



# K6 탐색적 부하 테스트

: SLI 측정과 SNS/Gmail 알람 연동 실습

김찬빈, DevRel





# K6 탐색적 부하 테스트: SLI 측정과 SNS/Gmail 알람 연동 실습

김찬빈, DevRel

## Content.

Chapter 1 소개 및 배경

---

Chapter 2 실습 환경 설정

---

Chapter 3 K6 탐색적 부하 테스트

---

Chapter 4 What is SLI?

---

Chapter 5 CloudWatch & SNS & Gmail 알림 시스템 구축

---

# 소개 및 배경

**What topics will be covered in this presentation?**



# K6 탐색적 부하 테스트: SLI 측정과 SNS/Gmail 알람 연동 실습

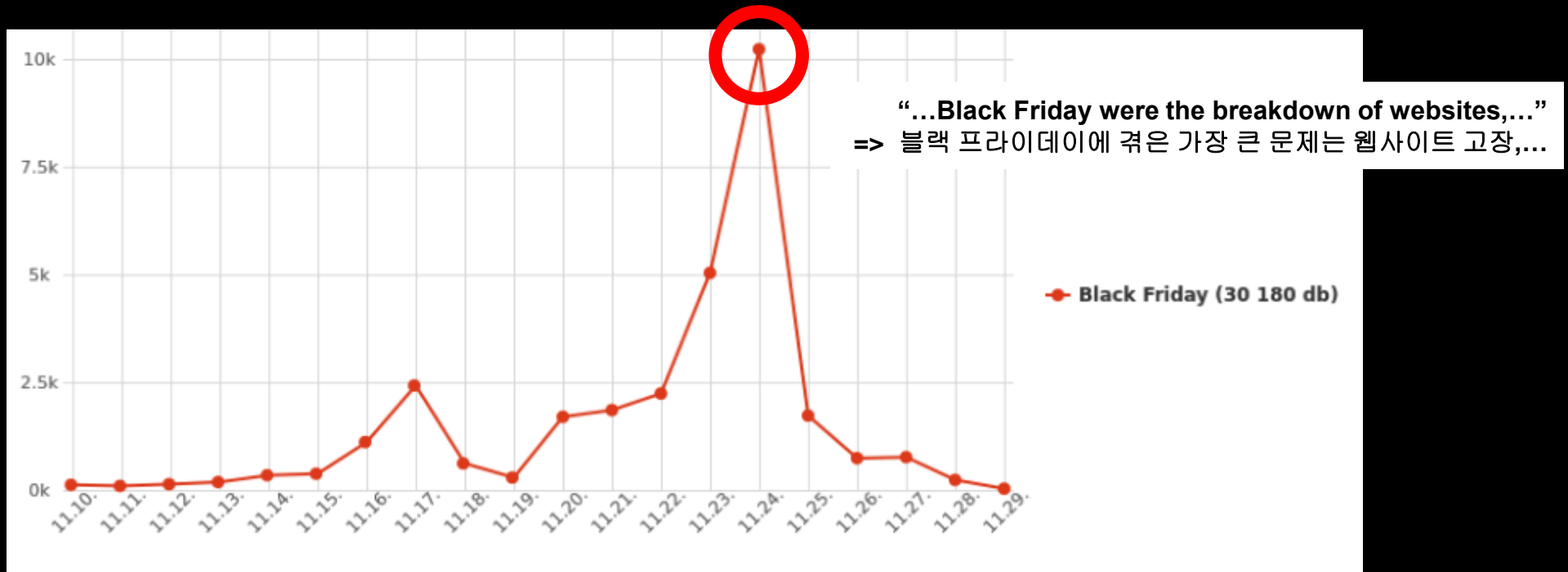
김찬빈, DevRel

## 왜 부하 테스트와 모니터링이 중요한가?

# K6 탐색적 부하 테스트: SLI 측정과 SNS/Gmail 알람 연동 실습

김찬빈, DevRel

## Black Friday 주문 트래픽 급증, 서비스 장애...매출과 직결



“How to successfully handle Black Friday in Hungary”, 2018.10.11. Expandeco  
<https://www.expandeco.com/en/how-to-successfully-handle-black-friday-in-hungary/>

# K6 탐색적 부하 테스트: SLI 측정과 SNS/Gmail 알람 연동 실습

김찬빈, DevRel



*부하 테스트를 왜 하는가?*

현 시스템의 한계를 미리 파악, 장애 예방

*모니터링은 왜 중요한가?*

문제가 발생하기 전, 조기 감지 및 대응을 위함



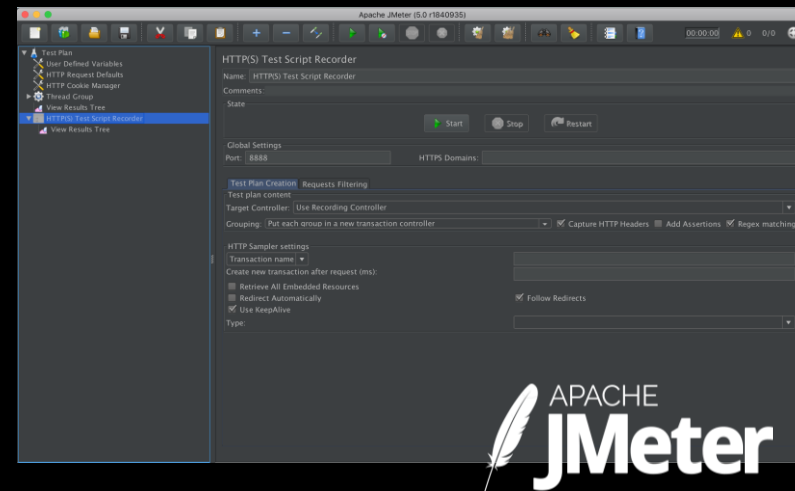
# K6 탐색적 부하 테스트: SLI 측정과 SNS/Gmail 알람 연동 실습

