



## 5. AWS

# ■ 목차

- Amazon Web Services
- *Account*
- *EC2*
- *S3*
- *CloudFront*
- *RDS*
- *Auto Scaling*
- *Elastic Load Balancer*
- *Lambda*

## ■ AWS EC2

- Root 계정
- IAM 계정 
- Region
- Availability Zone
- VPC
- Private IP, Publish IP, Elastic IP

## ■ AWS EC2

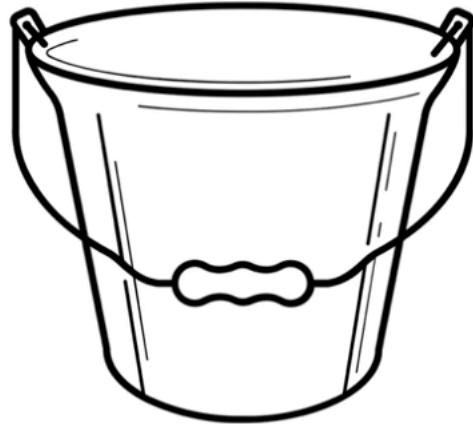
### ■ EC2 생성

- Node 설치
- *hello.js* 실행
- *Security Group*에 8000포트 추가
- 인스턴스 정지 → 종료

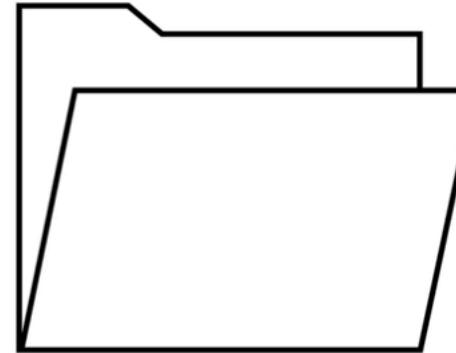
## AWS S3

- Simple Storage Service
- Amazon S3 Standard, S3 Standard-IA 및 S3 Glacier 스토리지 클래스는 SUCCESS 메시지를 반환하기 전에 하나의 Amazon S3 리전 내 **최소 3개의 AZ(Availability Zone, 가용 영역)**에 걸쳐 여러 디바이스에 객체를 **중복 저장**
- AZ(Availability Zone, 가용 영역)
  - 리전 내에 격리된 위치, 개별 데이터 센터
    - 1개의 데이터 센터에는 일반적으로 5만~8만대의 물리적 서버
  - 물리적으로는 떨어져 있지만, 논리적으로 연결
  - 1개의 리전당 2개 이상의 AZ (us-east-1는 5개)

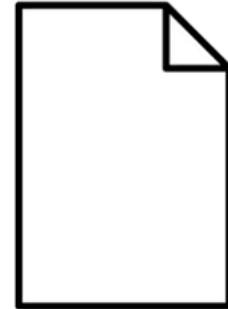




**Bucket**



**Folder**



**Object**



## 버킷 만들기

×

① 이름 및 지역      ② 옵션 구성      ③ 권한 설정      ④ 검토

이름 및 지역

버킷 이름 i

dowon-bucket

리전

미국 동부(버지니아 북부) ▼

---

기존 버킷에서 설정 복사

버킷을 선택합니다(선택 사항). 1버킷 ▼

# 버킷 만들기

■ 1 이름 및 지역   ■ 2 옵션 구성   ■ 3 권한 설정   ■ 4 검토

참고: 버킷 생성 후에는 특정 사용자에게 액세스 권한을 부여할 수 있습니다.

## 퍼블릭 액세스 차단(버킷 설정)

퍼블릭 액세스는 ACL(액세스 제어 목록), 버킷 정책, 액세스 지점 정책 또는 모두를 통해 버킷 및 객체에 부여됩니다. 모든 S3 버킷 및 객체에 대한 퍼블릭 액세스가 차단되었는지 확인하려면 [모든 퍼블릭 액세스 차단]을 활성화합니다. 이 설정은 이 버킷 및 해당 액세스 지점에만 적용됩니다. AWS에서는 [모든 퍼블릭 액세스 차단]을 활성화하도록 권장하지만, 이 설정을 적용하기 전에 퍼블릭 액세스가 없어도 애플리케이션이 올바르게 작동하는지 확인합니다. 버킷 또는 내부 객체에 어느 정도 수준의 퍼블릭 액세스가 필요한 경우 특정 스토리지 사용 사례에 맞게 아래 개별 설정을 사용자 지정할 수 있습니다. [세부 정보 ↗](#)

**모든 퍼블릭 액세스 차단**

이 설정을 활성화하면 아래 4개의 설정을 모두 활성화한 것과 같습니다. 다음 설정 각각은 서로 독립적입니다.

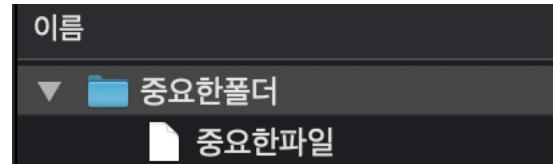
- 새 ACL(액세스 제어 목록)을 통해 부여된 버킷 및 객체에 대한 퍼블릭 액세스 차단

S3은 새로 추가된 버킷 또는 객체에 적용되는 퍼블릭 액세스 권한을 차단하며, 기존 버킷 및 객체에 대한 새 퍼블릭 액세스 ACL 생성을 금지합니다. 이 설정은 ACL을 사용하여 S3 리소스에 대한 퍼블릭 액세스를 허용하는 기존 권한을 변경하지 않습니다.
- 임의의 ACL(액세스 제어 목록)을 통해 부여된 버킷 및 객체에 대한 퍼블릭 액세스 차단

S3은 버킷 및 객체에 대한 퍼블릭 액세스를 부여하는 모든 ACL을 무시합니다.
- 새 퍼블릭 버킷 또는 액세스 지점 정책을 통해 부여된 버킷 및 객체에 대한 퍼블릭 액세스 차단

S3는 버킷 및 객체에 대한 퍼블릭 액세스를 부여하는 새 버킷 및 액세스 지점 정책을 차단합니다. 이 설정은 S3 리소스에 대한 퍼블릭 액세스를 허용하는 기존 정책을 변경하지 않습니다.
- 임의의 퍼블릭 버킷 또는 액세스 지점 정책을 통해 부여된 버킷 및 객체에 대한 퍼블릭 및 교차 계정 액세스 차단

S3는 버킷 및 객체에 대한 퍼블릭 액세스를 부여하는 정책을 사용하는 버킷 또는 액세스 지점에 대한 퍼블릭 및 교차 계정 액세스를 무시합니다.



업로드

파일 선택 권한 설정 속성 설정 검토

1 파일 크기: 26.0 B 대상 경로: dowon-bucket-new

### 스토리지 클래스

사용 사례 및 액세스 요구 사항을 기준으로 스토리지 클래스를 선택합니다. 자세히 알아보기 또는 다음 참조: Amazon S3 요금

스토리지 클래스	설계 대상	가용 영역	최소 스토리지 기간	청구 가능 체크	최소 객	모니터링 및 자동화	검색 요금
<input type="radio"/> 스탠다드	자주 액세스하는 데이터	≥ 3	-	-	-	-	-
<input checked="" type="radio"/> 인텔리전트-티어링:	변경 또는 알 수 없는 액세스 패턴으로 수명이 긴 데이터	≥ 3	30일	-	객체당 요금 적용	-	-
<input checked="" type="radio"/> 스탠다드-IA	수명이 길고 자주 액세스하지 않는 데이터	≥ 3	30일	128KB	-	GB당 요금 적용	-
<input checked="" type="radio"/> 단일 영역-IA	수명이 길고 자주 액세스하지 않으며 중요하지 않은 데이터	≥ 1	30일	128KB	-	GB당 요금 적용	-
<input checked="" type="radio"/> Glacier	검색 시간이 몇 분에서 몇 시간까지 걸리는 아카이브 데이터	≥ 3	90일	40KB	-	GB당 요금 적용	-
<input checked="" type="radio"/> Glacier Deep Archive	검색 시간이 몇 시간 걸리는 드물게 액세스되는 아카이브 데이터	≥ 3	180일	40KB	-	GB당 요금 적용	-

업로드 이전 다음

## 중요한파일

다운로드 경로 복사 Select 소스

최신 버전 ▾

개요	키	중요한파일
	크기	26.0 B
	만료 날짜	N/A
	만료 규칙	N/A
	ETag	ba84b0bfd353c28e260709d8261b0314
	마지막 수정	8월 5, 2020 12:00:01 오전 GMT+0900
객체 URL	<a href="https://dowon-bucket-new.s3-ap-northeast-1.amazonaws.com/%E1%84%8C%E1%85%AE%E1%86%BC%E1%84%8B%E1%85%AD%E1%84%92%E1%85%A1%E1%86%AB%E1%84%91%E1%85%A9%E1%86%AF%E1%84%83%E1%85%A5%EC%A4%91%EC%9A%94%ED%95%9C%ED%8C%8C%EC%9D%BC">https://dowon-bucket-new.s3-ap-northeast-1.amazonaws.com/%E1%84%8C%E1%85%AE%E1%86%BC%E1%84%8B%E1%85%AD%E1%84%92%E1%85%A1%E1%86%AB%E1%84%91%E1%85%A9%E1%86%AF%E1%84%83%E1%85%A5%EC%A4%91%EC%9A%94%ED%95%9C%ED%8C%8C%EC%9D%BC</a>	

down-bucket-new.s3-ap-northeast-1.amazonaws.com/중요한파일/중요한파일

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<Error>
  <Code>AccessDenied</Code>
  <Message>Access Denied</Message>
  <RequestId>EFB0D76F39513D1D</RequestId>
  <HostId>TYs2FCC7NZy11bbxAauCd+5i7FuII4Hlp4KGUDHGAbSiHyvqXtph7b/fIE8MGHFHxc0tIo8u0oA=</HostId>
</Error>
```



## dowon-bucket-new

액세스가 거부되었습니다

개요 속성 권한 관리 액세스 지점

퍼블릭 액세스 차단

액세스 제어 목록

버킷 정책

## 퍼블릭 액세스 차단(버킷 설정)

퍼블릭 액세스는 ACL(액세스 제어 목록), 버킷 정책, 액세스 지점 정책 또는 모두를 든 퍼블릭 액세스 차단]을 활성화합니다. 이 설정은 이 버킷 및 해당 액세스 지점에 퍼블릭 액세스가 없어도 애플리케이션이 올바르게 작동하는지 확인합니다. 버킷 또는 사용자 지정할 수 있습니다. 세부 정보 ↗

## 모든 퍼블릭 액세스 차단

활성화

- 새 ACL(액세스 제어 목록)을 통해 부여된 버킷 및 객체에 대한 퍼블릭 액세스 차단 활성화
- 임의의 ACL(액세스 제어 목록)을 통해 부여된 버킷 및 객체에 대한 퍼블릭 액세스 차단 활성화
- 새 퍼블릭 버킷 또는 액세스 지점 정책을 통해 부여된 버킷 및 객체에 대한 퍼블릭 액세스 차단 활성화
- 임의의 퍼블릭 버킷 또는 액세스 지점 정책을 통해 부여된 버킷 및 객체에 대한 퍼블릭 및 교차 계정 액세스 차단 활성화

## 다른 AWS 계정에 대한 액세스

+ 계정 추가

삭제

정식 ID ⓘ

객체 읽기 ⓘ

## 퍼블릭 액세스

그룹 ⓘ

객체 읽기 ⓘ

 Everyone

예

중요한파일 최신 버전 ▾

개요

속성

권한

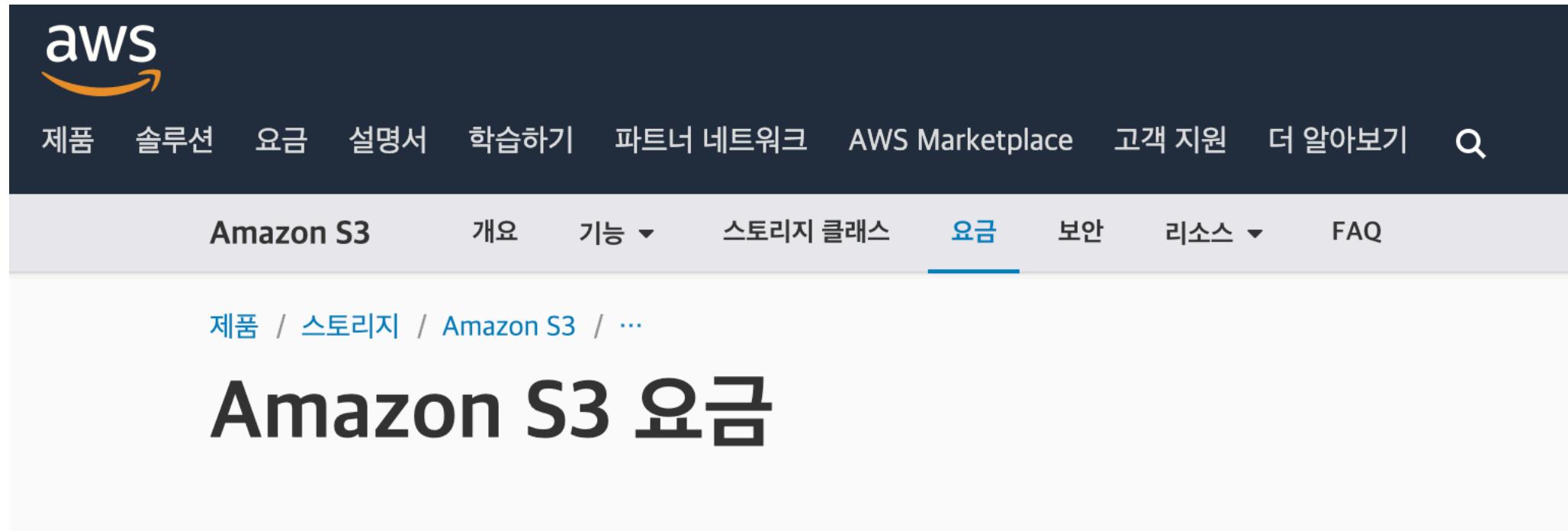
Select 소스

## 스토리지 클래스



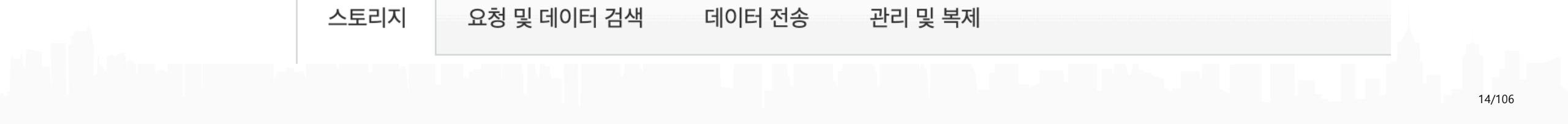
사용 사례 및 액세스 요구 사항을 기준으로 스토리지 클래스를 선택합니다. [자세히 알아보기](#) 또는 다음 참조: [Amazon S3 요금](#)

스토리지 클래스	설계 대상	가용 영역	최소 스토리지 기간	청구 가능한 체크 크기	최소 객	모니터링 및 자동화	검색 요금
<input checked="" type="radio"/> 스탠다드	자주 액세스하는 데이터	≥ 3	-	-	-	-	-
<input type="radio"/> 인텔리전트-티어링:	변경 또는 알 수 없는 액세스 패턴으로 수명이 긴 데이터	≥ 3	30일	-	객체당 요금 적용	-	-
<input type="radio"/> 스탠다드-IA	수명이 길고 자주 액세스하지 않는 데이터	≥ 3	30일	128KB	-	GB당 요금 적용	-
<input type="radio"/> 단일 영역-IA	수명이 길고 자주 액세스하지 않으며 중요하지 않은 데이터	≥ 1	30일	128KB	-	GB당 요금 적용	-
<input type="radio"/> Glacier	검색 시간이 몇 분에서 몇 시간까지 걸리는 아카이브 데이터	≥ 3	90일	40KB	-	GB당 요금 적용	-
<input type="radio"/> Glacier Deep Archive	검색 시간이 몇 시간 걸리는 드물게 액세스되는 아카이브 데이터	≥ 3	180일	40KB	-	GB당 요금 적용	-
<input type="radio"/> 중복 감소(권장 안 함)	자주 액세스하는 중요하지 않은 데이터	≥ 3	-	-	-	-	-



The screenshot shows the AWS S3 Pricing page. At the top, there's a navigation bar with links for 제품 (Products), 솔루션 (Solutions), 요금 (Pricing), 설명서 (Documentation), 학습하기 (Learn), 파트너 네트워크 (Partners), AWS Marketplace, 고객 지원 (Customer Support), 더 알아보기 (Learn More), and a search icon. Below the navigation bar, there's a secondary navigation bar with tabs for Amazon S3, 개요 (Overview), 기능 (Features), 스토리지 클래스 (Storage Classes), **요금** (Pricing) which is underlined, 보안 (Security), 리소스 (Resources), and FAQ. The main content area shows a breadcrumb trail: 제품 / 스토리지 / Amazon S3 / ... followed by a large title "Amazon S3 요금".

사용한 만큼만 비용을 지불합니다. 최소 요금이 없습니다. 데이터 프로필에 가장 적합한 S3 스토리지 클래스를 결정할 때는 스기능 요금이라는 네 가지 비용 요소를 고려해야 합니다.

A faint, stylized city skyline graphic is visible at the bottom of the slide.

요청 및 데이터 검색

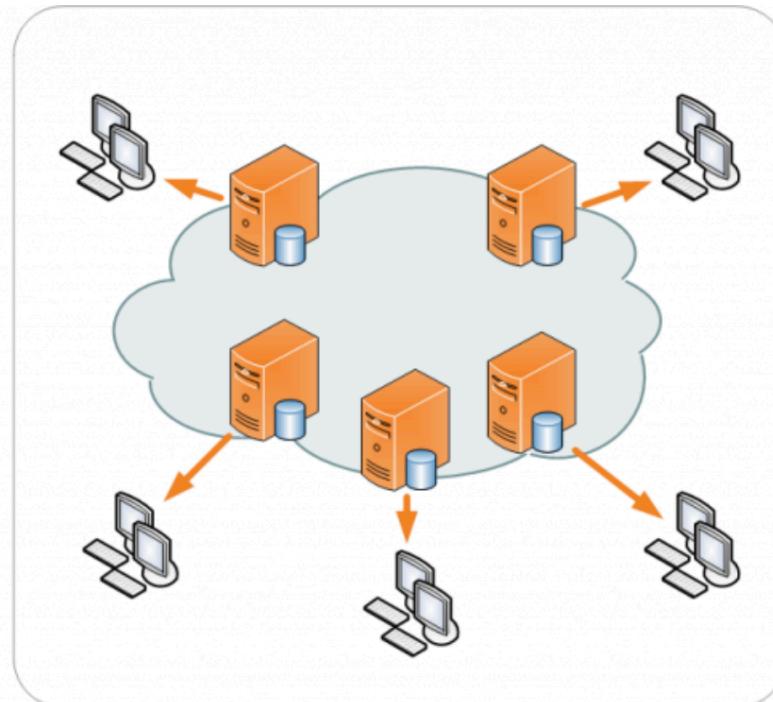
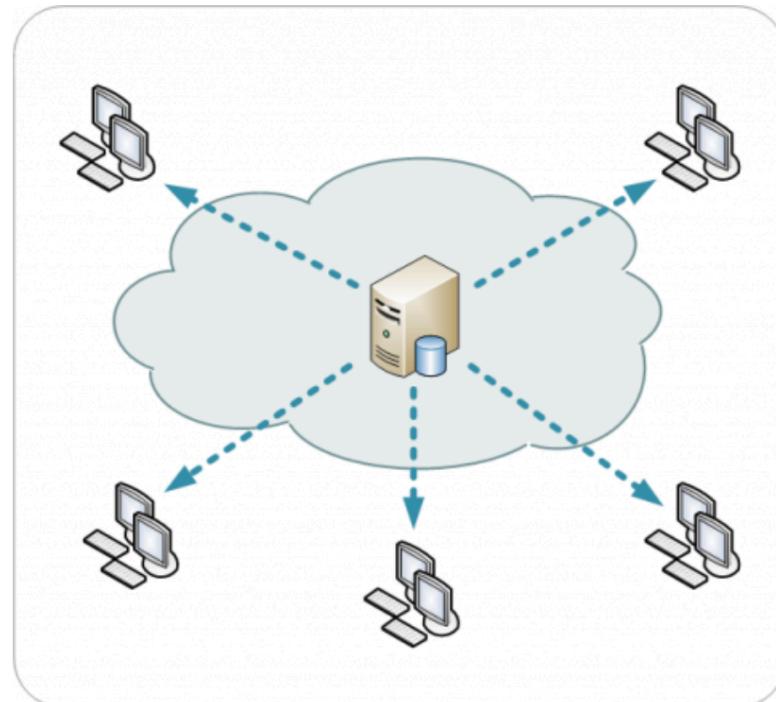
데이터 전송

관리 및 복제

- IAM에서 Role 생성
  - IAM Role → AWS Service → EC2
    - AmazonS3FullAccess 선택
- EC2 생성 시 IAM role 지정
  - EC2 생성 후
    - **(EC2) \$ aws s3 ls s3://dowon\_bucket**  
→ IAM Role을 등록하지 않았으면, 액세스 키와 시크릿 키를 등록해야만 S3에 접근 가능  
→ **(EC2) \$ aws configure**  
→ AWS Access Key ID, AWS Secret Access Key, Default region name, Default output format
    - **(EC2) \$ aws s3 cp ./test.txt s3://edu24-edowon-mybucket**
    - **(EC2) \$ aws s3 sync s3://edu24-edowon-mybucket/index.html ./index.html**

## AWS CloudFront

- Cache
  - Cache Server
  - 24 hours
- CDN



## AWS CloudFront

### Ec2에 nodejs 설치

