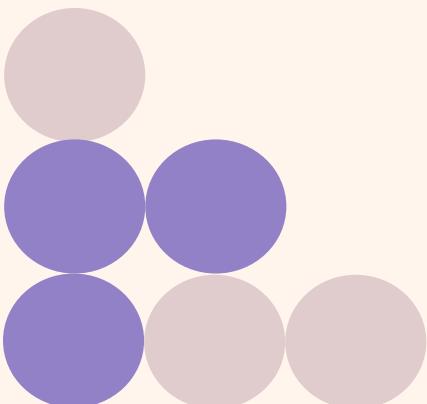


# MEDIAHUB

글로벌 미디어 콘텐츠 제공을 위한  
하이브리드 클라우드 솔루션



이수민 | lucyi030920@gmail.com

# 목차



## 프로젝트 개요

- 01-1 프로젝트 배경 및 필요성
- 01-2 프로젝트 구성 목표
- 01-3 프로젝트 기대 효과

## 프로젝트 구성

- 02-1 협업 도구 및 소프트웨어
- 02-2 주요 서비스
- 02-3 프로젝트 추진 일정 및 계획
- 02-4 프로젝트 구성도

## 프로젝트 구현

- 03-1 IAM
- 03-2 CloudFormation
- 03-3 Transit Gateway
- 03-4 Site to Site VPN
- 03-5 ACM
- 03-6 RDS
- 03-7 S3
- 03-8 Lambda
- 03-9 API Gateway
- 03-10 Amplify
- 03-11 WAF
- 03-12 Global Accelerator
- 03-13 Route 53
- 03-14 CloudWatch & SNS

