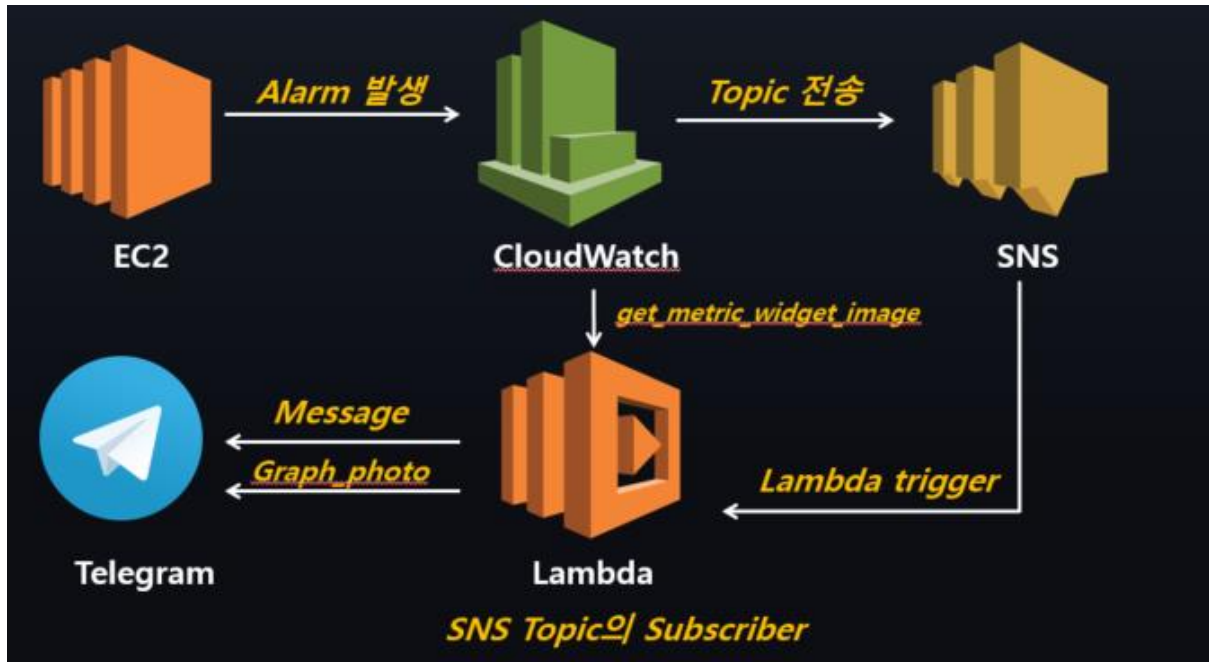


CloudWatch, SNS와 Lambda를 이용해서 텔레그램으로 알람받기(Python3 ver)

1. 진행방식



2. 텔레그램 설정

1) 봇 생성

텔레그램에서 @botfather검색 후 봇생성(/newbot)

2) 봇 별칭 및 username 지정

(예시 : 별칭-test_bot, Username - test_version3_bot)

3) 봇 정보 기록 (HTTP API TOKEN)

이 봇에 대한 정보를 HTTP api로 주고받을 때 식별할 수 있는 유니크한 값

4) chat id 기록 (UID)

주소창에 <https://api.telegram.org/bot<위에서 받은 HTTP API>/getUpdates> 입력

이때 'chat' value의 id key값 : 메시지를 보내는 채팅방(Chatbot내부)을 식별하는 id값

```
"chat":{"id":937211996,"first_name"
```

3. AWS 설정

1) SNS Topic 생성

Amazon SNS – Topic – Create Topic

Amazon SNS > 주제 > telegram_topic	
telegram_topic	
세부 정보	
이름	telegram_topic
ARN	arn:aws:sns:ap-northeast-2:271139208315:telegram_topic

2) CloudWatch 경보(alarm)생성

2-1) 인스턴스 및 지표선택

Amazon CloudWatch – Alarm – Create Alarm

알람명 : zip-alarm

네임스페이스 : AWS/ApplicationELB

타겟명 : targetgroup/web-eb12/606ae8d6296dc31a

지표이름 : RequestCount

통계 : 합계

기간 : 1분

2-2) 경보알림 조건선택

RequestCount >= 40

세부 정보		
이름	임계값	네임스페이스
zip-alarm	1 분 내 1개의 데이터 포인트에 대한 RequestCount >= 40	AWS/ApplicationELB
설명	마지막 변경	지표 이름
설명 없음	2020-04-01 00:57:08	RequestCount
상태	ARN	TargetGroup
☹ 데이터 부족	arn:aws:cloudwatch:ap-northeast-2:271139208315:alarm:zip-alarm	targetgroup/web-eb12/606ae8d6296dc31a
		LoadBalancer
		app/web-elb2/080d5df9178b848b
		AvailabilityZone
		ap-northeast-2a
		통계
		합계
		기간
		1분

2-3) 경보일 때, 알림 전송할 트리거 지정

SNS topic : telegram_topic선택

작업 구성

알림

경보 상태 트리거

Define the alarm state that will trigger this action.

