웹서비스를 위한 클라우드 아키텍처링

멀티 클라우드 DR 아키텍처 설계 및 자동 장애 전환 구현

4팀 GTA
백지영, 이충민, 최윤하, 한승규

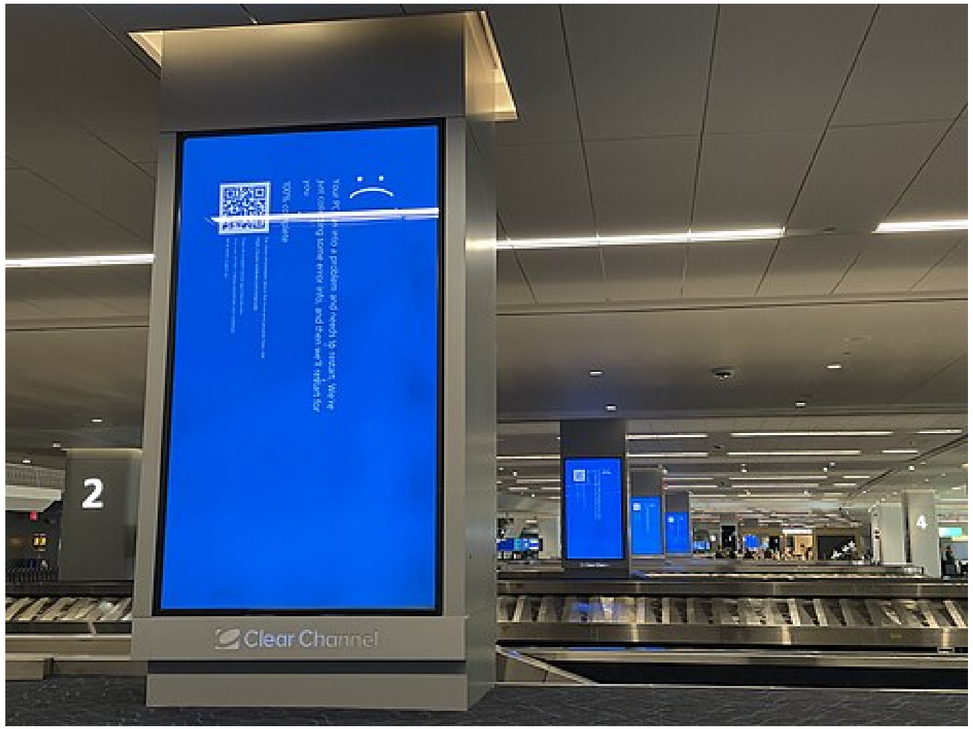
2025.07.04

목차

01 프로젝트 개요

주제선정

Azure 대규모 장애 사건



1차 장애 - Azure 자체 시스템 (24년 7월 18일)

- 원인 : Azure Central US 리전 네트워크 설정 오류
- 지속시간 : 12시간 ~ 24시간
- 영향 : 스토리지 및 Microsoft 365 서비스 중단

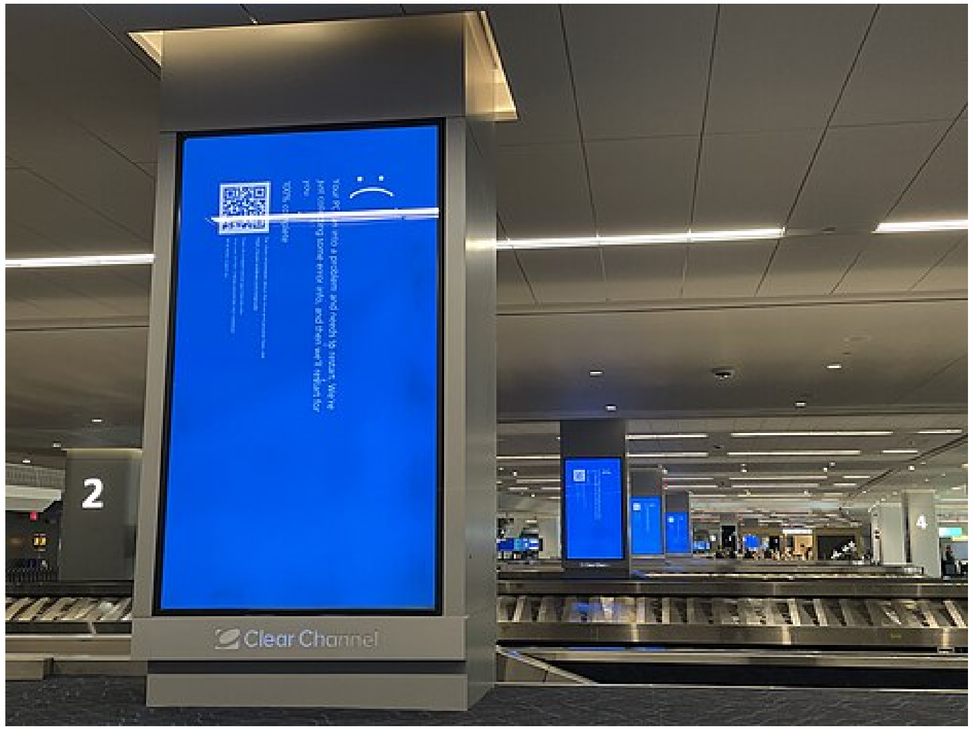
2차 장애 - 크라우드스트라이크 (24년 7월 19일)