## 프로젝트 결과

- 04-1 프로젝트 성과
- 04-2 보완점 및 향후 계획
- 04-3 결론

## Q & A

CHAPTER 01

# 프로젝트 개요

01-1 프로젝트 배경 및 필요성

01-2 프로젝트 구성 목표

01-3 프로젝트 기대 효과



비디오 저장 스토리지를 운영하는 글로벌 미디어 기업 A사

## Current Status

글로벌 미디어 기업 A사의 온프레미스 기반  
비디오 스토리지 서비스 운영 중 다양한 문제점 발생

## Key Challenges

- 높은 대기 시간과 비효율적인 콘텐츠 전송
- 보안과 신뢰성 부족
- 유연한 확장성과 비용절감 문제
- 다양한 언어와 현지화 요구



비디오 저장 스토리지를 운영하는 글로벌 미디어 기업 A사

### Why Hybrid Cloud?

- 글로벌 사용자에게 최적화된 서비스 제공
- 안전하고 확장 가능한 인프라 구축
- 효율적인 리소스 관리와 비용 최적화

### Expected Benefits

- 지역별 최적화된 콘텐츠 제공
- 보안성과 확장성 확보
- 운영 효율성 증대



### 하이브리드 클라우드 구축

온프레미스와 AWS 클라우드 환경의 원활한 통합을 통해 안정적이고 확장 가능한 인프라 구현



### 글로벌 서비스 성능 최적화

Global Accelerator를 활용한 지역별 최적화된 라우팅으로 전 세계 사용자에게 빠른 응답 속도 제공



### 자동 확장성 및 비용 효율성 확보

트래픽 패턴에 따른 자동 확장/축소로 리소스 최적화 및 비용 효율성 달성



### 보안 및 데이터 가용성 강화

멀티 리전 데이터 복제와 VPN 연결로 데이터 보안 및 가용성 보장



### 데이터 일관성 및 고가용성 확보

하이브리드 클라우드 환경에서의 실시간 데이터 동기화로 글로벌 서비스의 안정성 보장



### 글로벌 서비스 성능 향상

글로벌 사용자에게 지연 시간 최소화 및 안정적인 서비스 제공으로 만족도 향상



### 운영 효율성 향상

자동화된 리소스 관리와 모니터링으로 운영 비용 절감 및 관리 효율성 증대



### 보안 및 규정 준수 강화

엄격한 보안 정책 적용과 다중 계층 보안으로 글로벌 규정 준수 및 데이터 보호 강화

CHAPTER 02

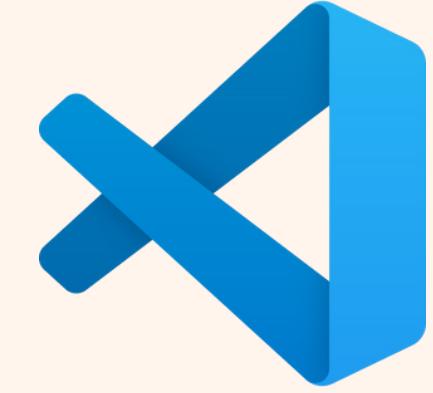
# 프로젝트 구성

02-1 협업 도구 및 소프트웨어

02-2 주요 서비스

02-3 프로젝트 추진 일정 및 계획

02-4 프로젝트 구성도

**Visual Studio Code**

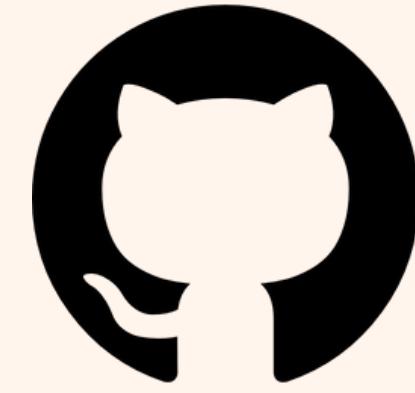
코드 작성 및 디버깅을 효율적으로 수행  
할 수 있는 경량화된 통합 개발 환경