김찬빈, DevRel



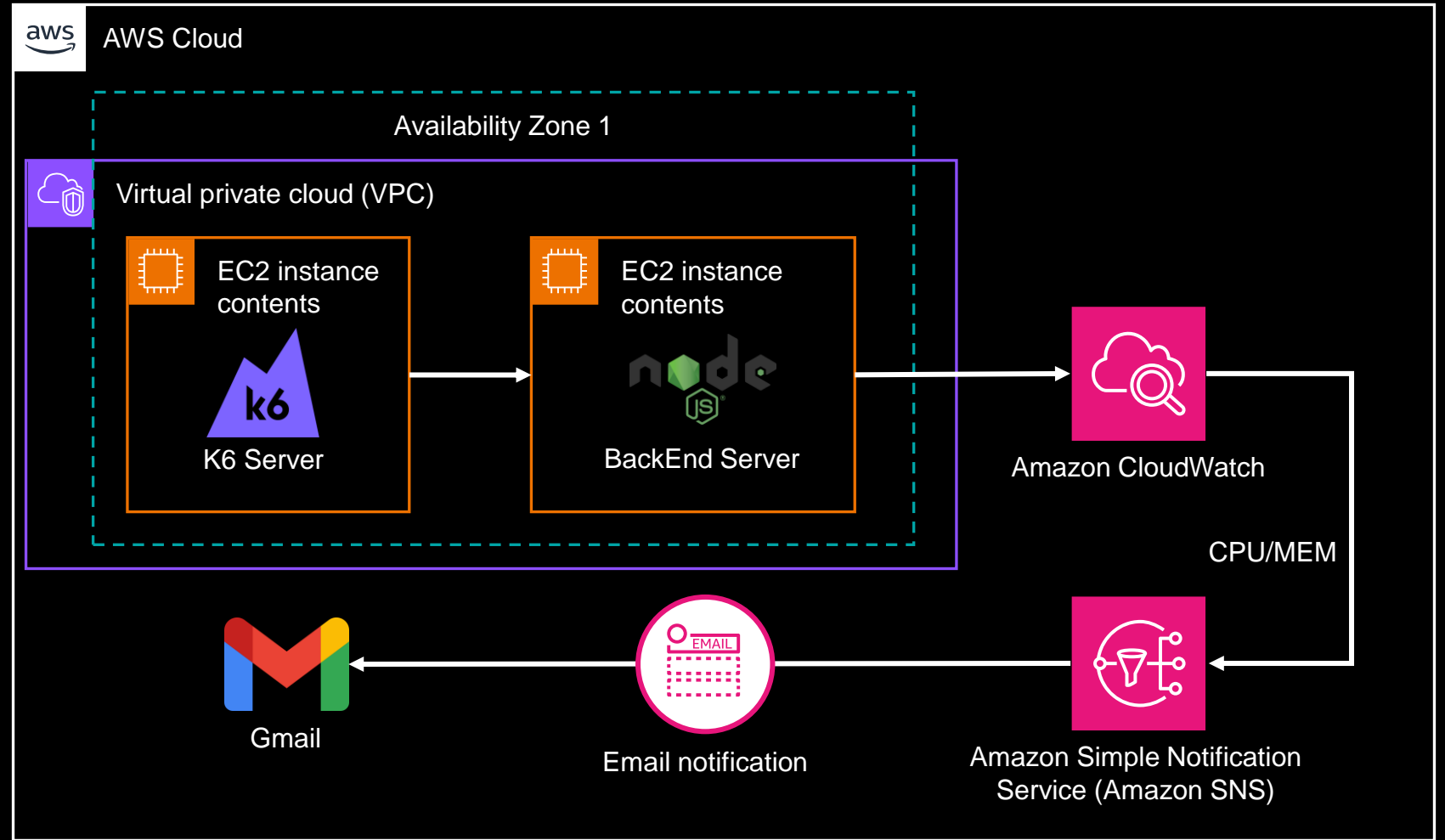
## Why, K6?

- 오픈소스, 무료
- JavaScript 기반 스크립트
- GUI 없이, 빠르게 CLI로 테스트
- 직관적, 개바개 취향



한정된 Meet-Up 시간 내 빠르게 테스트하기 적합하다 판단함.

## System Architecture

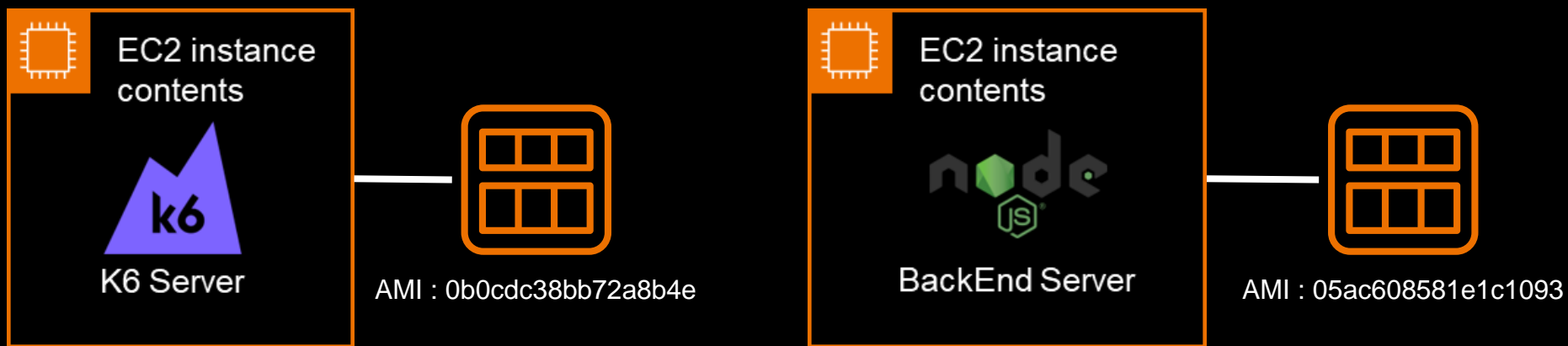




# 실습 환경 설정

Hands-on material preparation

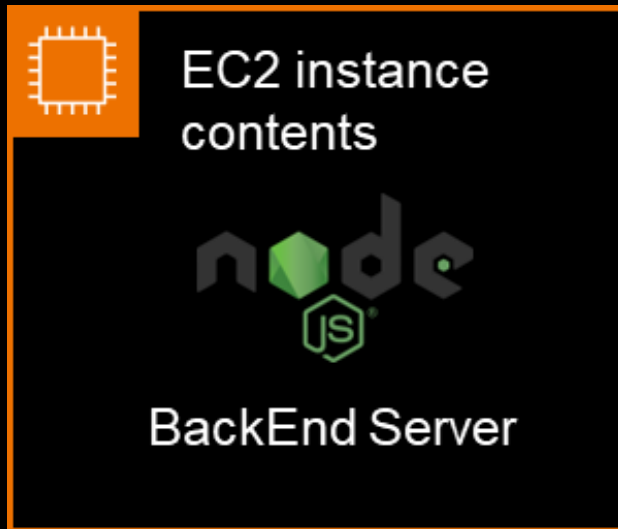
## AMI 기반 인스턴스 생성



- Ubuntu 24.x LTS
- 퍼블릭 IP 허용 (탄력적 IP 아님)

## AMI 기반 인스턴스 생성 (Node)

3000번 포트에 대한 인바운드 규칙 추가 필수



AMI : 05ac608581e1c1093  
NAME : ACC-DGU-Node-Server

### 인바운드 규칙 편집

인바운드 규칙은 인스턴스에 도달하도록 허용된 수신 트래픽을 제어합니다.

#### 인바운드 규칙

보안 그룹 규칙 ID  
sgr-0bc3381fe1b3d524d

유형 정보

사용자 지정 TCP

프로토콜 정보

TCP

포트 범위 정보

3000

소스 정보

사용자 지정

Q

0.0.0.0/0

선택된 AMI: (ami-05ac608581e1c1093) (커뮤니티 AMI)

ami-05ac608581e1c1093

빠른 시작 AMI(0)

일반적으로 사용하는 AMI

내 AMI(0)

내가 생성

AWS Marketplace AMI(3164)

AWS 및 신뢰할 수 있는 서드 파티 AMI

커뮤니티 AMI(500)

누구나 게시

#### 결과 구체화

필터 모두 지우기

#### 운영 체제

##### Linux/UNIX

- ☐ 모든 Linux/UNIX
- ☐ Amazon Linux
- ☐ CentOS
- ☐ Debian
- ☐ Fedora
- ☐ Gentoo

ami-05ac608581e1c1093(1개 필터링됨, 500개 필터링되지 않음)

#### 커뮤니티 AMI

커뮤니티 AMI에는 퍼블릭 AMI가 모두 포함되어 있습니다. 따라서 누구나 AMI를 게시할 수 있고 해당 AMI는 이 카탈로그에 표시됩니다. 이 카탈로그는 유료 제품도 포함할 수 있습니다. 커뮤니티 AMI를 사용할 때는 AMI를 시작하기 전에 게시자를 알고 신뢰할 수 있는지 확인하는 것이 좋습니다.