```
const express = require('express');
const app = express();
const port = process.env.PORT || 8000;

app.get('/', (req, res) => {
  setTimeout(function() {
    res.send('Hello, World! : ' + new Date());
    return console.log(new Date());
  }, 5000);
  //res.send('Hello, World! : ' + new Date())
});

app.listen(port, () => console.log(`Example app listening on port ${port}!`));
```

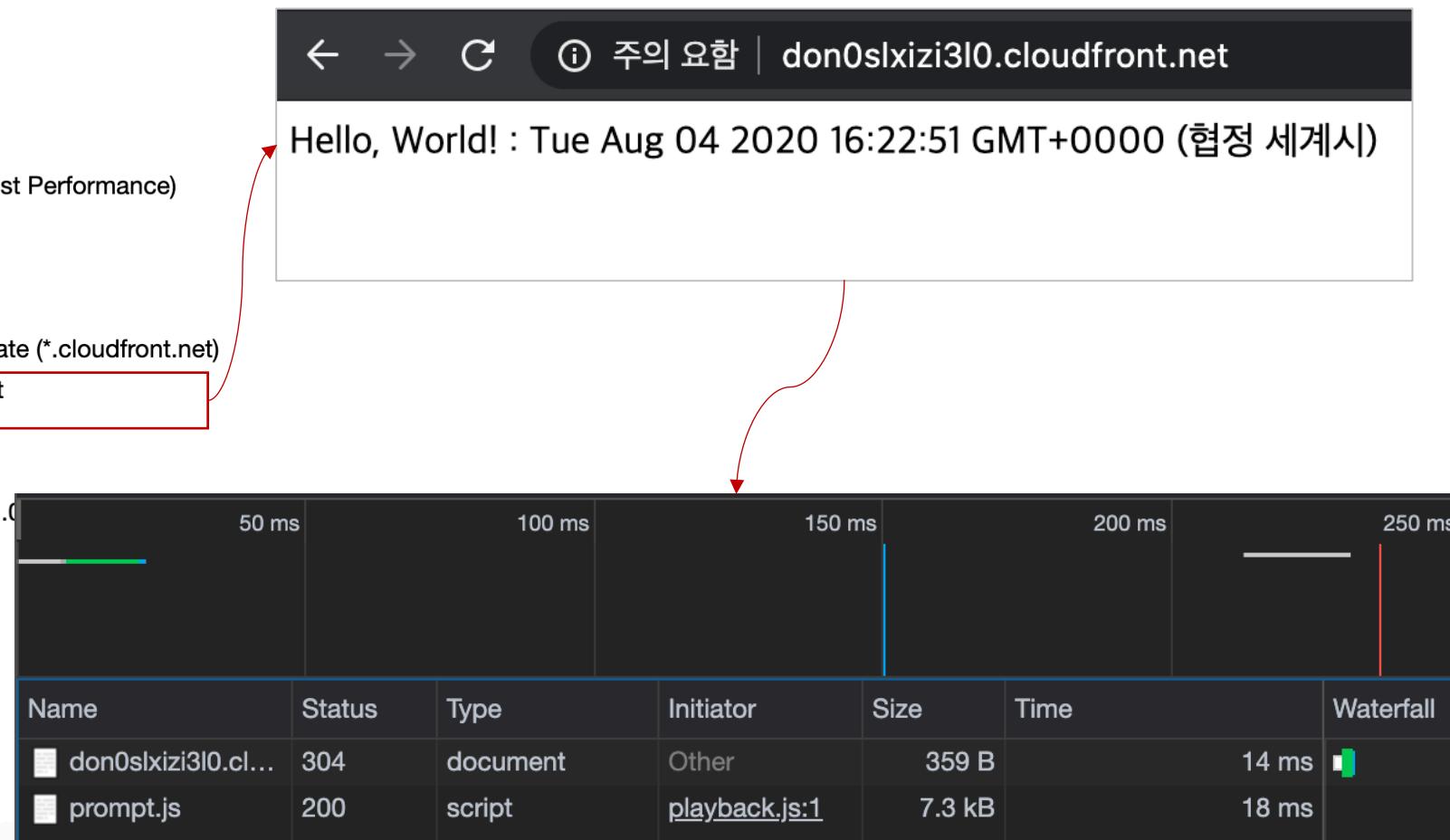
```
{
  "name": "ec2-user",
  "version": "1.0.0",
  "description": "",
  "main": "app.js",
  "scripts": {
    "start": "node app.js",
    "test": "echo \\\"Error: no test specified\\\" && exit 1"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "express": "^4.17.1"
  }
}
```

# AWS CloudFront



## CloudFront에 접속

<b>Distribution ID</b>	E2Z9FHNKETJE4J
<b>ARN</b>	arn:aws:cloudfront::735824579779:distribution/E2Z9FHNKETJE4J
<b>Log Prefix</b>	-
<b>Delivery Method</b>	Web
<b>Cookie Logging</b>	Off
<b>Distribution Status</b>	Deployed
<b>Comment</b>	-
<b>Price Class</b>	Use All Edge Locations (Best Performance)
<b>AWS WAF Web ACL</b>	-
<b>State</b>	Enabled
<b>Alternate Domain Names (CNAMEs)</b>	-
<b>SSL Certificate</b>	Default CloudFront Certificate (*.cloudfront.net)
<b>Domain Name</b>	don0slxizi3l0.cloudfront.net
<b>Custom SSL Client Support</b>	-
<b>Security Policy</b>	TLSv1
<b>Supported HTTP Versions</b>	HTTP/2, HTTP/1.1, HTTP/1.0
<b>IPv6</b>	Enabled
<b>Default Root Object</b>	-
<b>Last Modified</b>	2020-08-05 01:16 UTC+9
<b>Log Bucket</b>	-



- AWS CloudFront 

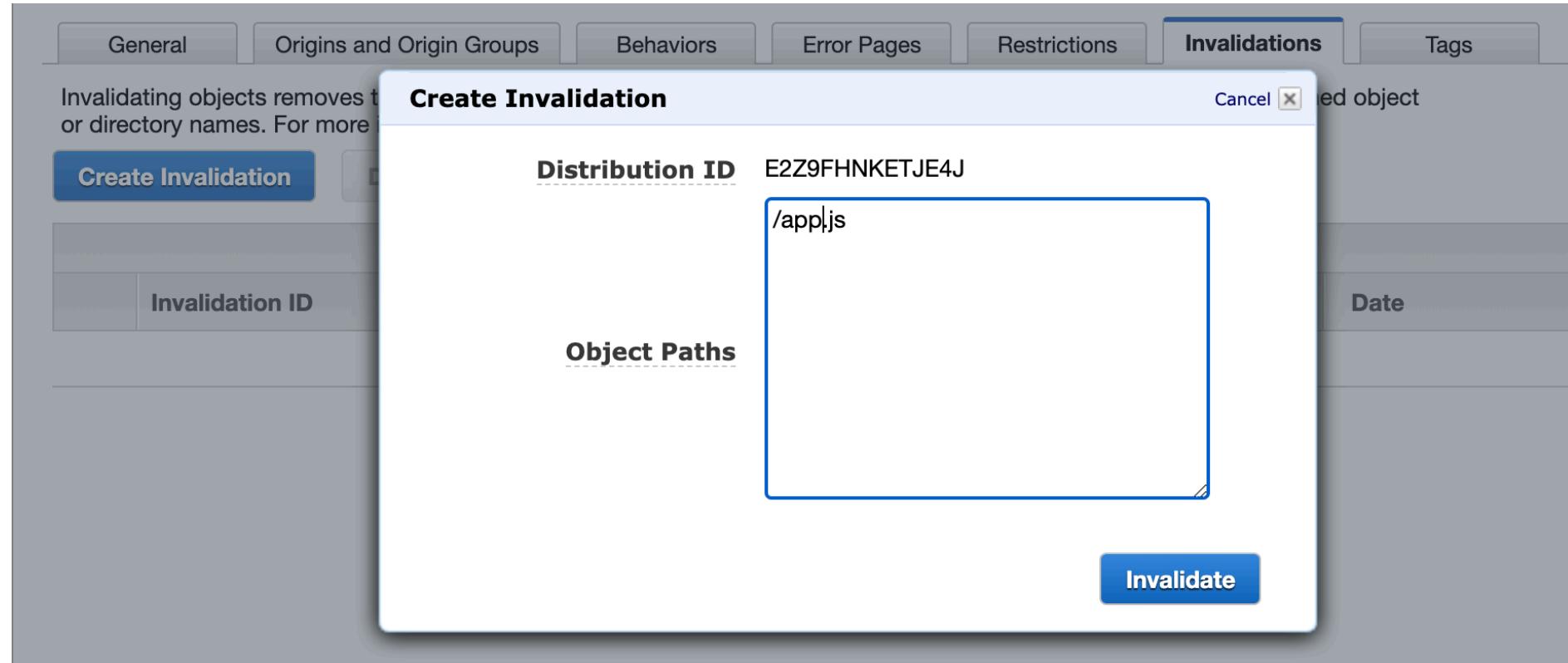
- CloudFront에 접속 → cache 사용

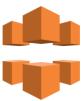
```
app.get('/', (req, res) => {
  var ip = req.headers['x-forwarded-for'] ||
    req.connection.remoteAddress ||
    req.socket.remoteAddress ||
    (req.connection.socket ? req.connection.socket.remoteAddress : null);