**Claude**

복잡한 문제 해결과 아이디어 제시를  
위한 인공지능 기반 대화형 도구

**Notion**

프로젝트 계획, 문서화, 협업을 한 곳에  
서 관리할 수 있는 생산성 도구

**GitHub**

소스 코드 관리 및 협업을 위한 효율적인  
개발 워크플로우 플랫폼



### Site-to-Site VPN

온프레미스 네트워크와 AWS 클라우드를  
안전하게 연결하는 VPN 터널 제공



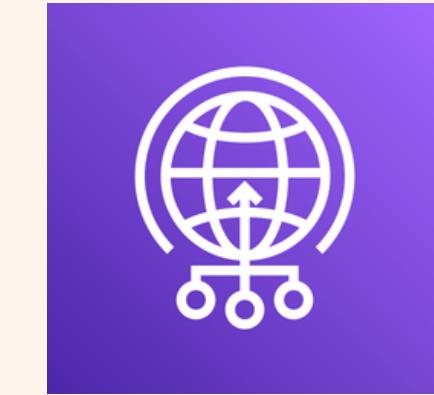
### Lambda

서버리스 컴퓨팅 서비스로 코드 실행과  
자동 확장을 관리하며, 사용한 컴퓨팅  
시간에 대해서만 비용 지불



### Amplify

풀스택 웹/모바일 애플리케이션의 빠른  
구축과 배포를 지원하는 개발 플랫폼으로  
CI/CD 파이프라인 자동화 제공

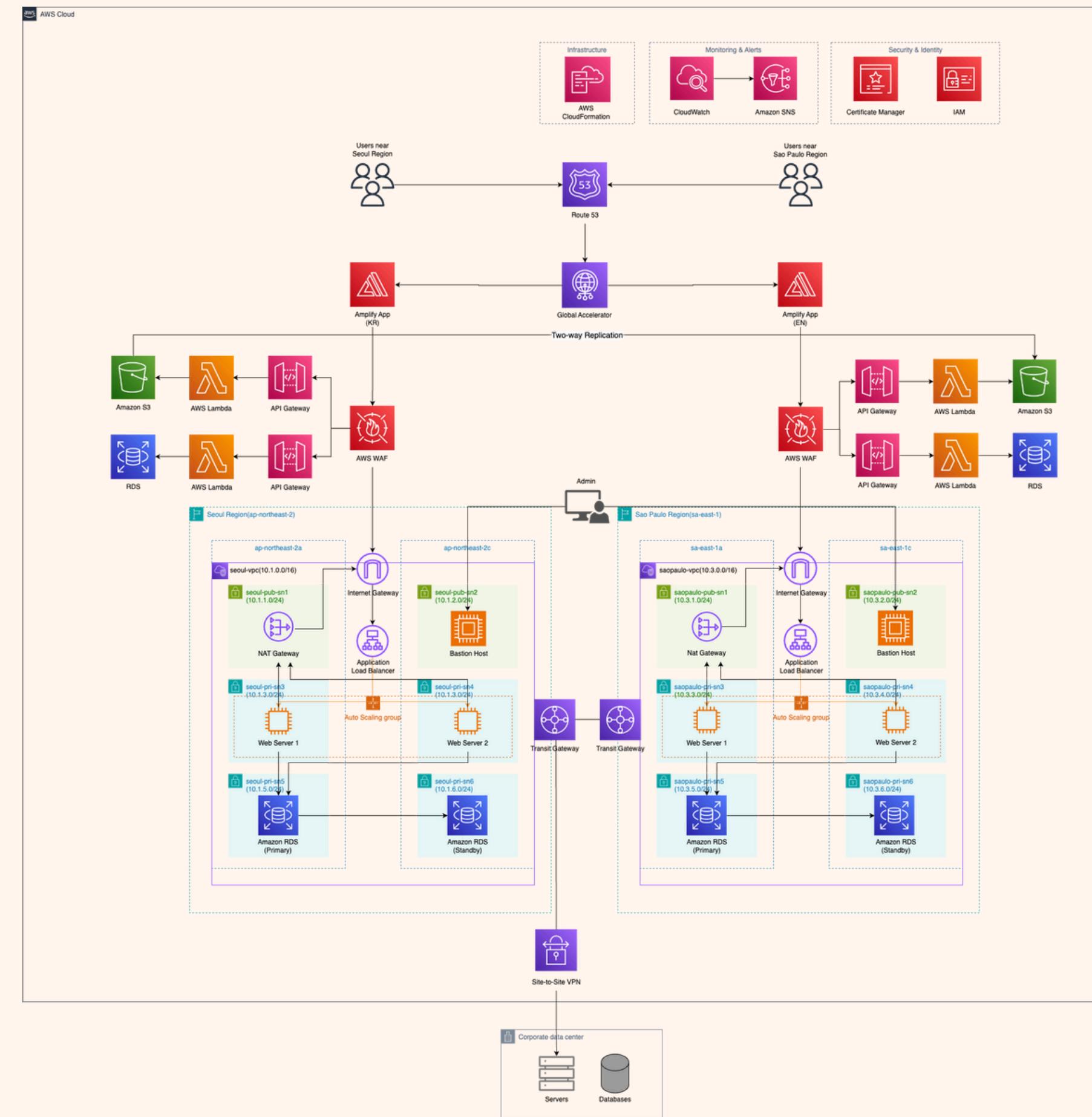


### Global Accelerator

전 세계 사용자에게 빠르고 안정적인  
네트워크 성능을 제공하기 위해 글로벌  
네트워크를 통해 애플리케이션의 트래픽 최적화



## 프로젝트 구성도



CHAPTER 03

# 프로젝트 구현

## IAM 사용자 생성 및 권한 부여

**사용자에게 콘솔 액세스 권한을 제공하고 있습니까?**

사용자 유형

- Identity Center에서 사용자 지정 - 권장  
Identity Center를 사용하여 사용자에게 콘솔 액세스 권한을 제공하는 것이 좋습니다. Identity Center를 사용하면 AWS 계정 및 클라우드 애플리케이션에 대한 사용자 액세스를 중앙에서 관리할 수 있습니다.
- IAM 사용자를 생성하고 싶음  
액세스 키, AWS CodeCommit이나 Amazon Keyspaces에 대한 서비스별 보안 인증 정보 또는 비상 계정 액세스를 위한 백업 보안 인증 정보를 통해 프로그래밍 방식 액세스를 활성화해야 하는 경우에만 IAM 사용자를 생성하는 것이 좋습니다.