#### ACC-DGU-Node-Server

ami-05ac608581e1c1093

Ubuntu-based ACC\_DGU\_Node API Server

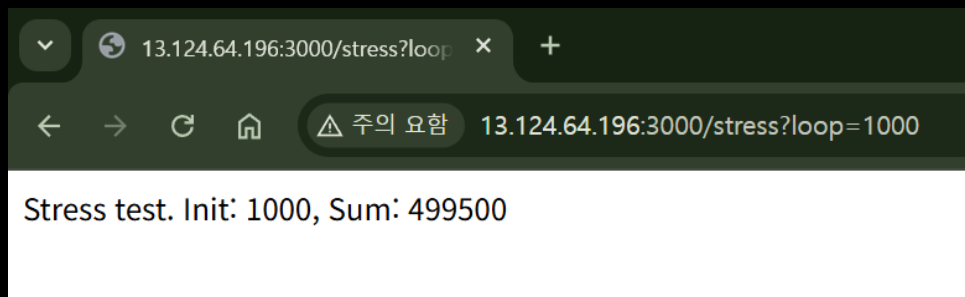
OwnerAlias: - 플랫폼: - 아키텍처: x86\_64 소유자: 294393683696 게시 날짜: 2025-07-14 루트 디바이스 유형: ebs 가상화: hvm ENA 활성화됨: 예 부트 모드: uefi-preferred

선택

## 간단한 node.js 서버 제공

### 접속 테스트

- `http://<public ip>:3000/ping`
- `http://<public ip>:3000/stress?loop=<x>`



/home/ACC\_DGU/app.js

```
const express = require('express');
const app = express();
const port = 3000;

app.get('/ping', (req, res) => {
  res.send('PONG!');
});

app.get('/stress', (req, res) => {
  const loop = parseInt(req.query.loop) || 10000000;
  let sum = 0;

  for (let i = 0; i < loop; i++) {
    sum += i;
  }

  res.send(`Stress test. Init: ${loop}, Sum: ${sum}`);
});

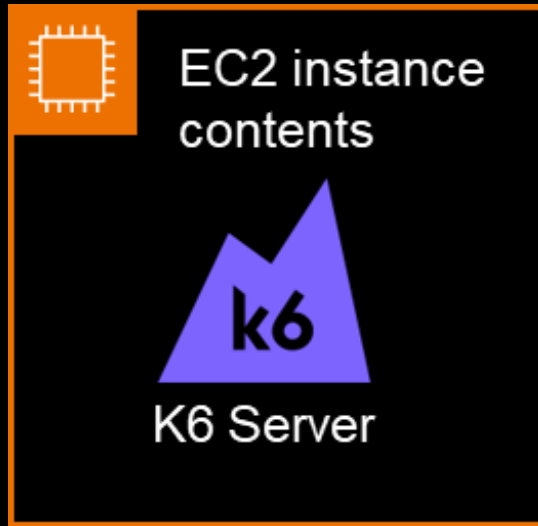
app.listen(port, () => {
  console.log(`port: ${port}`);
});
```



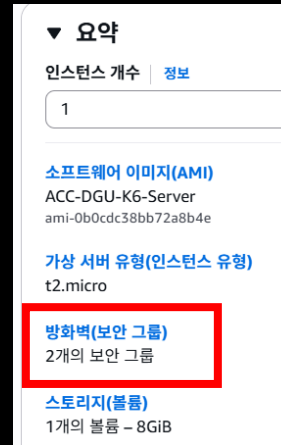
# K6 탐색적 부하 테스트: SLI 측정과 SNS/Gmail 알람 연동 실습

김찬빈, DevRel

## AMI 기반 인스턴스 생성 (K6)



AMI : 0b0cdc38bb72a8b4e  
NAME : ACC-DGU-K6-Server



아까 설정한 보안그룹...



## K6 설치 AMI, 샘플 코드 제공

### 설치 확인

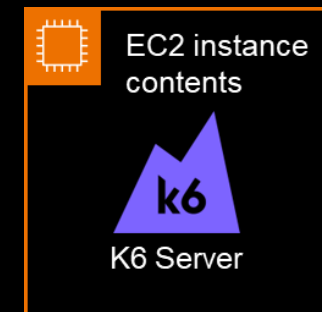
- K6 --version

```
root@ip-172-31-32-146:~# k6 --version
k6 v1.1.0 (commit/0e3fb953be, go1.24.4, linux/amd64)
```

```
import http from 'k6/http';
import { check, sleep } from 'k6';

export const options = {
  vus: 1,
  duration: '5s',
};

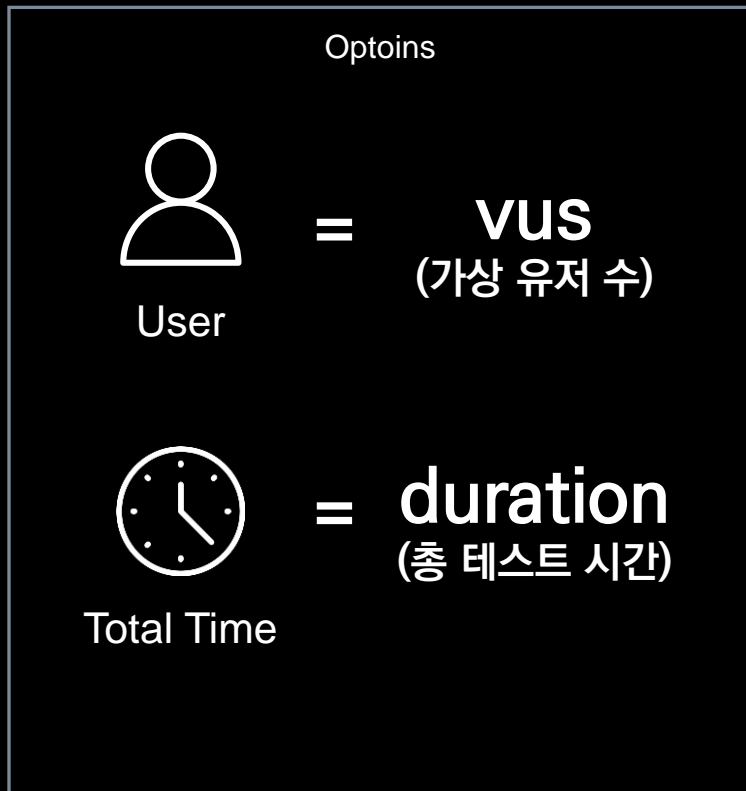
export default function () {
  const res = http.get('http://[TARGET]:3000/ping');
  check(res, {
    'status is 200': (r) => r.status === 200,
  });
  sleep(1);
}
```



