

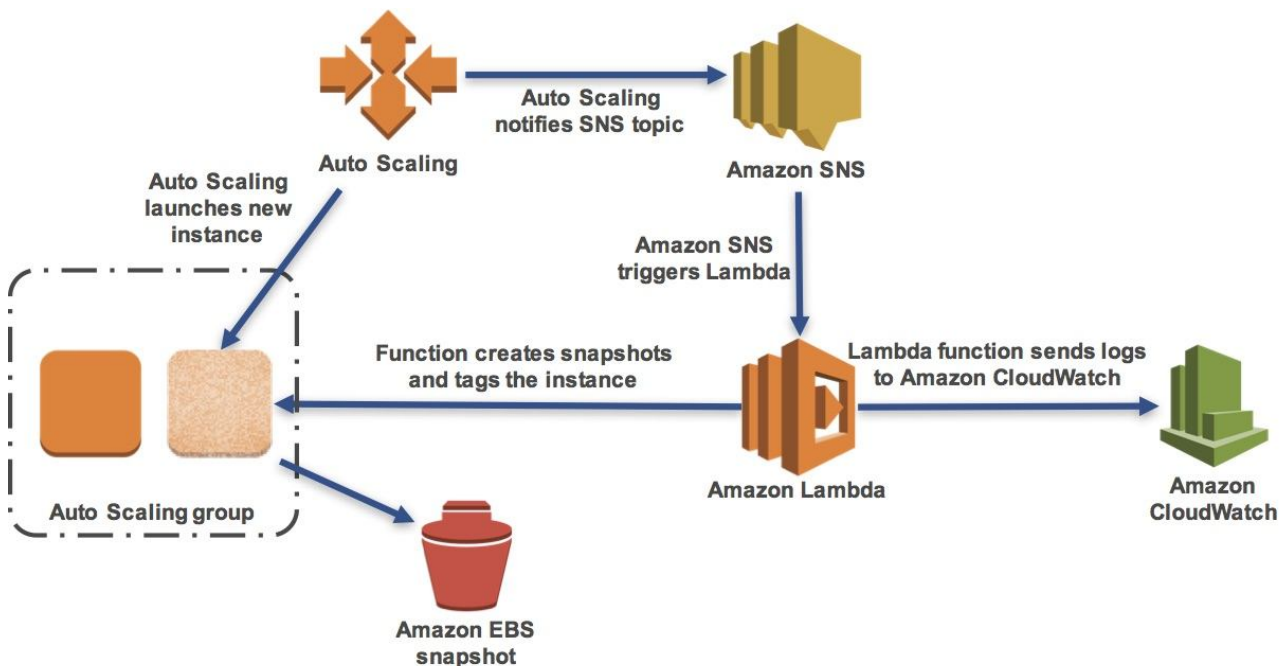
알림을 사용하여 AWS Lambda 트리거

상당수의 AWS 서비스는 이벤트가 발생하면 자동으로 알림을 생성할 수 있습니다. 이러한 알림을 사용하여 사람이 개입하지 않고도 자동화된 동작을 수행할 수 있습니다.

이번 실습에서는 AWS Lambda 함수를 생성합니다. AWS Lambda 함수는 자동으로 Auto Scaling이 실행하는 새 Amazon EC2 인스턴스를 태그하고 스냅샷을 남깁니다.

실습 시나리오는 다음과 같습니다.

- 이미 구성해둔 **Auto Scaling group**이 있습니다.
- Auto Scaling을 트리거하여 스케일아웃하고 새로운 **Amazon EC2 인스턴스**를 실행합니다.
- 이 인스턴스가 **Amazon Simple Notification Service(SNS)** 주제로 알림을 전송합니다.
- SNS 주제는 다음 작업을 수행하는 **AWS Lambda 함수**를 트리거합니다.
 - 인스턴스에 연결된 Amazon EBS 볼륨의 **스냅샷**을 생성합니다.
 - 인스턴스에 **태그**를 추가합니다.
 - Amazon CloudWatch 로그에 **로그 정보**를 전송합니다.



