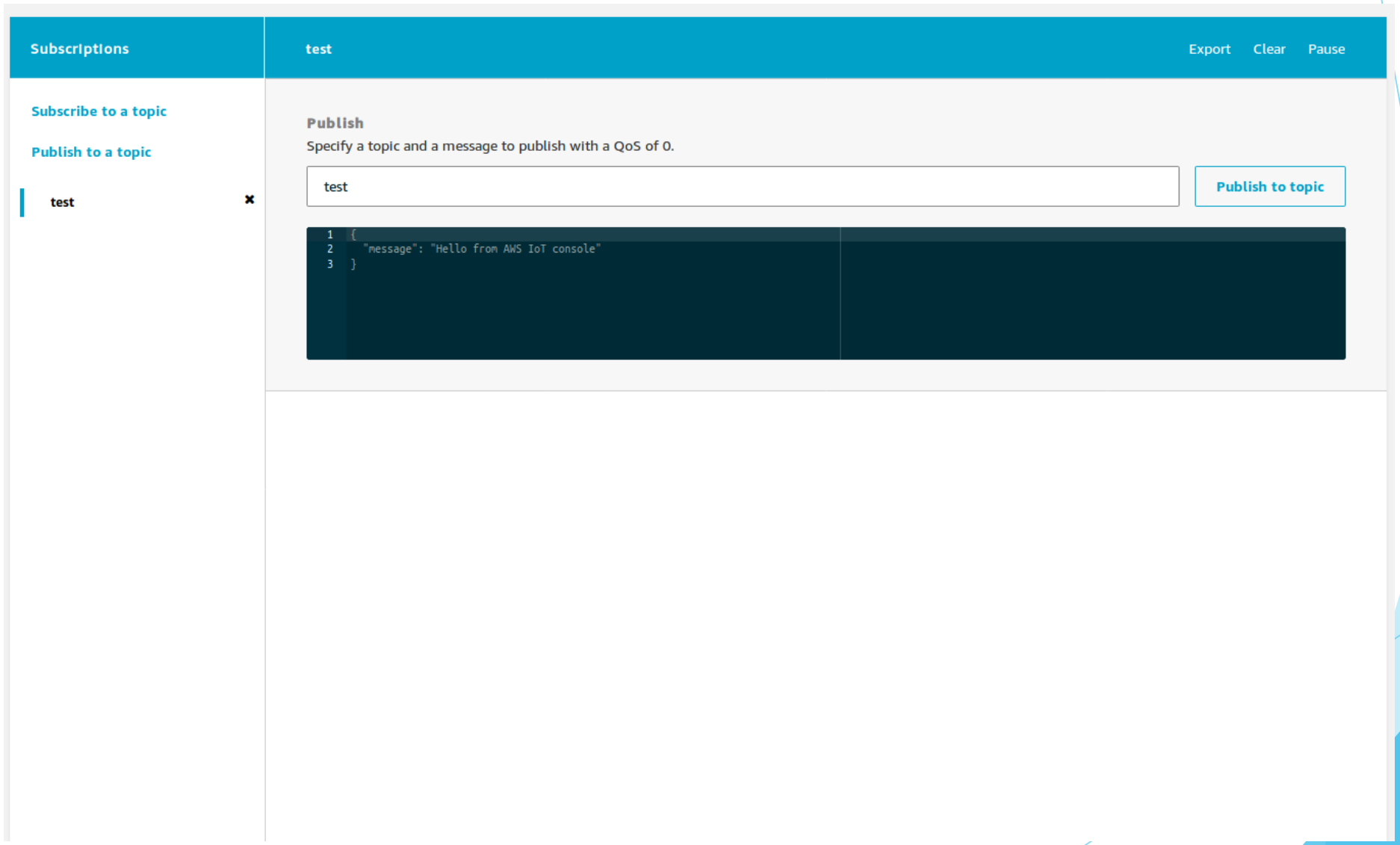
Subscribe to an IoT topic

1. Key in the topic name, choose a QoS, choose the MQTT payload format and then Click “Subscribe to topic”



Subscribe to an IoT topic

2. Keep the window open and move on to the next step which is connection to the thing via a python script



Connect to the thing via a python script

1. Use the following script to connect to the created thing using the appropriate account's endpoint and the downloaded certificates/keys
2. Call the sync function and pass the topic name and the message to be sent

```
from AWSIoTPythonSDK.MQTTLib import AWSIoTMQTTClient
import json


# Initialization
client = AWSIoTMQTTClient('id')
client.configureEndpoint('                .iot.ap-southeast-1.amazonaws.com', 8883)
client.configureCredentials('root.pem', 'private.pem.key', 'cert.pem.crt')
client.configureOfflinePublishQueueing(-1)  # Infinite offline publish queuing


def sync(topic, payload):
    try:
        client.connect()
        client.publish(topic, json.dumps(payload), 1)
        client.disconnect()
    except:
        return

sync("test", "khachapuri")
```