# K6 탐색적 부하 테스트

Exploratory load testing with K6

## 테스트 템플릿 K6 구조



200 OK!



## 기본 K6 구문 샘플 코드 핸들링

- options : 사용자 수(vus), 얼마간 (duration)
  - fun : 결과(res), 간격(sleep)
- > 1명의 가상 사용자가 5초간 1초 간격으로 전송

/home/ACC\_DGU/sample.js

```
import http from 'k6/http';
import { check, sleep } from 'k6';

export const options = {
  vus: 1,
  duration: '5s',
};

export default function () {
  const res = http.get('http://[TARGET]:3000/ping');
  check(res, {
    'status is 200': (r) => r.status === 200,
  });
  sleep(1);
}
```

## 기본 K6 구문 샘플 코드 핸들링

/home/ACC\_DGU/hands-on.js

- `cp ./sample.js ./hands-on.js`
- `sudo vim ./hands-on.js`
- 10명의 가상 사용자가 10초간 1초 간격으로 전송
- API 서버 IP 확인
- Stress API로 테스트 진행

```
import http from 'k6/http';
import { check, sleep } from 'k6';

export const options = {
  vus: 10,
  duration: '10s',
};

export default function () {
  const res = http.get('http://52.79.241.23:3000/stress?loop=1000000');
  check(res, {
    'status is 200': (r) => r.status === 200,
  });
  sleep(1);
}
```

## K6 테스트 실행

`k6 run ./hands-on.js`

1 요청 성공 여부

2 HTTP 호출 성능

평균 응답시간, 95 백분위 응답시간 확인

3 실행 통계

4 네트워크 통계

```
root@ip-172-31-32-146:/home/ACC_DGU# k6 run ./hands-on.js
```



```
execution: local
script: ./hands-on.js
output: -
```

```
scenarios: (100.00%) 1 scenario, 10 max VUs, 40s max duration (incl. graceful stop):
* default: 10 looping VUs for 10s (gracefulStop: 30s)
```

### TOTAL RESULTS

```
checks_total.....: 100      9.925729/s
checks_succeeded.....: 100.00% 100 out of 100
checks_failed.....: 0.00% 0 out of 100
```

✓ status is 200

### HTTP

```
http_req_duration.....: avg=5.08ms min=1.89ms med=2.32ms max=44.53ms p(90)=4.67ms p(95)=29.5ms
{ expected_response:true }.....: avg=5.08ms min=1.89ms med=2.32ms max=44.53ms p(90)=4.67ms p(95)=29.5ms
http_req_failed.....: 0.00% 0 out of 100
http_reqs.....: 100      9.925729/s
```

### EXECUTION

```
iteration_duration.....: avg=1s      min=1s      med=1s      max=1.04s    p(90)=1s    p(95)=1.03s
iterations.....: 100      9.925729/s
vus.....: 10      min=10      max=10
vus_max.....: 10      min=10      max=10
```

### NETWORK

```
data_received.....: 27 kB  2.7 kB/s
data_sent.....: 9.2 kB  913 B/s
```

```
running (10.1s), 00/10 VUs, 100 complete and 0 interrupted iterations
default ✓ [=====] 10 VUs 10s
root@ip-172-31-32-146:/home/ACC_DGU#
```

## 탐색적 부하 테스트

제한 시간(10초) 내에서 응답 지연으로 인해 처리된 요청이 줄어드는 결과, 그리고 평균 응답시간의 변화 확인.

계속해서, vus 수를 늘리거나 loop 값을 조절하는 등, 시스템 한계를 더 탐색할 수 있음

```
HTTP
http_req_duration.....: avg=5.08ms min=1.89ms med=2.32ms max=44.53ms p(90)=4.67ms p(95)=29.5ms
{ expected_response:true }.....: avg=5.08ms min=1.89ms med=2.32ms max=44.53ms p(90)=4.67ms p(95)=29.5ms
http_req_failed.....: 0.00% 0 out of 100
http_reqs.....: 100 9.925729/s
```



```
HTTP
http_req_duration.....: avg=174.15ms min=103.91ms med=121.88ms max=1.11s p(90)=144.78ms p(95)=621.15ms
{ expected_response:true }.....: avg=174.15ms min=103.91ms med=121.88ms max=1.11s p(90)=144.78ms p(95)=621.15ms
http_req_failed.....: 0.00% 0 out of 90
http_reqs.....: 90 8.113126/s
```



```
HTTP
http_req_duration.....: avg=7.89s min=1.05s med=4.59s max=17.81s p(90)=15.9s p(95)=16.85s
{ expected_response:true }.....: avg=7.89s min=1.05s med=4.59s max=17.81s p(90)=15.9s p(95)=16.85s
http_req_failed.....: 0.00% 0 out of 18
http_reqs.....: 18 0.834299/s
```

# What is SLI?

About SRE in DevOps...



# K6 탐색적 부하 테스트: SLI 측정과 SNS/Gmail 알람 연동 실습

김찬빈, DevRel

## SLI (Service Level Indicator)

“서비스 품질을 정량적으로 측정하는 지표”

“사용자가 직접 경험하는 성능 지표”

...

Example #1

요청 중 실패 비율 0.1% 미만

Example #2

24시간 내 서비스 가용성 99.9%

Example #3

서비스 응답 시간 평균 100ms,  
95%가 200ms 이내

SLI (Service Level Indicator) = 서비스 품질을 측정하는 지표 (현재 상태)



응답 시간 평균 200ms

SLO (Service Level Objective) = SLI를 기반으로 설정한 목표 (목표 상태)



응답 시간 평균 100ms 이내

SLA (Service Level Agreement) = 고객과 약속하는 서비스 계약 (보장 수준)

응답 시간 보장, 위반 시 보상 정의 등



# K6 탐색적 부하 테스트: SLI 측정과 SNS/Gmail 알람 연동 실습

김찬빈, DevRel

## 1 SLI (Service Level Indicator)

100명의 사용자가

10초간 1초 간격으로

Loop=10000을 호출한 결과

HTTP 응답 평균 7.19ms

```
import http from 'k6/http';
import { check, sleep } from 'k6';

export const options = {
  vus: 100,
  duration: '10s',
};

export default function () {
  const res = http.get('http://52.79.241.23:3000/stress?loop=10000');
  check(res, {
    'status is 200': (r) => r.status === 200,
  });
  sleep(1);
}
```

### HTTP

```
http_req_duration.....: avg=7.19ms min=681.64µs med=2.2ms max=55.21ms p(90)=18.43ms p(95)=32.61ms
{ expected_response:true }.....: avg=7.19ms min=681.64µs med=2.2ms max=55.21ms p(90)=18.43ms p(95)=32.61ms
http_req_failed.....: 0.00% 0 out of 1000
http_reqs.....: 1000 98.632011/s
```

## 2 SLO (Service Level Objective)

“100명 동시 접속, loop = 10000 시 HTTP 응답 평균 20ms 이내”

## 3 SLA (Service Level Agreement)

“목표 미달 시 지체보상금 지불 등”



# CloudWatch & SNS & Gmail 알림 시스템 구축

Hands-on for creating load problem alert notifications.