**콘솔 암호**

- 자동 생성된 암호  
사용자를 생성한 후 암호를 볼 수 있습니다.
- 사용자 지정 암호  
사용자의 사용자 지정 암호를 입력합니다.

8자 이상이어야 합니다.  
다음 문자 유형 중 세 가지 이상을 조합하여 포함해야 합니다. 대문자(A~Z), 소문자(a~z), 숫자(0~9), 기호 ! @ # \$ % ^ & \* ( ) \_ + -(하이픈) = [ ] { } | '

암호 표시

사용자는 다음 로그인 시 새 암호를 생성해야 합니다 - 권장  
사용자는 IAMUserChangePassword 정책을 자동으로 가져와 암호를 변경할 수 있도록 허용합니다.

**이 IAM 사용자를 생성한 후 액세스 키 또는 AWS CodeCommit이나 Amazon Keyspaces에 대한 서비스별 보안 인증 정보를 통해 프로그래밍 방**

**Identity and Access Management(IAM)**

IAM > 사용자

**사용자 (1) 정보**

IAM 사용자는 계정에서 AWS와 상호 작용하는 데 사용되는 장기 자격 증명을 가진 자격 증명입니다.

검색	사용자 이름	경로	그룹	마지막 활동	MFA	암호 수명	콘솔
	Lucy	/	0	-	-	-	-

**대시보드**

**액세스 관리**

- 사용자 그룹
- 사용자**
- 역할
- 정책
- 자격 증명 공급자
- 계정 설정

**보고서 액세스**

- Access Analyzer
- 외부 액세스
- 미사용 액세스
- 분석기 설정

## 각 리전 기본 인프라 배포

The screenshot shows the AWS CloudFormation console in the Seoul region. The left sidebar includes options for Stacks, StackSets, and Infrastructure Composer. The main area is titled 'CloudFormation > 스택' (Stacks) and shows a table of stacks. The table has columns for '스택 이름' (Stack Name), '상태' (Status), '생성 시간' (Creation Time), and '설명' (Description). One stack is listed: 'seoul-infra' with status 'CREATE\_COMPLETE', created on '2024-11-04 09:44:06 UTC+0900'. Buttons for 'Create', 'Delete', 'Update', 'Stack Operations', and 'Stack Creation' are visible at the top.

This screenshot shows the AWS CloudFormation console in the Paris region. The layout is identical to the Seoul screenshot, with the same sidebar and main 'Stacks' table. The table shows one stack named 'sao-infra' in a 'CREATE\_COMPLETE' state, created on '2024-11-04 09:42:16 UTC+0900'. The region dropdown at the top right is set to '상파울루' (Sao Paulo).