제거

☒

경보 상태

지표 또는 표현식이 정의된 임계값을 벗어났습니다.

☐

정상

지표 또는 표현식이 정의된 임계값 범위에 있습니다.

☐

데이터 부족

경보가 방금 시작되었거나 사용 가능한 데이터가 부족합니다.

SNS 주제 선택

Define the SNS (Simple Notification Service) topic that will receive the notification.

☒

기존 SNS 주제 선택

☐

새 주제 생성

☐

주제 ARN 사용

다음으로 알림 전송...

Q

telegram_topic_zip

X

Only email lists for this account are available.

이메일(엔드포인트)

arn:aws:lambda:ap-northeast-2:271139208315:function:tel-lambda-final-zip - SNS 콘솔에서 보기

새 알림 추가

4. Lambda 설정

4-1) Lambda 함수 생성(Python 3.8)

4-2) 전체zip 파일(py파일+packages파일)로 업로드

4-3) 환경변수 설정

CUSTOMER : 고객명

TOKEN : 텔레그램에서 챗봇 초기 생성시 발급받은 HTTP API TOKEN

UID : 메시지를 보낼 채널을 식별할 ID

4-4) Role추가

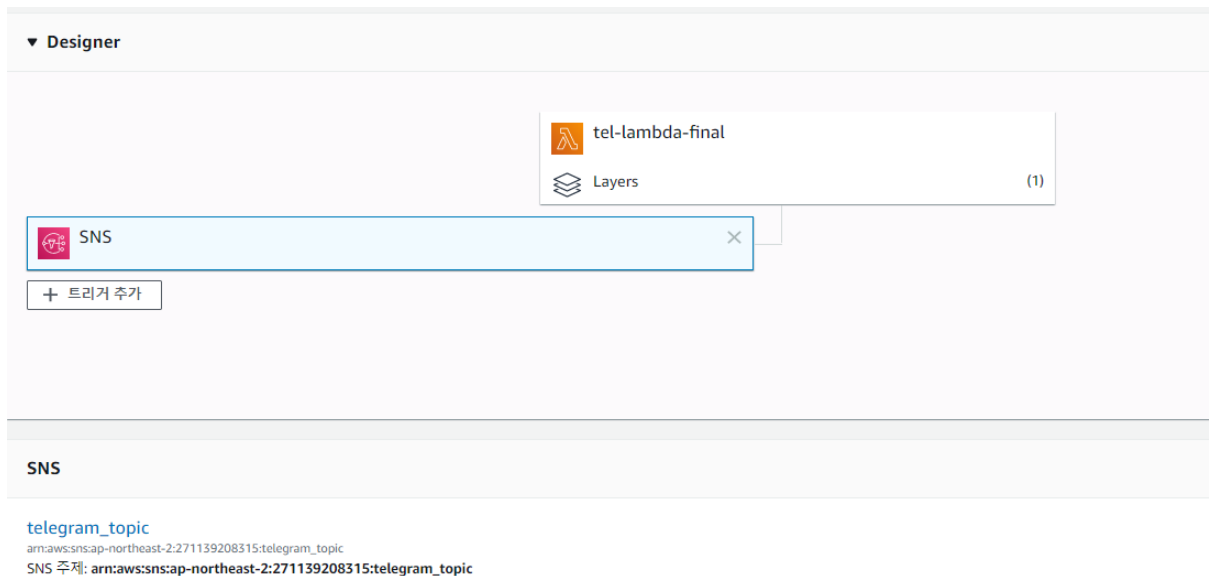
람다를 생성할 때 만든 Role : AWSLambdaBasicExecutionRole-868cc1bc-33b2-4947-b242-0209da381ed3