본 실습에서는 다음 작업을 수행합니다.

- Auto Scaling 이벤트에 대한 알림의 대상이 되는 **Amazon Simple Notification Service(Amazon SNS)** 주제를 생성합니다.
- 새 Amazon EC2 인스턴스가 시작되면 **알림을 전송**하도록 Auto Scaling 그룹을 구성합니다.
- Lambda 함수가 필요한 권한을 보유한 **IAM 역할**을 구성합니다. Amazon SNS 주제로부터 Auto Scaling 이벤트가 발생했다는 메시지를 수신하면 호출되는 **AWS Lambda 함수**를 생성합니다.

목표

본 실습을 완료하면 다음과 같은 작업을 수행할 수 있습니다.

- 알림을 전송하도록 Auto Scaling을 구성합니다.
- 알림에 응답하도록 AWS Lambda 함수를 생성합니다.

소요 시간

본 실습에는 약 **30분**이 소요됩니다.

AWS Management Console 액세스

1. [1]실습 제목 오른쪽에서 **Start Lab**을 클릭하여 Qwiklabs를 시작합니다.

Start Lab

1. [2] Qwiklabs 페이지의 **Connect** 탭에서 **Password**를 클립보드로 복사한 후 **Open Console**을 클릭합니다.

Open Console

1. [3] 다음 단계에 따라 AWS Management Console에 로그인합니다.

- **User Name**에 'awsstudent'를 입력합니다.
- **Password**에 클립보드에서 복사한 암호를 붙여넣습니다.
- **Sign In**을 클릭합니다.

작업 1: SNS 주제 생성

이 작업에서는 Auto Scaling 그룹이 알림 대상으로 사용할 **Amazon Simple Notification Service(SNS)** 주제를 생성합니다.

1. [4]AWS Management Console의 **Services** 메뉴에서 **Simple Notification Service**를 클릭합니다.
2. [5]**Get started**를 클릭합니다.

Get Started 버튼이 보이지 않으면 다음 단계로 건너뛰면 됩니다.

1. [6]**Create topic**을 클릭합니다.
2. [7]**Create new topic 대화 상자에서 다음과 같이 설정합니다.

Topic name	'ScaleEvent'
Display name	'ScaleEvent'

1. [8]**Create topic**을 클릭합니다.

이제 주제가 알림을 받을 준비가 되었습니다.

작업 2: Auto Scaling을 구성하여 이벤트 전송

이번 작업에서는 Auto Scaling 그룹을 구성하여 새로운 Amazon EC2 인스턴스가 그룹에서 시작될 때 SNS 주제에 알림을 전송합니다.

1. [9]AWS Management Console의 **Services** 메뉴에서 **EC2**를 클릭합니다.
2. [10]탐색 창에서 **Auto Scaling Groups**(아래로 스크롤해야 보일 수 있음)를 클릭합니다.

본 실습을 위해 자동으로 생성된 Auto Scaling 그룹이 보일 것입니다.

그룹 목록이 보이지 않으면 **Auto Scaling Group: 1**을 클릭하십시오.

1. [11]화면 아래쪽의 **Notifications** 탭을 클릭합니다.
2. [12]**Create notification**을 클릭합니다.

구분선을 위쪽으로 드래그하여 아래쪽 창을 더 크게 만들 수 있습니다.

1. [13] **Send a notification to**에서 **ScaleEvent**가 선택되었는지 확인합니다. (이것은 방금 생성한 알림 주제입니다.)
2. [14]**Whenever instances**에서 **launch**만 선택되었는지 확인합니다. 다른 옵션은 모드 선택이 해제되어야 합니다.
3. [15]**Save**를 클릭합니다.

이제 Auto Scaling 그룹에서 새 인스턴스가 시작될 때마다 Auto Scaling이 SNS 주제에 메시지를 전송합니다.