- 원인 : 보안 소프트웨어 업데이트 오류로 Windows 시스템 충돌
- 지속시간 : 원인 제거 6시간, 전체 복구 몇 주
- 영향 : Azure 가상머신 대량 블루스크린

01 프로젝트 개요

주제선정

Azure 대규모 장애 사건



피해 규모

전 세계적 영향

- 항공업계 : 5천편 이상 항공기 운항 중단
- 금융업계 : 은행, 거래소 시스템 마비
- 의료업계 : 병원 응급 시스템 중단
- 경제적 손실 : 1조 4천억원 이상

프로젝트 목표

GCP와 AWS를 활용한 고가용성 멀티 클라우드 환경 구축

자동 장애 전환 시스템 구현

운영 비용 최적화를 위한 Active-Passive DR 구성

통합 모니터링 및 보안 강화

01 프로젝트 개요

기술 스택

개발

클라우드

데이터베이스

도구

프로젝트 일정

6/27	팀 구성 및 아키텍처 설계
6/28 6/29	핵심 기술 학습 및 코드 작성
6/30	AWS 기본 인프라 구축
7/1	GCP 환경 구성 및 VPN 연결
7/2	DR 시스템 및 보안 설정
7/3	모니터링 시스템 구축 및 발표 준비
7/4	최종 발표 및 시연

핵심 아키텍처

네트워크 구성

글로벌 DNS : GCP Cloud DNS ↔ AWS Route 53
CDN : AWS CloudFront
WAF : GCP Cloud Armor ↔ AWS WAF
VPN : GCP HA VPN ↔ AWS Site-to-Site VPN

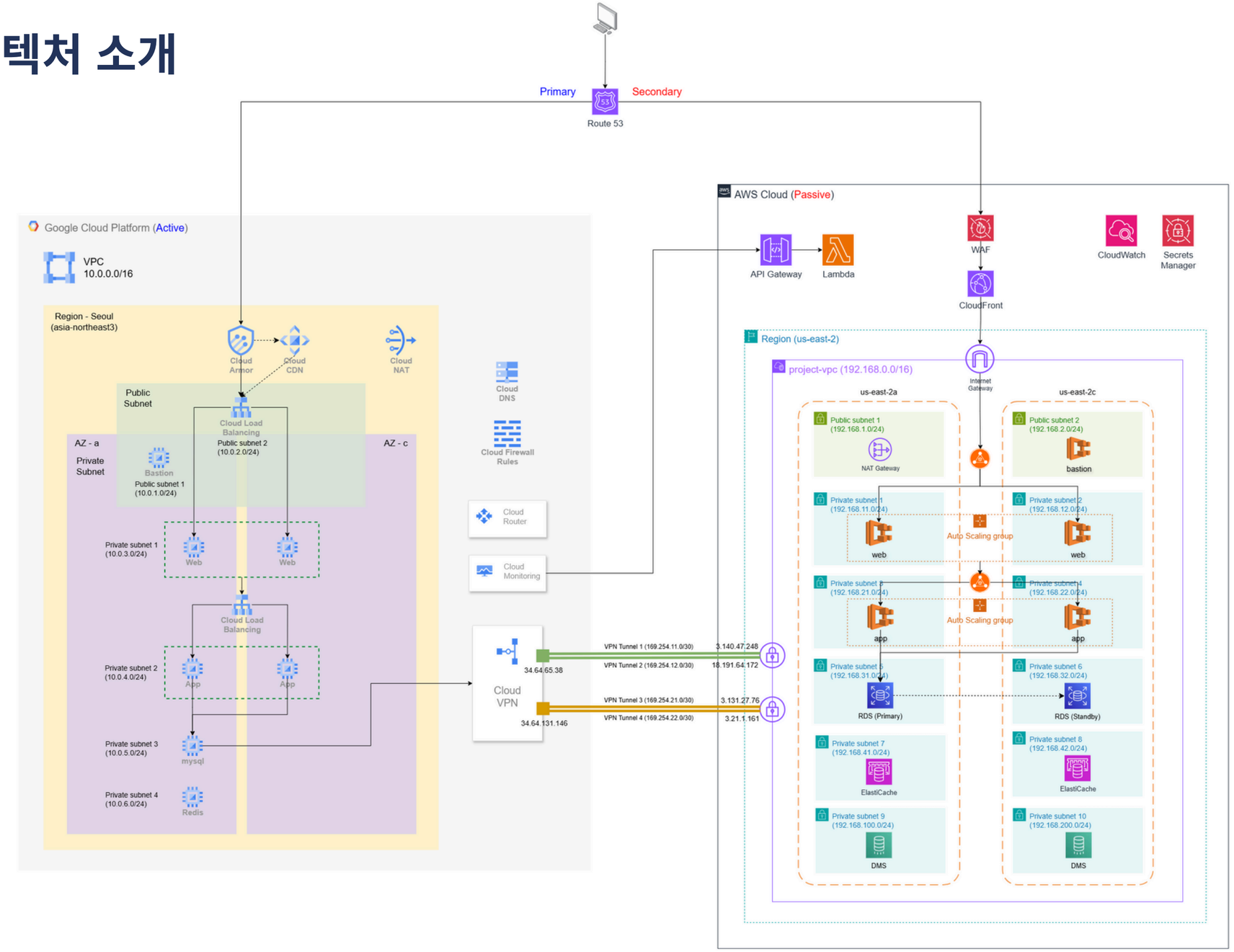
인프라 구성

3-Tier 아키텍처 : Web, App, Database 계층 분리
Auto Scaling : 자동 확장/축소 시스템
고가용성 DB : RDS Multi-AZ, ElastiCache
보안 관리 : Secrets Manager, IAM 역할 기반 접근 제어