# K6 탐색적 부하 테스트: SLI 측정과 SNS/Gmail 알람 연동 실습

김찬빈, DevRel



Amazon Simple Notification  
Service (Amazon SNS)

## AWS Simple Notification Service (SNS)

알림 전송 서비스, MQ와 같이 “구독자”들에게 메시지를 보낼 수 있다.



Amazon CloudWatch

## AWS CloudWatch

AWS 리소스(CPU 사용률, 네트워크 등) 모니터링 및 알람 설정 서비스(도구)

## AWS Simple Notification Service

AWS Console에서 AWS SNS로 접속한 후,  
새로운 주제 생성.

그 후, “표준” 형태로 주제를 생성한다.

### 주제 생성

**주제 이름**  
주제는 메시지 채널입니다. 메시지를 주제에 게시하면 이는 모든 구독 엔드포인트에 전송됩니다.

**다음 단계**

[개요로 시작](#)

### 세부 정보

**유형** 정보  
주제를 생성한 후에는 주제 유형을 수정할 수 없음

☐ FIFO(선입선출)

- 엄격하게 보존된 메시지 순서 지정
- 정확히 1회 메시지 전송
- 구독 프로토콜: SQS

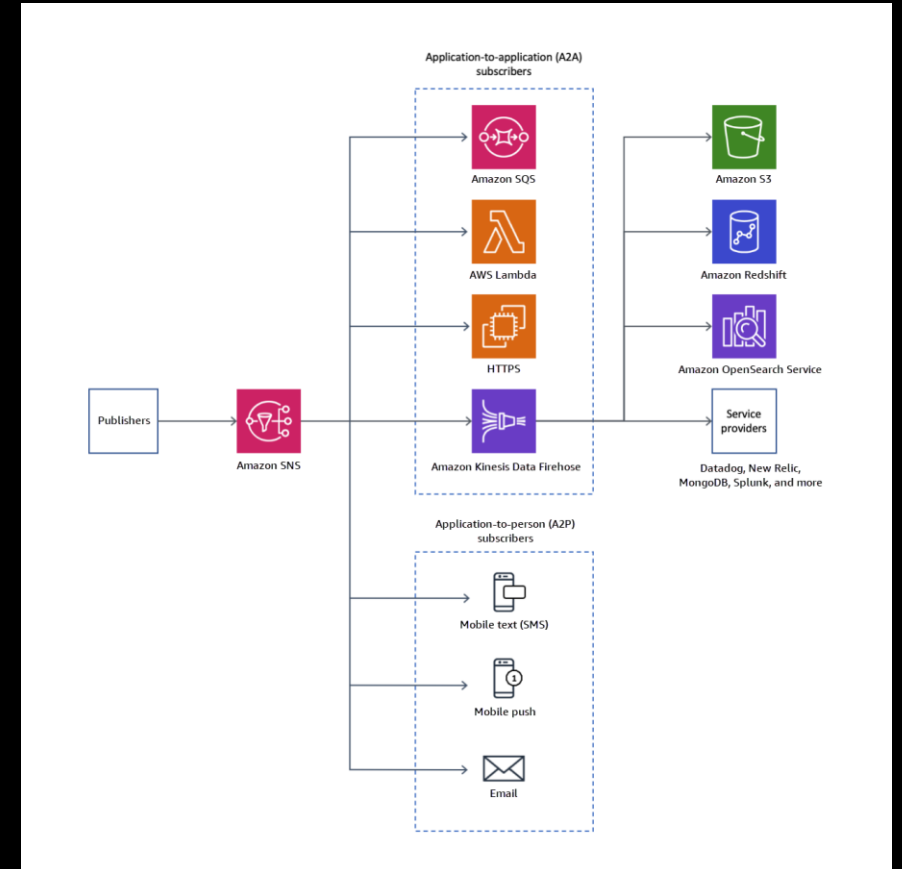
☒ 표준

- 최상의 메시지 순서 지정
- 최소 1회 메시지 전송
- 구독 프로토콜: SQS, Lambda, Data Firehose, HTTP, SMS, 이메일, 모바일 애플리케이션 엔드포인트

**이름**  
  
최대 256자이며 영숫자, 하이픈(-) 및 밑줄(\_)을 포함할 수 있습니다.

**표시 이름 - 선택 사항** 정보  
이 주제를 SMS 구독과 함께 사용하려면 표시 이름을 입력하십시오. 처음 10자만 SMS 메시지에 표시됩니다.

최대 100자.

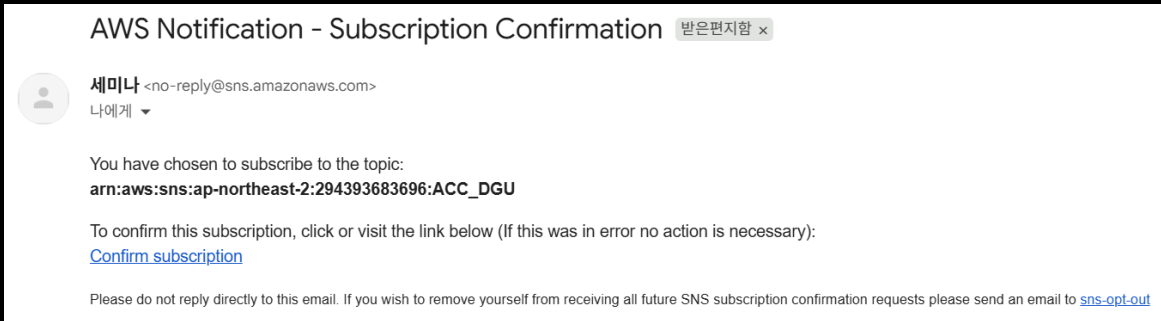


[https://docs.aws.amazon.com/ko\\_kr/sns/latest/dg/welcome.html](https://docs.aws.amazon.com/ko_kr/sns/latest/dg/welcome.html)

## AWS Simple Notification Service

새로운 “구독 생성”을 선택한 뒤,  
이메일 프로토콜을 사용하여 개인 메일을 입력하면 된다.

이후, 아래와 같은 확인 메일을 확인한다.





# K6 탐색적 부하 테스트: SLI 측정과 SNS/Gmail 알람 연동 실습

김찬빈, DevRel

## AWS Simple Notification Service

“확인됨” 상태로 변경되면, SNS 기본 작업은 끝난다.

