serverless arch. on aws - 1

오늘 다룰 주제

serverless architecture on aws
- lambda, s3, api gateway, cloudfront, +…

AWS Lambda

Run code without thinking about servers.
Pay only for the compute time you consume.

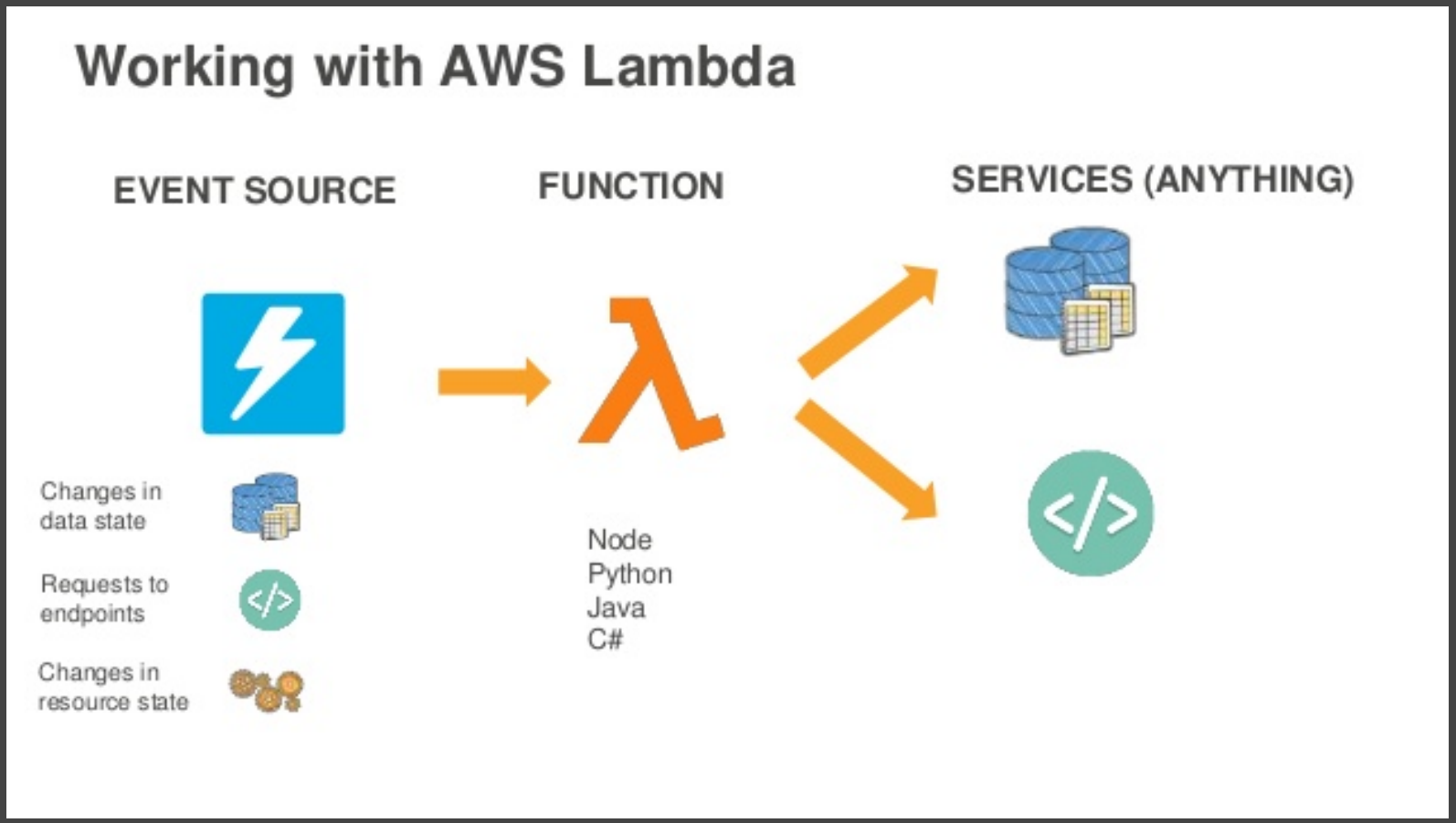
서버를 고려하지 않고 애플리케이션과 서비스 구축, 실행

간단한 설정
저렴한 비용
충분한 성능
AWS 타 기능 통합

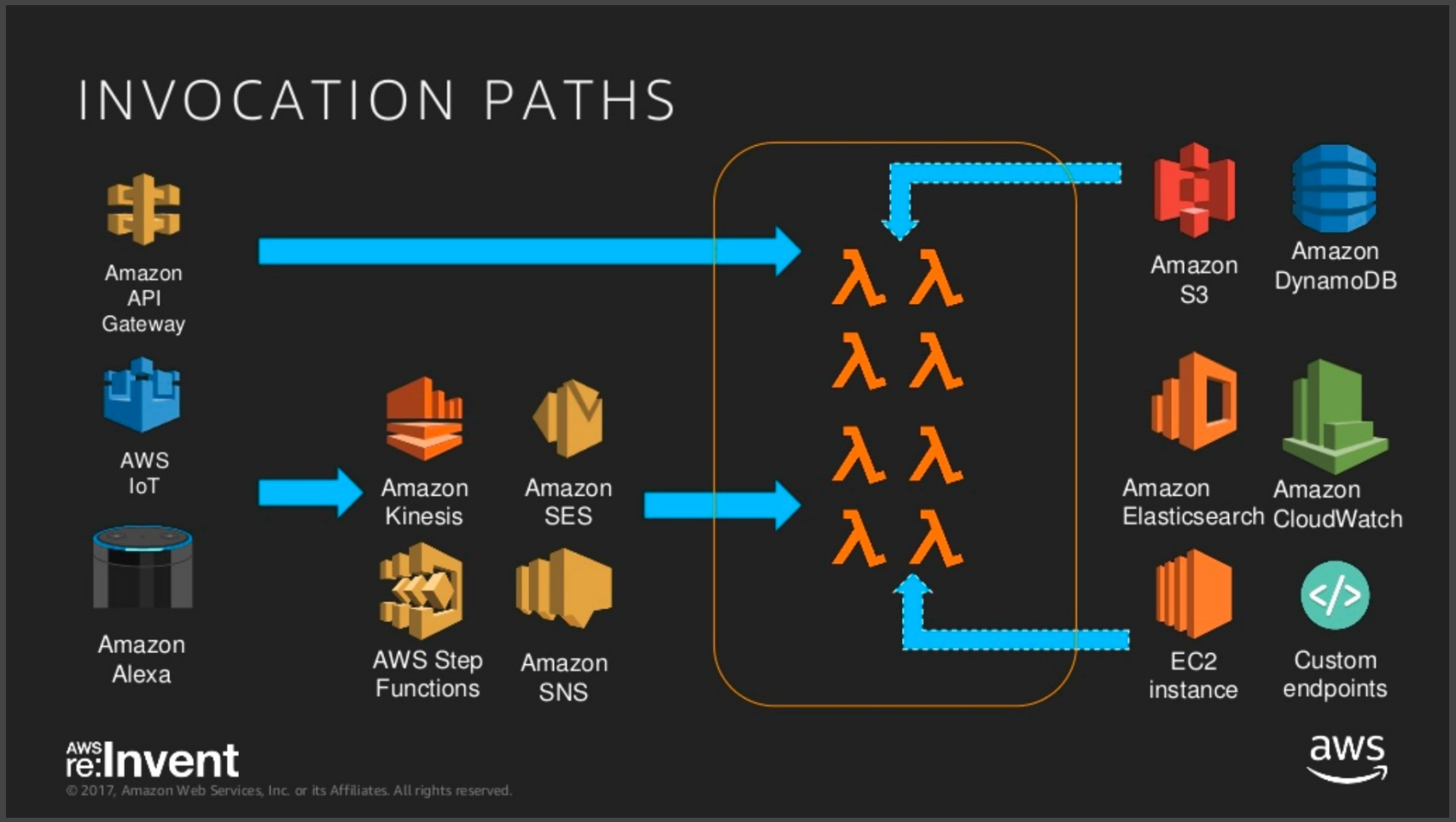
EC2 vs Elastic Beanstalk vs Lambda

많은 강점을 가진 서비스인데…

AWS Lambda


Working with AWS Lambda
EVENT SOURCE
FUNCTION
SERVICES (ANYTHING)
Changes in data state
Requests to endpoints
Changes in resource state
Node
Python
Java
C#

AWS Lambda


INVOCATION PATHS
Amazon API Gateway
AWS IoT
Amazon Alexa
Amazon Kinesis
Amazon SES
AWS Step Functions
Amazon SNS
Amazon S3
Amazon DynamoDB
Amazon Elasticsearch
Amazon CloudWatch
EC2 instance
Custom endpoints
aws re:Invent
© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.
aws

AWS Lambda

Amazon linux 기반 (64bit)
Node.js – v8.10, v6.10
Java – 8
Python – 3.6, 2.7
.NET Core – 1.0.1, 2.0, 2.1
Go – 1.x

예약/지정된 환경 변수, 직접 컴파일한 모듈도 사용 가능

AWS Lambda

메모리 : 128MB ~ 3008MB(CPU 할당은 메모리 비례)
실행 시간 : 100ms ~ 15min

128MB 제한 함수 100ms 실행 시 : $0.000000208

ex) 512MB 제한 함수 월 3백만 회, 매회 1초 실행 시
- 3,000,000 회 * 1초 = 3,000,000 초
- 3,000,000 초 * 512MB 100ms 당 $0.000000834
  = $25.02 + 요청 1백만 건당 $0.2
  = $25.62 (무료 구간 제외), $18.34 (무료 구간 포함)