  console.log(ip);
```

## AWS CloudFront

- CloudFront → cache 무효화



- AWS CloudFront 

- Content Delivery Network



## Edit Distribution

### Distribution Settings

**Price Class**

Use All Edge Locations (Best Performance) 

**AWS WAF Web ACL**

None 

**Alternate Domain Names  
(CNAMEs)****SSL Certificate**

Default CloudFront Certificate (\*.cloudfront.net)

# AWS CloudFront



## ■ Edge Locations

Amazon CloudFront	엣지 로케이션 포함 리전	미국 및 캐나다	유럽 및 이스라엘	남아프리카, 캐나 및 중동	남아메리카	일본	호주	싱가포르, 대한민국, 대만, 홍콩 및 필리핀	인도
	요금 계층 전체	예	예	예	예	예	예	예	예
페이지 콘텐츠	요금 계층 200	예	예	예	x	예	x	예	예
Amazon CloudFront 인프라 보안	요금 계층 100	예	예	x	x	x	x	x	x
최종 사용자 캐시(11개)의 글로벌 네트워크를 사용하고 있습니다. Amazon CloudFront 엣지 로케이션의 위치는 다음과 같습니다.									
가용성									
성능									
프로그래밍 가능성과 DevOps 지원성									
Lambda@Edge									
비용 효율성									
새로운 소식									

● Edge Locations  
● Multiple Edge Locations  
○ Regional Edge Caches

22/106

# AWS CloudFront



## Dotcom-tools

**dotcom-tools**

Pro Solutions    Free Tools    About Dotcom-Monitor    Blog    [LOGIN](#)    [FREE TRIAL](#)

**Website Speed Test** X

URL: [http://ec2-54-238-250-143.ap-northeast-1.co...](http://ec2-54-238-250-143.ap-northeast-1.amazonaws.com/)

First Visit     Repeat Visit



**Denver**

AGENT / LOCATION	FIRST VISIT	REPEAT VISIT	LIGHTHOUSE
<a href="#">Denver</a>	 1.4 sec	 1.4 sec	 100
<a href="#">Dallas</a>	 1.4 sec	 1.4 sec	 100
<a href="#">Amsterdam</a>	 1.7 sec	 1.7 sec	 100
<a href="#">Tokyo</a>	 1.1 sec	 1.1 sec	 100
<a href="#">Mumbai</a>	 1.6 sec	 1.6 sec	 100
<a href="#">Warsaw</a>	 1.6 sec	 1.6 sec	 100

# AWS CloudFront



## Dotcom-tools

**dotcom-tools** [Pro Solutions](#) [Free Tools](#) [About Dotcom-Monitor](#) [Blog](#) [LOGIN](#) [FREE TRIAL](#)

### Website Speed Test

URL: <http://don0slxizi3l0.cloudfront.net/>

First Visit Repeat Visit

 Denver

CHECKS COMPLETE  
**6 of 6 Locations**

ERRORS FROM  
**0 Locations**

AVG: 1<sup>st</sup> VISIT  
**904.2 ms**

AVG: 2<sup>nd</sup> VISIT  
**158.2 ms**

AGENT / LOCATION	FIRST VISIT	REPEAT VISIT	LIGHTHOUSE
<a href="#">Denver</a>	 1.4 sec	 161.0 ms	 100
<a href="#">Dallas</a>	 1.3 sec	 148.0 ms	 100
<a href="#">Amsterdam</a>	 1.6 sec	 133.0 ms	 99
<a href="#">Tokyo</a>	 128.0 ms	 134.0 ms	 100
<a href="#">Mumbai</a>	 795.0 ms	 216.0 ms	 99
<a href="#">Warsaw</a>	 148.0 ms	 157.0 ms	 100

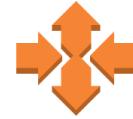
## ■ AWS RDS



## ■ Relational Database Service

- Database 생성
- Public으로 접속 (Public 접속 허용, SG에 3306 추가)
- Private으로 접속, (Only 같은 VPC의 EC2를 통해서만 접속)
  - SG에 해당 EC2의 SG 등록해야 함
- 모니터링
- 백업과 복원
- 가격

## AWS Auto Scaling



### Auto Scaling

- 같은 사양, 같은 환경, 같은 코드를 가지고 있는 똑같은 EC2 인스턴스들의 묶음
- 자동으로 생성할 EC2 인스턴스 선택
  - **Snapshot → AMI**

The screenshot shows the AWS CloudWatch Metrics console interface. At the top, there are tabs: '인스턴스 시작' (selected), '연결', and '작업'. Below the tabs is a search bar with placeholder text '태그 및 속성별 필터 또는 키워드별 검색'. The main area displays a table of EC2 instances:

Name	인스턴스 ID	인스턴스 유형	가용 영역	인스턴스 상태	상태 검사
downon	i-008d47e21d94647...	t2.micro	us-east-1c	terminated	
ec2-nodejs	i-03b1ea1d6...	t2.micro	us-east-1e	stopped	

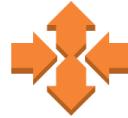
A context menu is open over the 'ec2-nodejs' instance, listing the following options:

- Windows 암호 가져오기
- Create Template From Instance
- 기준 인스턴스를 기반으로 시작
- 인스턴스 상태
- 인스턴스 설정
- 이미지
- 네트워킹
- CloudWatch 모니터링

On the right side of the menu, there are two additional options:

- 이미지 생성
- 번들 인스턴스(인스턴스 스토어 AMI)

## AWS Auto Scaling



### 이미지 생성

이미지 생성 ×

인스턴스 ID: i-03b1ea1d6d850bcd2

이미지 이름: exercise-image

이미지 설명:

재부팅 안 함:

인스턴스 볼륨

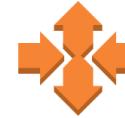
볼륨 유형	디바이스	스냅샷	크기(GiB)	볼륨 유형	IOPS	처리량(MB/초)	종료 시 삭제	암호화됨
루트	/dev/xvda	snap-0ed3f0f331ab4cbc7	8	범용 SSD(gp2)	100/3000	해당 사항 없음	<input checked="" type="checkbox"/>	암호화되지 않음

[새 볼륨 추가](#)

EBS 볼륨의 전체 크기: 8 GiB  
EBS 이미지를 생성할 때 위의 각 볼륨에 대해 EBS 스냅샷이 생성됩니다.

[취소](#) [이미지 생성](#)

# AWS Auto Scaling



## 이미지 생성

시작하기 EC2 Image Builder 작업 ▾

사용자 소유 태그 및 속성별 필터 또는 키워드별 검색

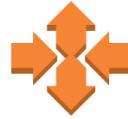
	Name	AMI 이름	AMI ID	소스	소유자	표시 여부	상태
	exercise-image	ami-009a4ba4e4cb74899	735824579779/...	735824579779	프라이빗		pending

시작하기 EC2 Image Builder 작업 ▾

사용자 소유 태그 및 속성별 필터 또는 키워드별 검색

	Name	AMI 이름	AMI ID	소스	소유자	표시 여부	상태
	exercise-image	ami-009a4ba4e4cb74899	735824579779/...	735824579779	프라이빗		available

# AWS Auto Scaling



## Instance

### Launch Template > Template 생성

aws 서비스 리소스 그룹

EC2 > 시작 템플릿 > 시작 템플릿 생성

#### 시작 템플릿 생성

시작 템플릿을 생성하면 저장된 인스턴스 구성을 만들어 두었다가 나중에 이를 재사용하고, 공유하고, 시작할 수 있습니다. 여러 버전의 템플릿을 저장할 수 있습니다.

##### 시작 템플릿 이름 및 설명

시작 템플릿 이름 - 필수

이 계정에 대해 고유해야 합니다. 최대 128자입니다. '&', '\*', '@' 등의 특수 문자나 공백은 사용할 수 없습니다.

템플릿 버전 설명

최대 255자

Auto Scaling 지원 정보  
EC2 Auto Scaling에 이 템플릿을 사용하려면 이 항목을 선택합니다.  
 EC2 Auto Scaling에 사용할 수 있는 템플릿을 설정하는 데 도움이 되는 지침 제공

▶ 템플릿 태그  
▶ 원본 템플릿

#### 시작 템플릿 콘텐츠

아래에서 시작 템플릿의 세부 정보를 지정합니다. 비워 둔 필드는 시작 템플릿에 포함되지 않습니다.

##### Amazon Machine Image(AMI) 정보

AMI  
**exercise-image**  
ami-009a4ba4e4cb74899  
카탈로그: 내 AMI 아키텍처: 64비트(x86) 가상화: hvm

##### 인스턴스 유형 정보

인스턴스 유형  
**t2.micro**  
프리 티어 사용 가능  
페일리: General purpose 1 vCPU 1 GiB 메모리  
온디맨드 Linux 요금: 0.0116 USD 시간당 온디맨드 Windows 요금: 0.0162 USD 시간당

##### 키 페어(로그인) 정보

키 페어 이름

새 키 페어 생성

##### 네트워크 설정

###### 네트워킹 플랫폼 정보

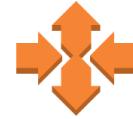
Virtual Private Cloud(VPC)  
AWS 클라우드 내에서 논리적으로 격리된 자체 영역에 있는 가상 네트워크로 시작

EC2-Classic  
다른 고객과 공유하는 단일 플랫 네트워크에서 시작

###### 보안 그룹 정보

보안 그룹 선택  
  
VPC: vpc-3f62d545

# AWS Auto Scaling



## Auto Scaling Group

Screenshot of the AWS Auto Scaling Groups creation wizard.

The navigation bar shows: aws, 서비스 ▾, 리소스 그룹 ▾, and a bell icon.

The sidebar menu includes:

- New EC2 Experience (Tell us what you think)
- EC2 Dashboard (새로 만들기)
- Events (새로 만들기)
- Tags
- Limits
- INSTANCES
- IMAGES
- ELASTIC BLOCK STORE
- NETWORK & SECURITY
- LOAD BALANCING
- AUTO SCALING
  - Launch Configurations (새로 만들기)
  - Auto Scaling Groups (새로 만들기)

The main content area shows the steps for creating an Auto Scaling group:

- 1단계: 시작 템플릿 또는 구성 선택
- 2단계: 설정 구성
- 3단계 (선택 사항): 고급 옵션 구성
- 4단계 (선택 사항): 그룹 크기 및 조정 정책 구성
- 5단계 (선택 사항): 알림 추가
- 6단계 (선택 사항): 태그 추가
- 7단계: 검토

**시작 템플릿 또는 구성 선택** (Info)

이 Auto Scaling 그룹에서 시작하는 모든 EC2 인스턴스에 공통된 설정이 포함된 시작 템플릿을 지정합니다. 현재 시작 구성을 사용하는 경우 시작 템플릿으로 마이그레이션할 수 있습니다.

**이름**

Auto Scaling 그룹 이름  
그룹을 식별할 이름을 입력합니다.  
[Input field]

현재 리전에서 이 계정에 대해 고유해야 하며 255자를 넘지 않아야 합니다.

**시작 템플릿** (Info) 시작 구성으로 전환

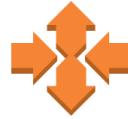
시작 템플릿  
Amazon Machine Image(AMI), 인스턴스 유형, 키 페어 및 보안 그룹과 같은 인스턴스 수준 설정이 포함된 시작 템플릿을 선택합니다.

시작 템플릿 선택 ▼ C

시작 템플릿 생성 [Create button]

취소 다음

# AWS Auto Scaling



## Auto Scaling Group

**설정 구성** [Info](#)

아래 설정을 구성합니다. 시작 템플릿을 선택했는지에 따라 이러한 설정에는 EC2 리소스를 최적화하는 데 유용한 옵션이 포함될 수 있습니다.

**구매 옵션 및 인스턴스 유형** [Info](#)

시작 템플릿 준수  
시작 템플릿에 따라 구매 옵션(온디맨드 또는 스팟)과 인스턴스 유형이 결정됩니다.

구매 옵션과 인스턴스 유형 조합  
시작할 온디맨드 및 스팟 용량과 선택적으로 여러 인스턴스 유형을 지정합니다. 이 옵션은 인스턴스 풀의 규모와 비용을 최적화하는데 매우 유용합니다.

**네트워크** [Info](#)

대부분의 애플리케이션에서는 여러 가용 영역을 사용할 수 있으며 EC2 Auto Scaling이 여러 영역 간에 인스턴스를 균일하게 분산할 수 있습니다. 기본 VPC와 기본 서브넷은 빠르게 시작하는 데 적합합니다.

VPC

vpc-3f62d545  
172.31.0.0/16 Default

VPC 생성 [\[+\]](#)

서브넷

서브넷 선택 [\[▼\]](#) [\[C\]](#)

us-east-1a | subnet-33deb81d X  
172.31.80.0/20 Default

us-east-1b | subnet-20765c6a X  
172.31.16.0/20 Default

us-east-1c | subnet-ecfd99b0 X  
172.31.32.0/20 Default

서브넷 생성 [\[+\]](#)

취소 [이전](#) [검토로 이동](#) [다음](#)

### 그룹 크기 및 조정 정책 구성

Auto Scaling 그룹의 원하는 용량, 최소 용량 및 최대 용량을 설정합니다. 선택적으로 조정 정책을 추가하여 그룹의 인스턴스 수를 동적으로 조정할 수 있습니다.

#### 그룹 크기 - 선택 사항

원하는 용량을 변경하여 Auto Scaling 그룹의 크기를 지정합니다. 최소 및 최대 용량 한도를 지정할 수도 있습니다. 원하는 용량은 한도 범위 내에 있어야 합니다.

원하는 용량

1

최소 용량

1

최대 용량

2

#### 조정 정책 - 선택 사항

조정 정책을 사용하여 수요의 변화를 충족하도록 Auto Scaling 그룹의 크기를 동적으로 조정할지 여부를 선택합니다. [Info](#)

대상 추적 조정 정책

원하는 결과를 선택하고 조정 정책에 따라 필요한 경우 용량을 추가 및 제거하여 해당 결과를 달성합니다.

없음

조정 정책 이름

Target Tracking Policy

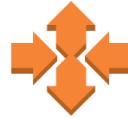
지표 유형

평균 CPU 사용률

대상 값

50

# AWS Auto Scaling



## Auto Scaling Group

### 태그 추가 Info

태그를 추가하면 AWS에서 Auto Scaling 그룹의 검색, 필터링 및 추적에 도움이 됩니다. 인스턴스가 시작될 때 이러한 태그를 인스턴스에 자동으로 추가하도록 선택할 수도 있습니다.

시작 템플릿에 태그를 지정하여 인스턴스(및 연결된 EBS 볼륨)에 태그를 추가할 수도 있습니다. 하지만 Auto Scaling 그룹에 대해 중복된 키가 지정되어 있는 경우 시작 템플릿의 인스턴스 태그 값이 재정의되므로 주의해야 합니다. X

#### 태그(1)

키	값 - 선택 사항	새 인스턴스에 태그 지정
Namw	exercise-grp	<input checked="" type="checkbox"/>

[태그 추가](#)

49 남은 수

취소

이전

다음

### 검토 Info

#### 1단계: 시작 템플릿 또는 구성 선택

##### 그룹 세부 정보

Auto Scaling 그룹 이름  
EXERCISE\_GROUP

##### 시작 템플릿

시작 템플릿  
[exercise-launch-template](#) 버전  
lt-0171d27ae3721dfa3

Default

설명  
initial version

#### 6단계: 태그 추가

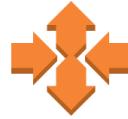
##### 태그 (1)

키	값	새 인스턴스에 태그 지정
Namw	exercise-grp	예

취소

Auto Scaling 그룹 생성

# AWS Auto Scaling



✓ EXERCISE\_GROUP, 1 조정 정책을(를) 생성함 X

EC2 > Auto Scaling 그룹

### Auto Scaling 그룹 (1/1)

이름	시작 템플릿/구성	인스턴스	상태	원하는 용량	최소	최대	가용 영역
EXERCISE_GROUP	exercise-launch-template   버전 기본값	1	-	1	1	2	us-east-1a, us-east-1b, us-e...

세부 정보 활동 자동 조정 인스턴스 관리 모니터링 인스턴스 새로 고침

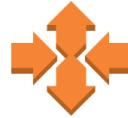
#### 그룹 세부 정보

최대 용량	Amazon 리소스 이름(ARN)
2	arn:aws:autoscaling:us-east-1:735824579779:autoScalingGroup:48f6dcaa-df96-4e42-9dba-e8d6829f0589:autoScalingGroupName/EXERCISE_GROUP

#### 시작 템플릿

시작 템플릿	AMI ID	인스턴스 유형
<a href="#">exercise-launch-template</a> lt-0171d27ae3721dfa3	ami-009a4ba4e4cb74899	t2.micro
버전	보안 그룹	보안 그룹 ID
Default	-	sg-01eeef8c02c64e3657
설명	키 페어 이름	스토리지(블룸)
initial version	dowon-key	

# AWS Auto Scaling



인스턴스 시작 ▾      연결      작업 ▾

태그 및 속성별 필터 또는 키워드별 검색

	Name	인스턴스 ID	인스턴스 유형	가용 영역	인스턴스 상태	상태 검사	경보 상태	퍼블릭 DNS(IPv4)	IPv4 퍼블릭 IP
<input type="checkbox"/>	dowon	i-008d47e21d94647...	t2.micro	us-east-1c	<span style="color:red;">●</span> terminated		없음		-
<input type="checkbox"/>	ec2-nodejs for ami	i-03b1ea1d6d850bcd2	t2.micro	us-east-1e	<span style="color:red;">●</span> stopped		없음		-
<input checked="" type="checkbox"/>		i-0e7f4fcddc23ce644	t2.micro	us-east-1c	<span style="color:green;">●</span> running	<span style="color:green;">✓</span> 2/2개 검사 ...	없음		ec2-18-234-121-82.co... 18.234.121.82