ACC\_DGU

편집삭제메시지 게시

세부 정보

이름

ACC\_DGU

표시 이름

세미나

ARN

arn:aws:sns:ap-northeast-2:294393683696:ACC\_DGU

주제 소유자

294393683696

유형

표준

구독

액세스 정책

데이터 보호 정책

전송 정책(HTTP/S)

전송 상태 로깅

암호화

태그

통합

구독 (1)

편집삭제확인 요청구독 확인구독 생성

Q 검색

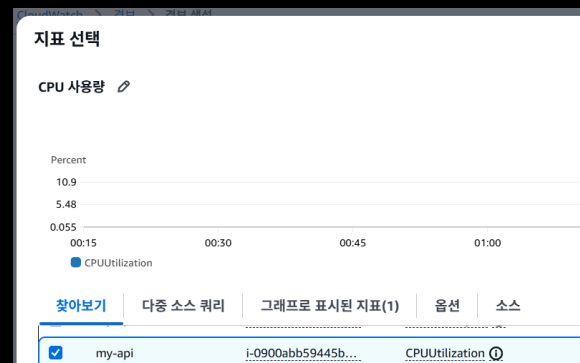
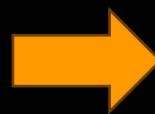
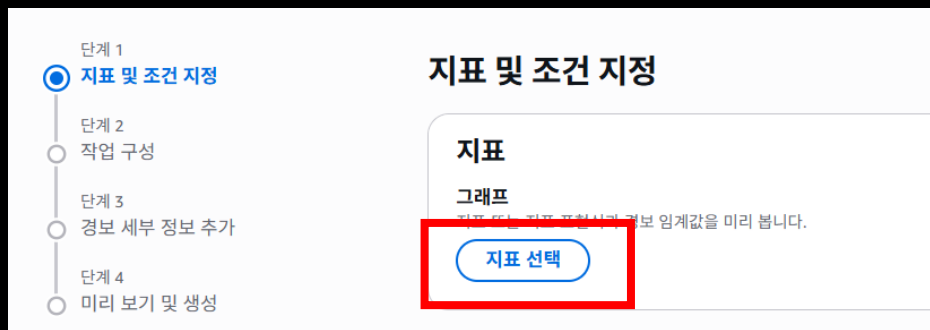
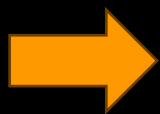
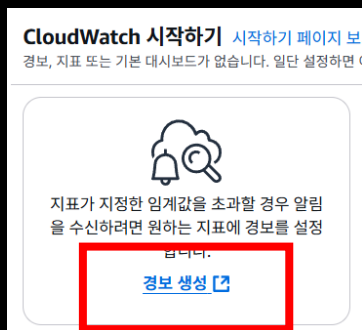
< 1 > ⚙

ID	엔드포인트	상태	프로토콜
<input type="radio"/> <a href="#">e3abc6de-5928-48fd-b9f8-32d670ea9ae7</a>	flqld86851@gmail.com	✓ 확인됨	EMAIL

## AWS CloudWatch

AWS Console에서 CloudWatch로 접속한 후,  
새로운 경보 생성으로 진입.

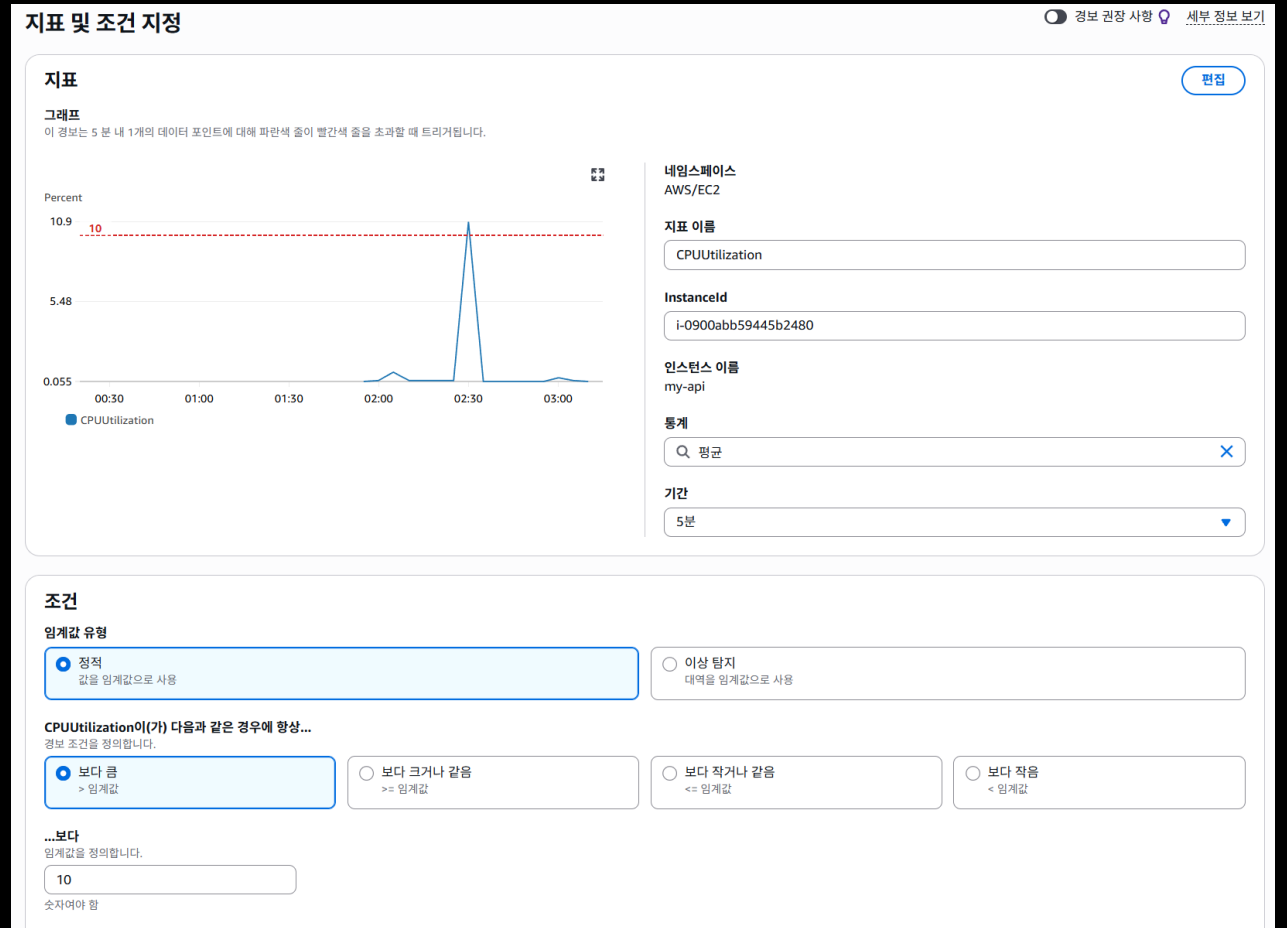
그 후, API 서버(Node)의 **CPU 사용량**을 선택 (자유)



## AWS CloudWatch

원하는 기준으로 지표를 생성할 것.

예시는 CPU 사용량이 5분 평균  
“10%”이상인 경우 발생





# K6 탐색적 부하 테스트: SLI 측정과 SNS/Gmail 알람 연동 실습

김찬빈, DevRel

## AWS CloudWatch

앞서 생성한 SNS 주제를 선택.  
이후, 경보 이름을 설정한 뒤  
생성을 완료한다.

### 작업 구성

#### 알림

##### 경보 상태 트리거

이 작업을 트리거하는 경보 상태를 정의합니다.

##### ☒ 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.

제거

##### 다음 SNS 주제에 알림을 보냅니다.

알림을 수신할 SNS(Simple Notification Service) 주제를 정의합니다.

##### ☒ 기존 SNS 주제 선택

##### ☐ 새 주제 생성

☐ 주제 ARN을 사용하여 다른 계정에 알림

##### 다음으로 알림 전송...

Q SNS 주제 선택

ACC\_DGU

알림 추가

### 경보 세부 정보 추가

#### 이름 및 설명

##### 경보 이름

ACC\_DGU\_CPU\_CHECKER

##### 경보 설명 - 선택 사항 [서식 가이드라인 보기](#)

##### 편집

##### 미리 보기

# 이것은 H1입니다

\*\*이중 별표는 강조 문자를 제공합니다.\*\*

이것은 [예제](https://example.com/) 인라인 링크입니다.