03-3

# Transit Gateway

## 각 리전 Transit Gateway 생성

서울 리전

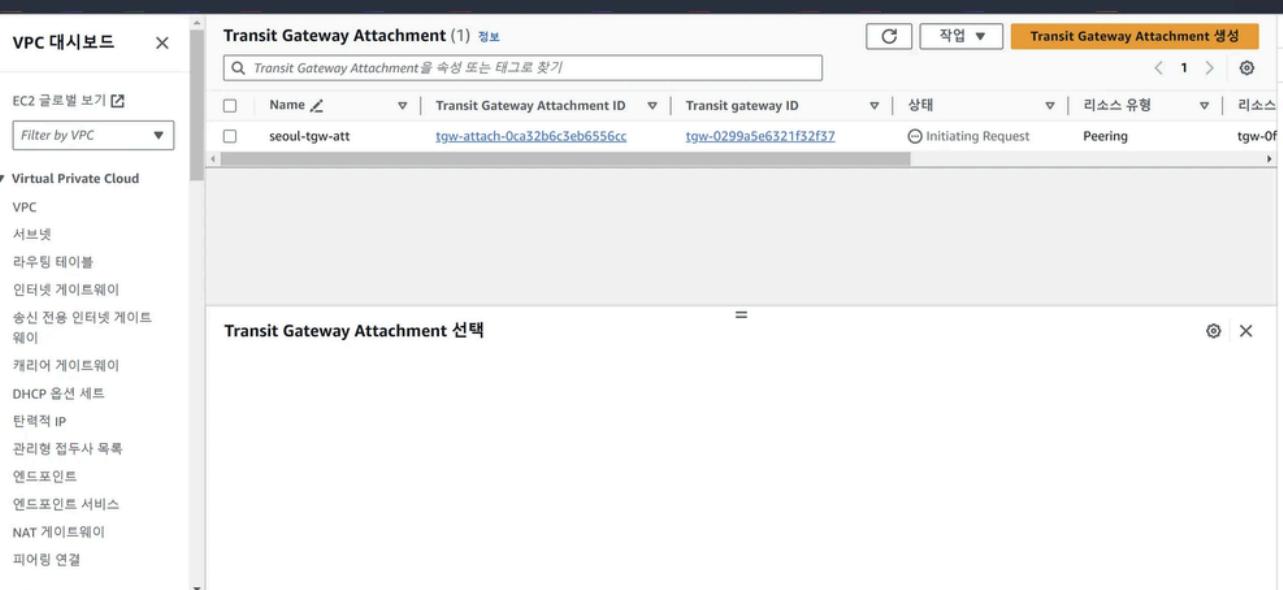
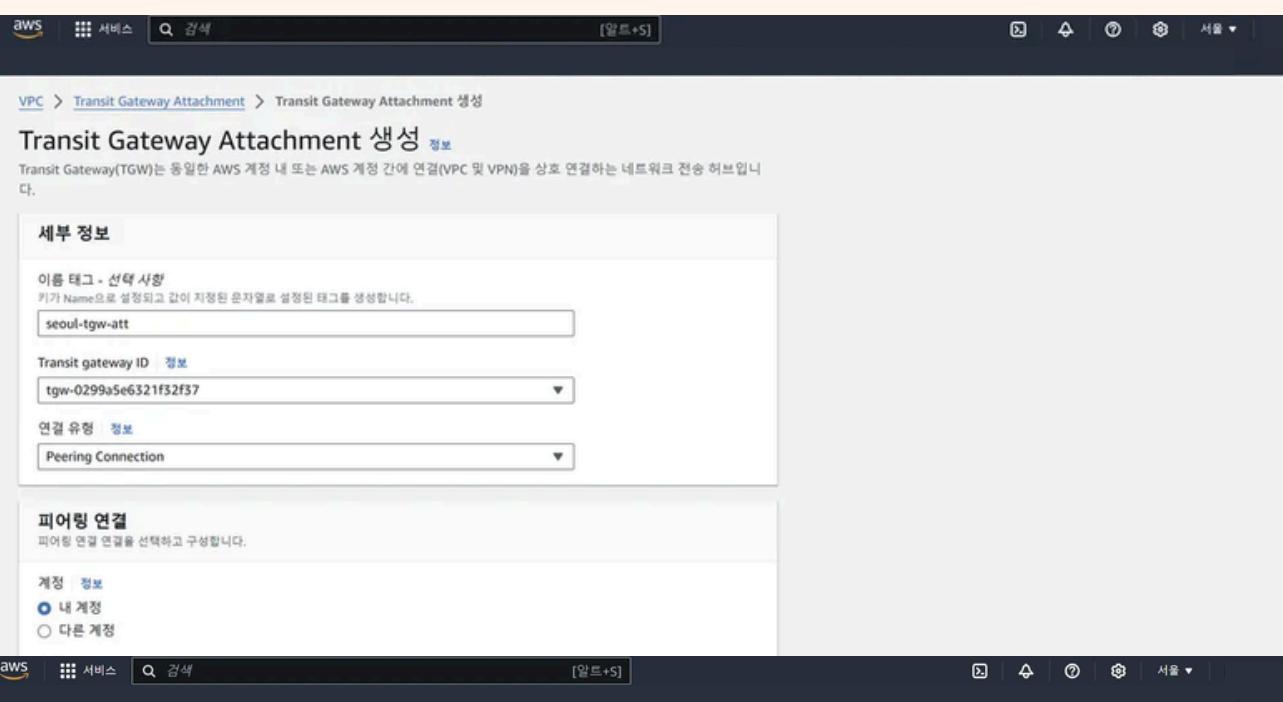
The screenshot shows the 'Transit Gateway 생성' (Create Transit Gateway) wizard in the AWS VPC console. In the '세부 정보 - 선택 사항' (Advanced Information - Selection Options) step, the '이름 태그' (Name Tag) is set to 'seoul-tgw'. In the '설명' (Description) field, it says '나중에 식별하는 데 도움이 되도록 Transit Gateway에 대한 설명을 설정합니다.' (Set a description for the Transit Gateway to help identify it later.) Below this, the 'Transit Gateway 구성' (Transit Gateway Configuration) section includes fields for 'Amazon 측 ASN(자율 시스템 번호)' (ASN) set to '64512', and checkboxes for 'DNS 지원' (DNS Support), '보안 그룹 참조 지원' (Security Group Reference Support), and 'VPN ECMP 지원' (VPN ECMP Support). The final step, 'Transit Gateway 선택' (Select Transit Gateway), shows the newly created 'seoul-tgw' entry in the list.