```
▶ ssh -i "dowon-key.pem" ec2-user@ec2-18-234-121-82.compute-1.amazonaws.com
The authenticity of host 'ec2-18-234-121-82.compute-1.amazonaws.com (18.234.121.82)'
ECDSA key fingerprint is SHA256:hZ6lyPZ1vA/QJ5/78sihw8Fd3xtPxYf9Xk5ttLS8PiM.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-18-234-121-82.compute-1.amazonaws.com,18.234.121.82'
Last login: Wed Aug  5 13:29:52 2020 from 175.192.16.174
```

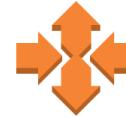
```
[ec2-user@ip-172-31-45-135 ~]$ ll
합계 28
drwxrwxr-x 52 ec2-user ec2-user 4096 8월  5 13:31 node_modules
-rw-rw-r--  1 ec2-user ec2-user 14284 8월  5 13:31 package-lock.json
-rw-rw-r--  1 ec2-user ec2-user  282  8월  5 13:31 package.json
[ec2-user@ip-172-31-45-135 ~]$ npm start
```

```
> ec2-user@1.0.0 start /home/ec2-user
> node app.js

Example app listening on port 8000!
::ffff:175.192.16.174
```

← → C ⓘ 주의 요함 | ec2-18-234-121-82.compute-1.amazonaws.com:8000  
Hello, World! : ::ffff:175.192.16.174,Wed Aug 05 2020 14:15:06 GMT+0000 (협정 세계시)

# AWS Auto Scaling



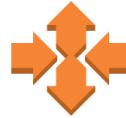
Auto Scaling 그룹 (1/1)

		C	편집	삭제	Auto Scaling 그룹 생성				
<input type="text"/> Auto Scaling 그룹 검색		<	1	>					
<input checked="" type="checkbox"/>	이름	시작 템플릿/구성	인스턴스	상태	원하는 용량	최소	최대	가용 영역	▼
<input checked="" type="checkbox"/>	EXERCISE_GROUP	exercise-launch-template   버전 기본값	1	-	1	1	2	us-east-1a, us-east-1b, us-e...	

세부 정보      활동      자동 조정      **인스턴스 관리**      모니터링      인스턴스 새로 고침

인스턴스 (1)

		C	작업						
<input type="text"/> 인스턴스 필터링		<	1	>					
<input type="checkbox"/>	인스턴스 ID	수명 주기	인스턴스 유형	가중치 기반 용량	시작 템플릿/구성	가용 영역	상태	다음으로부터 보호됨	▼
<input type="checkbox"/>	i-0e7f4fcddc23ce...	InService	t2.micro	-	exercise-launch-template	us-east-1c	Healthy		



## ■ Stress test

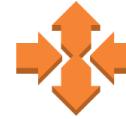
- \$ sudo apt-get install stress -y
- \$ stress --cpu 1 --timeout 600

```
ec2-user@ip-172-31-45-135:~ (ssh)
stress.x86_64 0:1.0.4-4.2.amzn1

Complete!
[ec2-user@ip-172-31-45-135 ~]$ clear
[ec2-user@ip-172-31-45-135 ~]$ stress --cpu 1 --timeout 600
stress: FAIL: [3105] (244) unrecognized option: --timeout
[ec2-user@ip-172-31-45-135 ~]$ stress --cpu 1 --timeout 6^C
[ec2-user@ip-172-31-45-135 ~]$
[ec2-user@ip-172-31-45-135 ~]$ clear
[ec2-user@ip-172-31-45-135 ~]$ stress --cpu 1 --timeout 600
stress: info: [3107] dispatching hogs: 1 cpu, 0 io, 0 vm, 0 hdd

ec2-user@ip-172-31-45-135:~ (ssh)
top - 14:26:26 up 16 min,  2 users,  load average: 0.83, 0.31, 0.11
Tasks: 80 total,   2 running, 54 sleeping,   0 stopped,   0 zombie
Cpu(s): 100.0%us,  0.0%sy,  0.0%ni  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem: 1009132K total, 219348K used, 789784K free, 13488k buffers
Swap: 0k total,       0k used,       0k free, 142596k cached
PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
3108 ec2-user  20   0  7320     92    0 R 99.9  0.0   1:43.91 stress
3424 ec2-user  20   0 15356  2144 1864 R  0.3  0.2   0:00.01 top
  1 root      20   0 19696  2580 2252 S  0.0  0.3   0:01.14 init
  2 root      20   0     0     0    0 S  0.0  0.0   0:00.00 kthreadd
  3 root      20   0     0     0    0 I  0.0  0.0   0:00.00 kworker/0:0
```

# AWS Auto Scaling



## ■ Stress test

세부 정보      활동      자동 조정      **인스턴스 관리**      모니터링      인스턴스 새로 고침

### 인스턴스 (2)

인스턴스 필터링

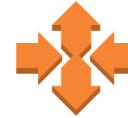
<input type="checkbox"/>	인스턴스 ID	수명 주기	인스턴스 유형	가중치 기반 용량	시작 템플릿/구성
<input type="checkbox"/>	i-05baf92594233...	InService	t2.micro	-	exercise-launch-template
<input type="checkbox"/>	i-0e7f4fcddc23ce...	InService	t2.micro	-	exercise-launch-template

### 작업 기록 (2)

활동 기록 필터링

상태	설명	원인	시작 시간	종료 시간
Successful	Launching a new EC2 instance: i-05baf92594233d49c	At 2020-08-05T14:30:05Z a monitor alarm TargetTracking-EXERCISE_GROUP-AlarmHigh-d96d6d1a-1862-4e46-a901-f3a6f939a972 in state ALARM triggered policy Target Tracking Policy changing the desired capacity from 1 to 2. At 2020-08-05T14:30:27Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 1 to 2.	2020 8월 05, 11:30:30 오후 +09:00	2020 8월 05, 11:36:02 오후 +09:00
Successful	Launching a new EC2 instance: i-0e7f4fcddc23ce644	At 2020-08-05T14:09:47Z a user request created an AutoScalingGroup changing the desired capacity from 0 to 1. At 2020-08-05T14:09:52Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 1.	2020 8월 05, 11:09:55 오후 +09:00	2020 8월 05, 11:10:27 오후 +09:00

# AWS Auto Scaling



**그룹 크기**

원하는 용량을 변경하여 Auto Scaling 그룹의 크기를 지정합니다. 최소 및 최대 용량 한도를 지정할 수도 있습니다. 원하는 용량은 한도 범위 내에 있어야 합니다.

원하는 용량  
1

최소 용량  
1

최대 용량  
1

취소 업데이트

인스턴스 시작 ▾ 연결 작업 ▾

태그 및 속성별 필터 또는 키워드별 검색

Name	인스턴스 ID	인스턴스 유형	가용 영역	인스턴스 상태	상태
ec2-nodejs for ami	i-03b1ea1d6d850bcd2	t2.micro	us-east-1e	stopped	stopped
	i-05baf92594233d49c	t2.micro	us-east-1a	running	running
	i-0e7f4fcddc23ce644	t2.micro	us-east-1c	shutting-down	shutting-down

작업 기록 (3)

활동 기록 필터링

상태	설명	원인	시작 시간	종료 시간
InProgress	Terminating EC2 instance: i-0e7f4fcddc23ce644	At 2020-08-05T14:45:08Z a user request update of AutoScalingGroup constraints to min: 1, max: 2, desired: 1 changing the desired capacity from 2 to 1. At 2020-08-05T14:45:31Z an instance was taken out of service in response to a difference between desired and actual capacity, shrinking the capacity from 2 to 1. At 2020-08-05T14:45:31Z instance i-0e7f4fcddc23ce644 was selected for termination.	2020 8월 05, 11:45:31 오후 +09:00	

# AWS Load Balancer



New EC2 Experience  Tell us what you think X

Load Balancer 생성 작업

EC2 대시보드 New

이벤트 New

태그

제한

▶ 인스턴스

▶ 이미지

▶ Elastic Block Store

▶ 네트워크 및 보안

▼ 로드 밸런싱

로드밸런서 New

대상 그룹 New

Load Balancer 유형 선택

Elastic Load Balancing은 다음 세 유형의 Load Balancer를 지원합니다 - Application Load Balancer, Network Load Balancer(신규) 및 Classic Load Balancer. 요구 사항을 충족하는 Load Balancer 유형을 선택하십시오. [내게 적합한 Load Balancer에 대해 자세히 알아보십시오.](#)

**Application Load Balancer** 생성

HTTP  
HTTPS

HTTP 및 HTTPS 트래픽을 사용하는 웹 애플리케이션을 위한 유연한 기능이 필요한 경우 Application Load Balancer를 선택합니다. 요청 수준에 따라 작동하는 Application Load Balancer는 마이크로서비스 및 컨테이너를 비롯한 애플리케이션 아키텍처를 대상으로 하는 고급 라우팅 및 표시 기능을 제공합니다.

[자세히 알아보기 >](#)

**Network Load Balancer** 생성

TCP  
TLS  
UDP

애플리케이션에 초고성능, 대규모 TLS 오프로딩, 중앙 집중화된 인증서 배포, UDP에 대한 지원 및 고정 IP 주소가 필요한 경우 Network Load Balancer를 선택합니다. 연결 수준에서 작동하는 Network Load Balancer는 안전하게 초당 수백만 개의 요청을 처리하면서도 극히 낮은 지연 시간을 유지할 수 있습니다.

[자세히 알아보기 >](#)

**Classic Load Balancer** 생성

이전 세대  
HTTP, HTTPS 및 TCP용

EC2-Classic 네트워크에서 구축된 기존 애플리케이션이 있는 경우 Classic Load Balancer를 선택합니다.

[자세히 알아보기 >](#)

# AWS Load Balancer



## 1단계: Load Balancer 구성

### 기본 구성

Load Balancer를 구성하려면 이름을 입력하고, 체계를 선택하고, 하나 이상의 리스너를 지정하고, 네트워크를 선택합니다. 기본 구성은 선택한 네트워크에서의 인터넷 경계 Load Balancer입니다.

이름 <span style="color: #808080;">(i)</span>	<input type="text" value="exercise-lb"/>
체계 <span style="color: #808080;">(i)</span>	<input checked="" type="radio"/> 인터넷 경계 <input type="radio"/> 내부
IP 주소 유형 <span style="color: #808080;">(i)</span>	<input type="text" value="ipv4"/>

### 리스너

리스너는 구성한 프로토콜 및 포트를 사용하여 연결 요청을 확인하는 프로세스입니다.

#### Load Balancer 프로토콜

#### Load Balancer 포트

### 가용 영역

Load Balancer에서 활성화할 가용 영역을 지정합니다. Load Balancer는 지정한 가용 영역에 위치한 대상으로만 트래픽을 라우팅합니다. 가용 영역은 VPC 내부에 있는 서버나 애플리케이션을 표시합니다.

VPC <span style="color: #808080;">(i)</span>	<input type="text" value="vpc-3f62d545 (172.31.0.0/16) (기본값)"/>
가용 영역	<input checked="" type="checkbox"/> us-east-1a <input type="text" value="subnet-33deb81d"/> IPv4 주소 <span style="color: #808080;">(i)</span> AWS에서 할당
	<input checked="" type="checkbox"/> us-east-1b <input type="text" value="subnet-20765c6a"/> IPv4 주소 <span style="color: #808080;">(i)</span> AWS에서 할당
	<input checked="" type="checkbox"/> us-east-1c <input type="text" value="subnet-ecfd99b0"/> IPv4 주소 <span style="color: #808080;">(i)</span> AWS에서 할당

## 1. Load Balancer 구성

## 2. 보안 설정 구성

## 3. 보안 그룹 구성

## 4. 라우팅 구성

## 5. 대상 등록

## 6. 검토

## 3단계: 보안 그룹 구성

보안 그룹은 Load Balancer에 대한 트래픽을 제어하는 방화벽 규칙 세트입니다. 이 페이지에서는 특정 트래픽을 Load Balancer에 허용하는 방화벽 규칙을 설정합니다.

보안 그룹 할당:  새 보안 그룹 생성

기존 보안 그룹 선택

#### 보안 그룹 ID

#### 이름

sg-53273410

default

sg-01eeef8c02c64e3657

launch-wizard-2

# AWS Load Balancer



## 4단계: 라우팅 구성

Load Balancer는 지정된 프로토콜 및 포트를 사용하여 이 대상 그룹의 대상으로 요청을 라우팅하며,

### 대상 그룹

대상 그룹	<input type="button" value="새 대상 그룹"/>
이름	exercise-target-group
대상 유형	<input checked="" type="radio"/> 인스턴스 <input type="radio"/> IP <input type="radio"/> Lambda 함수
프로토콜	HTTP
포트	8000

### 상태 검사

프로토콜	HTTP
경로	/

or /health

### ▶ 고급 상태 검사 설정

## 5단계: 대상 등록

대상 그룹에 대상을 등록합니다. 활성화된 가용 영역에서 대상을 등록할 경우, Load Balancer는 등록 프로세스가 완료되고 대상이 초기 상태 검사를 통과하자마자 해당 대상으로 요청을 라우팅하기 시작합니다.

### 등록된 대상

인스턴스를 등록 취소하려면 등록된 인스턴스를 하나 이상 선택한 다음 제거를 클릭합니다.

<input type="button" value="제거"/>	인스턴스	이름	포트	상태	보안 그룹
사용 가능한 인스턴스가 없습니다.					

### 인스턴스

추가 인스턴스를 등록하려면 하나 이상의 실행 인스턴스를 선택하고 포트를 지정한 다음 추가를 클릭하십시오. 기본 포트는 대상 그룹에 지정된 포트입니다. 인스턴스가 지정된 포트에 이미 등록되어 있는 경우, 등록은 실패합니다.

등록된 항목에 추가 포트: 8000

<input type="button" value="인스턴스 검색"/>		X				
	인스턴스	이름	상태	보안 그룹	영역	서브넷 ID
	i-05baf92594233d49c		<span style="color: green;">running</span>	launch-wizard-2	us-east-1a	subnet-33deb81d

## AWS Load Balancer



- Load Balancer 대상 그룹에 Auto Scaling의 "EXERCISE\_GROUP"을 등록

The screenshot shows the AWS CloudFormation console interface. On the left, there's a navigation pane with 'CloudFormation' selected. The main area displays a stack named 'HelloWorld' with the following resources:

- HelloWorldFunction**: AWS Lambda Function (Type: AWS::Lambda::Function)
  - Description: Hello World Lambda Function
  - Runtime: Node.js 12.x
  - Code Size: 1.1 MB
  - Role: arn:aws:iam::123456789012:lambda执行角色 (arn:aws:lambda:ap-northeast-1:123456789012:执行角色)
- HelloWorldPolicy**: AWS Lambda Execution Policy (Type: AWS::Lambda::Permission)
  - Principal: service-role/AWSLambdaBasicExecutionRole
  - Action: lambda:InvokeFunction
  - Resource: !GetAtt HelloWorldFunction.Arn

At the bottom right, there are buttons for 'Create Stack' and 'Edit'.

# AWS Load Balancer



- Load Balancer 대상 그룹에 Auto Scaling의 "EXERCISE\_GROUP"을 등록

EC2 > 대상 그룹 > exercise-target-group

## exercise-target-group

arn:aws:elasticloadbalancing:us-east-1:735824579779:targetgroup/exercise-target-group/46330c83df382ec3

기본 구성			
대상 유형 instance	프로토콜 : 포트 HTTP : 8000	VPC vpc-3f62d545	로드 밸런서 exercise-lb

그룹 세부 정보 대상 모니터링 태그

### 등록된 대상 (1)

인스턴스 ID	포트	영역	상태	상태 세부 정보
i-05baf92594233d49c	8000	us-east-1a	healthy	

# AWS Load Balancer



Load Balancer 생성 작업

태그 및 속성별 필터 또는 키워드별 검색

이름	DNS 이름	상태	VPC ID	가용 영역
exercise-lb	exercise-lb-1844322574.us-east-1.elb.amazonaws.com	active	vpc-3f62d545	us-east-1b, us-east-1a, ...

로드 밸런서: exercise-lb

설명 리스너 모니터링 통합 서비스 태그

기본 구성

이름	exercise-lb
ARN	arn:aws:elasticloadbalancing:us-east-1:735824579779:loadbalancer/app/exercise-lb/d5d1c8f3ede29b42
DNS 이름	exercise-lb-1844322574.us-east-1.elb.amazonaws.com (A 레코드)
상태	active
유형	application
체계	internet-facing

← → ⏪ ⓘ 주의 요함 | exercise-lb-1844322574.us-east-1.elb.amazonaws.com:8000

Hello, World! : 175.192.16.174, Wed Aug 05 2020 15:14:07 GMT+0000 (협정 세계시)

# AWS Fault Tolerance



- 장애 발생 시 문제가 생긴 서버에는 요청을 보내지 않고, 정상적인 서버에만 요청을 전달
- 로드 밸런싱 > 대상 그룹

상태 검사 프로토콜

HTTP ▼

상태 검사 경로

비본 경로 "/" 를 사용하여 루트를 ping하거나 원하는 경우 사용자 지정 경로를 지정합니다.

/health

최대 1024자까지 허용됩니다.

기본값 복원

▼ 고급 상태 검사 설정

포트

대상에 대한 상태 검사를 수행할 때 로드 밸런서가 사용하는 포트입니다. 기본값은 각 대상이 로드 밸런서로부터 트래픽을 수신하는 포트이지만, 다른 포트를 지정할 수 있습니다.

트래픽 포트

재정의

정상 임계 값

비정상 상태의 대상을 정상으로 간주하기까지 필요한 연속적 상태 검사 성공 횟수입니다(2~10).

2

2-10

비정상 임계값

대상을 비정상 상태로 간주하기까지 필요한 연속적인 상태 검사 실패 횟수입니다(2~10).

2

2-10

제한 시간

상태 검사 실패를 의미하는 응답이 없는 기간입니다(2~120초).

2 초

2-120

간격

개별 대상의 상태 검사 사이의 대략적인 시간입니다(5~300초).

5 초

5-300

성공 코드

대상으로부터 응답 성공을 확인할 때 사용하는 HTTP 코드입니다. 값 범위(예: "200-299")에서 여러 값(예: "200,202")을 지정할 수 있습니다.

200

## AWS Fault Tolerance



- Auto Scaling 그룹에서 EXERCISE\_GROUP의 그룹 크기를 2로 조정

그룹 크기

원하는 용량을 변경하여 Auto Scaling 그룹의 크기를 지정합니다. 최소 및 최대 용량 한도를 지정할 수도 있습니다. 원하는 용량은 한도 범위 내에 있어야 합니다.

원하는 용량  
2

최소 용량  
1

최대 용량  
2

인스턴스 (2)

인스턴스 필터링

<input type="checkbox"/>	인스턴스 ID	▲	수명 주기	▼	인스턴스 유형	▼
<input type="checkbox"/>	i-05baf92594233...		InService		t2.micro	
<input type="checkbox"/>	i-0cce3e2cc297a...		InService		t2.micro	

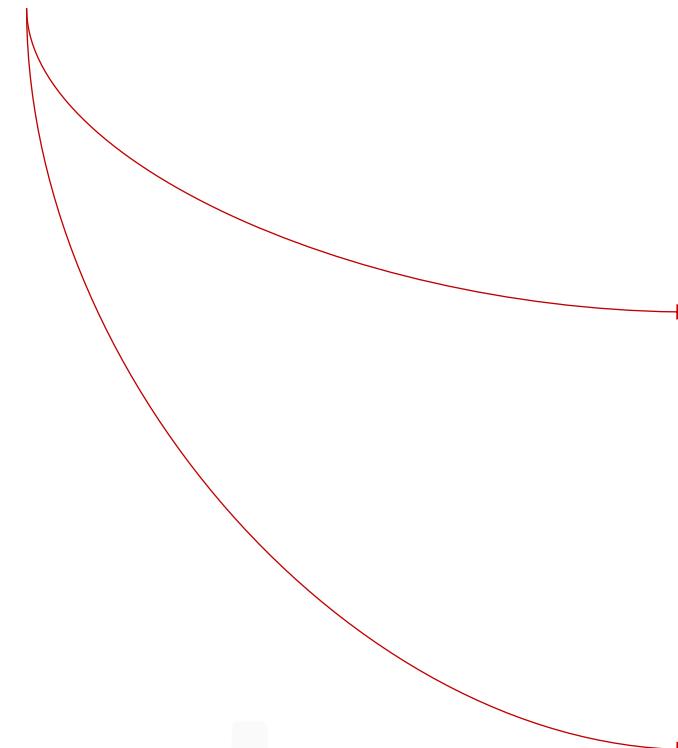
# AWS Fault Tolerance



## 로드 밸런서에서 2개의 서버로 정상 요청

← → ⌂ ⓘ 주의 요함 | exercise-lb-1844322574.us-east-1.elb.amazonaws.com:8000

Hello, World! : 175.192.16.174,Wed Aug 05 2020 15:25:50 GMT+0000 (협정 세계시)