DR 구성

Active-Passive 구성으로 비용 최적화
Route 53 Failover를 통한 자동 트래픽 전환
AWS DMS를 활용한 데이터 연속성 유지

02 아키텍처 소개


Primary
Secondary
Route 53
Google Cloud Platform (Active)
VPC
10.0.0.0/16
Region - Seoul
(asia-northeast3)
Cloud Armor
Cloud CDN
Cloud NAT
Public Subnet
Cloud Load Balancing
Public subnet 2
(10.0.2.0/24)
AZ - a
AZ - c
Private Subnet
Bastion
Public subnet 1
(10.0.1.0/24)
Private subnet 1
(10.0.3.0/24)
Web
Web
Cloud Load Balancing
Private subnet 2
(10.0.4.0/24)
App
App
Private subnet 3
(10.0.5.0/24)
mysql
Private subnet 4
(10.0.6.0/24)
Redis
Cloud DNS
Cloud Firewall Rules
Cloud Router
Cloud Monitoring
Cloud VPN
34.64.65.38
34.64.131.146
VPN Tunnel 1 (169.254.11.0/30)
VPN Tunnel 2 (169.254.12.0/30)
VPN Tunnel 3 (169.254.21.0/30)
VPN Tunnel 4 (169.254.22.0/30)
3.140.47.248
18.191.64.172
3.131.27.76
3.21.1.161
AWS Cloud (Passive)
API Gateway
Lambda
WAF
CloudFront
CloudWatch
Secrets Manager
Region (us-east-2)
project-vpc (192.168.0.0/16)
Internet Gateway
us-east-2a
us-east-2c
Public subnet 1
(192.168.1.0/24)
NAT Gateway
Public subnet 2
(192.168.2.0/24)
bastion
Private subnet 1
(192.168.11.0/24)
web
Auto Scaling group
Private subnet 2
(192.168.12.0/24)
web
Private subnet 3
(192.168.21.0/24)
app
Auto Scaling group
Private subnet 4
(192.168.22.0/24)
app
Private subnet 5
(192.168.31.0/24)
RDS (Primary)
Private subnet 6
(192.168.32.0/24)
RDS (Standby)
Private subnet 7
(192.168.41.0/24)
ElastiCache
Private subnet 8
(192.168.42.0/24)
ElastiCache
Private subnet 9
(192.168.100.0/24)
DMS
Private subnet 10
(192.168.200.0/24)
DMS

02 아키텍처 소개

레코드 세부 정보

레코드 편집

레코드 이름
www.choiyunha.com

레코드 유형
A

값
35.201.106.123

별칭
아니요

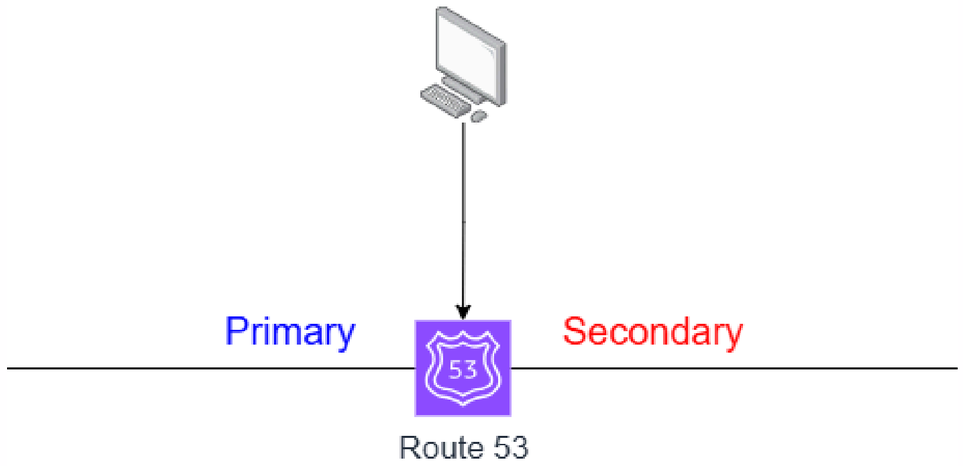
TTL(초)
300

라우팅 정책
Failover

장애 조치 레코드 유형
기본

상태 확인 ID
4fc82202-8b78-4473-851b-0c9688d82

레코드 ID
gcp-primary



레코드 세부 정보

레코드 편집

레코드 이름
www.choiyunha.com

레코드 유형
A

값
d36vqg3xcdb804.cloudfront.net.