람다에서 Cloudwatch 리소스에 접근하고자 하는 Role : CloudWatchReadOnlyAccess

관련	신뢰 관계	태그	엑세스 관리자	세션 취소
▼ Permissions policies (2 정책이 적용됨)				
정책 연결				
정책 이름				정책 유형
CloudWatchReadOnlyAccess				AWS 관리형 정책
AWSLambdaBasicExecutionRole-868cc1bc-33b2-4947-b242-0209da381ed3				관리형 정책

4-5) 트리거 등록

subscribe 할 topic 선택하여 트리거 활성화



참고) packages파일을 따로 Layer로 추가하는 경우 필요한 작업

Layer : Python의 Package 모듈을 따로 import하기 위해서 사용

1) 현재 생성한 Layer : python3-packages-final

Lambda함수가 run하기 위해 필요한 모듈파일들 (request,urlib 외 5개모듈)

호환 런타임 : python 3.6/ python 3.7/ python 3.8



2) Layers 람다 함수에 적용

Add Layers 함수

Lambda > 계층 > Add layer 함수

Add layer 함수

계층 선택
함수의 호환되는 계층을 선택할 런타임 또는 Amazon 리소스 이름(ARN)을 지정해야 계층에 버전.

☒ 실행 시간 호환 계층 목록에서 선택
☐ 계층 버전 ARN 제공

호환되는 계층

이름
python3-pacakges-final ▼

버전
1 ▼

Layer 또한 zip파일처럼 적용되서 내부적으로 지정된 Python의 path대로 라이브러리 import
현재 layer zip파일의 상위디렉토리 /python

+ 만약 Layer의 상위 디렉토리가 없다면 패키지가 unzip되는 루트 디렉토리는 /opt 추가

sys.path.append('/opt')

```
9 sys.path.append('/opt')
10 import requests
```

5. 최종 테스트 결과(텔레그램)

test_bot

[🔴 TEST CUSTOMER NAME(ZIP)]

알람명: zip-alarm

변경상태: 🔵 > 🔴

상태변경 시간: 2020-03-30 22:28:08 (한국시간)

타겟: LoadBalancer=app/web-elb2/080d5df9178b848b

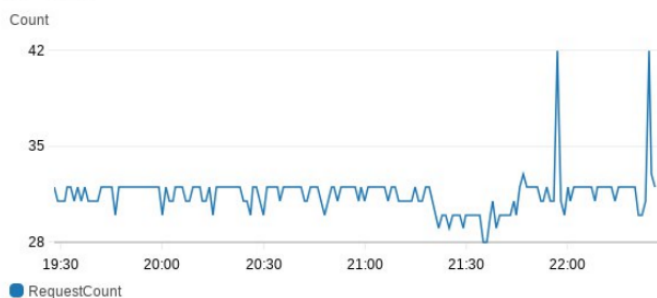
리전: Asia Pacific (Seoul)

알람상세: Threshold Crossed: 1 out of the last 1 datapoints [42.0 (30/03/20 13:24:00)] was greater than or equal to the threshold (40.0) (minimum 1 datapoint for OK -> ALARM transition).

오후 10:28

test_bot
[🔴 TEST CUSTOMER NAME(ZIP)] 알람명: zip-alarm 변경상태: 🔵 ...

zip-alarm



6. Telegram API

텔레그램 공식 API : <https://core.telegram.org/bots/api>

사용한 method

1) getUpdates : 메시지들의 결과 저장 (chat-id 확인가능)

test request URL : <https://api.telegram.org/bot<bot 토큰명>/getUpdates>

2) sendMessage 이벤트 : text 이하의 메시지를 텔레그램으로 전송

test request URL : <https://api.telegram.org/bot<bot 토큰명>/sendMessage?>

chat_id=<전송할 채널 id>&text=<보낼 msg>

3) sendPhoto 이벤트 : file id 에 해당하는 photo 를 텔레그램으로 전송

test request URL : <https://api.telegram.org/bot<bot 토큰명>/sendPhoto?>

chat_id=<전송할 채널 id>&&photo=<보낼 phot 파일>

(추가 요구사항) Group안에 test_bot을 초대하여 알림을 받는 경우

/getUpdates로 response확인시, 이때 'chat' value의 id key값이 -로 시작하는 UID로 변경

```
"chat":{"id":-437055380,
```

```
"title":"test_group","type":"group","all_members_are_administrators":true},
```

(추가 요구사항) Channel안에 test_bot을 초대하여 알림을 받는 경우

```
"chat":{"id":-1001407838341,"title":"test_channel","type":"channel"}
```

이 UID 에 따라서 람다의 환경변수 변경가능