```
✖ ec2-user@ip-172-31-94-216:~ (ssh)
> ec2-user@1.0.0 start /home/ec2-user
> node app.js

Example app listening on port 8000!
175.192.16.174
175.192.16.174
^[[175.192.16.174

✖ ec2-user@ip-172-31-29-53:~ (ssh)
> node app.js

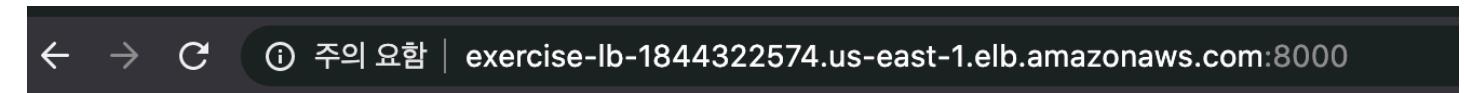
Example app listening on port 8000!
175.192.16.174
175.192.16.174
175.192.16.174
175.192.16.174
```

# AWS Fault Tolerance

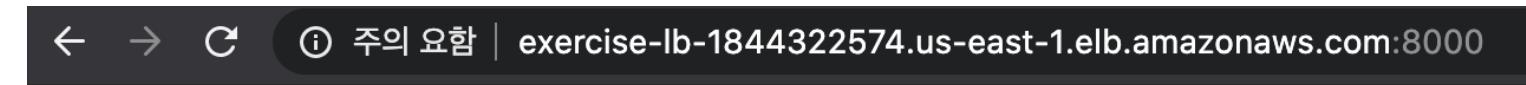


## 한개의 서버 종료

```
● ● ●  
X ec2-user@ip-172-31-94-216:~ (ssh)  
> ec2-user@1.0.0 start /home/ec2-user  
> node app.js  
  
Example app listening on port 8000!  
175.192.16.174  
175.192.16.174  
^[[175.192.16.174  
^C[ec2-user@ip-172-31-94-216 ~]$  
  
X ec2-user@ip-172-31-29-53:~ (ssh)  
> node app.js  
  
Example app listening on port 8000!  
175.192.16.174  
175.192.16.174  
175.192.16.174  
175.192.16.174
```



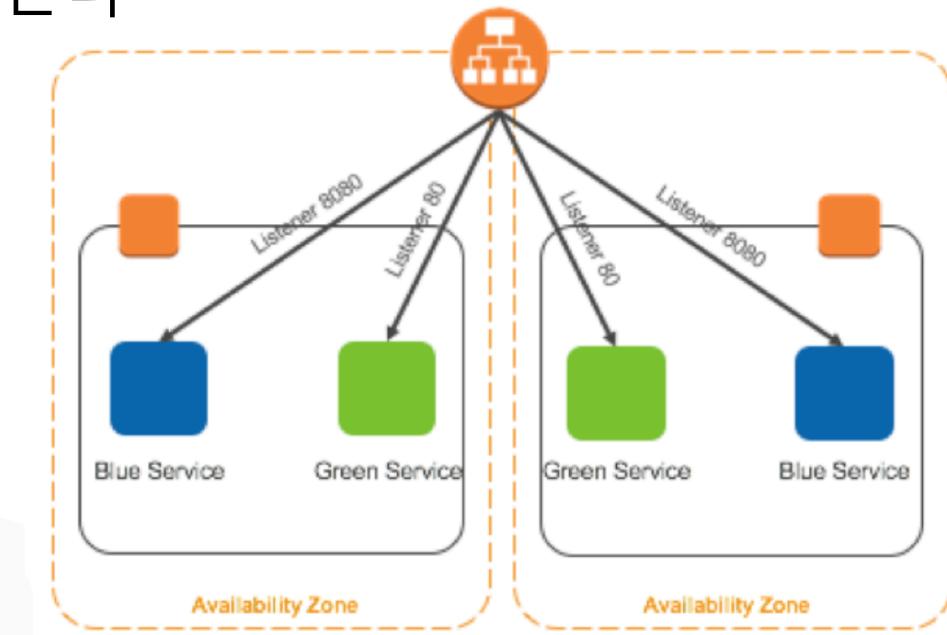
## 502 Bad Gateway



Hello, World! : 175.192.16.174,Wed Aug 05 2020 15:25:50 GMT+0000 (협정 세계시)

## ■ AWS 블루/그린 배포

- 구, 신버전이 동시에 떠 있는 시간을 매우 짧게 처리할 수 있다
  - 모든 인스턴스가 사전에 준비 되어 있다
- 롤백을 빨리 처리할 수 있다
  - 로드 밸런서에 등록, 해제만 하면 쉽게 롤백 가능
  - 서비스에 문제가 없을 경우, 하나의 그룹에 있는 인스턴스를 모두 종료
- 배포 과정에서 서비스되는 인스턴스의 수가 줄지 않는다
  - 요청량을 처리하는 데 장애가 없음



# AWS 블루/그린 배포

## 블루/그린 배포를 위한 Auto Scaling 그룹 만들기

EC2 > Auto Scaling 그룹 > Auto Scaling 그룹 생성

1단계 시작 템플릿 또는 구성 선택

2단계 설정 구성

3단계 (선택 사항) 고급 옵션 구성

4단계 (선택 사항) 그룹 크기 및 조정 정책 구성

5단계 (선택 사항) 알림 추가

6단계 (선택 사항) 태그 추가

7단계 검토

### 시작 템플릿 또는 구성 선택 Info

이 Auto Scaling 그룹에서 시작하는 모든 EC2 인스턴스에 공통된 설정이 포함된 시작 템플릿을 지정합니다. 현재 시작 구성을 사용하는 경우 시작 템플릿으로 마이그레이션할 수 있습니다.

**이름**

**Auto Scaling 그룹 이름**  
그룹을 식별할 이름을 입력합니다.

**DEMO-GRP-BLUE**

현재 리전에서 이 계정에 대해 고유해야 하며 255자를 넘지 않아야 합니다.

### 시작 템플릿 Info

**시작 템플릿** Info **시작 구성으로 전환**

Amazon Machine Image(AMI), 인스턴스 유형, 키 페어 및 보안 그룹과 같은 인스턴스 수준 설정이 포함된 시작 템플릿을 선택합니다.

**dowon-exercise-template**

**버전**

**Default (1)** ▾ **C**

**시작 템플릿 버전 생성** Info

**시작 템플릿 버전 생성** Info

# AWS 블루/그린 배포

## 블루/그린 배포를 위한 Auto Scaling 그룹 만들기

EC2 > Auto Scaling 그룹 > Auto Scaling 그룹 생성

1단계  
시작 템플릿 또는 구성 선택

2단계  
설정 구성

3단계 (선택 사항)  
고급 옵션 구성

4단계 (선택 사항)  
그룹 크기 및 조정 정책 구성

5단계 (선택 사항)  
알림 추가

6단계 (선택 사항)  
태그 추가

7단계  
검토

### 설정 구성 Info

아래 설정을 구성합니다. 시작 템플릿을 선택했는지에 따라 이러한 설정에는 EC2 리소스를 최적화하는 방법이 달라집니다.

#### 구매 옵션 및 인스턴스 유형 Info

시작 템플릿 준수

시작 템플릿에 따라 구매 옵션(온디맨드 또는 스팟)과 인스턴스 유형이 결정됩니다.

구매 옵션과 함께 시작할 온디맨드 인스턴스 유형을 정합니다. 이 옵션은 매우 유용합니다.

#### 네트워크 Info

대부분의 애플리케이션에서는 여러 가용 영역을 사용할 수 있으며 EC2 Auto Scaling이 여러 가용 영역에서 인스턴스를 배포합니다. 기본 VPC와 기본 서브넷은 빠르게 시작하는 데 적합합니다.

#### VPC

vpc-3f62d545  
172.31.0.0/16 Default



#### VPC 생성 Info

#### 서브넷

서브넷 선택  
us-east-1a | subnet-33deb81d X  
172.31.80.0/20 Default



us-east-1b | subnet-20765c6a X  
172.31.16.0/20 Default



us-east-1c | subnet-ecfd99b0 X  
172.31.32.0/20 Default



#### 서브넷 생성 Info

EC2 > Auto Scaling 그룹 > Auto Scaling 그룹 생성

1단계  
시작 템플릿 또는 구성 선택

2단계  
설정 구성

3단계 (선택 사항)  
고급 옵션 구성

4단계 (선택 사항)  
그룹 크기 및 조정 정책 구성

5단계 (선택 사항)  
알림 추가

6단계 (선택 사항)  
태그 추가

7단계  
검토

### 그룹 크기 및 조정 정책 구성 Info

Auto Scaling 그룹의 원하는 용량, 최소 용량 및 최대 용량을 설정합니다. 선택적으로 조정 정책을 추가하여 그룹의 인스턴스 수를 동적으로 조정할 수 있습니다.

#### 그룹 크기 - 선택 사항 Info

원하는 용량을 변경하여 Auto Scaling 그룹의 크기를 지정합니다. 최소 및 최대 용량 한도를 지정할 수도 있습니다. 원하는 용량은 한도 범위 내에 있어야 합니다.

원하는 용량

1

최소 용량

1

최대 용량

1

#### 조정 정책 - 선택 사항

조정 정책을 사용하여 수요의 변화를 충족하도록 Auto Scaling 그룹의 크기를 동적으로 조정할지 여부를 선택합니다. Info

대상 추적 조정 정책

원하는 결과를 선택하고 조정 정책에 따라 필요한 경우 용량을 추가 및 제거하여 해당 결과를 달성합니다.

없음

# AWS 블루/그린 배포

## 블루/그린 배포를 위한 Auto Scaling 그룹 만들기

EC2 > Auto Scaling 그룹 > Auto Scaling 그룹 생성

1단계  
시작 템플릿 또는 구성 선택

2단계  
설정 구성

3단계 (선택 사항)  
고급 옵션 구성

4단계 (선택 사항)  
그룹 크기 및 조정 정책 구성

5단계 (선택 사항)  
알림 추가

6단계 (선택 사항)  
태그 추가

7단계  
검토

### 태그 추가 Info

태그를 추가하면 AWS에서 Auto Scaling 그룹의 검색, 필터링 및 추적에 도움이 됩니다. 인스턴스가 시작될 때 이러한 태그를 인스턴스에 자동으로 추가하도록 선택할 수도 있습니다.

ⓘ 시작 템플릿에 태그를 지정하여 인스턴스(및 연결된 EBS 블륨)에 태그를 추가할 수도 있습니다. 하지만 Auto Scaling 그룹에 대해 중복된 키가 지정되어 있는 경우 시작 템플릿의 인스턴스 태그 값이 재정의되므로 주의해야 합니다. X

태그(1)			
키 <input type="text" value="Name"/>	값 - 선택 사항 <input type="text" value="demo-grp-blue"/>	새 인스턴스에 태그 지정 <input checked="" type="checkbox"/>	<button>제거</button>
<button>태그 추가</button>			

49 남은 수

취소 이전 다음

# AWS 블루/그린 배포

## 기존에 생성 해둔 “exercise-target-group” 대상 그룹에 등록

EC2 > Auto Scaling 그룹

Auto Scaling 그룹 (1/2)

Auto Scaling 그룹 검색

이름 시작 템플릿/구성 인스턴스 상태 원하는 용량 최소 최대 가용 영역

<input checked="" type="checkbox"/> DEMO-GRP-BLUE	dowon-exercise-template   버전 기본값	0	● 용량 업데이트 중	1	1	1	us-east-1a, us-east-1b, u...
<input type="checkbox"/> Dowon-Exercise-Group	dowon-exercise-template   버전 기본값	1	-	1	1	2	us-east-1a, us-east-1b, u...

EC2 > Auto Scaling 그룹 > DEMO-GRP-BLUE

편집 DEMO-GRP-BLUE Info

로드 밸런싱 - 선택 사항

로드 밸런서의 대상 그룹 선택

대상 그룹 선택

exercise-target-group

대상 그룹 생성

로드 밸런서 선택

로드 밸런서 선택

Cancel 업데이트

# AWS 블루/그린 배포

## 로드 밸런서 테스트

Load Balancer 생성 작업

태그 및 속성별 필터 또는 키워드별 검색

이름 DNS 이름 상태

exercise-lb exercise-lb-159534608.us-e... active

로드 밸런서: exercise-lb

설명 리스너 모니터링 통합 서비스 태그

기본 구성

이름	exercise-lb
ARN	arn:aws:elasticloadbalancing:us-east-1:735824579779:loadbalancer/exercise-lb-159534608.us-east-1.elb.amazonaws.com
DNS 이름	exercise-lb-159534608.us-east-1.elb.amazonaws.com (A 레코드)

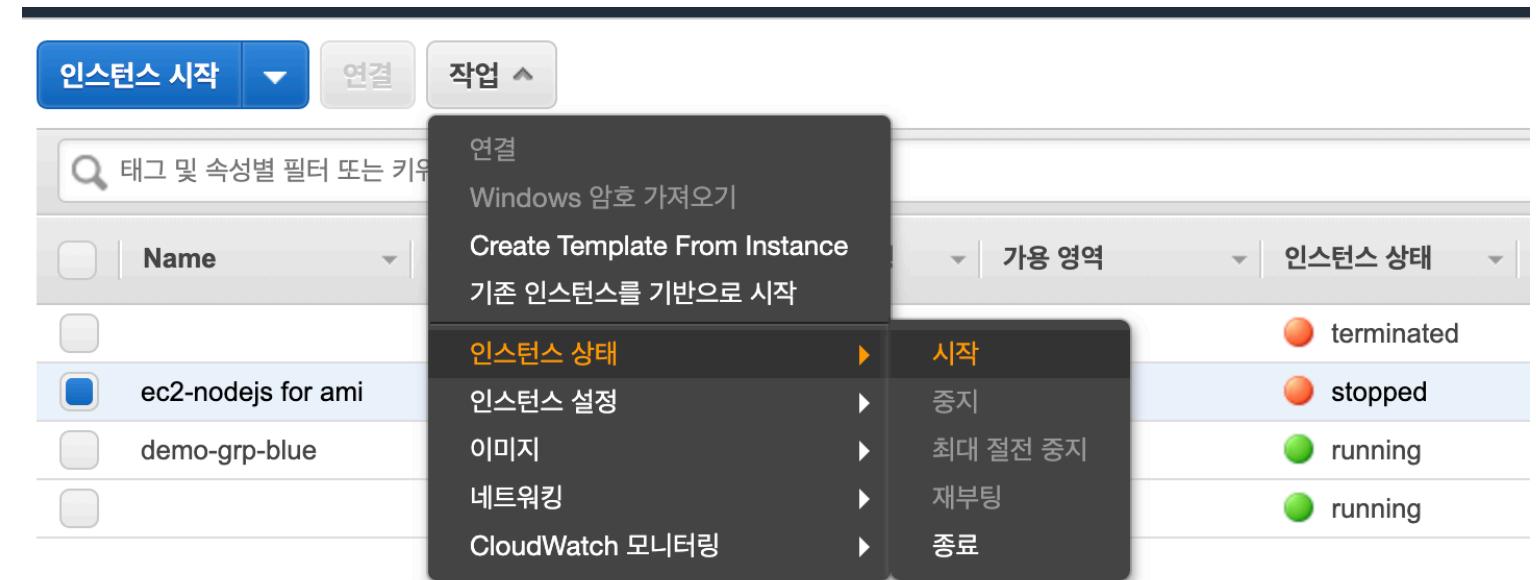
```
x ec2-user@ip-172-31-44-215:~ (ssh)
User info is: [ RowDataPacket { id: 'kenneth', password: '1234' } ]
User info is: [ RowDataPacket { id: 'kenneth', password: '1234' } ]
User info is: [ RowDataPacket { id: 'kenneth', password: '1234' } ]
User info is: [ RowDataPacket { id: 'kenneth', password: '1234' } ]
User info is: [ RowDataPacket { id: 'kenneth', password: '1234' } ]
User info is: [ RowDataPacket { id: 'kenneth', password: '1234' } ]
User info is: [ RowDataPacket { id: 'kenneth', password: '1234' } ]
175.192.16.174
[]