별칭
예

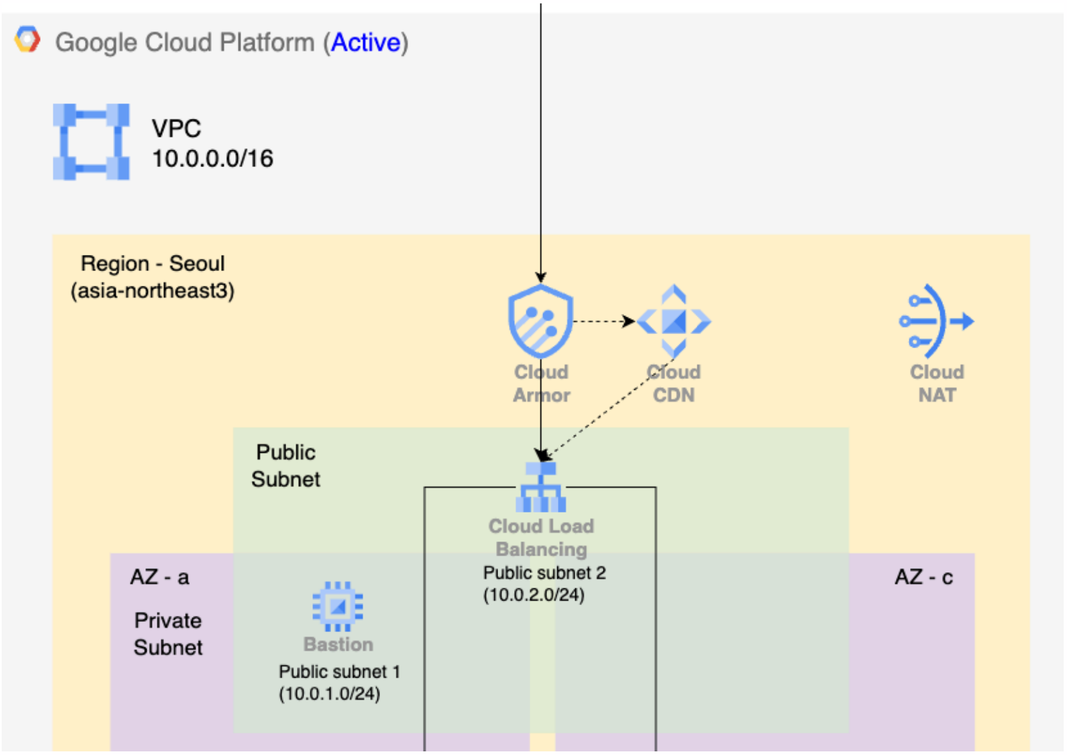
TTL(초)
-

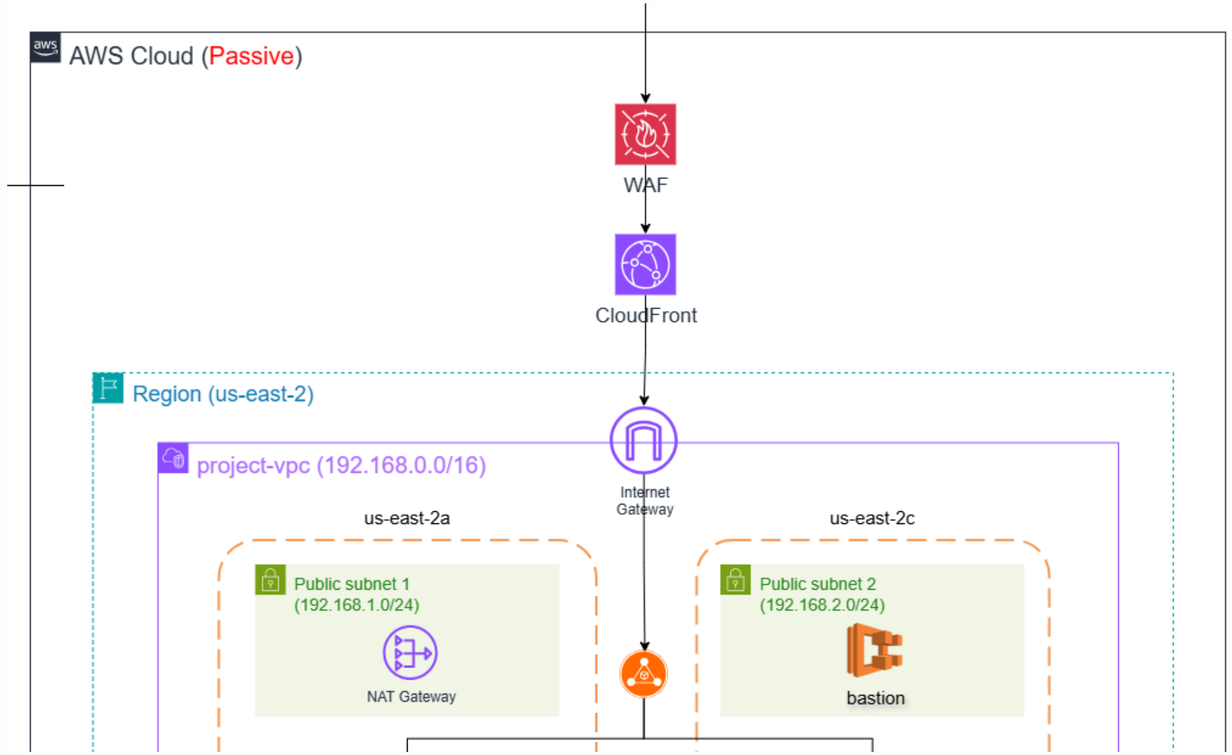
라우팅 정책
Failover

장애 조치 레코드 유형
보조

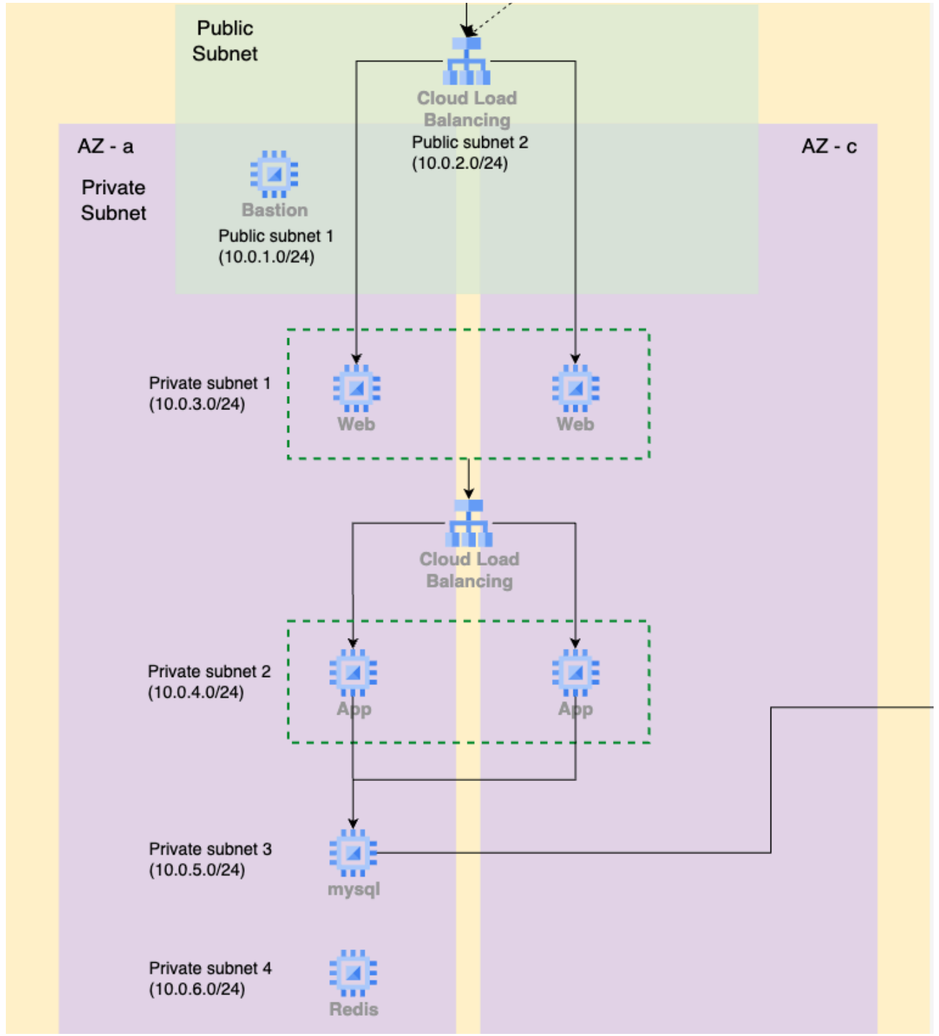
레코드 ID
aws-secondary

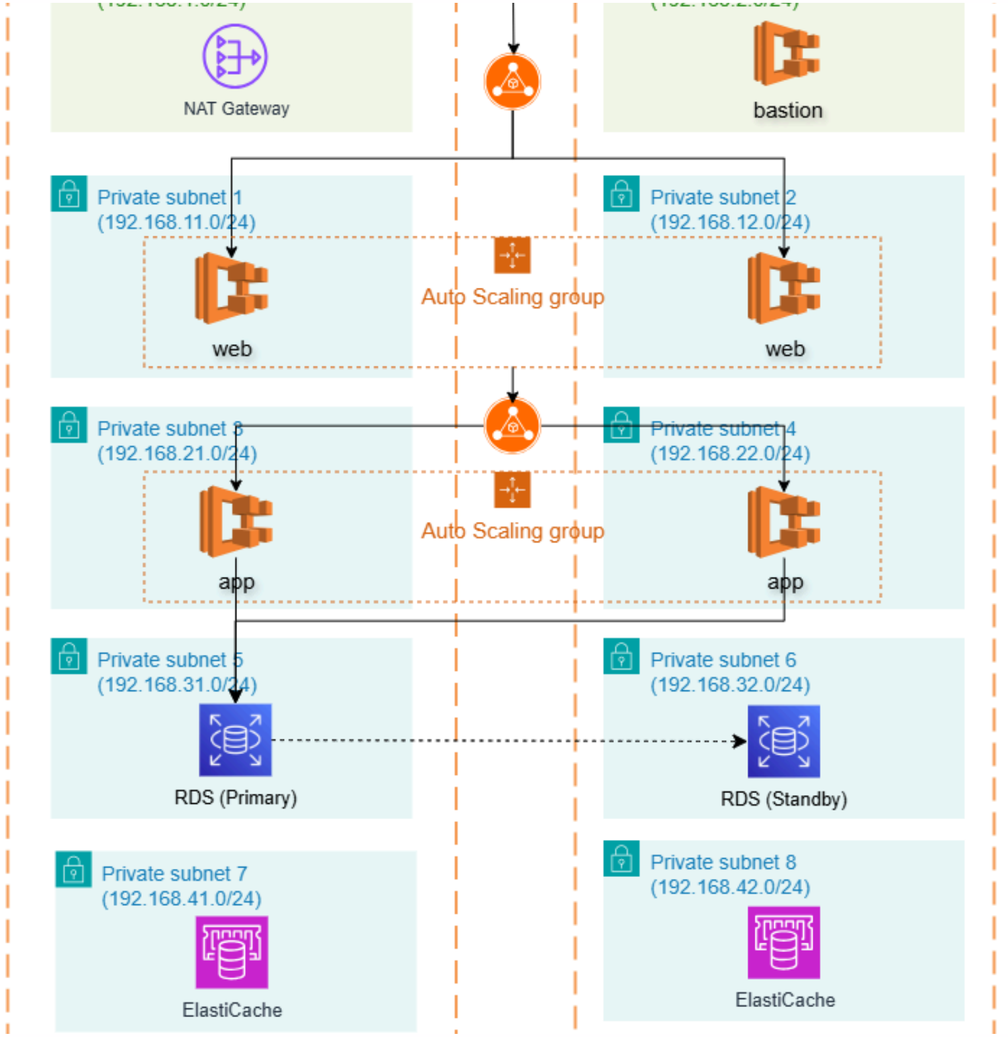
02 아키텍처 소개