상파울루 리전

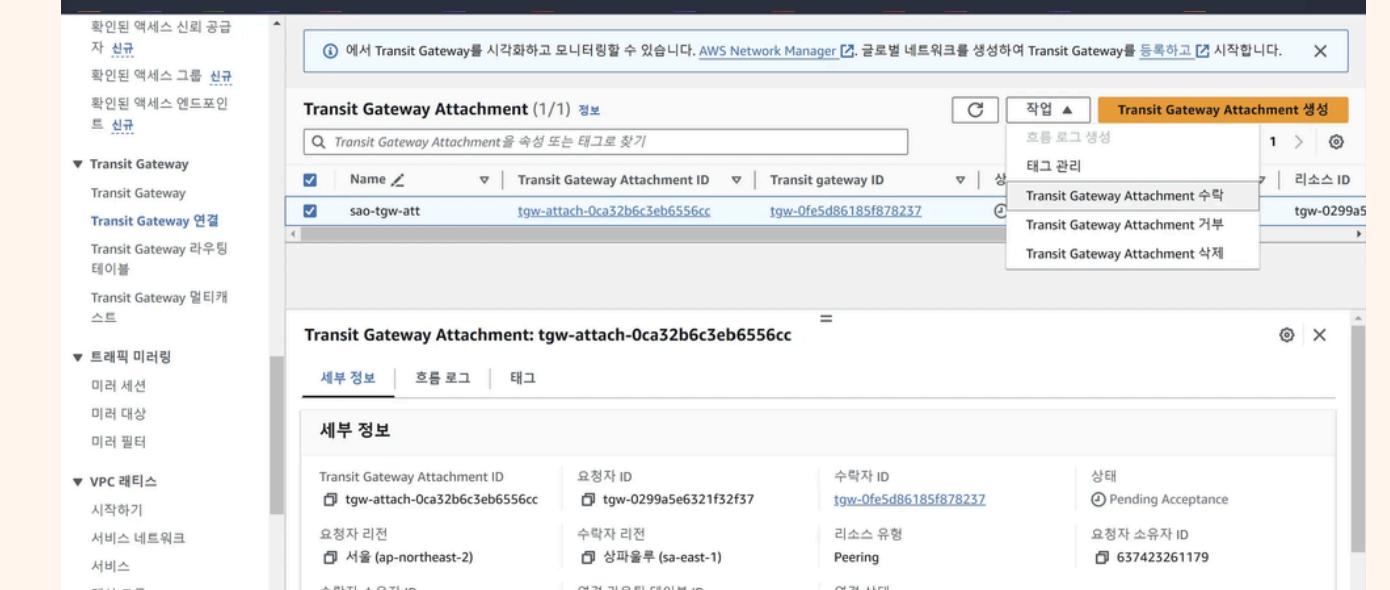
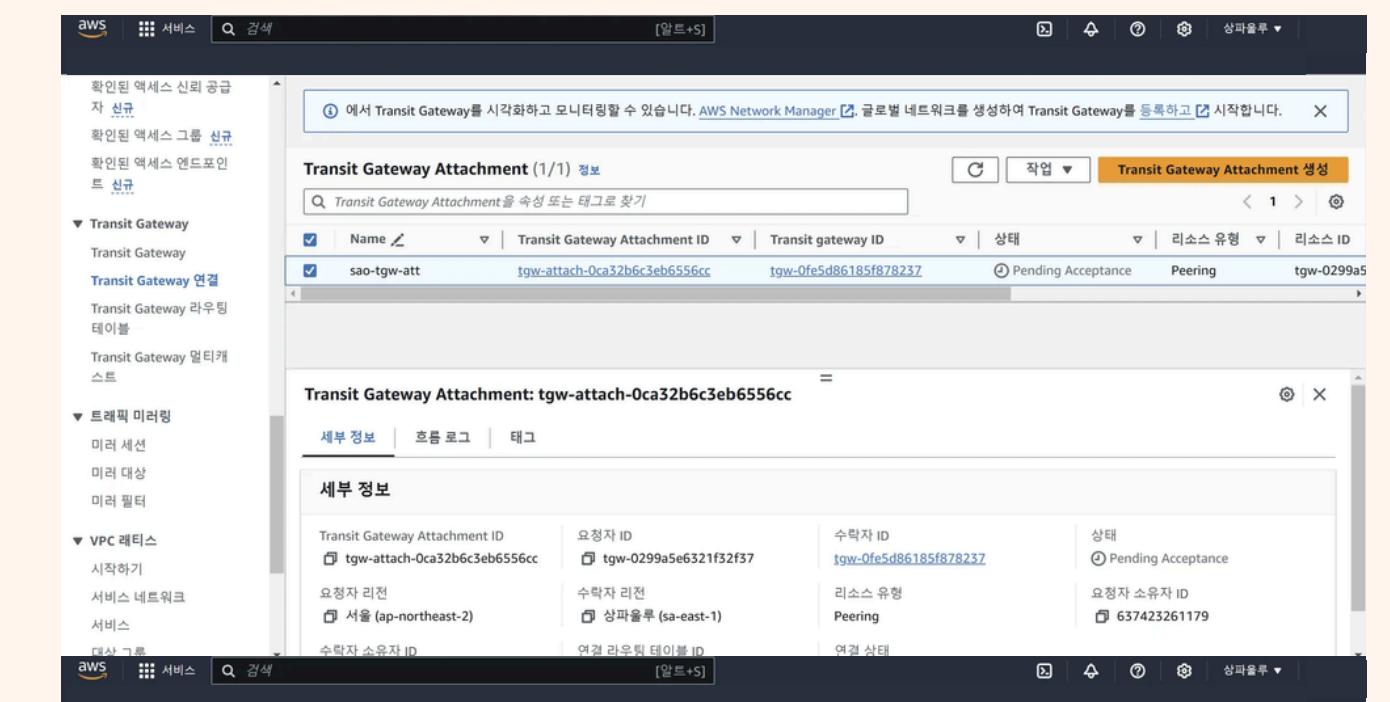
The screenshot shows the 'Transit Gateway 생성' (Create Transit Gateway) wizard in the AWS VPC console for the São Paulo region. It follows a similar structure to the Seoul region screen. The '세부 정보 - 선택 사항' step has '이름 태그' set to 'sao-tgw'. The '설명' field contains '나중에 식별하는 데 도움이 되도록 Transit Gateway에 대한 설명을 설정합니다.' The 'Transit Gateway 구성' section includes 'Amazon 측 ASN' set to '64513', and checkboxes for 'DNS 지원', '보안 그룹 참조 지원', and 'VPN ECMP 지원'. The final step, 'Transit Gateway 선택' (Select Transit Gateway), shows the newly created 'sao-tgw' entry in the list. A note at the top right of the page says '에서 Transit Gateway를 시작하고 모니터링할 수 있습니다. AWS Network Manager, 글로벌 네트워크를 생성하여 Transit Gateway를 등록하고 시작합니다.' (You can start and monitor the Transit Gateway. Create a global network and register it to start the Transit Gateway.)