최대 1024자(0/1024)





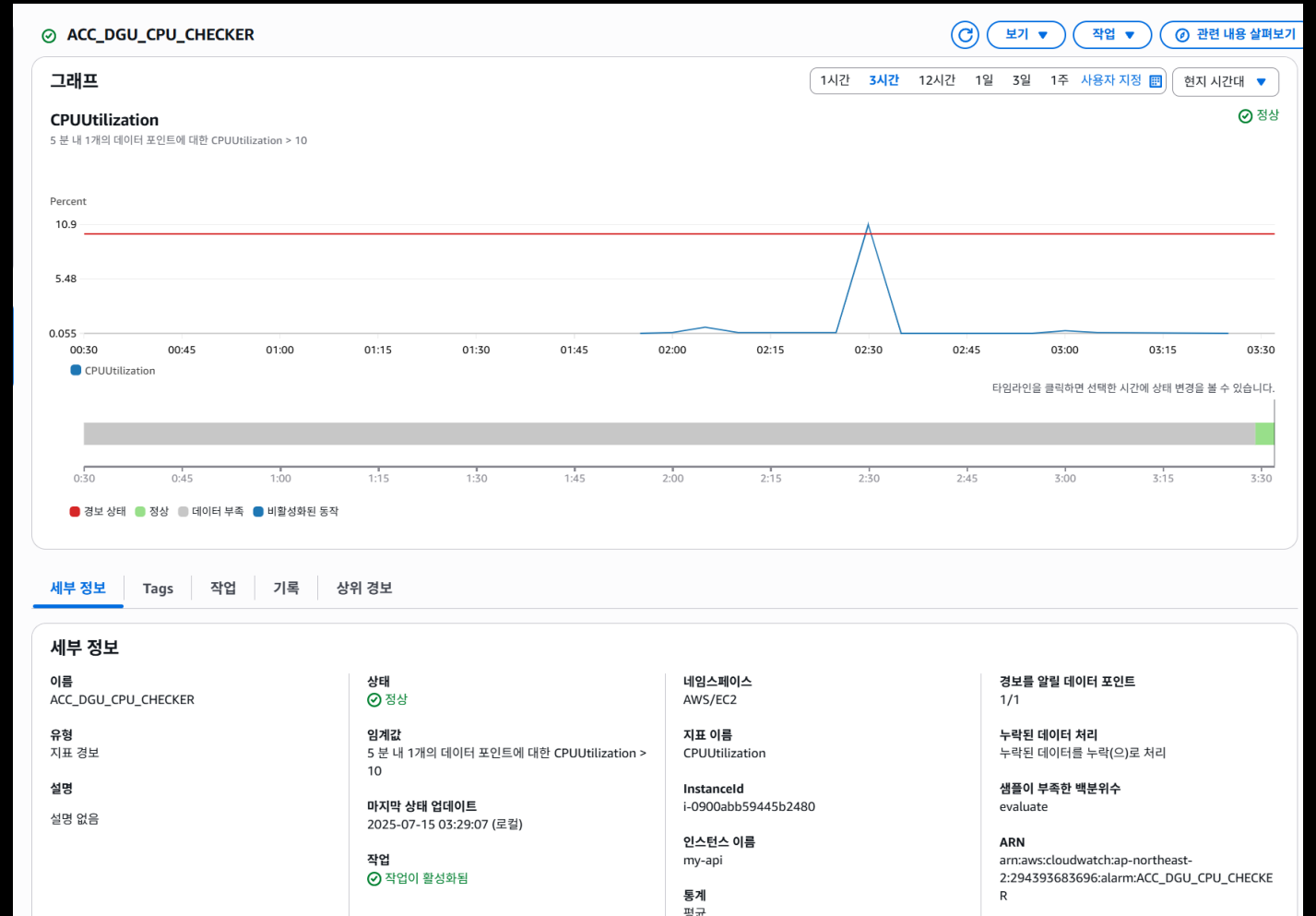


# K6 탐색적 부하 테스트: SLI 측정과 SNS/Gmail 알람 연동 실습

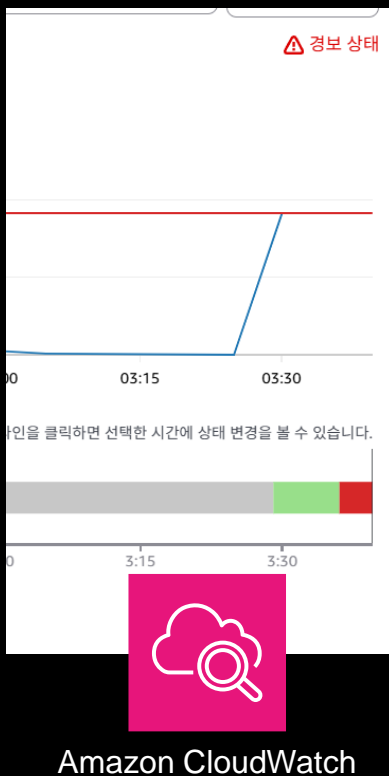
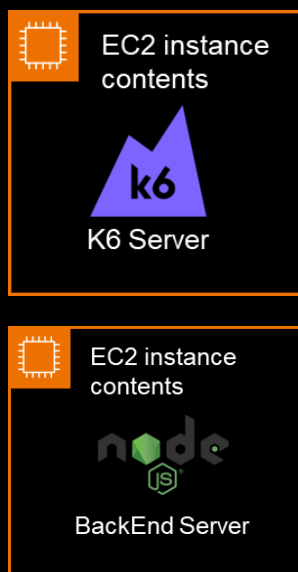
김찬빈, DevRel

## AWS CloudWatch

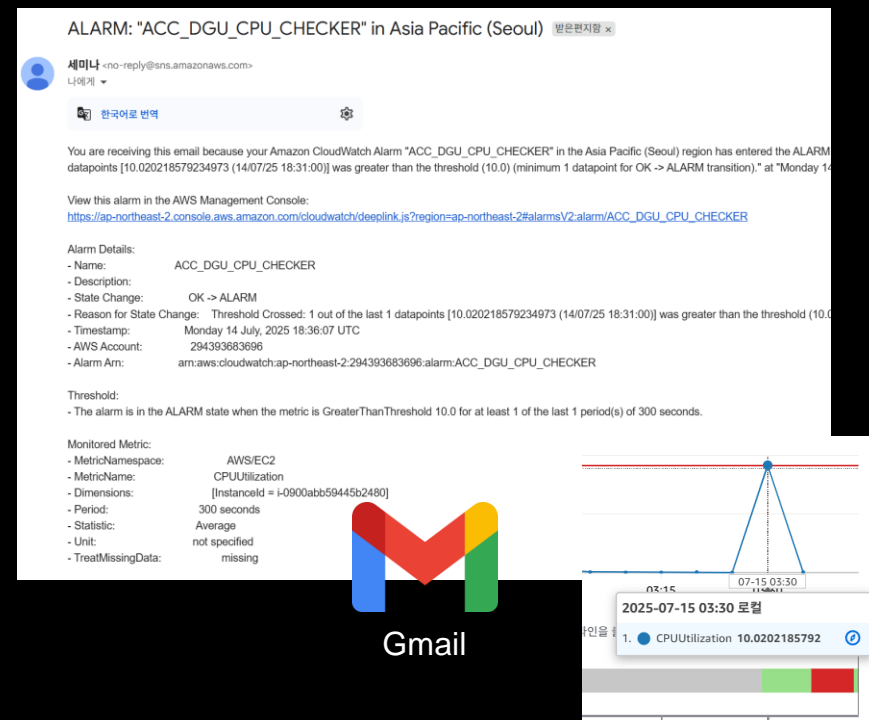
경보 생성이 완료된 모습



## 최종 테스트



Amazon Simple Notification Service (Amazon SNS)