작업 3: Lambda 함수를 위한 IAM 역할 생성하기

이번 작업에서는 EC2 인스턴스에서 작업을 수행하고 Amazon CloudWatch 로그에 메시지를 기록할 권한이 있는 **IAM role**을 생성합니다. 추후 이 역할을 Lambda 함수와 연결할 것입니다.

1. [16]**Services** 메뉴에서 **IAM**을 클릭합니다.
2. [17]탐색 창에서 **Roles**를 클릭합니다.
3. [18]**Create new role**을 클릭합니다.
4. [19]**Select role type** 페이지의 **AWS Lambda** 행에서 **Select**를 클릭합니다.

실습에서는 Amazon EC2 명령을 호출하는 모든 권한을 포함하여 *사전에 구성된 정책*을 사용할 것입니다. 이러한 유형의 정책은 시스템 관리자가 유지 관리할 수 있습니다.

1. [20]**Attach Policy** 페이지의, **Filter** 유형에 'AmazonEC2FullAccess'를 입력합니다.
2. [21]**AmazonEC2FullAccess**를 선택(체크)합니다.
3. [22]**Next Step**을 클릭합니다.
4. [23]**Role name**에 'SnapandTag'를 입력합니다.
5. [24]**Create role**을 클릭합니다.

이제 IAM 역할에는 스냅샷 생성이나 태그 추가 등의 Amazon EC2 인스턴스에서 작업을 수행할 권한이 포함됩니다.

Lambda 함수도 디버깅을 위해 로그 항목을 Amazon CloudWatch 로그에 기록할 권한이 필요합니다. 역할에 직접 적용되는 사용자 지정 정책인 **inline policy**를 생성하여 이러한 권한을 부여할 수 있습니다. 이러한 유형의 정책은 일치하는 사전 구성 정책이 없을 때 사용할 수 있습니다.

1. [25]**SnapshotTag**를 클릭합니다 (이름을 직접 클릭).
2. [26]화면 아래쪽의 **Inline Policies**를 클릭한 다음 **click here**를 클릭합니다.

정책 생성기에서 간단한 포인트 앤 클릭 인터페이스를 통해 정책을 생성할 수 있습니다. 하지만, 실습에서는 제공된 정책을 사용합니다.

1. [27]**Custom Policy**를 선택하고 **Select**를 클릭합니다.
2. [28]**Policy Name**에 'WriteLogs'를 입력합니다.
3. [29]아래의 **Copy Code Block**을 클릭하고 정책 문서 필드에 다음을 붙여넣습니다.

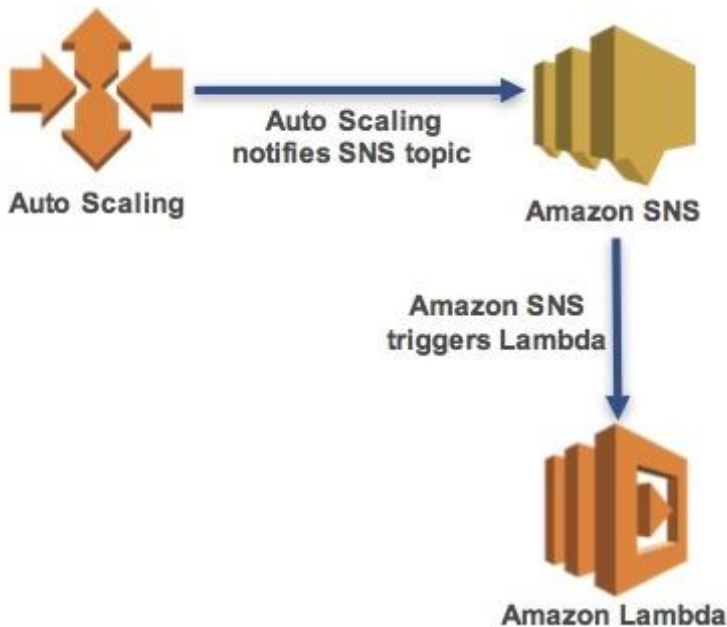
```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents"
      ],
      "Resource": "arn:aws:logs:*:*:*",
      "Effect": "Allow"
    }
  ]
}
```

이 정책은 Amazon CloudWatch 로그에 로그 그룹을 만들고 로그에 로그 이벤트를 기록하는 권한을 부여합니다. Lambda 함수 디버깅에 매우 유용한 정책입니다.