AWS Lambda

Lambda 컨테이너 생명 주기를 AWS가 관리
- 개발자가 컨테이너 생성/소멸 시점을 알 수 없음

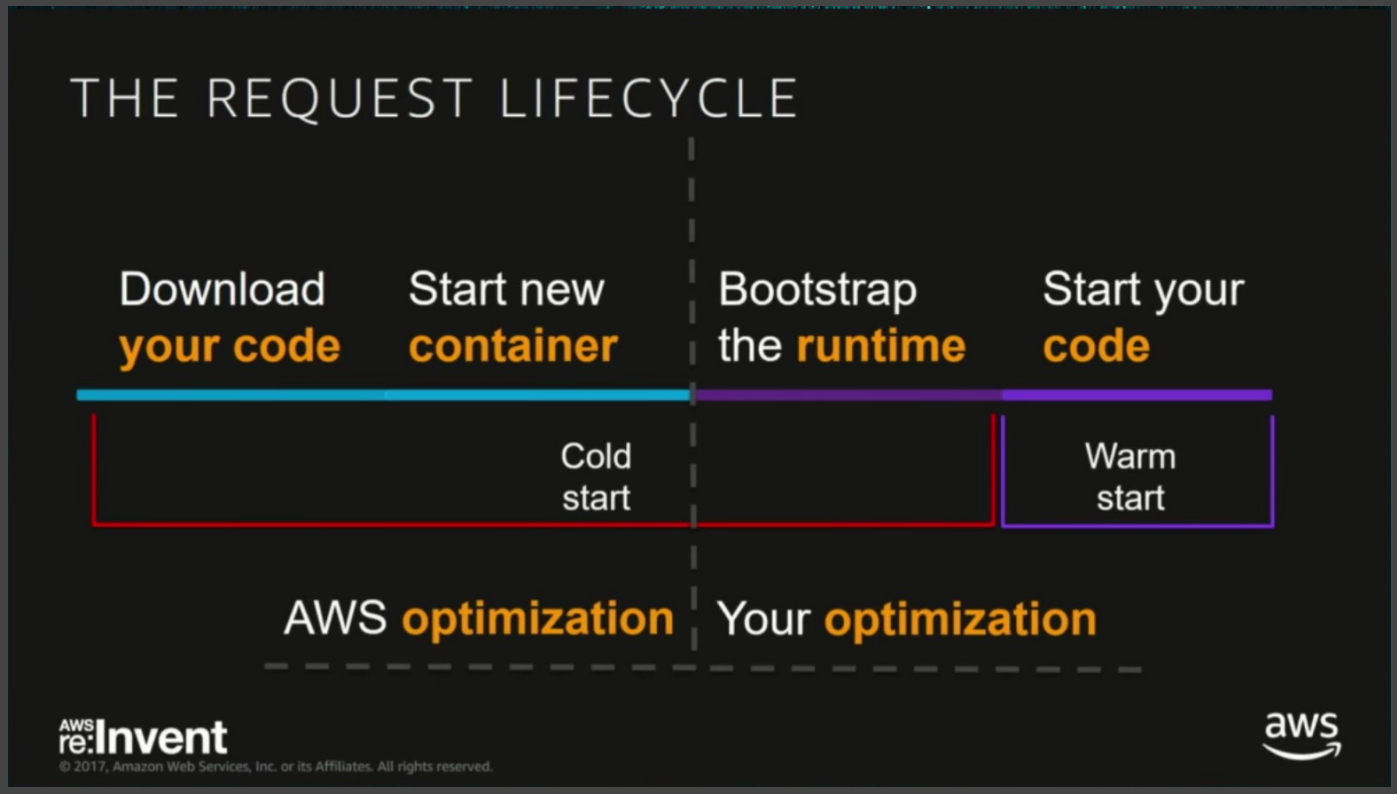
하나의 컨테이너는 하나의 lambda 함수 실행만 가능.
- 반복적으로 실행되면 컨테이너를 재활용

한번 실행 후, 일정 시간 실행되지 않으면 컨테이너는 소멸
- 소멸 후 재 실행 시 재생성, 지연 발생 (cold-start)

기본 설정 값은 VPC 사용 안함. 사용 시 추가 지연 및 제한
- 사설망 접근 필요 시 IP 할당 제한 및 ENI 생성 지연

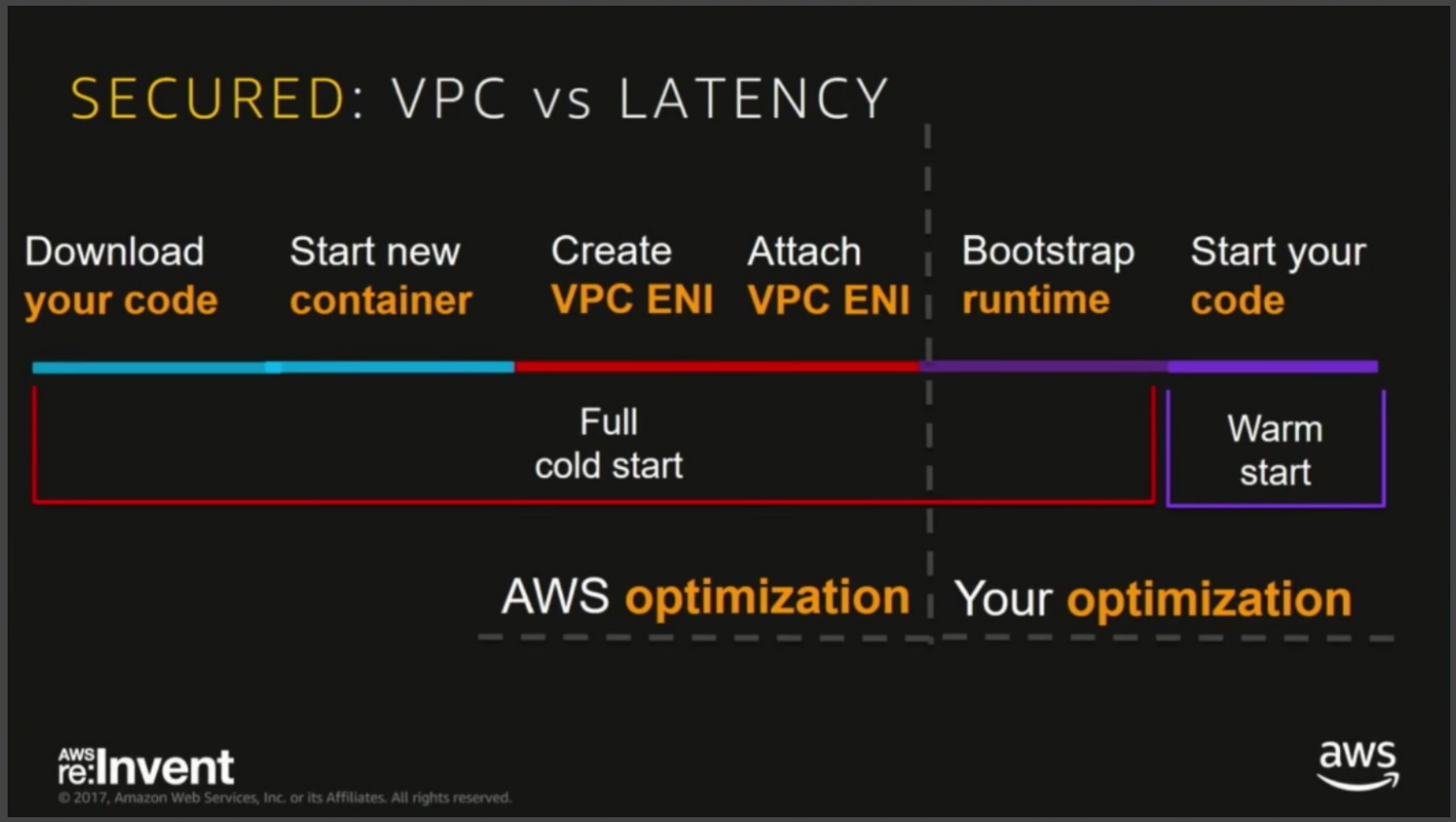
AWS Lambda

한번 실행 후, 일정 시간 실행되지 않으면 컨테이너는 소멸
- 소멸 후 재 실행 시 재생성, 지연 발생 (cold-start)



AWS Lambda

기본 설정 값은 VPC 사용 안함. 사용 시 추가 지연 및 제한
- 사설망 접근 필요 시 IP 할당 제한 및 ENI 생성 지연



AWS Lambda

메모리 최대 3008MB, 설정한 제한 초과 시 종료
임시 디스크 /tmp 512MB
file descriptor 1024개
process + thread 1024개
최대 실행 시간 15분(5분 -> 15분)
실행 요청/응답 페이로드 – 6MB(동기), 256KB(비동기)
배포 패키지 크기 – 50MB(zip 압축), 250MB(압축 해제)

리전 별 동시 실행 – 1000
배포 패키지 총 크기 – 75GB

AWS Lambda

Run code without thinking about servers.
Pay only for the compute time you consume.