AWS account endpoint

Downloaded certifcates/keys

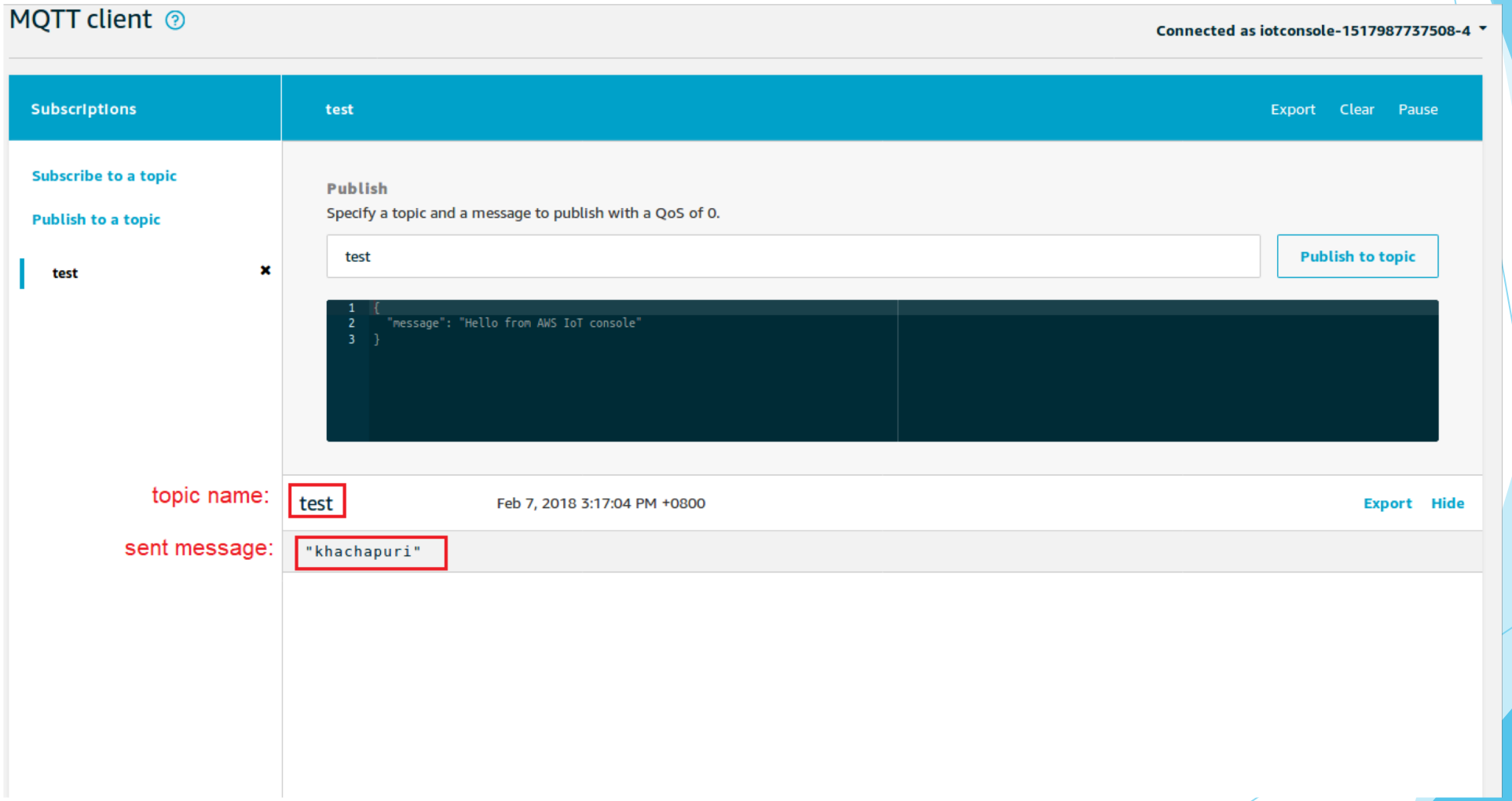
topic name

message to be sent

Note: The endpoint can