1. [30]**Apply Policy**를 클릭합니다.

Lambda 함수에 사용할 정책입니다.

작업 4: Lambda 함수 생성



이번 작업에서는 Auto Scaling이 새 EC2 인스턴스를 시작할 때 **Amazon SNS가 호출할 AWS Lambda 함수**를 생성합니다. Lambda 함수는 인스턴스에 연결된 Amazon EBS 볼륨의 스냅샷을 생성한 뒤 인스턴스에 태그를 추가합니다.

1. [31]**Services** 메뉴에서 **Lambda**를 클릭합니다.
2. [32]**Get Started Now**를 클릭합니다. (**Get Started Now**가 나타나지 않으면 다른 단계로 진행하십시오.)

Blueprints는 Lambda 함수 기록용 코드 템플릿입니다. 청사진은 Alexa skills 생성이나 Amazon Kinesis Firehose 스트림 처리와 같은 표준 Lambda 트리거용으로 제공됩니다. 이 실습에서는 사전에 작성된 Lambda 함수를 제공하므로 청사진을 사용하지 않습니다.

1. [33]**Select blueprint** 페이지에서 **Blank Function**을 클릭합니다.
2. [34]**Configure triggers** 페이지에서 Lambda 아이콘 왼쪽의 빈 칸을 클릭합니다.
3. [35]드롭다운 목록이 나타나면 **SNS**(아래로 스크롤해야 보일 수 있음)를 클릭합니다.
4. [36]**SNS topic**에 **ScaleEvent**가 선택되었는지 확인합니다.

Amazon SNS는 *ScaleEvent* 주제가 Auto Scaling으로부터 알림을 받을 때 이 Lambda 함수를 호출합니다.

1. [37]**Enable trigger** 체크 박스를 클릭하여 선택합니다.
2. [38]**Next**를 클릭합니다.

"수행 권한 없음"에 대한 오류는 모두 무시해도 됩니다. 무시해도 실습에는 영향이 없습니다.

1. [39]**Configure triggers** 페이지에서 다음과 같이 설정합니다(목록에 없는 설정은 모두 무시).

Name	'SnapandTag'
Description	'Snapshot and tag EC2 instance'
Runtime	Python 2.7

1. [40]**Code entry type** 섹션 아래에 나타나는 **모든 코드를 삭제**하여 필드를 비웁니다.

2. [41]아래의 **Copy Code Block**을 클릭하고 코드 필드에 붙여넣습니다.

```
# -*- coding: utf-8 -*-
# Snap_and_Tag Lambda 함수
#
# 이 함수는 Auto Scaling이 새 인스턴스를 시작하면 트리거됩니다.
# EBS 볼륨의 스냅샷이 생성되고 태그가 추가됩니다.

from __future__ import print_function

import json, boto3

def lambda_handler(event, context):
    print("Received event: " + json.dumps(event, indent=2))

    # Auto Scaling 이벤트 알림에서 EC2 인스턴스 ID 추출
    message = event['Records'][0]['Sns']['Message']
    autoscalingInfo = json.loads(message)
    ec2InstanceId = autoscalingInfo['EC2InstanceId']

    # 인스턴스에 연결된 모든 EBS 볼륨의 스냅샷
    ec2 = boto3.resource('ec2')
    for v in ec2.volumes.filter(Filters=[{'Name': 'attachment.instance-id', 'Values': [ec2InstanceId]}]):
        description = 'Autosnap-%s-%s' % ( ec2InstanceId, v.volume_id )

        if v.create_snapshot(Description = description):
            print("WtWtSnapshot created with description [%s]" % description)

    # EC2 인스턴스에 태그 추가: Key = Snapshots, Value = Created
    ec2 = boto3.client('ec2')
    response = ec2.create_tags(
        Resources=[ec2InstanceId],
        Tags=[{'Key': 'Snapshots', 'Value': 'Created'}]
    )
    print ("***Tag added to EC2 instance with id: " + ec2InstanceId)

    # 완료!
    return ec2InstanceId
```

코드를 검사합니다. 다음 단계를 수행합니다.