Google Cloud Platform (Active)
VPC
10.0.0.0/16
Region - Seoul
(asia-northeast3)
Cloud
Armor
Cloud
CDN
Cloud
NAT
Public
Subnet
Cloud Load
Balancing
Public subnet 2
(10.0.2.0/24)
AZ - a
Private
Subnet
Bastion
Public subnet 1
(10.0.1.0/24)
AZ - c

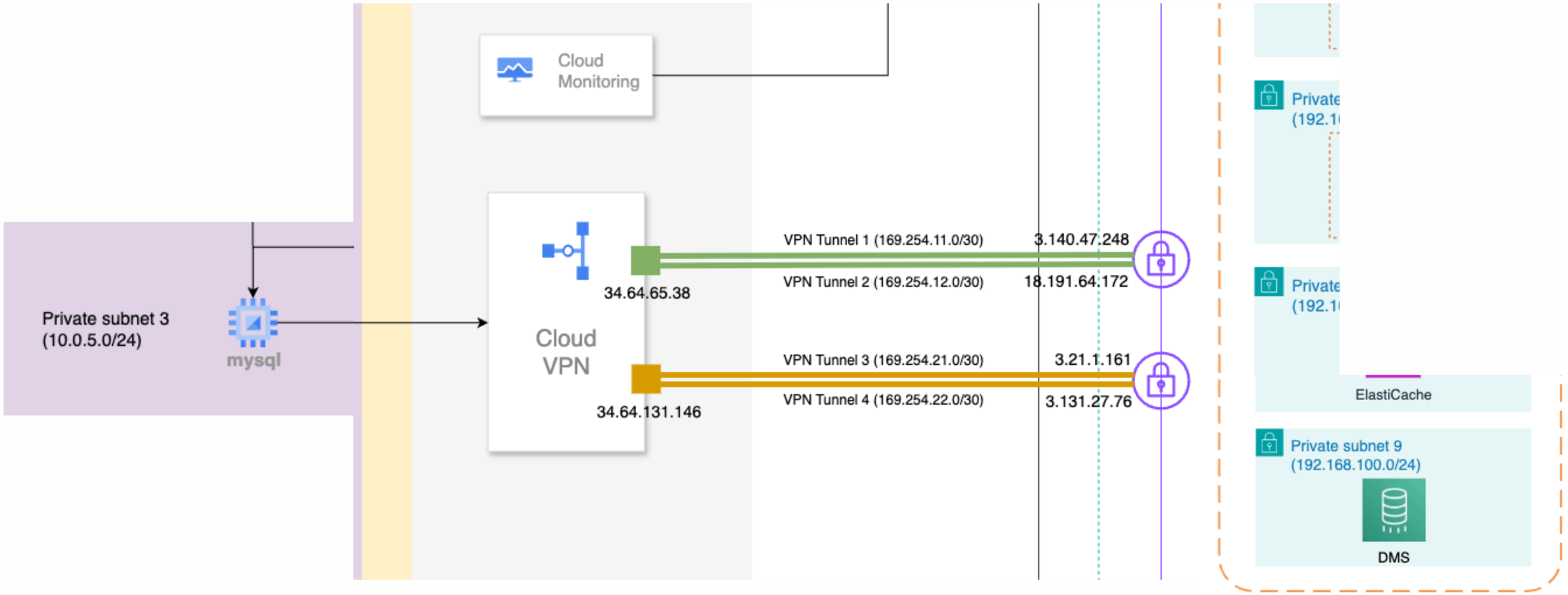

AWS Cloud (Passive)
WAF
CloudFront
Region (us-east-2)
project-vpc (192.168.0.0/16)
Internet
Gateway
us-east-2a
us-east-2c
Public subnet 1
(192.168.1.0/24)
NAT Gateway
Public subnet 2
(192.168.2.0/24)
bastion