서버를 고려하지 않고 애플리케이션과 서비스 구축, 실행

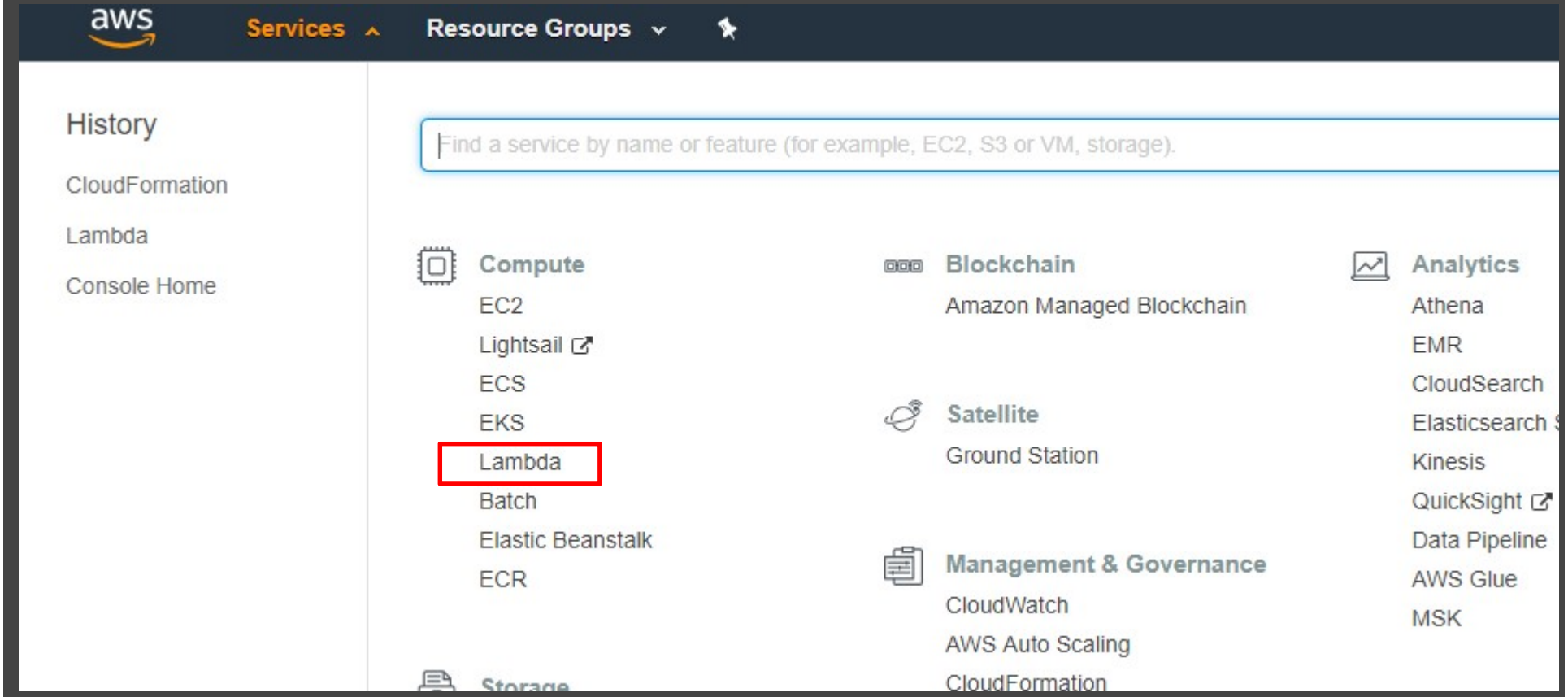
간단한 설정
저렴한 비용
충분한 성능
AWS 타 기능 통합

많은 강점을 가진 서비스인데…

여러 제한 조건도 따르니, 적절하게 사용

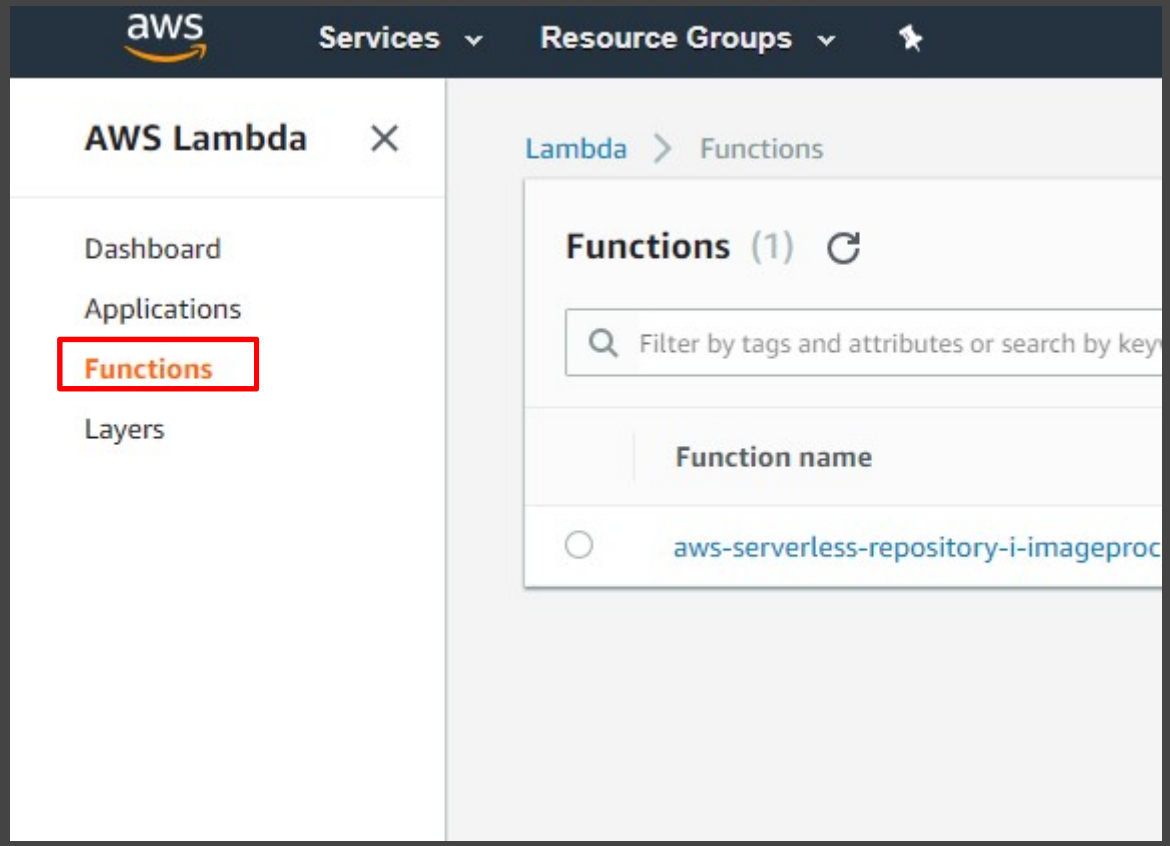
AWS Lambda

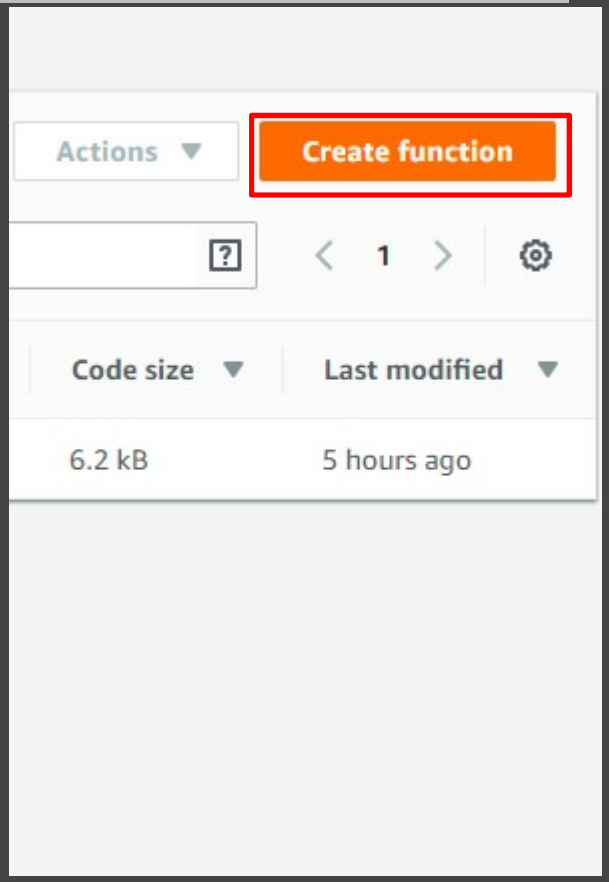
AWS Console -> Services -> Lambda 선택



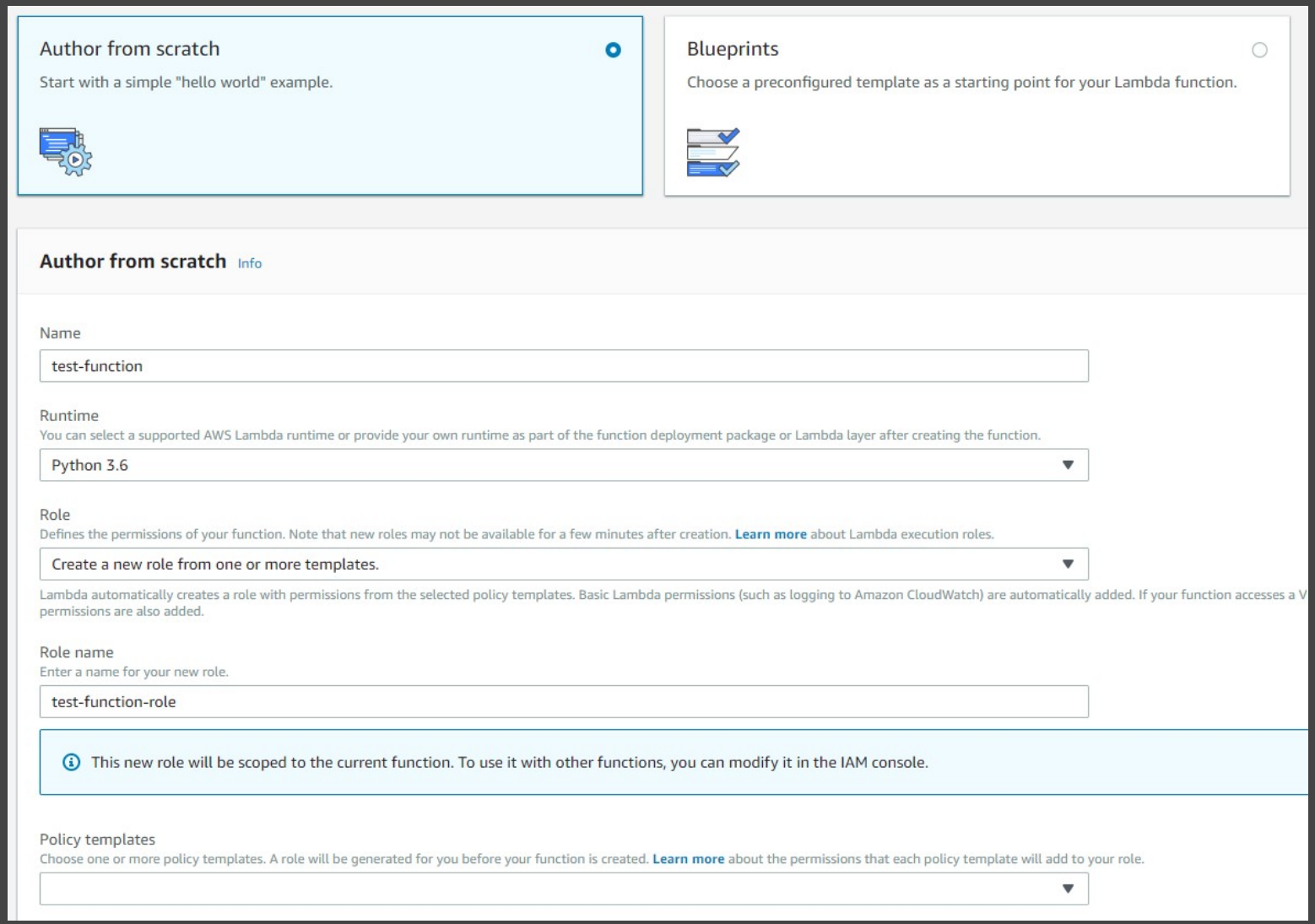
AWS Lambda

좌측 메뉴에서 Functions 선택 후
우측 Create function 클릭





AWS Lambda


Author from scratch
Start with a simple "hello world" example.
Blueprints
Choose a preconfigured template as a starting point for your Lambda function.
Author from scratch Info
Name
test-function
Runtime
You can select a supported AWS Lambda runtime or provide your own runtime as part of the function deployment package or Lambda layer after creating the function.
Python 3.6
Role
Defines the permissions of your function. Note that new roles may not be available for a few minutes after creation. Learn more about Lambda execution roles.
Create a new role from one or more templates.
Lambda automatically creates a role with permissions from the selected policy templates. Basic Lambda permissions (such as logging to Amazon CloudWatch) are automatically added. If your function accesses a V
permissions are also added.
Role name
Enter a name for your new role.
test-function-role
This new role will be scoped to the current function. To use it with other functions, you can modify it in the IAM console.
Policy templates
Choose one or more policy templates. A role will be generated for you before your function is created. Learn more about the permissions that each policy template will add to your role.

