

Lambda, Cloudwatch ec2 stop/start

Lambda ?

ec2 Lambda 1ms

1. Lambda IAM

** Lambda

[Policy]

: Lambda_policy

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "ec2:Describe*",
        "ec2:Start*",
        "ec2:Stop*"
      ],
      "Resource": "*"
    }
  ]
}
```

[Role]

: Lambda_role

policy(Lambda_policy) 가

2. Lambda

Author from scratch

Lambda Function : StartEC2Instance / StopEC2Instance

Runtime : Python3.8

Permissions - Use an existing role - role

[Lambda Code]

```
import boto3
region = 'ap-northeast-2'
instances = []
ec2_r = boto3.resource('ec2')
ec2 = boto3.client('ec2', region_name=region)
//
for instance in ec2_r.instances.all():
    instances.append(instance.id)

def lambda_handler(event, context):
    ec2.start_instances(InstanceIds=instances)
    print('started your instances: ' + str(instances))
```

ec2

```
import boto3
region = 'ap-northeast-2'
instances = []
ec2_r = boto3.resource('ec2')
ec2 = boto3.client('ec2', region_name=region)
```

```
for instance in ec2_r.instances.all():
    for tag in instance.tags:
        if tag['Key'] == 'auto-schedule':
            if tag['Value'] == 'auto':
                instances.append(instance.id)
```

```
def lambda_handler(event, context):
    ec2.start_instances(InstanceIds=instances)
    print('start your instances: ' + str(instances))
```

가

Lambda - Deploy - Test

- 1) IAM Policy, Lambda code
- 2) Lambda-General configuration - Timeout 3(default) -> 15
- 3) region ec2 Instance state가 1EA terminate
가

3. CloudWatch

Event - Rules - Go to Amazon EventBridge

step1)

 : StartEC2Instance
 Event bus : default
 Rule type : Schedule

step2)

GMT cron

step3)

target types : AWS service
 Select a target : Lambda function
 Function : Lambda Function (StartEC2Instance)
 ++ 가 가

step4)

() 4. Lambda code /

[SNS]

topic
 subscription
 Protocol : Email
 Endpoint :

[Lambda]

Configuration - Destinations - Add destination
 source : Asynchronous invocation
 Condition : On Failure / On Success
 Destination Type : SNS topic
 Destination : topic