02 아키텍처 소개

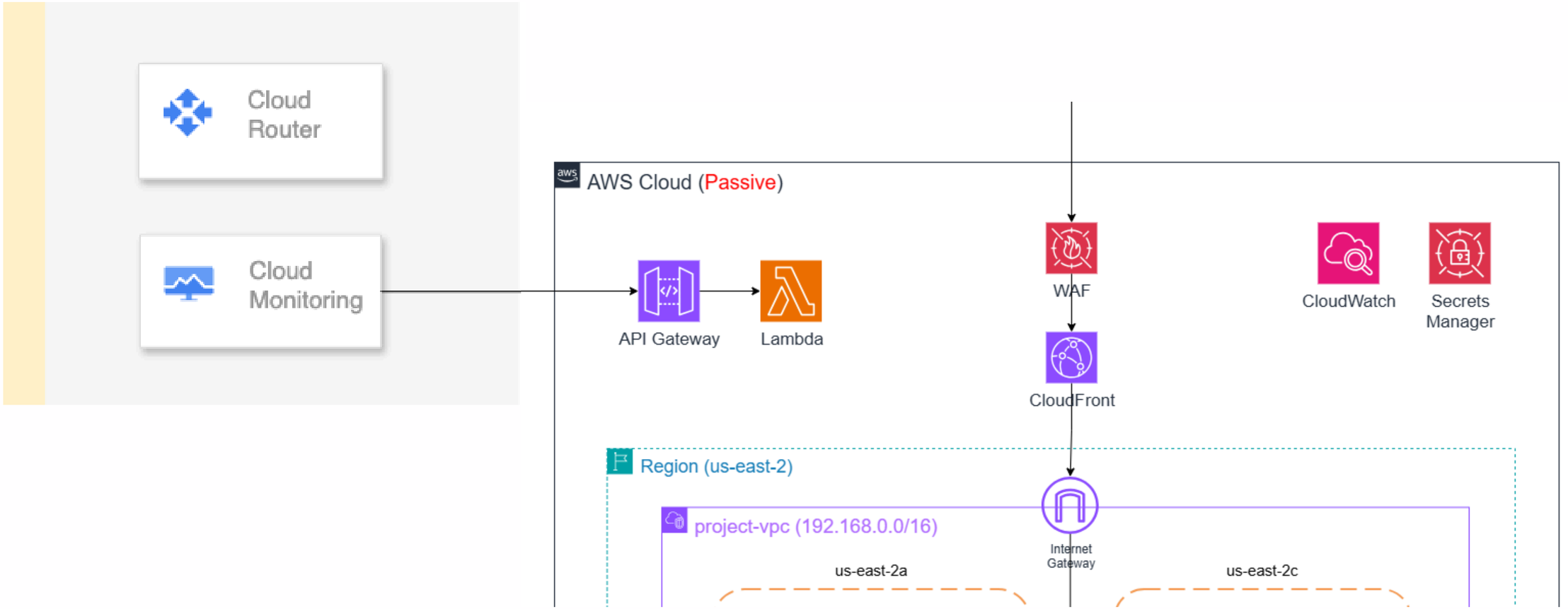

Public Subnet
Cloud Load Balancing
Public subnet 2 (10.0.2.0/24)
AZ - a
AZ - c
Private Subnet
Bastion
Public subnet 1 (10.0.1.0/24)
Private subnet 1 (10.0.3.0/24)
Web
Web
Cloud Load Balancing
Private subnet 2 (10.0.4.0/24)
App
App
Private subnet 3 (10.0.5.0/24)
mysql
Private subnet 4 (10.0.6.0/24)
Redis


NAT Gateway
bastion
Private subnet 1 (192.168.11.0/24)
Private subnet 2 (192.168.12.0/24)
Auto Scaling group
web
web
Private subnet 3 (192.168.21.0/24)
Private subnet 4 (192.168.22.0/24)
Auto Scaling group
app
app
Private subnet 5 (192.168.31.0/24)
Private subnet 6 (192.168.32.0/24)
RDS (Primary)
RDS (Standby)
Private subnet 7 (192.168.41.0/24)
Private subnet 8 (192.168.42.0/24)
ElastiCache
ElastiCache

02 아키텍처 소개


Cloud Monitoring
Private subnet 3
(10.0.5.0/24)
mysql
Cloud VPN
34.64.65.38
34.64.131.146
VPN Tunnel 1 (169.254.11.0/30)
VPN Tunnel 2 (169.254.12.0/30)
VPN Tunnel 3 (169.254.21.0/30)
VPN Tunnel 4 (169.254.22.0/30)
3.140.47.248
18.191.64.172
3.21.1.161
3.131.27.76
Private
(192.1
Private
(192.1
ElastiCache
Private subnet 9
(192.168.100.0/24)
DMS