AWS Lambda

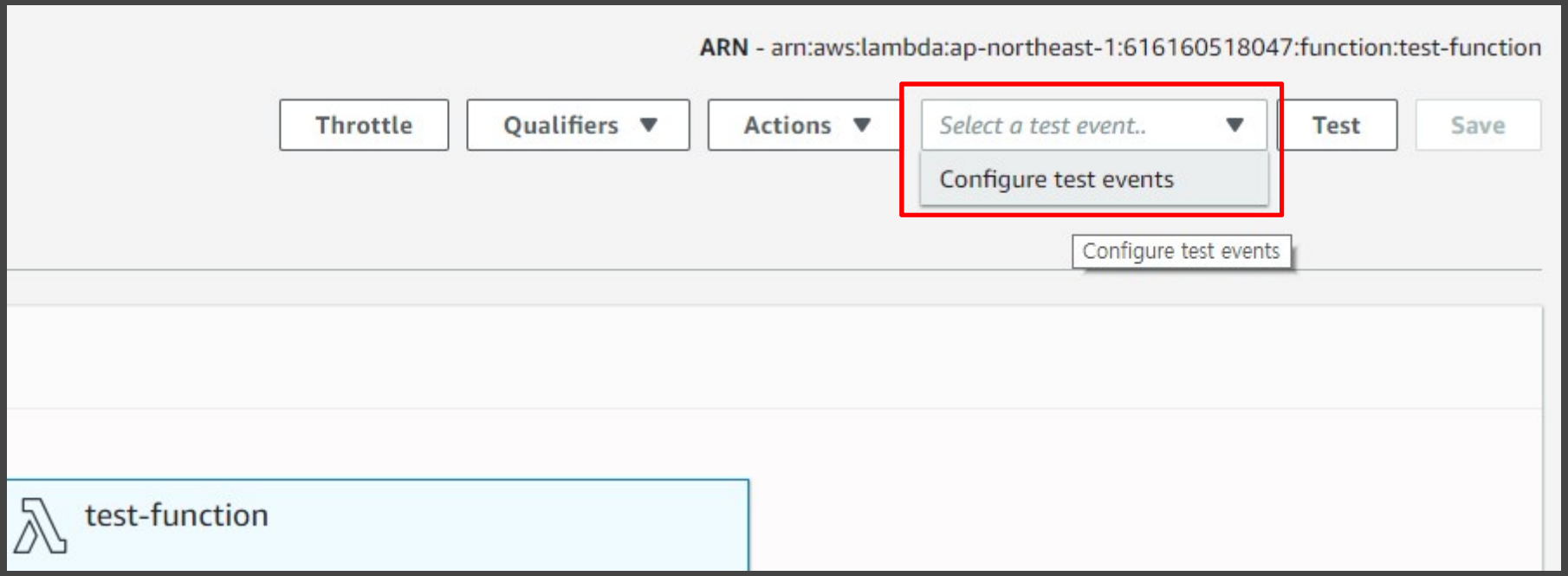
우측 상단 Author from scratch 선택 확인

Name : 함수 이름(관리용)
Runtime : 함수 런타임 선택(샘플이니 자유롭게 선택)
Role : 함수가 사용할 IAM Role
- Create a new role from one or more templates
Role Name : 함수 생성 시 만들어질 Role 이름(관리용)

입력 후 Create function 클릭

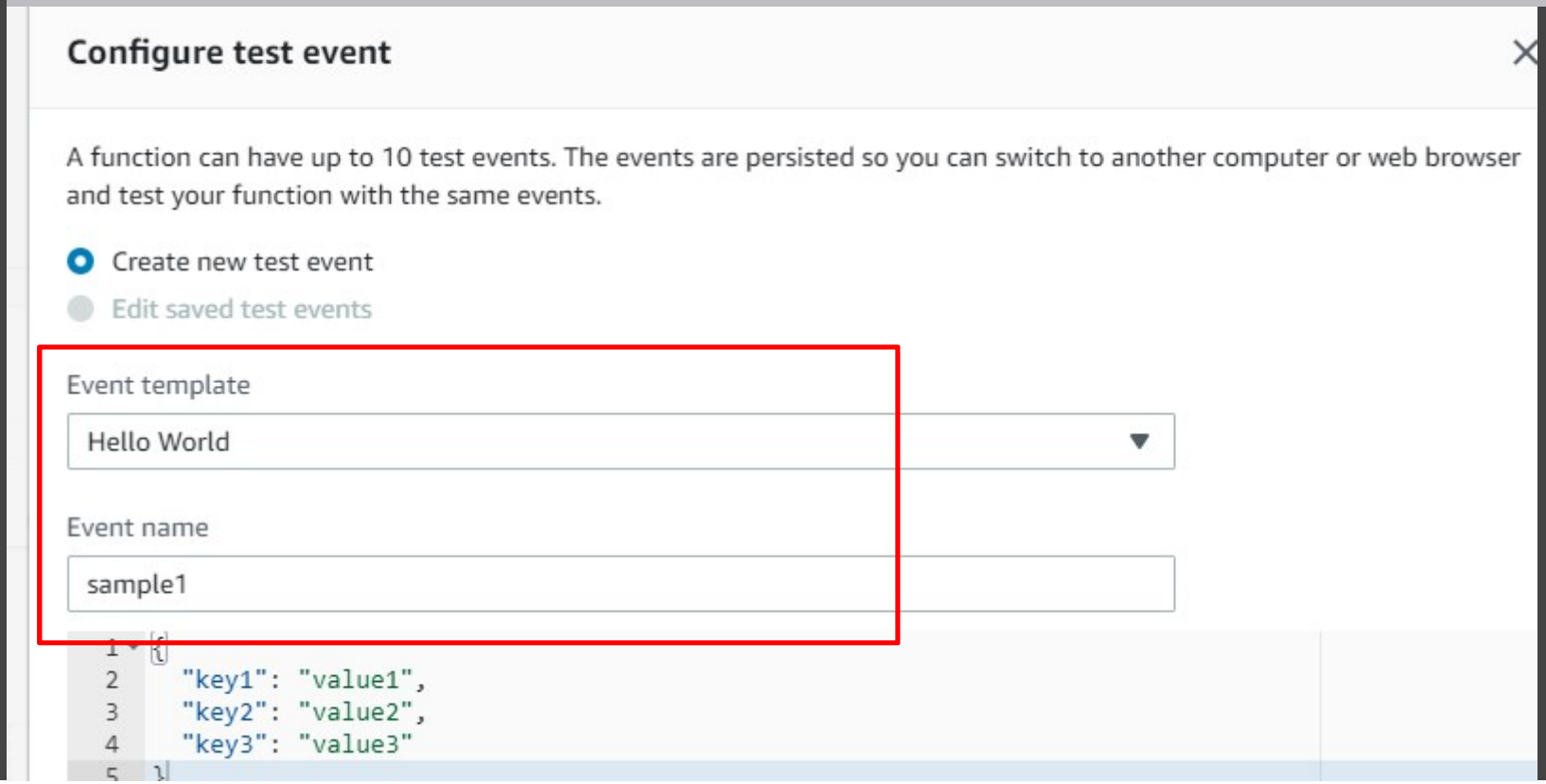
AWS Lambda

우측 상단의 Select a test event.. 클릭 후
Configure test events 클릭



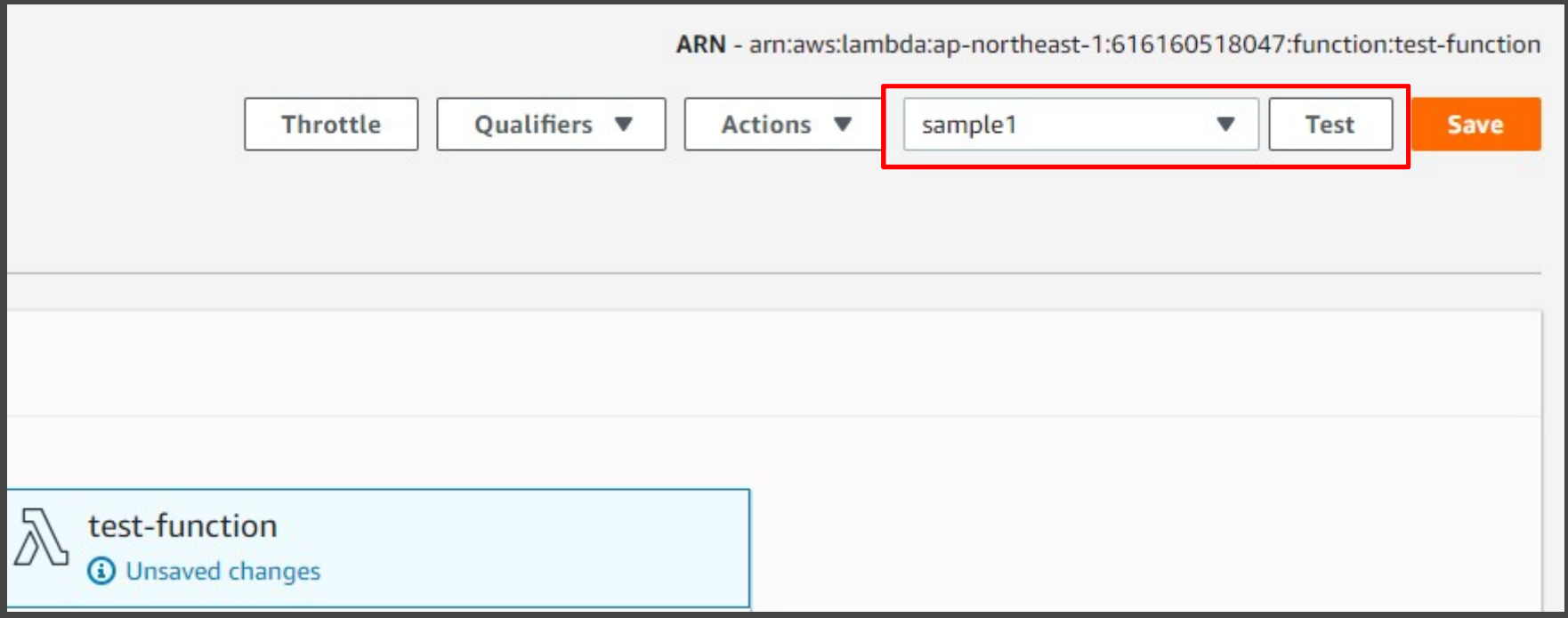
AWS Lambda

Event template - Hello World 혹은 임의 지정
Event name – 테스트 이벤트 명(관리용)
우측 하단 Create 클릭