be found in the "Settings" menu option in the bottom left corner

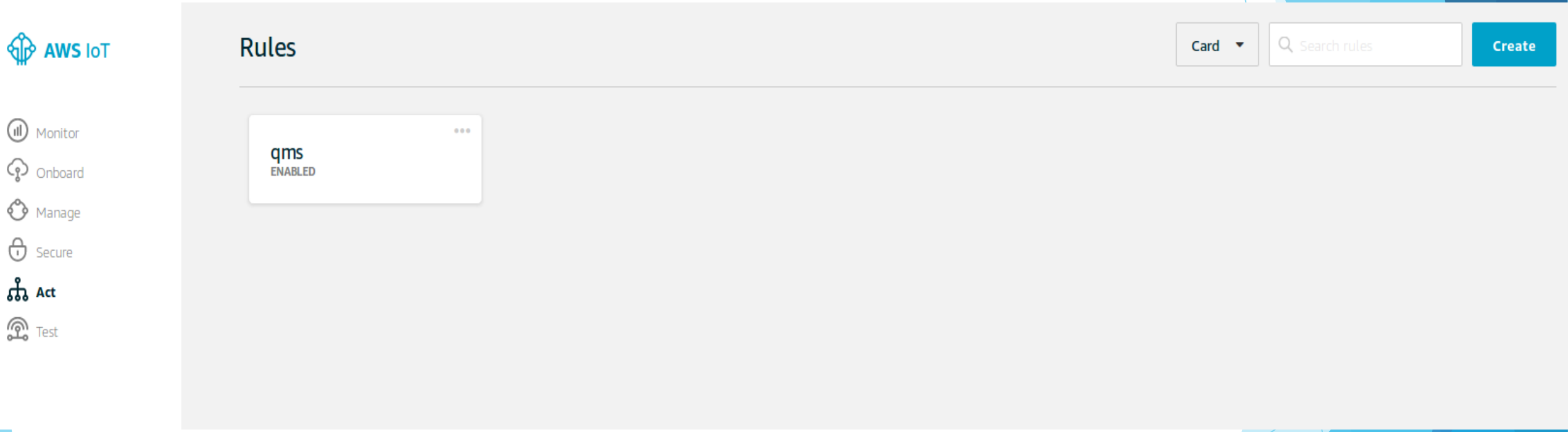
Connect to the thing via a python script