02 아키텍처 소개


Cloud Router
Cloud Monitoring
AWS Cloud (Passive)
API Gateway
Lambda
WAF
CloudFront
CloudWatch
Secrets Manager
Region (us-east-2)
project-vpc (192.168.0.0/16)
Internet Gateway
us-east-2a
us-east-2c

03 시연 영상

https://youtu.be/lxeubdhbO1k?si=wts_6042jry7hWL1

04 Trouble Shooting

기존 AWS DRS => AWS Lambda 선택 이유

AWS DRS의 한계점

- **동일 이미지 중복 복제 :** Auto Scaling으로 생성된 인스턴스 모두 개별 복제
- **불필요한 리소스 낭비 :** 같은 애플리케이션 코드를 여러 번 복제
- **높은 비용 :** 지속적인 볼륨 동기화로 인한 네트워크 비용 증가

➡ **Lambda 사용 : Auto Scaling Scale out**

04 Trouble Shooting

기존 AWS DRS => AWS Lambda 선택 이유

	AWS DRS	Lambda
복제 대상	VM 디스크 × 인스턴스 수	함수 코드 (경량)
비용	지속적 동기화	실행 시에만 과금
복구 시간	중간 (부팅 필요)	빠름
관리 복잡도	높음	낮음 (서버리스)

04 Trouble Shooting

AWS만 CloudFront 적용한 이유

GCP Cloud CDN vs Cloud Storage Direct
응답 속도 비교 테스트

```
(venv) choiyunha@choeyunhas-MacBook-Pro flask_app-main
nloads/flask_app-main/venv/bin/python /Users/choiyunha/
in/cdn_test.py
===== CDN =====
CDN | Request 1: 404 | X-Cache: N/A | Time: 0.058 s
CDN | Request 2: 404 | X-Cache: N/A | Time: 0.054 s
CDN | Request 3: 404 | X-Cache: N/A | Time: 0.041 s

===== ORIGIN (GCS) =====
ORIGIN | Request 1: 403 | X-Cache: N/A | Time: 0.305 s
ORIGIN | Request 2: 403 | X-Cache: N/A | Time: 0.300 s
ORIGIN | Request 3: 403 | X-Cache: N/A | Time: 0.325 s
```

AWS CloudFront vs WEB ALB Direct
응답 속도 비교 테스트

```
CloudFront 성능 테스트:
  Run 1: 1182.10 ms (HTTP 200)
  Run 2: 482.07 ms (HTTP 200)
  Run 3: 473.81 ms (HTTP 200)
  Run 4: 472.86 ms (HTTP 200)
  Run 5: 467.16 ms (HTTP 200)
  평균: 615.60 ms
  최소/최대: 467.16 / 1182.10 ms
  측정 성공: 5/5회

ALB Direct 성능 테스트:
  Run 1: 1273.96 ms (HTTP 200)
  Run 2: 5407.72 ms (HTTP 200)
  Run 3: 8385.45 ms (HTTP 200)
  Run 4: 2770.48 ms (HTTP 200)
  Run 5: 1406.56 ms (HTTP 200)
  평균: 3848.84 ms
  최소/최대: 1273.96 / 8385.45 ms
  측정 성공: 5/5회

==========
=== 최종 결과 요약 ===
CloudFront (HTTPS): 615.60 ms
ALB Direct (HTTP): 3848.84 ms
CloudFront가 84.0% 더 빠름
CDN 캐싱 효과로 3233.24ms 단축
```

05 활용방안 및 기대효과

적용 대상

- 전자상거래, 금융, 헬스케어, 게임 등 서비스 중단 민감 산업
- 중소기업 → 대기업 단계별 확장 가능한 아키텍처

도입 전략

- **1단계 :** Active-Passive DR 구성으로 시작
- **2단계 :** 무중단 서비스 필요 시 Active-Active 확장

05 활용방안 및 기대효과

비즈니스 연속성

- RTO 15분 이내, RPO 1분 이내 목표 달성
- 자동 장애 전환으로 서비스 중단 최소화

비용 최적화

- Active-Passive 구성으로 DR 리소스 최소화
- Auto Scaling으로 리소스 효율성 극대화
- 기존 대비 비용 절감

보안 강화

- 이중 보안 체계 (Cloud Armor + AWS WAF)
- 통합 권한 관리 (Secrets Manager + IAM)

05 활용방안 및 기대효과

성능 향상

- CDN 활용으로 글로벌 콘텐츠 전송 최적화
- 지연 시간 40-60% 단축

운영 효율성

- 통합 모니터링 (Cloud Monitoring + CloudWatch)
- 장애 감지 시간 70% 단축
- 사전 예방적 운영 가능

벤더 종속성 탈피

- 멀티 클라우드로 단일 업체 의존도 감소
- 운영 유연성 확보 및 서비스 안정성 동시 확보

06 팀 구성 및 역할

이충민

코드 담당
AWS 3 tier 구축
DR
CloudFront, WAF

최윤하

팀장
AWS 3 tier 구축
GCP CDN, Armor
VPN 터널링

백지영

GCP 3 tier 구축
GCP Armor
GCP Monitoring
AWS Lambda

한승규

GCP 3 tier 구축
GCP Monitoring
Lambda 및 DR
AWS DMS

Q&A