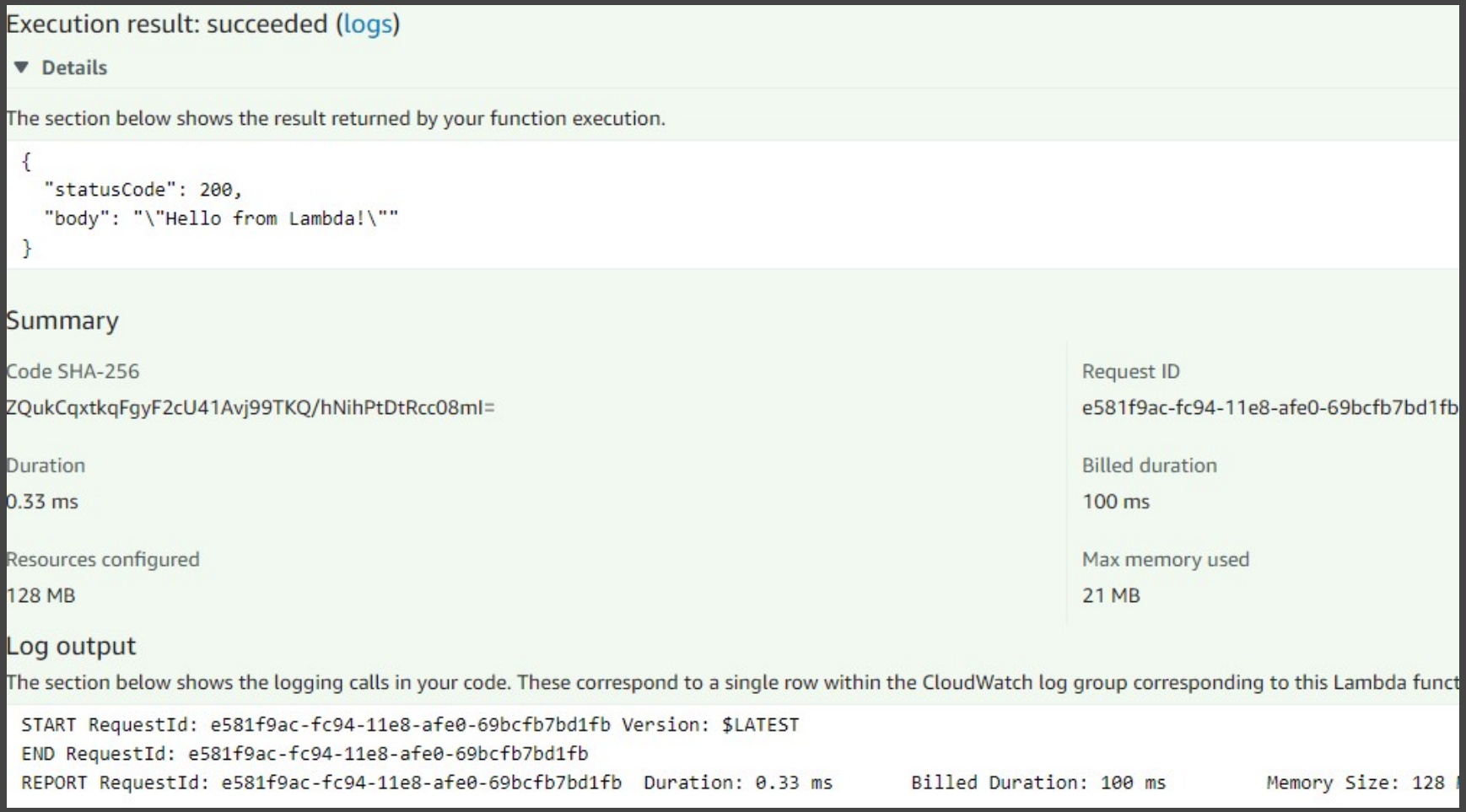


AWS Lambda

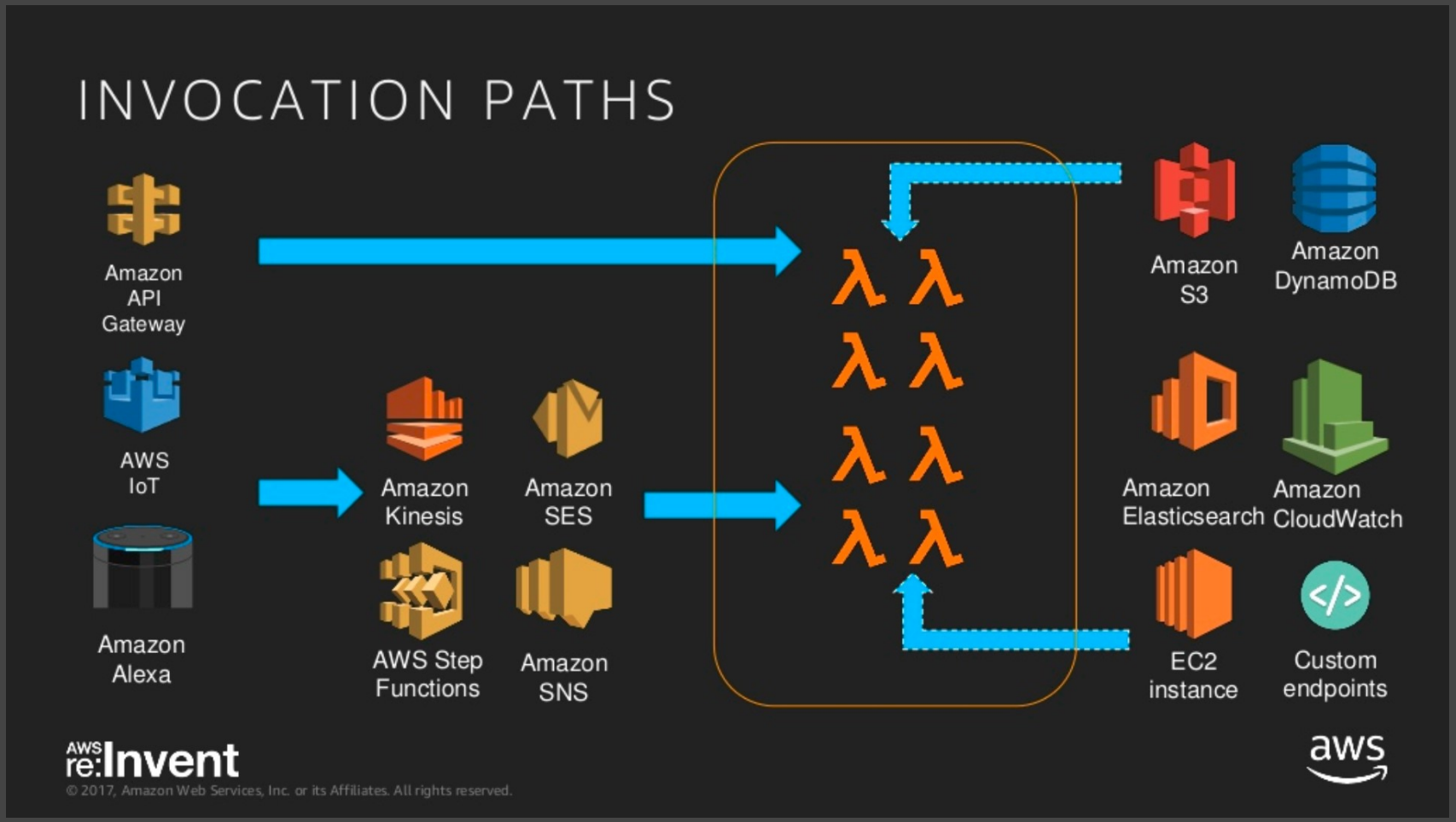
테스트 이벤트 생성 확인 후 Test 버튼 클릭



AWS Lambda

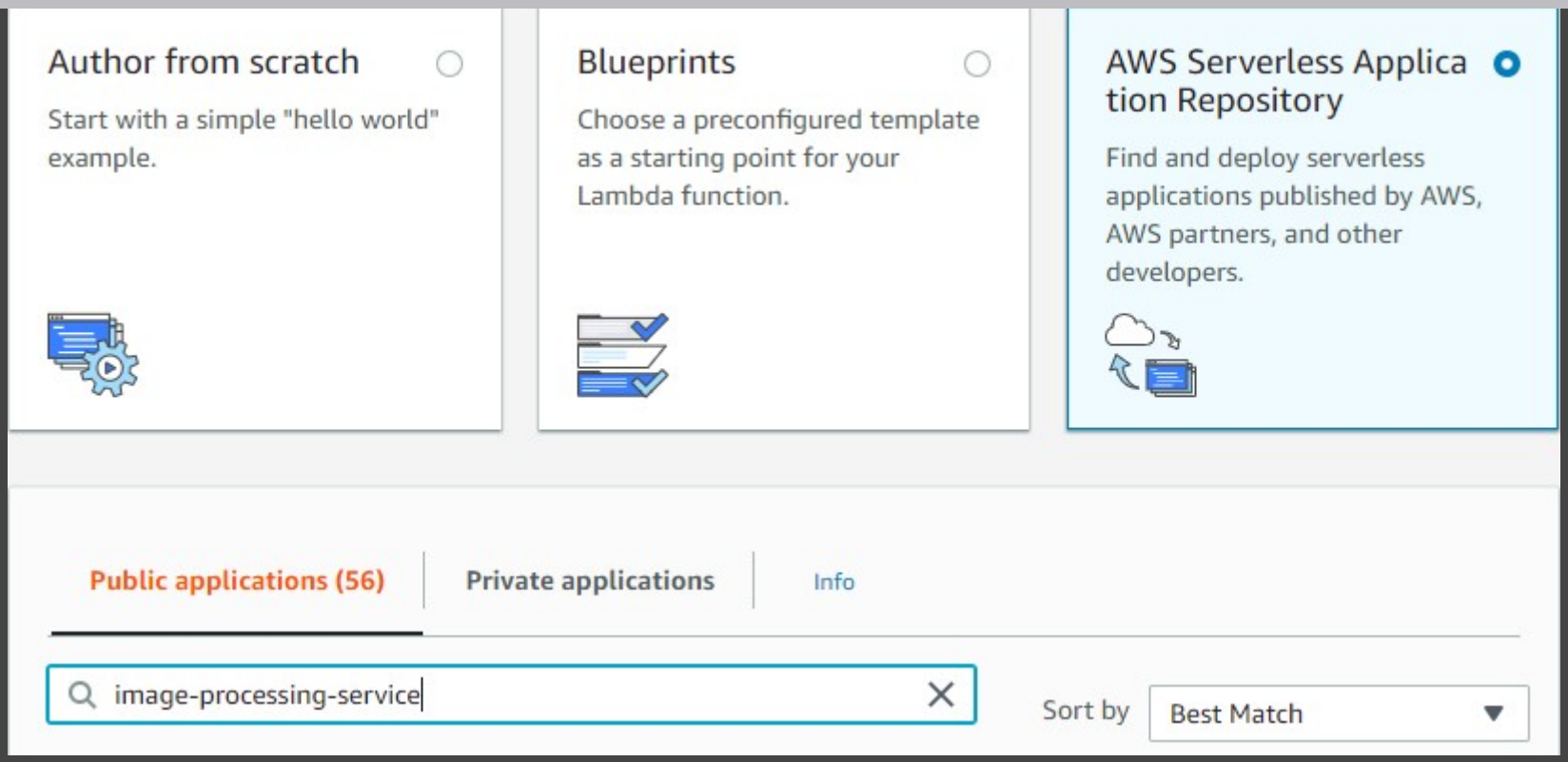

Execution result: succeeded (logs)
Details
The section below shows the result returned by your function execution.
{
"statusCode": 200,
"body": "\"Hello from Lambda!\""
}
Summary
Code SHA-256
ZQukCqxtkqFgyF2cU41Avj99TKQ/hNihPtDtRcc08ml=
Request ID
e581f9ac-fc94-11e8-afe0-69bcfb7bd1fb
Duration
0.33 ms
Billed duration
100 ms
Resources configured
128 MB
Max memory used
21 MB
Log output
The section below shows the logging calls in your code. These correspond to a single row within the CloudWatch log group corresponding to this Lambda funct
START RequestId: e581f9ac-fc94-11e8-afe0-69bcfb7bd1fb Version: $LATEST
END RequestId: e581f9ac-fc94-11e8-afe0-69bcfb7bd1fb
REPORT RequestId: e581f9ac-fc94-11e8-afe0-69bcfb7bd1fb Duration: 0.33 ms Billed Duration: 100 ms Memory Size: 128