x ec2-user@ip-172-31-85-53:~ (ssh)
> node index.js

Express server listening on port 8000
User info is: [ RowDataPacket { id: 'kenneth', password: '1234' } ]
User info is: [ RowDataPacket { id: 'kenneth', password: '1234' } ]
User info is: [ RowDataPacket { id: 'kenneth', password: '1234' } ]
User info is: [ RowDataPacket { id: 'kenneth', password: '1234' } ]
175.192.16.174
```

## AWS 블루/그린 배포

- 새로운 버전의 코드를 적용한 AMI와 시작 템플릿 생성
- 이미지 생성을 위한 EC2 시작



## AWS 블루/그린 배포

- 새로운 버전의 코드를 적용한 AMI와 시작 템플릿 생성
- 코드 수정
  - / API 수정 → **Hello, New World!**
  - **/users/:id** API 추가

```
app.get('/users/:id', (req, res) => {
  connection.query('SELECT * from Users WHERE id=\'' + req.params.id + '\'', (error, rows) => {
    if (error) throw error;
    console.log('User detail info is: ', rows);
    res.send(rows);
  });
});
```

- EC2 인스턴스 중지

# AWS 블루/그린 배포

## ▪ 새로운 버전의 코드를 적용한 AMI와 시작 템플릿 생성

EC2 > 시작 템플릿 > 시작 템플릿 생성

### 시작 템플릿 생성

시작 템플릿을 생성하면 저장된 인스턴스 구성을 만들어 두었다가 나중에 이를 재사용하고, 공유하고, 시작할 수 있습니다. 여러 버전의 템플릿을 저장할 수 있습니다.

#### 시작 템플릿 이름 및 설명

##### 시작 템플릿 이름 - 필수

dowon-exercise-template-beta

이 계정에 대해 고유해야 합니다. 최대 128자입니다. '&', '\*', '@' 등의 특수 문자나 공백은 사용할 수 없습니다.

##### 템플릿 버전 설명

beta version

최대 255자

##### Auto Scaling 지침 정보

EC2 Auto Scaling에 이 템플릿을 사용하려면 이 항목을 선택합니다.

EC2 Auto Scaling에 사용할 수 있는 템플릿을 설정하는 데 도움이 되는 지침 제공

#### ▶ 템플릿 태그

#### ▼ 원본 템플릿

기존의 다른 템플릿을 토대로 템플릿을 만들고자 하는 경우, 원한다면 원본 템플릿을 지정할 수 있습니다.

##### 시작 템플릿 이름

dowon-exercise-template

ID: lt-08ae366c253092303

#### 시작 템플릿 콘텐츠

아래에서 시작 템플릿의 세부 정보를 지정합니다. 비워 둔 필드는 시작 템플릿에 포함되지 않습니다.

#### Amazon Machine Image(AMI) 정보

##### AMI

dowon-exercise-ami-beta

ami-066e296233ffea9ad

카탈로그: 내 AMI 아키텍처: 64비트(x86) 가상화: hvm

# AWS 블루/그린 배포

- 새로운 버전의 코드를 적용한 AMI와 시작 템플릿 생성

The screenshot shows the 'Start Template (1/2) Information' step of the AWS CloudFormation template creation process. It lists two available start templates:

시작 템플릿 ID	시작 템플릿 이름	기본 버전
lt-0d6769e8b80b44908	dowon-exercise-template-beta	1
lt-08ae366c253092303	dowon-exercise-template	1

Below the table, the selected template is shown in detail:

**dowon-exercise-template-beta (lt-0d6769e8b80b44908)**

**시작 템플릿 세부 정보**

시작 템플릿 ID	시작 템플릿 이름	기본 버전
lt-0d6769e8b80b44908	dowon-exercise-template-beta	1

# AWS 블루/그린 배포

## 블루/그린 배포를 위한 새로운 Auto Scaling 그룹 만들기

EC2 > Auto Scaling 그룹 > Auto Scaling 그룹 생성

1단계  
시작 템플릿 또는 구성 선택

2단계  
설정 구성

3단계 (선택 사항)  
고급 옵션 구성

4단계 (선택 사항)  
그룹 크기 및 조정 정책 구성

5단계 (선택 사항)  
알림 추가

6단계 (선택 사항)  
태그 추가

7단계  
검토

### 시작 템플릿 또는 구성 선택 Info

이 Auto Scaling 그룹에서 시작하는 모든 EC2 인스턴스에 공통된 설정이 포함된 시작 템플릿을 지정합니다. 현재 시작 구성을 사용하는 경우 시작 템플릿으로 마이그레이션할 수 있습니다.

**이름**

**Auto Scaling 그룹 이름**  
그룹을 식별할 이름을 입력합니다.

DEMO-GRP-GREEN

현재 리전에서 이 계정에 대해 고유해야 하며 255자를 넘지 않아야 합니다.

### 시작 템플릿 Info

시작 템플릿

Amazon Machine Image(AMI), 인스턴스 유형, 키 페어 및 보안 그룹과 같은 인스턴스 수준 설정이 포함된 시작 템플릿을 선택합니다.

dowon-exercise-template-beta

시작 템플릿 생성

버전

Default (1)

시작 템플릿 버전 생성

시작 구성으로 전환

# AWS 블루/그린 배포

- 블루/그린 배포를 위한 새로운 Auto Scaling 그룹 만들기
  - DEMO-GROUP-BLUE와 똑같이 생성

EC2 > Auto Scaling 그룹 > Auto Scaling 그룹 생성

1단계 시작 템플릿 또는 구성 선택

2단계 설정 구성

3단계 (선택 사항) 고급 옵션 구성

4단계 (선택 사항) 그룹 크기 및 조정 정책 구성

5단계 (선택 사항) 알림 추가

6단계 (선택 사항) 태그 추가

7단계 검토

### 그룹 크기 및 조정 정책 구성 Info

Auto Scaling 그룹의 원하는 용량, 최소 용량 및 최대 용량을 설정합니다. 선택적으로 조정 정책을 추가하여 그룹의 인스턴스 수를 동적으로 조정할 수 있습니다.

#### 그룹 크기 - 선택 사항 Info

원하는 용량을 변경하여 Auto Scaling 그룹의 크기를 지정합니다. 최소 및 최대 용량 한도를 지정할 수도 있습니다. 원하는 용량은 한도 범위 내에 있어야 합니다.

원하는 용량	1
최소 용량	1
최대 용량	1

#### 태그(1)

새 인스턴스에 태그 지정

키	값 - 선택 사항
Name	demo-grp-blue

제거

#### 조정 정책 - 선택 사항

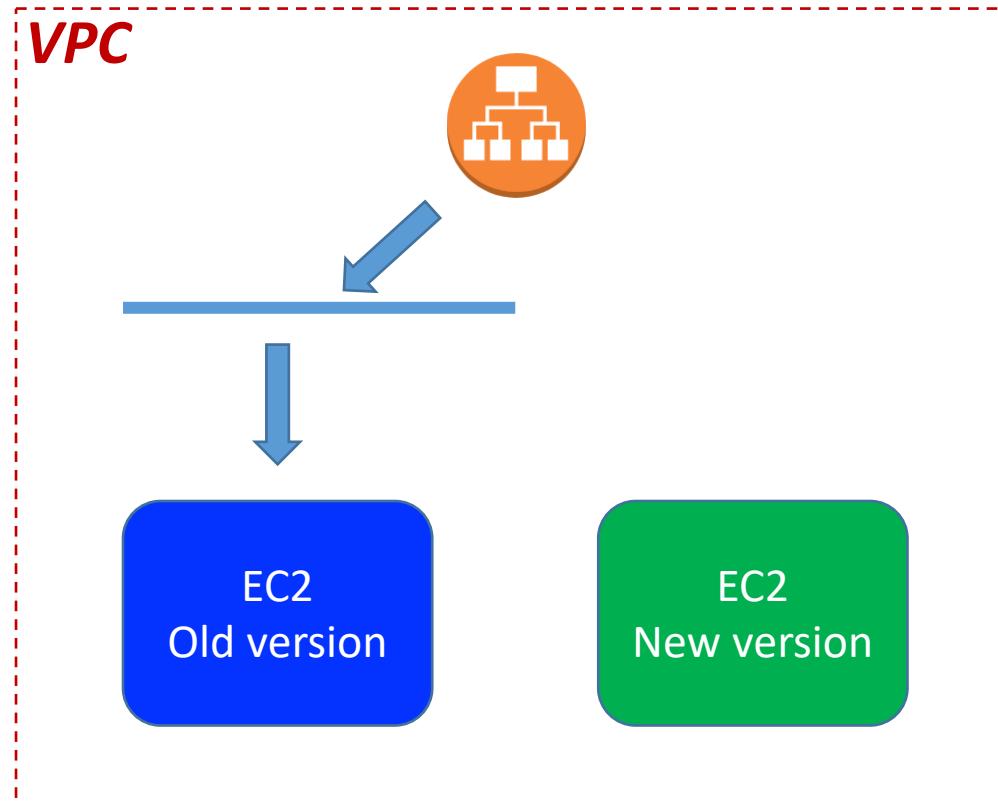
조정 정책을 사용하여 수요의 변화를 충족하도록 Auto Scaling 그룹의 크기를 동적으로 조정할지 여부를 선택합니다. Info

대상 추적 조정 정책  
원하는 결과를 선택하고 조정 정책에 따라 필요한 경우 용량을 추가 및 제거하여 해당 결과를 달성합니다.

없음

## AWS 블루/그린 배포

- 블루/그린 배포를 위한 새로운 Auto Scaling 그룹 만들기



1. GREEN 그룹을 대상 그룹에 포함
2. 두 서버에 요청이 골고루 넘어가도록 로드 밸런서에 등록
3. 두 버전에 문제가 없으면 BLUE 그룹을 대상 그룹에서 제외
4. GREEN 그룹만 남김

## AWS 블루/그린 배포

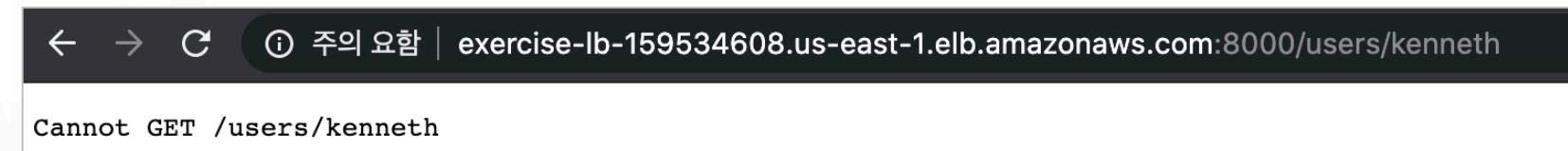
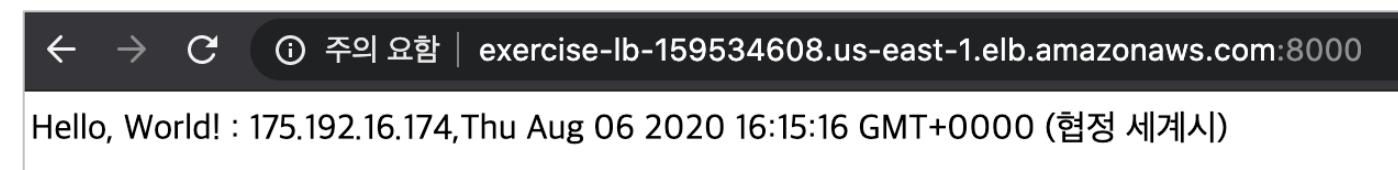
- 로드 밸런서에 **그린** 그룹 등록
  - 로드 밸런서에서 **블루** 그룹으로만 요청이 전달되는지 확인