3. Go back to the AWS console to see the sent message



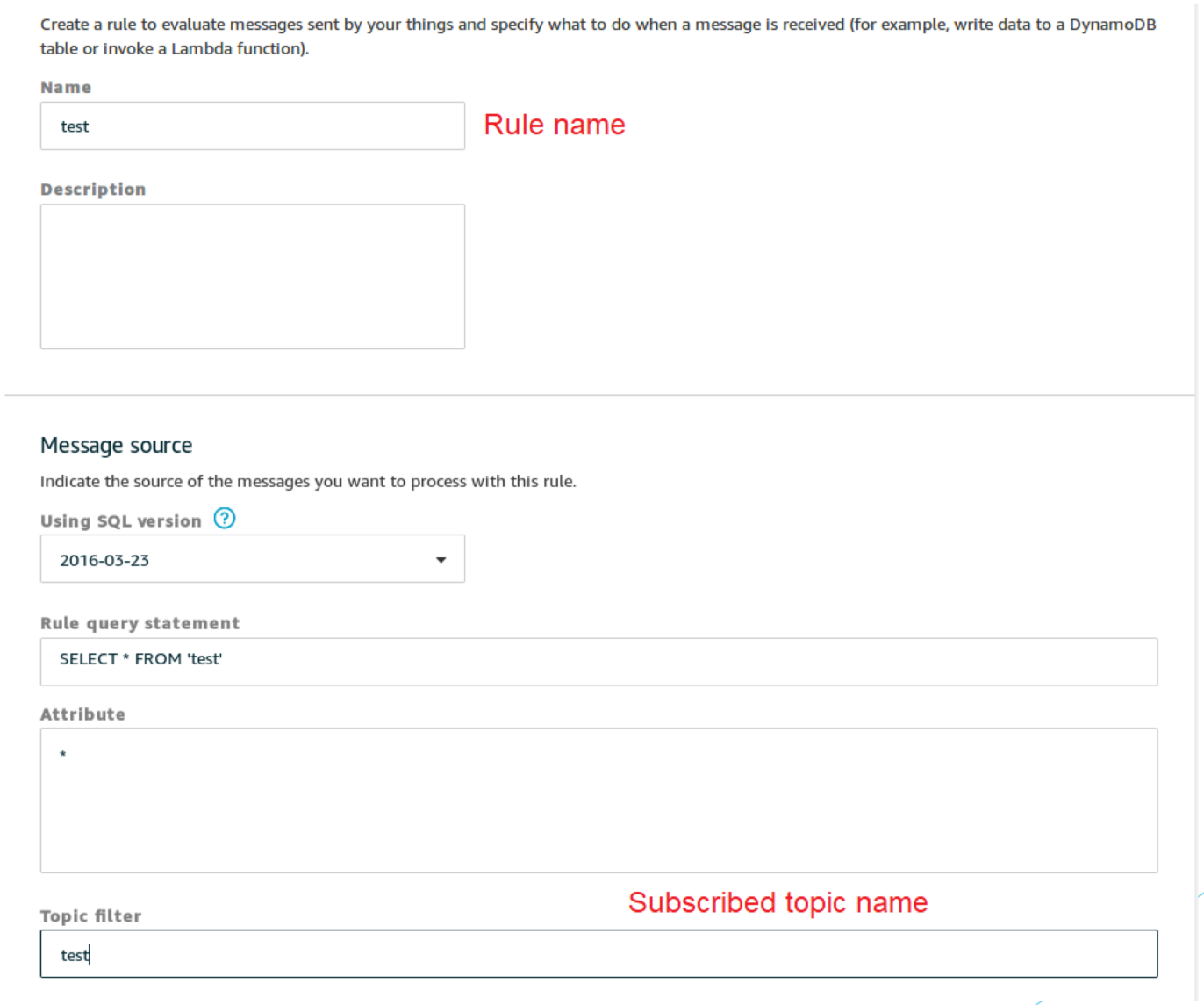
Add an action on receiving an event

1. Navigate to the “Act” menu option and then click “Create”



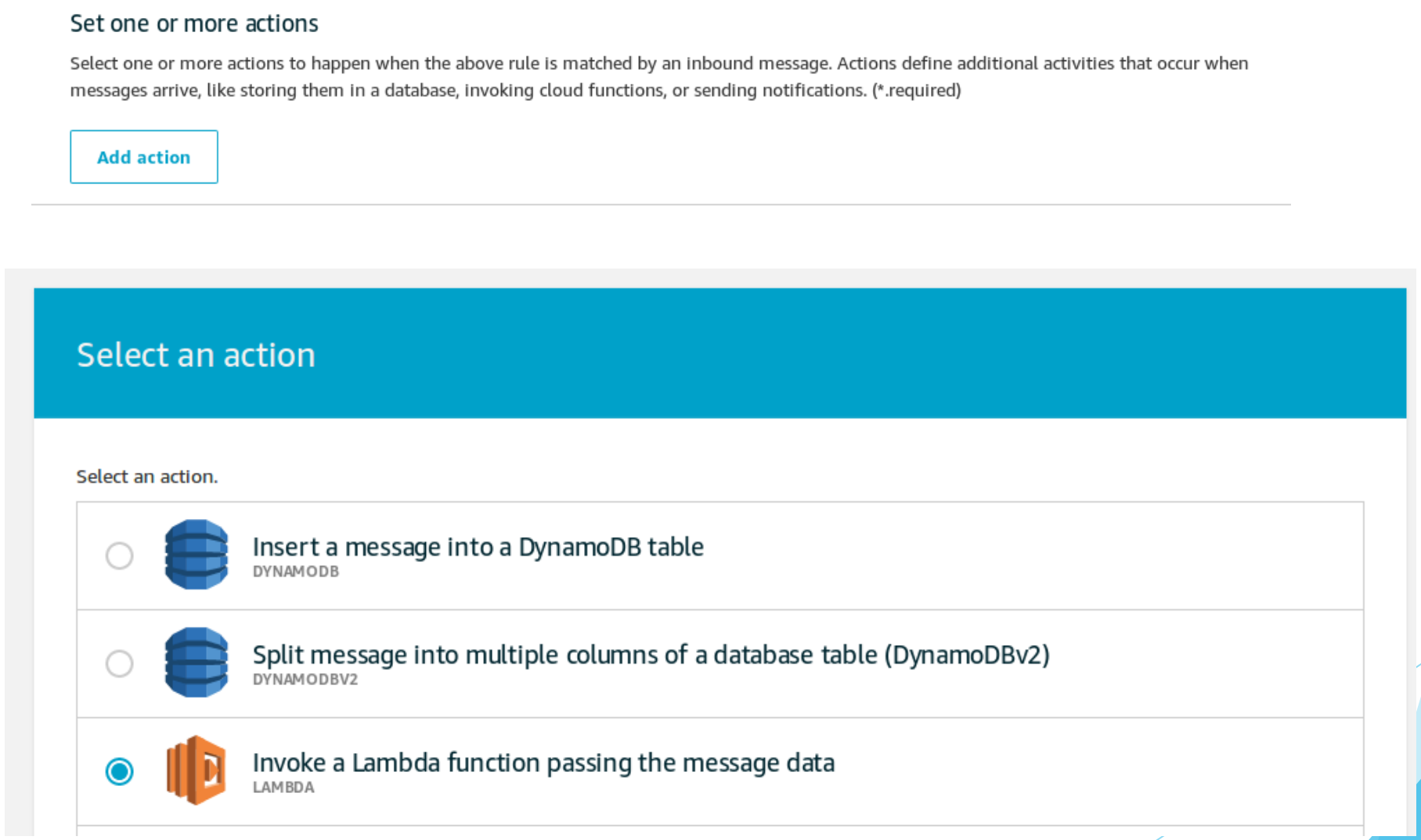
Add an action on receiving an event