AWS Lambda


INVOCATION PATHS
Amazon API Gateway
AWS IoT
Amazon Alexa
Amazon Kinesis
Amazon SES
AWS Step Functions
Amazon SNS
Amazon S3
Amazon DynamoDB
Amazon Elasticsearch
Amazon CloudWatch
EC2 instance
Custom endpoints
aws re:Invent
© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.
aws

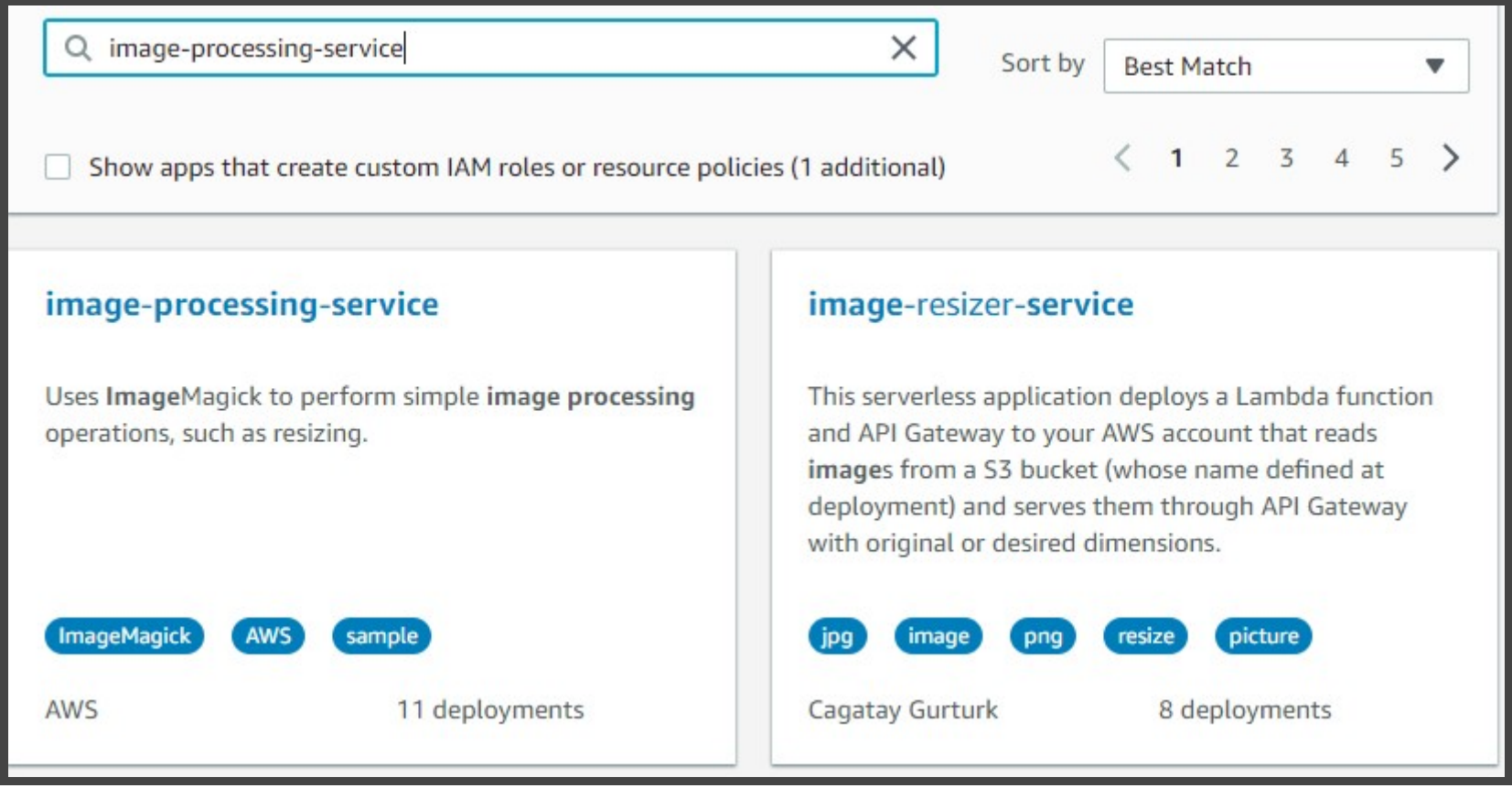
AWS Lambda

함수 생성 페이지에서
AWS Serverless Application Repository
선택 후 image-processing-service 로 검색