```
X ec2-user@ip-172-31-44-215:~ (ssh)
[ec2-user@ip-172-31-44-215 ~]$ npm start

> ec2-user@1.0.0 start /home/ec2-user
> node index.js

Express server listening on port 8000
175.192.16.174
175.192.16.174
□
```

**BLUE 그룹의 EC2**



## AWS 블루/그린 배포

### 로드 밸런서에 **그린** 그룹 등록

The screenshot shows the AWS Auto Scaling console with the path: EC2 > Auto Scaling 그룹 > DEMO-GRP-GREEN. The main title is 편집 DEMO-GRP-GREEN Info. Below it, there's a section titled "로드 밸런싱 - 선택 사항".  
**로드 밸런서의 대상 그룹 선택**: A dropdown menu labeled "대상 그룹 선택" contains the entry "exercise-target-group" which is highlighted with a red border. To the right of the dropdown are a "▼" icon and a "C" icon.  
**대상 그룹 생성**: A blue button labeled "대상 그룹 생성" with a small icon.  
**로드 밸런서 선택**: Another dropdown menu labeled "로드 밸런서 선택" with a "▼" icon and a "C" icon.  
**로드 밸런서 선택**: A blue button labeled "로드 밸런서 선택" with a small icon.  

At the bottom right, there are two buttons: "Cancel" and a large orange "업데이트" button.

# AWS 블루/그린 배포

## 로드 밸런서에 블루/그린 테스트

<input checked="" type="checkbox"/>	DEMO-GRP-GREEN	dowon-exercise-template-beta   버전 기본값	1	-	1
<input type="checkbox"/>	DEMO-GRP-BLUE	dowon-exercise-template   버전 기본값	1	-	1
<input type="checkbox"/>	Dowon-Exercise-Group	dowon-exercise-template   버전 기본값	0	-	0

로드 밸런싱

로드 밸런서 대상 그룹  
exercise-target-group

Classic Load Balancer

← → C ⓘ 주의 요함 | exercise-lb-159534608.us-east-1.elb.amazonaws.com:8000

Hello, New World! : 175.192.16.174, Thu Aug 06 2020 16:20:47 GMT+0000 (협정 세계시)

← → C ⓘ 주의 요함 | exercise-lb-159534608.us-east-1.elb.amazonaws.com:8000

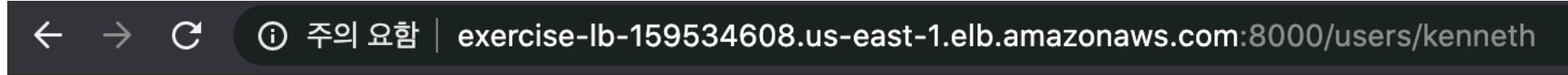
Hello, World! : 175.192.16.174, Thu Aug 06 2020 16:20:57 GMT+0000 (협정 세계시)

로드 밸런서에서 요청 시

BLUE/GREEN이 정상적으로 실행

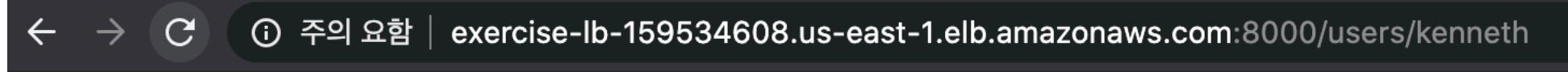
## ■ AWS 블루/그린 배포

### ■ 로드 밸런서에 블루/그린 테스트



Cannot GET /users/kenneth

*BLUE 그룹 응답 → Old version*



```
[{"id": "kenneth", "password": "1234"}]
```

*GREEN 그룹 응답 → New version*

# AWS 블루/그린 배포

## ■ 블루 그룹을 대상 그룹에서 제거

EC2 > Auto Scaling 그룹 > DEMO-GRP-BLUE

### 편집 DEMO-GRP-BLUE Info

#### 로드 밸런싱 - 선택 사항

로드 밸런서의 대상 그룹 선택

대상 그룹 선택



exercise-target-group X

삭제

대상 그룹 생성

#### 로드 밸런서 선택

로드 밸런서 선택



로드 밸런서 선택

로드 밸런서에서 요청 시  
**GREEN**만 정상적으로 실행