2. Key in the rule name, the attribute, and the topic name (Make sure the topic name is the same one we published to)



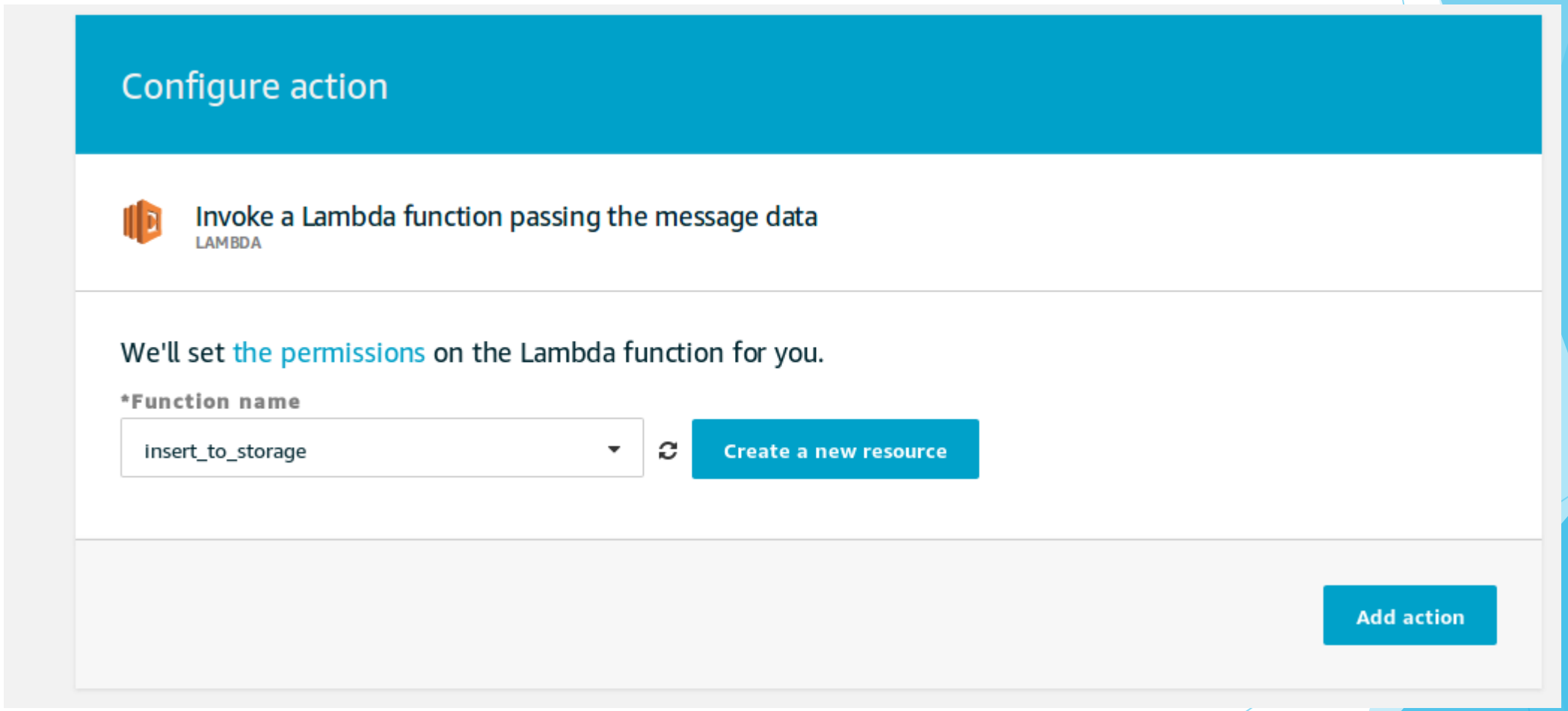
Add an action on receiving an event

3. Click on “Add action” and choose (Invoke a Lambda function option)