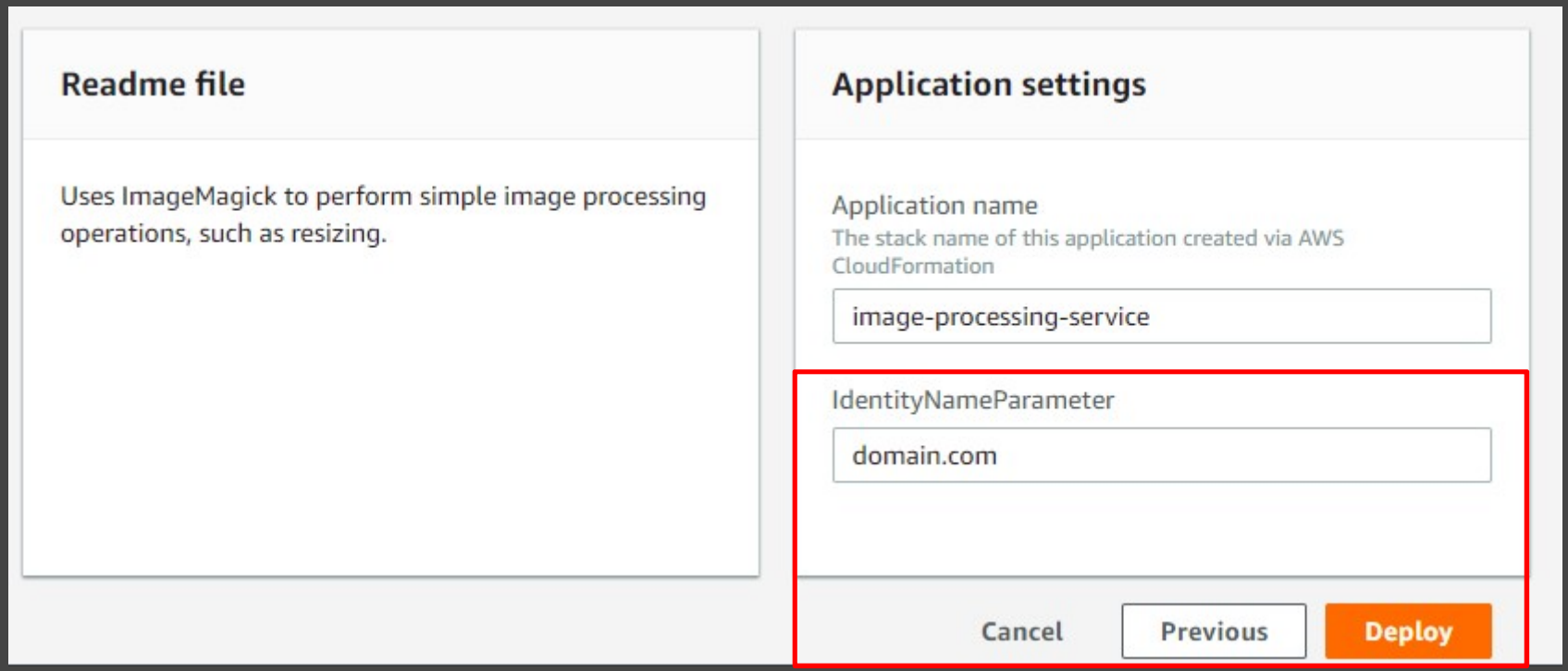
AWS Lambda

검색 결과 중 AWS 가 작성자인 image-processing-service 선택



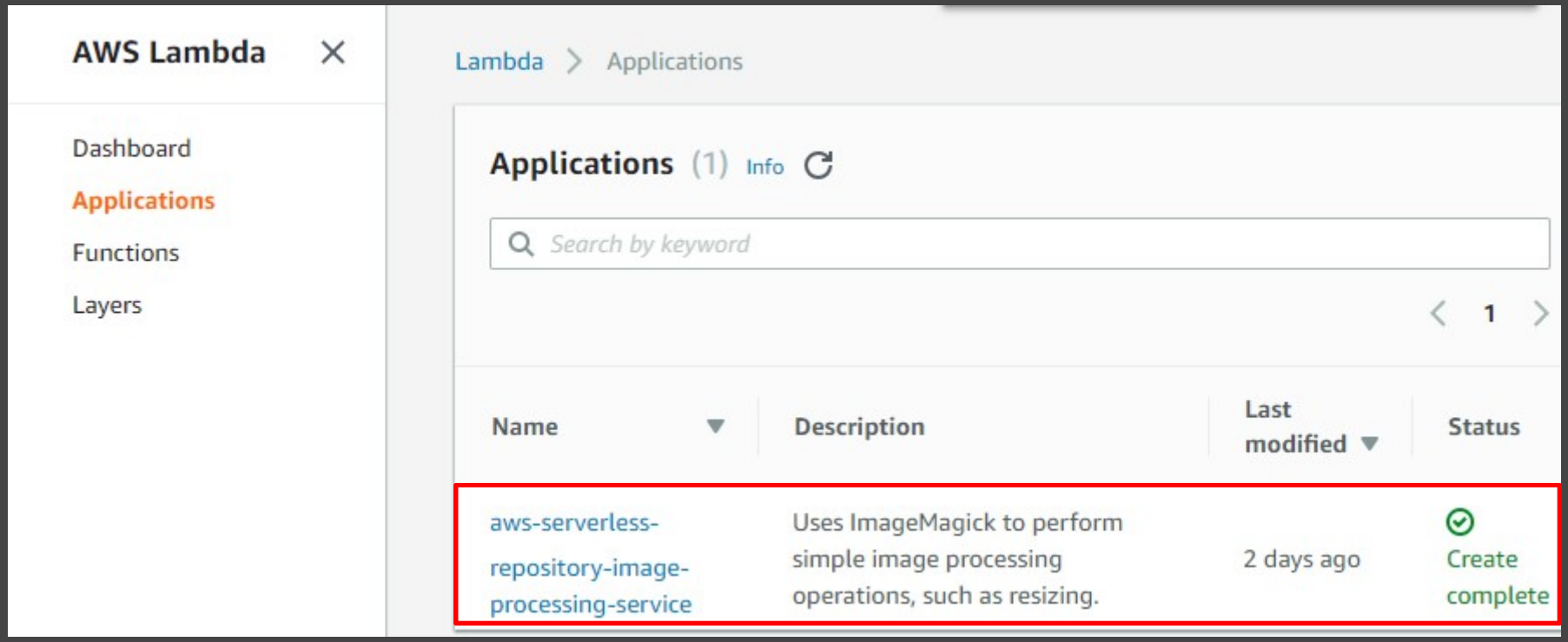
AWS Lambda

우측 하단 IdentityNameParameter 에 임의의 도메인 주소를 입력하고 Deploy 선택



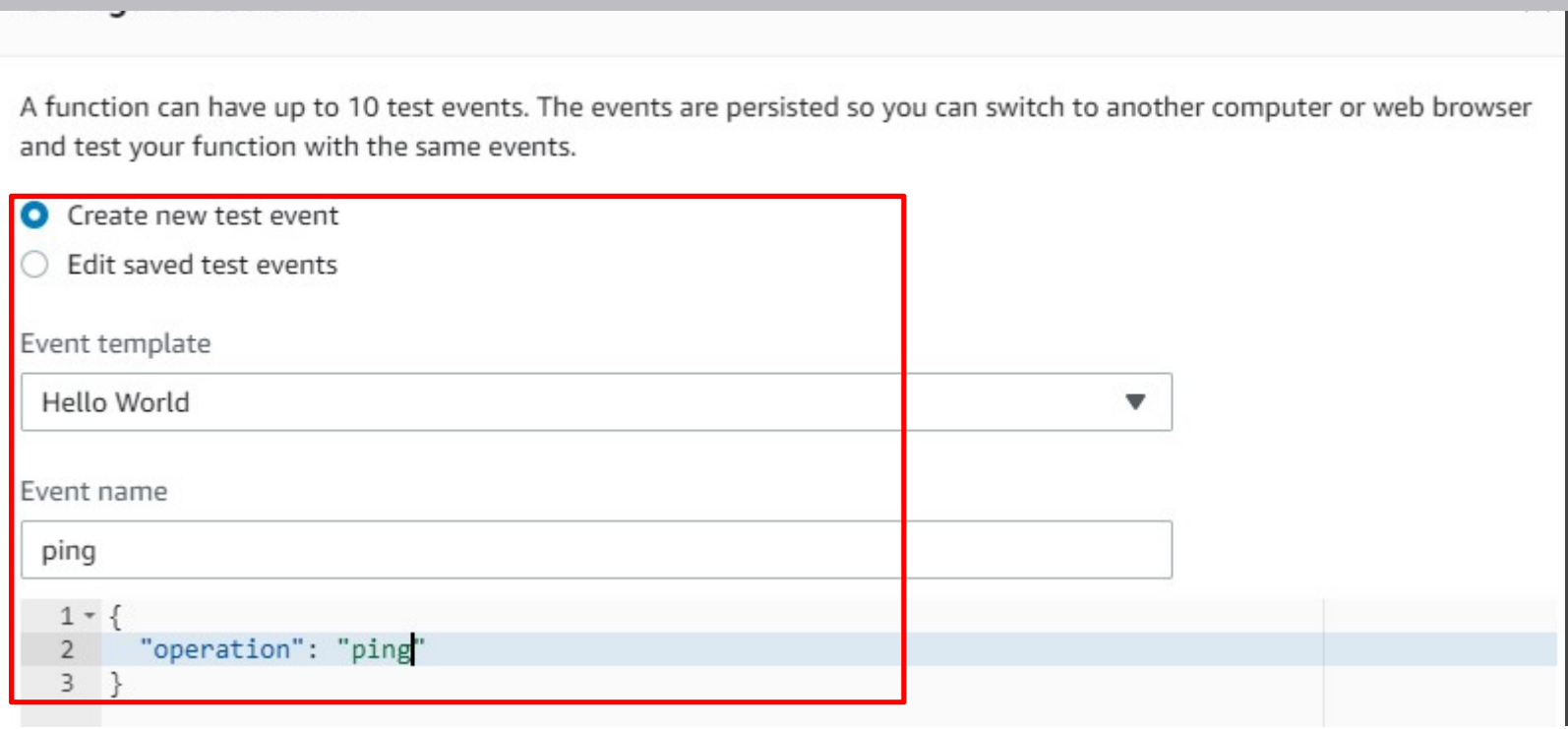
AWS Lambda

좌측 메뉴 Application 선택 후
생성 요청한 리소스들이 생성 완료 상태인지 확인



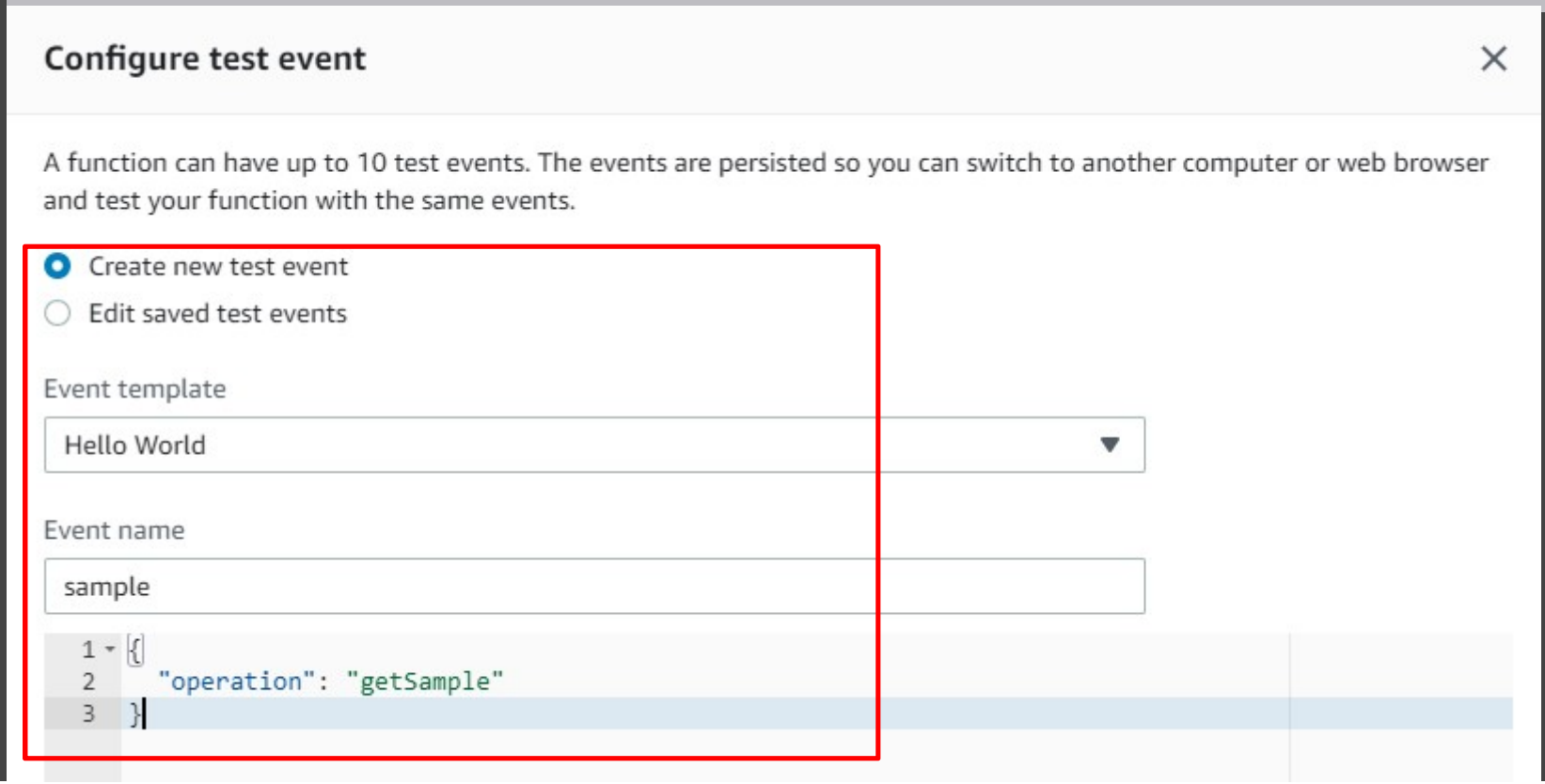
AWS Lambda

제공된 코드가 정상 동작하도록 테스트 이벤트 생성
임의의 Event template 선택 후 아래 body json 에
{“operation”: “ping”} 입력 후
우측 하단 Create 클릭