```
← → C ⓘ 주의 요함 | exercise-lb-159534608.us-east-1.elb.amazonaws.com:8000/users/kenneth
[{"id": "kenneth", "password": "1234"}]
```

## AWS 블루/그린 배포

- **블루** 그룹의 그룹 크기를 0으로 조정 → EC2 삭제

### 그룹 크기

원하는 용량을 변경하여 Auto Scaling 그룹의 크기를 지정합니다. 최소 및 최대 용량 한도를 지정할 수도 있습니다. 원하는 용량은 한도 범위 내에 있어야 합니다.

원하는 용량

최소 용량

최대 용량

인스턴스 시작 ▾      연결      작업 ▾

태그 및 속성별 필터 또는 키워드별 검색

Name	인스턴스 ID	인스턴스 유형	가용 영역	인스턴스 상태	상태 검사
demo-grp-green	i-024b25470108e43...	t2.micro	us-east-1b	running	2/2개 검사 ...
ec2-nodejs for ami	i-03b1ea1d6d850bcd2	t2.micro	us-east-1e	stopped	
demo-grp-blue	i-05c99c73d9fbdd4a5	t2.micro	us-east-1c	terminated	

- *AWS Code Deploy* 

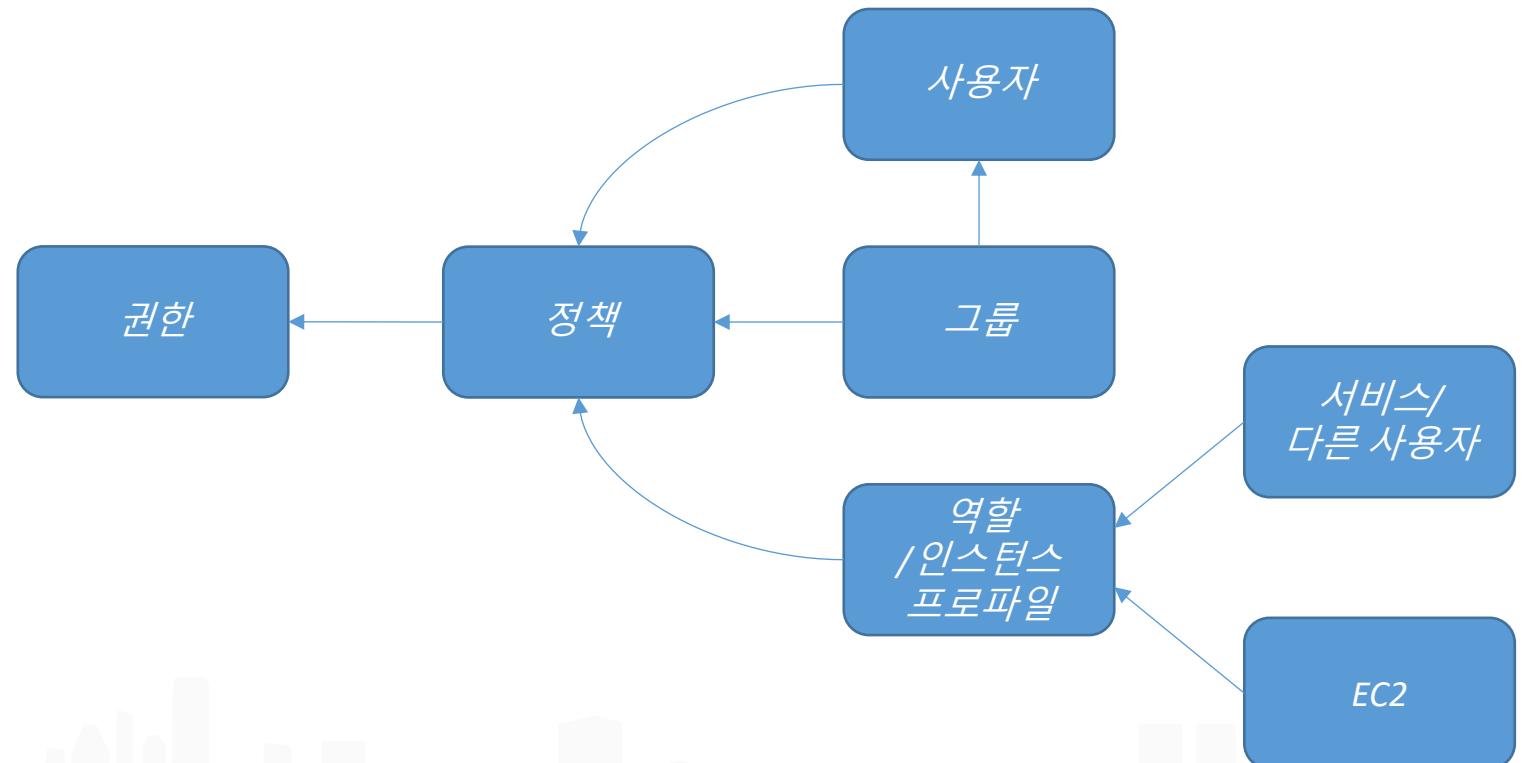
- 작업 순서

- 1) IAM Role 추가 (CodeDeploy, EC2)
- 2) 신규 EC2에 Code Deploy Agent 설치
  - Auto Scaling을 위해서는 AMI 생성



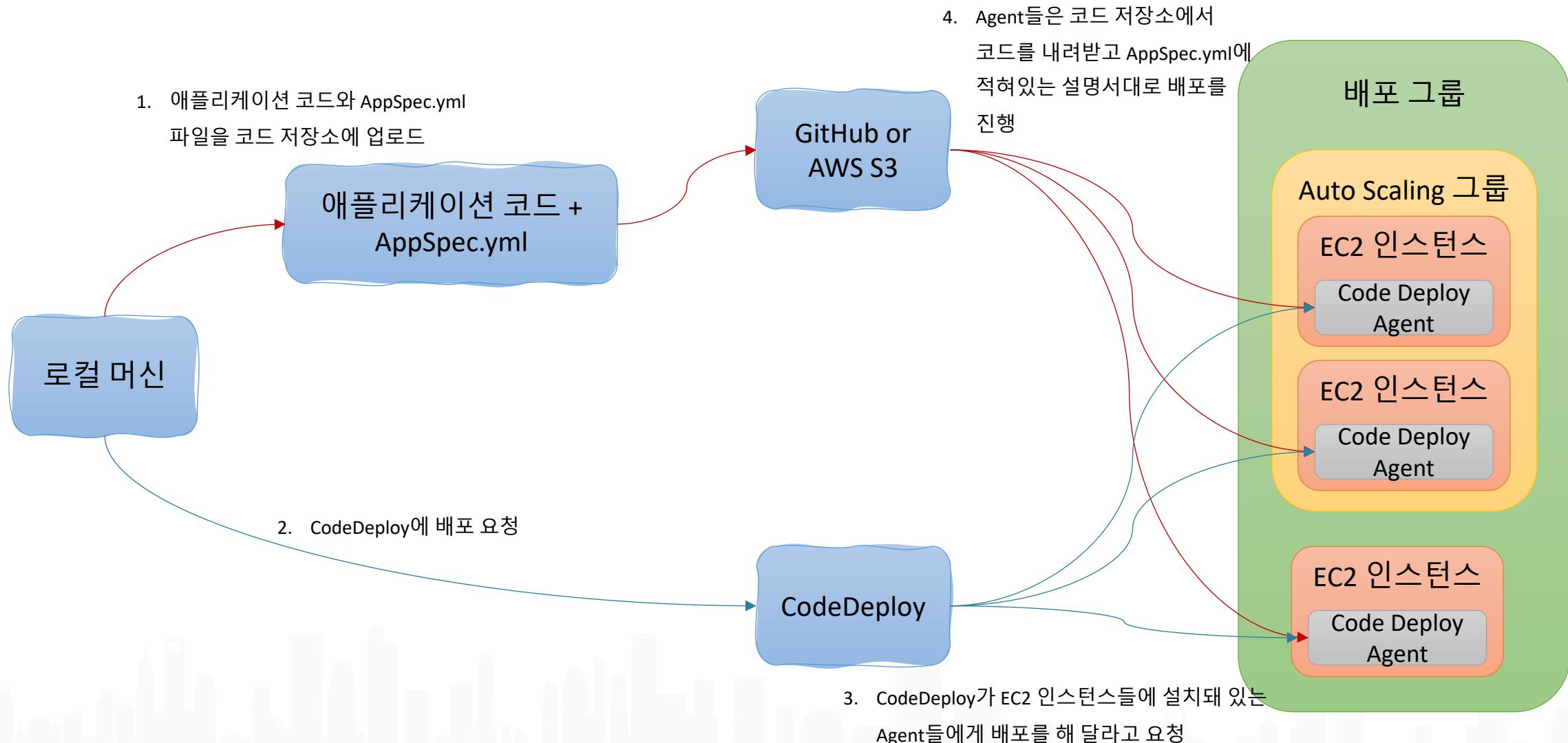
## ■ AWS Code Deploy

- 권한
- 정책
- 사용자
- 그룹
- 역할
- 인스턴스 프로파일



# AWS Code Deploy

## CodeDeploy 배포 과정



## ■ AWS Code Deploy

- CodeDeploy를 위한 서비스 역할 생성
  - IAM Role 생성 → AWS 서비스 → **CodeDeploy** → AWSCodeDeployRole 추가
    - Role name: **exercise-code-deploy-service-role**
- EC2를 위한 역할, 인스턴스 프로파일 생성
  - IAM 정책 생성 → s3에 대한 읽기, 목록보기
    - 정책 이름: exercise-code-deploy-ec2-policy
  - IAM Role 생성 → AWS 서비스 → **EC2** → **exercise-code-deploy-ec2-policy** 추가
    - Role name: exercise-code-deploy-ec2-role
    - 인스턴스 프로파일 ARN 값을 저장
      - *arn:aws:iam::735824579779:instance-profile/exercise-code-deploy-ec2-role*

## ■ AWS Code Deploy

- CodeDeploy로 배포 가능한 인스턴스 생성
  - EC2 인스턴스 생성 → EC2 인스턴스에 CodeDeploy Agent 설치
    - \$ wget <https://aws-codedeploy-ap-northeast-2.s3.amazonaws.com/latest/install>
    - \$ chmod +x ./install
    - \$ sudo ./install auto
    - \$ sudo service codedeploy-agent status

*The AWS CodeDeploy agent is running as PID 2955*

## ■ AWS Code Deploy

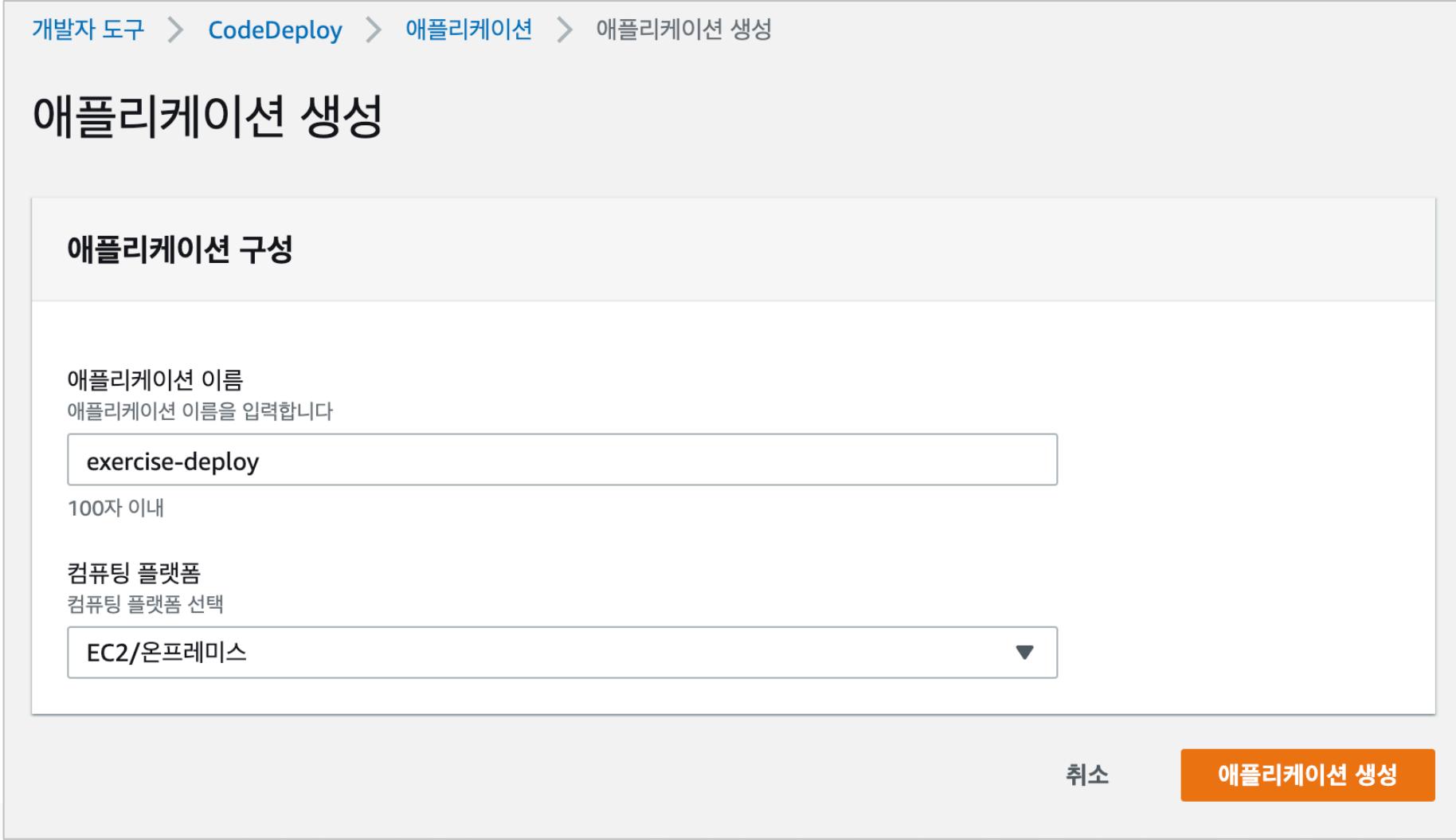
- CodeDeploy로 배포 가능한 인스턴스 생성

- (Auto Scaling의 경우)

- EC2 중지 후 AMI 생성
      - 이미지 이름: ***exercise-image-code-deploy***
    - Launch Template 생성
      - Launch template name: ***exercise-launch-template-code-deploy***
      - AMI ID 선택, Key pair 선택, Security Group 선택
      - 고급 세부 정보
        - IAM 인스턴스 프로파일 지정
          - *arn:aws:iam::735824579779:instance-profile/exercise-code-deploy-ec2-role*
    - Auto Scaling 그룹 생성
      - 인스턴스의 수는 2개씩

## ■ AWS Code Deploy

### ■ CodeDeploy 애플리케이션 생성



개발자 도구 > CodeDeploy > 애플리케이션 > 애플리케이션 생성

## 애플리케이션 생성

### 애플리케이션 구성

애플리케이션 이름  
애플리케이션 이름을 입력합니다

exercise-deploy

100자 이내

컴퓨팅 플랫폼  
컴퓨팅 플랫폼 선택

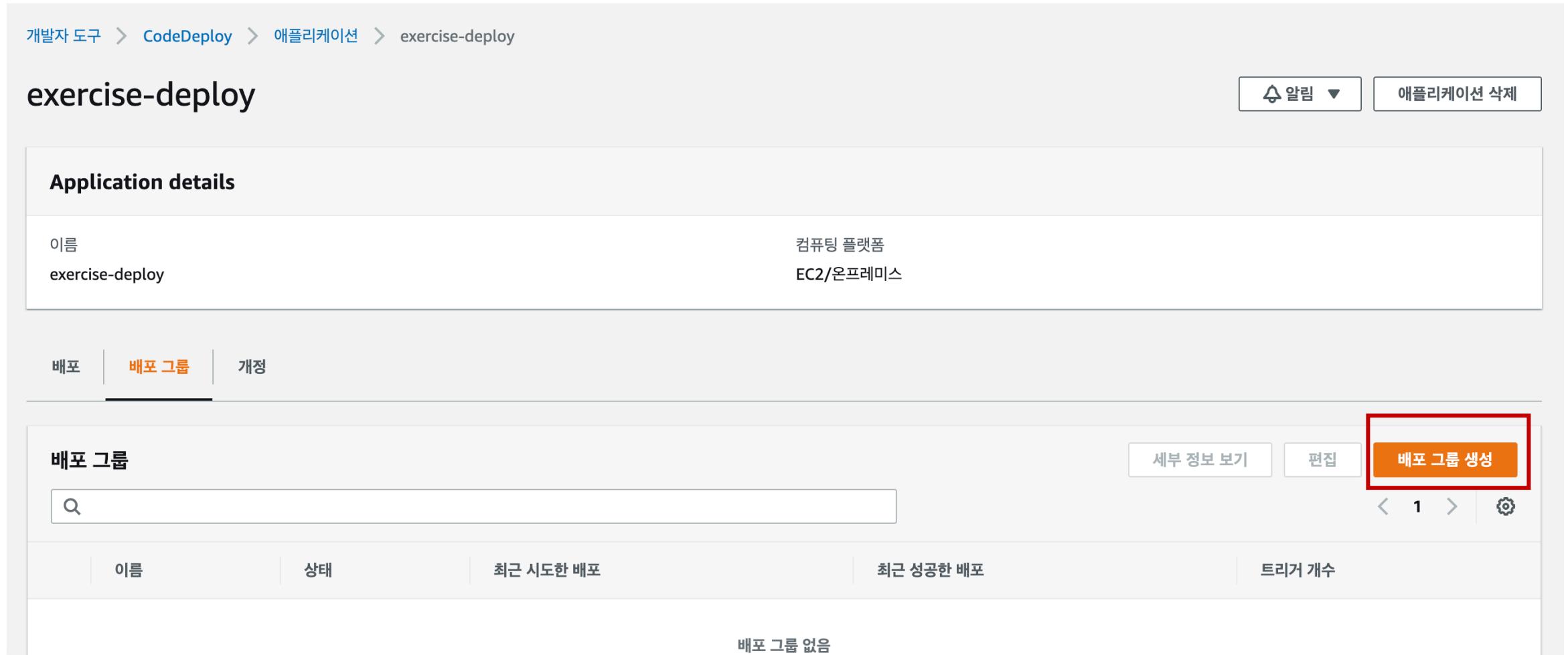
EC2/온프레미스

취소

애플리케이션 생성

# AWS Code Deploy

## CodeDeploy 애플리케이션 생성



개발자 도구 > CodeDeploy > 애플리케이션 > exercise-deploy

### exercise-deploy

Application details

이름	컴퓨팅 플랫폼
exercise-deploy	EC2/온프레미스

배포      **배포 그룹**      개정

#### 배포 그룹

세부 정보 보기      편집      **배포 그룹 생성**

이름	상태	최근 시도한 배포	최근 성공한 배포	트리거 개수
배포 그룹 없음				

# AWS Code Deploy



## CodeDeploy 애플리케이션 생성

### 배포 그룹 생성

**애플리케이션**

- 애플리케이션
- exercise-deploy
- 컴퓨팅 유형
- EC2/온프레미스

**배포 그룹 이름**

배포 그룹 이름 입력  
production-in-place  
100자 제한

production-in-place

**서비스 역할**

서비스 역할 입력  
AWS CodeDeploy 대상 인스턴스에 액세스하도록 허용하는 CodeDeploy 권한이 있는 서비스 역할을 입력합니다.

**배포 유형**

애플리케이션 배포 방법 선택

- 현재 위치  
배포 그룹의 인스턴스를 최신 애플리케이션 개정으로 업데이트합니다.  
배포 중에 각 인스턴스가 업데이트를 위해 잠시 오프라인 상태로 전환됩니다.
- 블루/그린  
배포 그룹의 인스턴스를 새 인스턴스로 교체하고 최신 애플리케이션 개정을 해당 인스턴스에 배포합니다. 대체 환경의 인스턴스가 로드 밸런서에 등록된 후 원본 환경의 인스턴스는 등록 취소되고 종료할 수 있습니다.

**환경 구성**

이 배포에 추가할 Amazon EC2 Auto Scaling 그룹, Amazon EC2 인스턴스 및 온프레미스 인스턴스의 조합 선택

Amazon EC2 Auto Scaling 그룹  
2개의 일치하는 고유한 인스턴스. 자세한 내용을 보려면 여기를 클릭하십시오. [\[링크\]](#)

애플리케이션 개정을 배포할 Amazon EC2 Auto Scaling 그룹을 10개까지 선택할 수 있습니다.

Amazon EC2 인스턴스

온프레미스 인스턴스

**일치하는 인스턴스**

2개의 일치하는 고유한 인스턴스. 자세한 내용을 보려면 여기를 클릭하십시오. [\[링크\]](#)

**배포 설정**

**배포 구성**

기본 및 사용자 지정 배포 구성 목록에서 선택합니다. 배포 구성은 애플리케이션이 배포되는 속도와 배포 성공 또는 실패 조건을 결정하는 규칙 세트입니다.

또는

CodeDeployDefault.AllAtOnce

**로드 밸런서**

배포 프로세스 중에 수신 트래픽을 관리할 로드 밸런서를 선택합니다. 로드 밸런서는 배포 중인 각 인스턴스에서 트래픽을 차단하고 배포 성공 후 인스턴스에 대한 트래픽을 다시 허용합니다.

로드 밸런싱 활성화

▶ 고급 - 선택 사항

[취소](#) [배포 그룹 생성](#)

# AWS Code Deploy

## CodeDeploy 애플리케이션 생성

개발자 도구 > CodeDeploy > 애플리케이션 > exercise-deploy > production-in-place

### production-in-place

편집      삭제      **배포 생성**

#### 배포 그룹 세부 정보

배포 그룹 이름  
production-in-place

애플리케이션 이름  
[exercise-deploy](#)

컴퓨팅 플랫폼  
EC2/온프레미스

배포 유형  
현재 위치

서비스 역할 ARN  
[arn:aws:iam::440263241313:role/exercise-code-deploy-service-role](#)

배포 구성  
[CodeDeployDefault.AllAtOnce](#)

루트 활성

에이전트 업데이트 스케줄러  
[AWS Systems Manager에서 업데이트 일정 예약 방법 알아보기](#)

# AWS Code Deploy

## CodeDeploy 애플리케이션 생성

배포 그룹

production-in-place X

컴퓨팅 플랫폼

EC2/온프레미스

배포 유형

현재 위치

개정 유형

애플리케이션을 Amazon S3에 저장

애플리케이션을 GitHub에 저장

GitHub 토큰 이름

이미 연결한 계정으로 연결된 토큰의 이름을 선택하거나 AWS CodeDeploy에 다른 계정에 액세스할 수 있는 권한을 부여합니다. GitHub 계정에 처음 연결하려면 계정의 별칭을 입력한 다음 「GitHub에 연결」을 선택합니다.

joneconsulting X

GitHub에 연결

리포지토리 이름

joneconsulting/new-simple-nodejs

커밋 ID

7c459a11e4297a1e1af155782bf409b4cd16d2cb

개정 유형

애플리케이션을 Amazon S3에 저장  애플리케이션을 GitHub에 저장

개정 위치  
개정이 저장된 Amazon S3 버킷을 복사하여 붙여 넣습니다.

s3://edu24-edowon-codedeploy/new-simple-nodejs.zip X

s3://bucket-name/folder/object.[zip|tar|tgz]

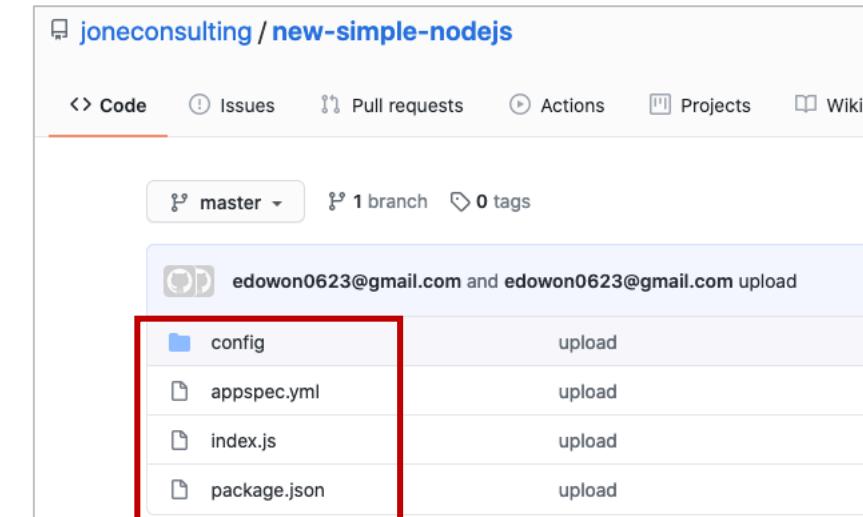
개정 파일 형식

.zip ▼

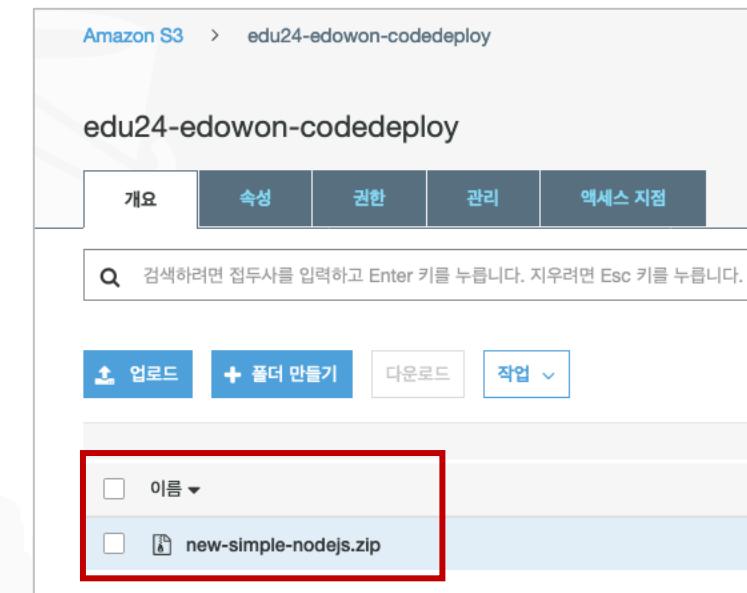
# AWS Code Deploy

## CodeDeploy 애플리케이션 생성

```
[ec2-user@ip-172-31-9-58 ~]$ ls -l
합계 20
drwxr-xr-x 3 root      root      4096  8월 11 15:24 build
-rwxrwxr-x 1 ec2-user  ec2-user  14099  7월 30 22:48 install
[ec2-user@ip-172-31-9-58 ~]$ ls -l ./build
합계 16
-rw-r--r-- 1 root      root      81   8월  7 03:53 appspec.yml
drwxr-xr-x 2 root      root      4096  8월 11 15:20 config
-rw-r--r-- 1 root      root    1205   8월  7 03:53 index.js
-rw-r--r-- 1 root      root     351   8월  7 03:53 package.json
[ec2-user@ip-172-31-9-58 ~]$
```



github



AWS S3