- 알림 메시지에서 EC2 인스턴스 ID 추출
- 인스턴스에 연결된 모든 EBS 볼륨의 스냅샷 생성
- 인스턴스에 태그를 추가하여 스냅샷이 생성된 것을 나타냄

1. [42]**Existing role**을 아래로 스크롤하고 전에 생성해둔 **SnapandTag** 역할을 선택합니다.
2. [43]**Advanced settings**를 클릭합니다.
3. [44]**Timeout** 값을 **3 min 0 sec**로 변경합니다.
4. [45]다른 값은 기본 설정 그대로 두고 **Next**를 클릭합니다.
5. [46]함수 설정을 검토하고 **Create function**을 클릭합니다.

이제 Auto Scaling이 새 인스턴스를 시작하면 Lambda 함수가 자동으로 실행됩니다.

작업 5: Auto Scaling 그룹을 스케일아웃하여 Lambda 함수 트리거

이번 작업에서는 **Auto Scaling 그룹의 원하는 용량을 증가**시킵니다. 이렇게 하면 Auto Scaling 그룹은 증가한 용량 요구 사항을 충족하기 위해 Amazon EC2 인스턴스를 시작합니다. 그런 다음 Auto Scaling 이 *ScaleEvent* SNS 주제에 알림을 전송합니다. 알림을 전송하고 나면 Amazon SNS가 *SnapandTag* Lambda 함수를 호출합니다.

1. [47]**Services** 메뉴에서 **EC2**를 클릭합니다.
2. [48]탐색 창에서 **Auto Scaling Groups**(아래로 스크롤해야 보일 수 있음)를 클릭합니다.
3. [49]화면 아래쪽의 **Details** 탭에서 **Edit**를 클릭합니다.
4. [50]**Desired**에 '2'를 입력합니다.
5. [51]**Save**를 클릭합니다.

그러면 Auto Scaling이 추가 Amazon EC2 인스턴스를 시작합니다.

1. [52]**Activity History** 탭을 클릭하고 시작된 새 EC2 인스턴스 진행률을 모니터링합니다.

상태가 변경되어 **성공**이 두 줄 표시될 때까지 대기합니다. 새로 고침 아이콘을 클릭하여 상태를 업데이트할 수 있습니다.

상태가 업데이트되면 Lambda 함수가 제대로 실행되었는지 확인할 수 있습니다.

1. [53](#) 탐색 창에서 **Instances**를 클릭합니다.
2. [54]시작 시간이 가장 최근인 인스턴스의 행을 클릭합니다. 인스턴스의 **Launch Time** 열을 보려면 오른쪽으로 스크롤해야 할 수도 있습니다.
3. [55]화면 아래쪽의 **Tags** 탭을 클릭합니다.

키가 **Snapshots**이고 값이 **Created**인 태그가 보일 것입니다. Lambda 함수가 EC2 인스턴스에 추가한 태그입니다.

1. [56]탐색 창에서 **Snapshots**를 클릭합니다.

스냅샷 창에서 Lambda 함수가 생성한 두 개의 스냅샷이 보일 것입니다.

태그나 스냅샷이 생성되지 않았다면, Lambda 함수에 오류가 발생했거나 실행되지 않은 것입니다. 강사의 지원을 받아 구성을 확인하십시오.

Auto Scaling 그룹이 정상적으로 Lambda 함수를 트리거하여 태그와 스냅샷을 생성했습니다. 지금까지가 AWS에서 서버 없는 솔루션을 생성하는 방법에 대한 예제입니다.

실습 완료

축하합니다! 실습을 완료했습니다.

본 실습용 qwikLABS 페이지로 돌아가 **End**를 클릭하여 실습 환경을 종료해 주십시오.