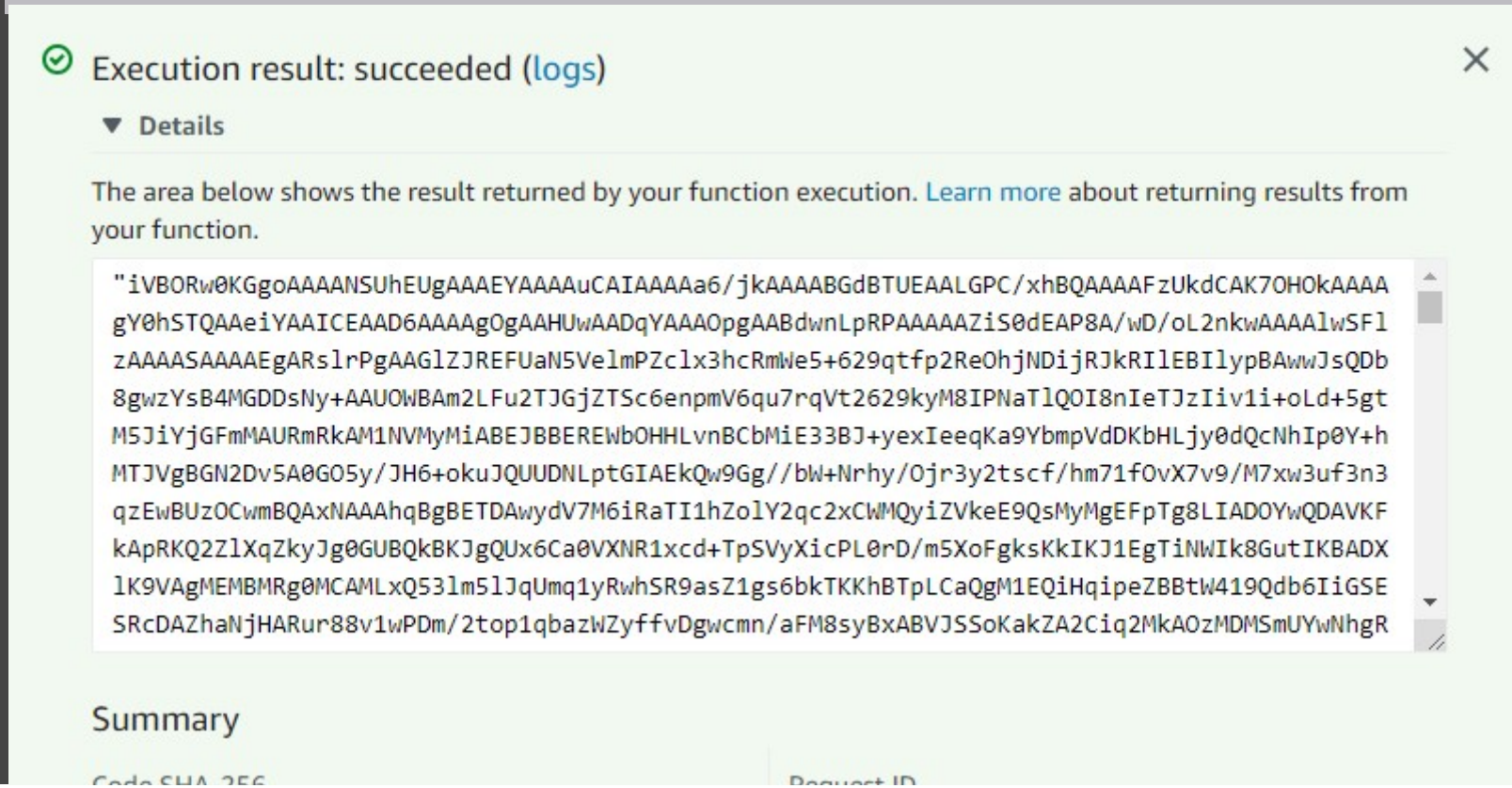
AWS Lambda

위에서 생성한 ping 테스트 이벤트로 동작 확인 후
위와 유사하게 sample 테스트 이벤트 생성.
Body json - {“operation”: “getSample”}



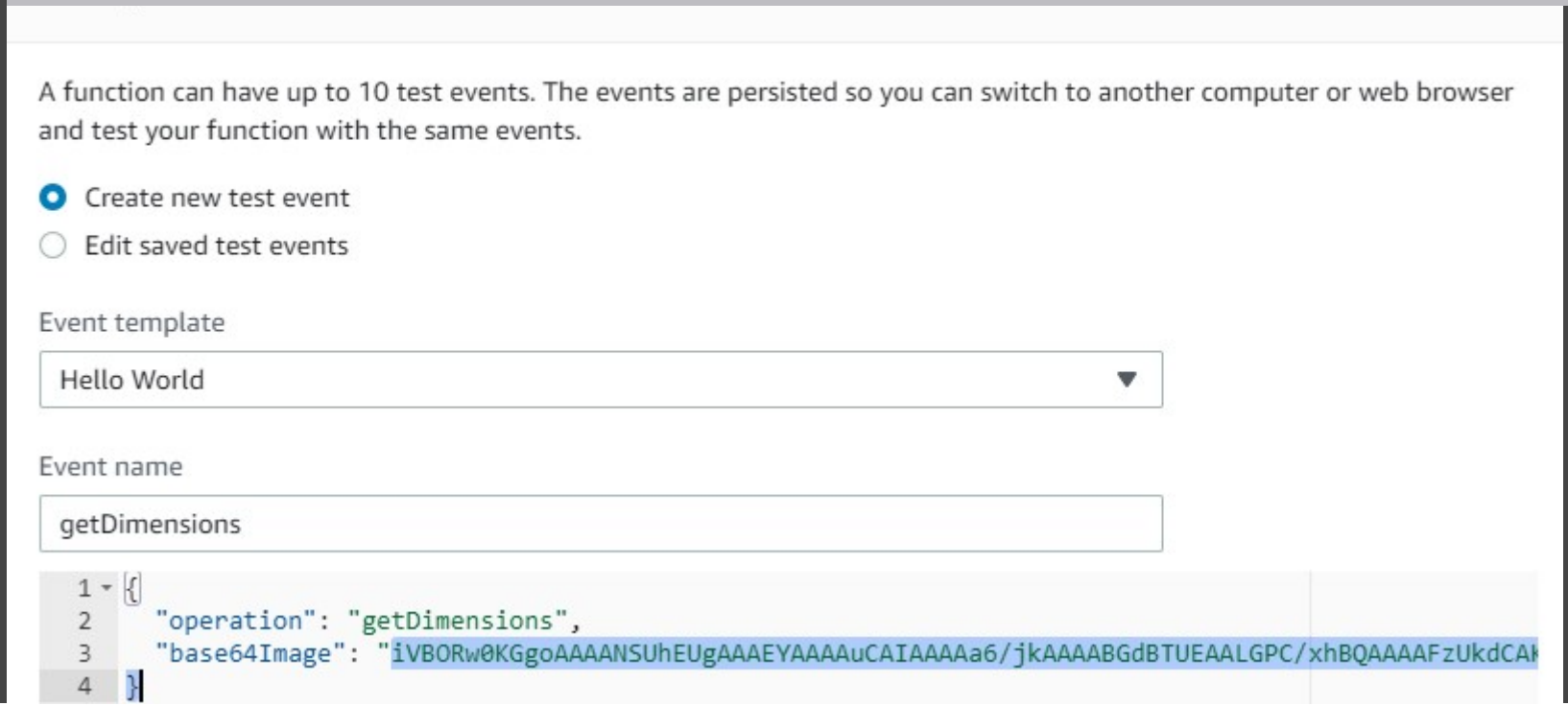
AWS Lambda

앞서 생성한 sample 이벤트는 내장 장미 이미지를 반환
https://codebeautify.org/base64-to-image-converter

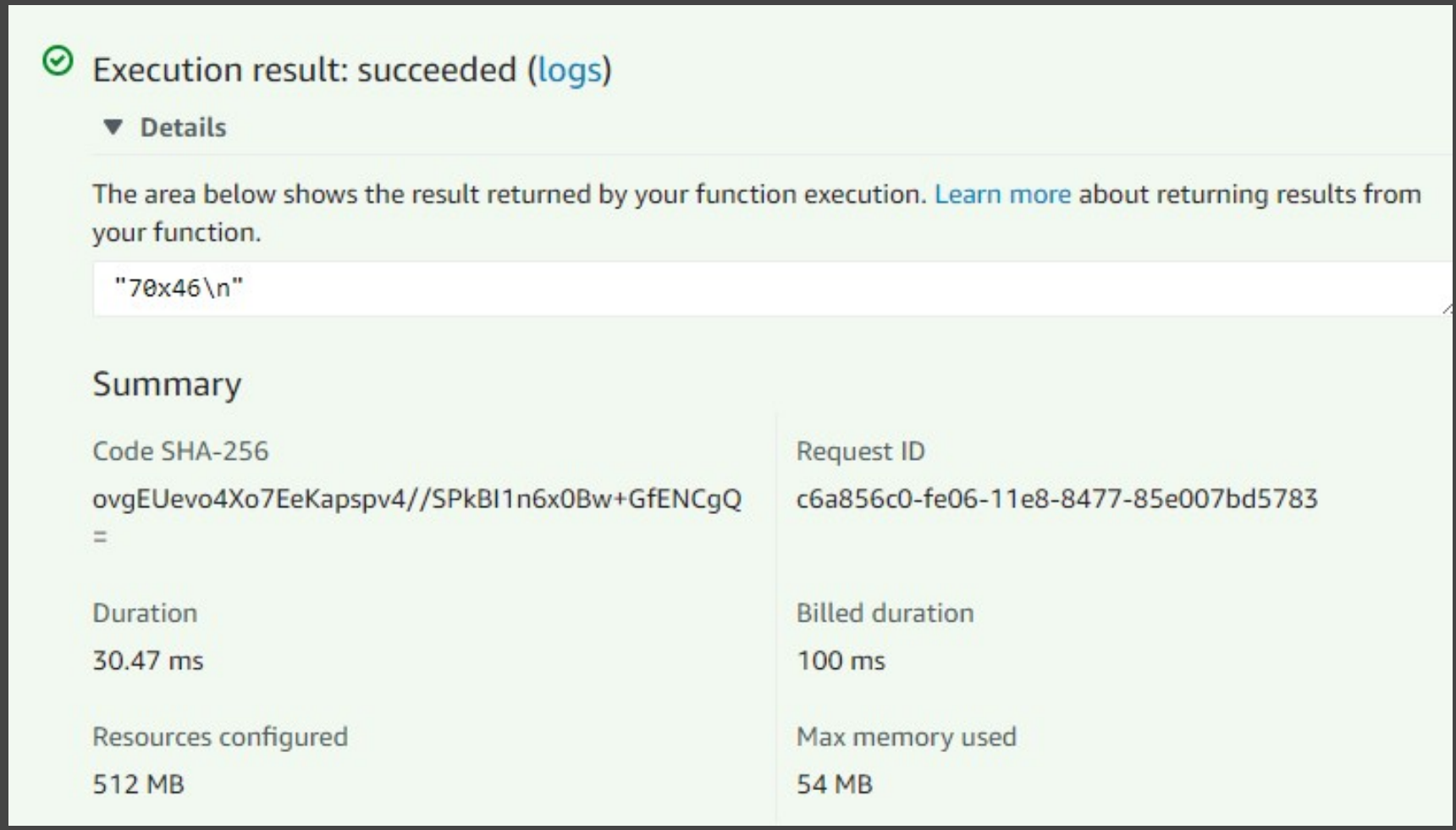


AWS Lambda

sample 이벤트가 출력한 base64 값을 인자로 사용하여 새 테스트 이벤트 생성
{“operation”: “getDimensions”, “base64Image”: “sample 이벤트의 출력 값”}



AWS Lambda


Execution result: succeeded (logs)
Details
The area below shows the result returned by your function execution. Learn more about returning results from your function.
"70x46\n"
Summary
Code SHA-256
ovgEUevo4Xo7EeKapspv4//SPkBI1n6x0Bw+GfENCgQ=
Request ID
c6a856c0-fe06-11e8-8477-85e007bd5783
Duration
30.47 ms
Billed duration
100 ms
Resources configured
512 MB
Max memory used
54 MB

AWS Lambda

SAM
- Serverless application model
- Cloudformation +

IAM
- Identity and Access Management
- AWS 내 자원에 대한 액세스/인증 제어

Cloudwatch
- AWS 자원 및 애플리케이션 모니터링
- 지표 수집/경보/로그