ALARM: "ACC\_DGU\_CPU\_CHECKER" in Asia Pacific (Seoul)

세미나 <no-reply@sns.amazonaws.com>  
나에게

한국어로 번역

You are receiving this email because your Amazon CloudWatch Alarm "ACC\_DGU\_CPU\_CHECKER" in the Asia Pacific (Seoul) region has entered the ALARM state. Datapoints [10.020218579234973 (14/07/25 18:31:00)] was greater than the threshold (10.0) (minimum 1 datapoint for OK -> ALARM transition)." at "Monday 14 July 2025 18:36:07 UTC".

View this alarm in the AWS Management Console:  
[https://ap-northeast-2.console.aws.amazon.com/cloudwatch/deeplink.js?region=ap-northeast-2#alarmsV2:alarm/ACC\\_DGU\\_CPU\\_CHECKER](https://ap-northeast-2.console.aws.amazon.com/cloudwatch/deeplink.js?region=ap-northeast-2#alarmsV2:alarm/ACC_DGU_CPU_CHECKER)

Alarm Details:

- Name: ACC\_DGU\_CPU\_CHECKER
- Description: OK -> ALARM
- State Change: OK -> ALARM
- Reason for State Change: Threshold Crossed: 1 out of the last 1 datapoints [10.020218579234973 (14/07/25 18:31:00)] was greater than the threshold (10.0) (minimum 1 datapoint for OK -> ALARM transition)." at "Monday 14 July 2025 18:36:07 UTC"
- Timestamp: Monday 14 July, 2025 18:36:07 UTC
- AWS Account: 294393683696
- Alarm ARN: arn:aws:cloudwatch:ap-northeast-2:294393683696:alarm:ACC\_DGU\_CPU\_CHECKER

Threshold:

- The alarm is in the ALARM state when the metric is GreaterThanThreshold 10.0 for at least 1 of the last 1 period(s) of 300 seconds.

Monitored Metric:

- MetricNamespace: AWS/EC2
- MetricName: CPUUtilization
- Dimensions: [InstanceId = i-0900abb59445b2480]
- Period: 300 seconds
- Statistic: Average
- Unit: not specified
- TreatMissingData: missing

Gmail

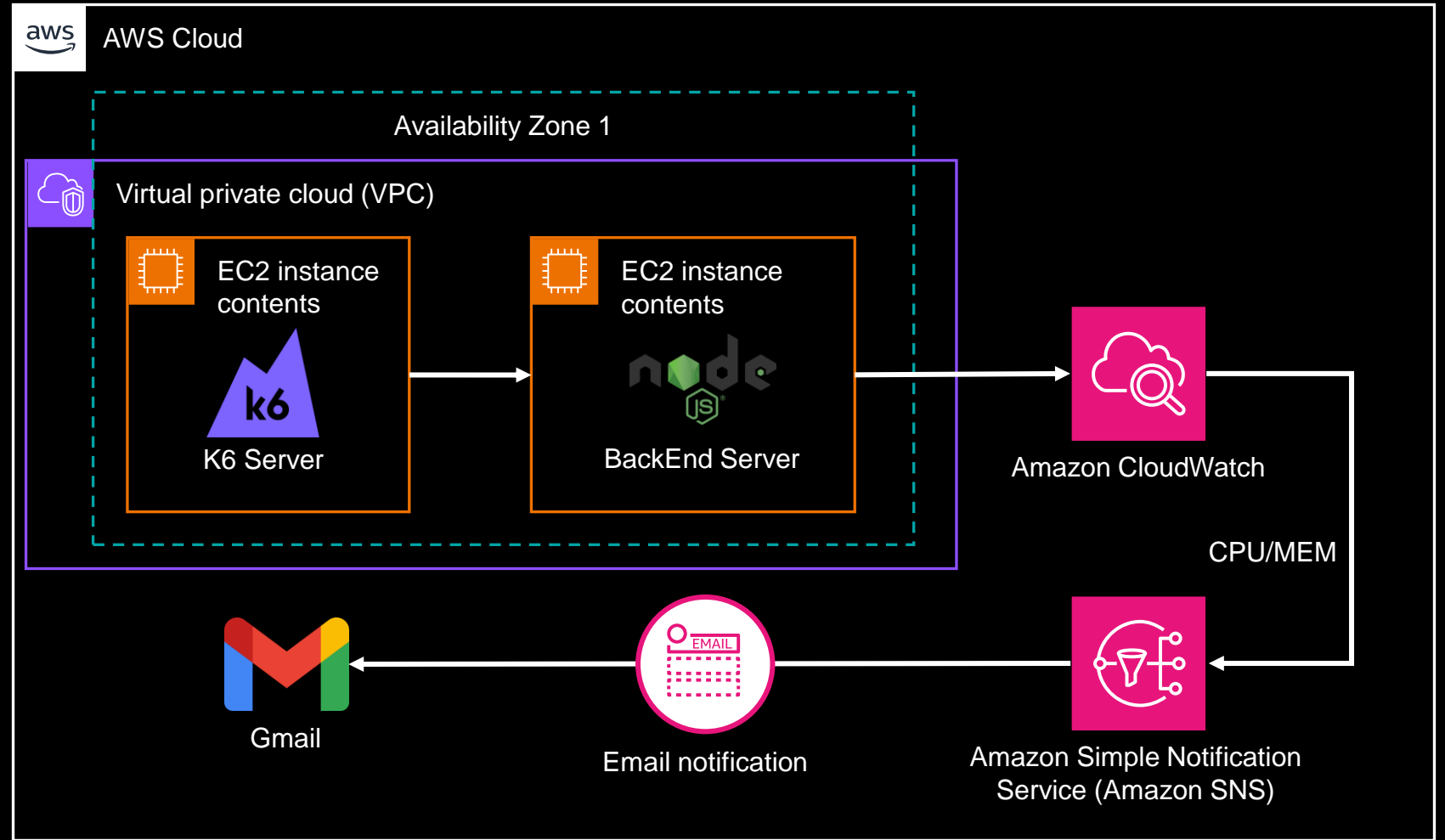
2025-07-15 03:30 로컬

1. CPUUtilization 10.0202185792

# K6 탐색적 부하 테스트: SLI 측정과 SNS/Gmail 알람 연동 실습

김찬빈, DevRel

고생하셨습니다.



# Q&A

**I hope you enjoyed it.**