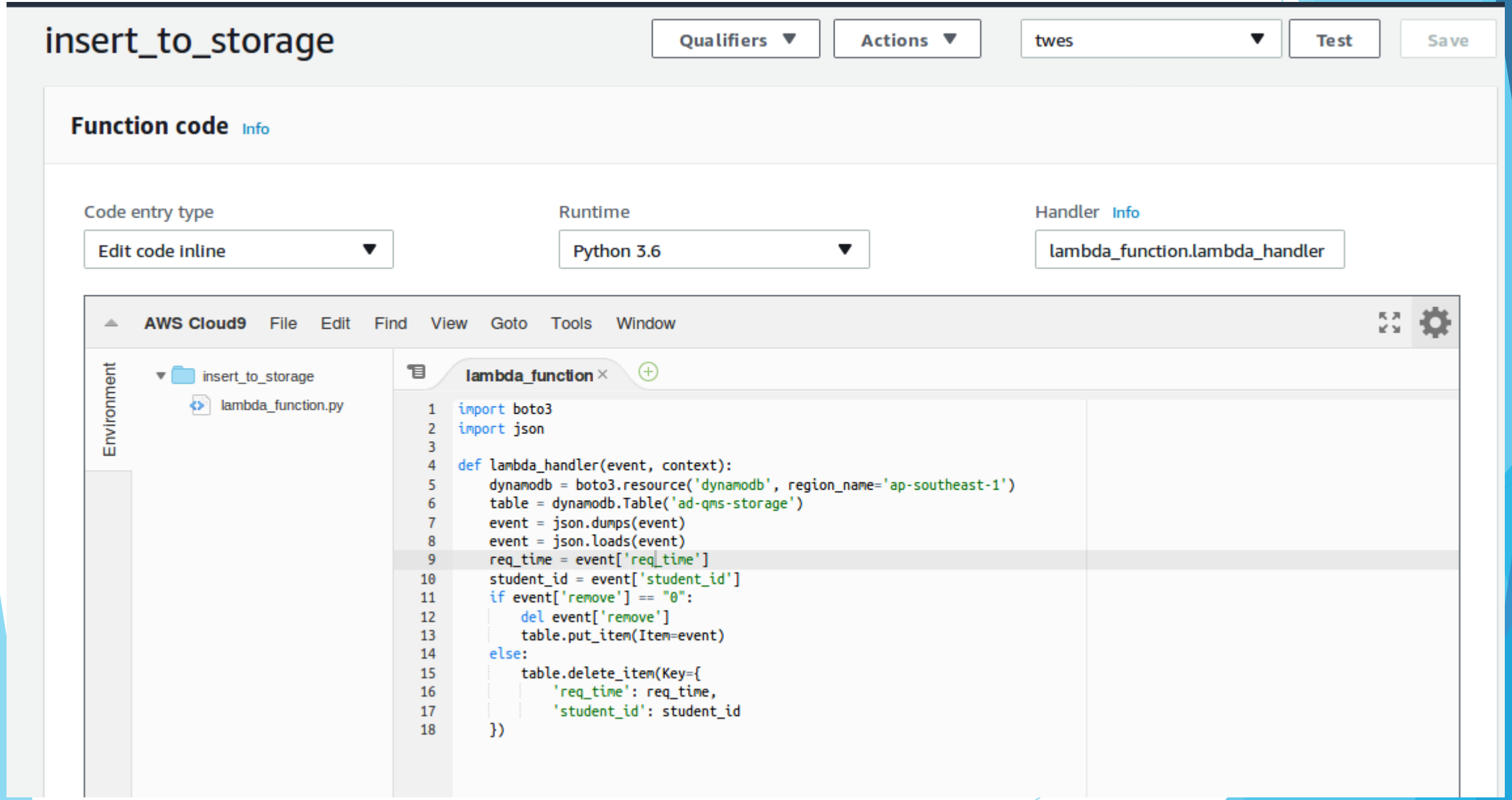


Add an action on receiving an event

4. Choose the lambda function name to be executed from the dropdown list and click on “Add action”



Lambda function code sample



View the DynamoDB table

Scan: [Table] ad-qms-storage: req_time, student_id

Viewing 1 to 1 items

Scan | [Table] ad-qms-storage: req_time, student_id

Add filter

Start search

req_time	student_id	canceled	counter	employee_username	end_serving_time	services	serving_time	start_serving_time
07/02/18 04:58:	TP040902	0	Null	Null	Null	5	Null	Null

Q&A

Thank You