## 각 리전 Transit Gateway attachment 생성 후 연결

서울 리전



상파울루 리전



# Transit Gateway

각 리전 Transit Gateway 구성 완료

**Screenshot 1: Seoul Region (서울)**

The screenshot shows the 'Transit Gateway Attachment' list in the Seoul region. A single attachment named 'seoul-tgw-att' is listed, showing it is available and associated with a Peering connection.

Name	Transit Gateway Attachment ID	Transit gateway ID	상태	리소스 유형	리소스 ID
seoul-tgw-att	tgw-attach-0ca32b6c3eb6556cc	tgw-0299a5e6321f32f37	Available	Peering	tgw-0fe5d86185f878237

**Screenshot 2: São Paulo Region (상파울루)**

The screenshot shows the 'Transit Gateway Attachment' list in the São Paulo region. A single attachment named 'sao-tgw-att' is listed, showing it is available and associated with a Peering connection.

Name	Transit Gateway Attachment ID	Transit gateway ID	상태	리소스 유형	리소스 ID
sao-tgw-att	tgw-attach-0ca32b6c3eb6556cc	tgw-0fe5d86185f878237	Available	Peering	tgw-0299a5e6321f32f37

**Details for Attachment: tgw-attach-0ca32b6c3eb6556cc**

The details page for the attachment shows the following information:

세부 정보	요청자 ID	수락자 ID	상태
Transit Gateway Attachment ID	tgw-attach-0ca32b6c3eb6556cc	tgw-0299a5e6321f32f37	Pending Acceptance
요청자 리전	서울 (ap-northeast-2)	수락자 리전	리소스 유형
서비스 네트워크	상파울루 (sa-east-1)	Peering	요청자 소유자 ID
서비스		연결 라우팅 테이블 ID	637423261179
대상 그룹			

## 서울 리전 오프레미스 환경 설정

```
[root@ip-10-10-1-204 ~]# yum install -y quagga
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package quagga.x86_64 0:0.99.22.4-5.amzn2 will be installed
--> Processing Dependency: net-snmp for package: quagga-0.99.22.4-5.amzn2.x86_64
--> Processing Dependency: libnetsnmp.so.31()(64bit) for package: quagga-0.99.22.4-5.amzn2.x86_64
--> Processing Dependency: libnetsnmpagent.so.31()(64bit) for package: quagga-0.99.22.4-5.amzn2.x86_64
```

Complete!

```
[root@ip-10-10-1-204 ~]# systemctl start strongswan
[root@ip-10-10-1-204 ~]# systemctl enable strongswan
Created symlink from /etc/systemd/system/multi-user.target.wants/strongswan.service to /usr/lib/systemd/system/strongswan.service.
[root@ip-10-10-1-204 ~]# █
```

## 서울 리전 고객 게이트웨이 및 가상 프라이빗 게이트웨이 생성

**고객 게이트웨이 생성**

고객 게이트웨이는 AWS에서 생성하는 리소스로, 온프레미스 네트워크의 고객 게이트웨이 디바이스를 나타냅니다.

**세부 정보**

**이름 태그 - 선택 사항**  
'이름' 키와 사용자가 지정하는 값을 사용하여 태그를 생성합니다.

**BGP ASN** **정보**  
고객 게이트웨이 디바이스의 ASN입니다.

**IP 주소** **정보**  
고객 게이트웨이 디바이스의 외부 인터페이스에 대한 IP 주소를 지정합니다.

**인증서 ARN - 선택 사항**  
AWS Certificate Manager(ACM)에 프로비저닝된 프라이빗 인증서의 ARN입니다.

**디바이스 - 선택 사항**  
고객 게이트웨이 디바이스의 이름을 입력합니다.

**가상 프라이빗 게이트웨이 생성**

가상 프라이빗 게이트웨이는 Site-to-Site VPN 연결의 Amazon 측에 있는 VPN 집선기입니다.

**세부 정보**

**이름 태그 - 선택 사항**  
'이름' 키와 사용자가 지정하는 값을 사용하여 태그를 생성합니다.

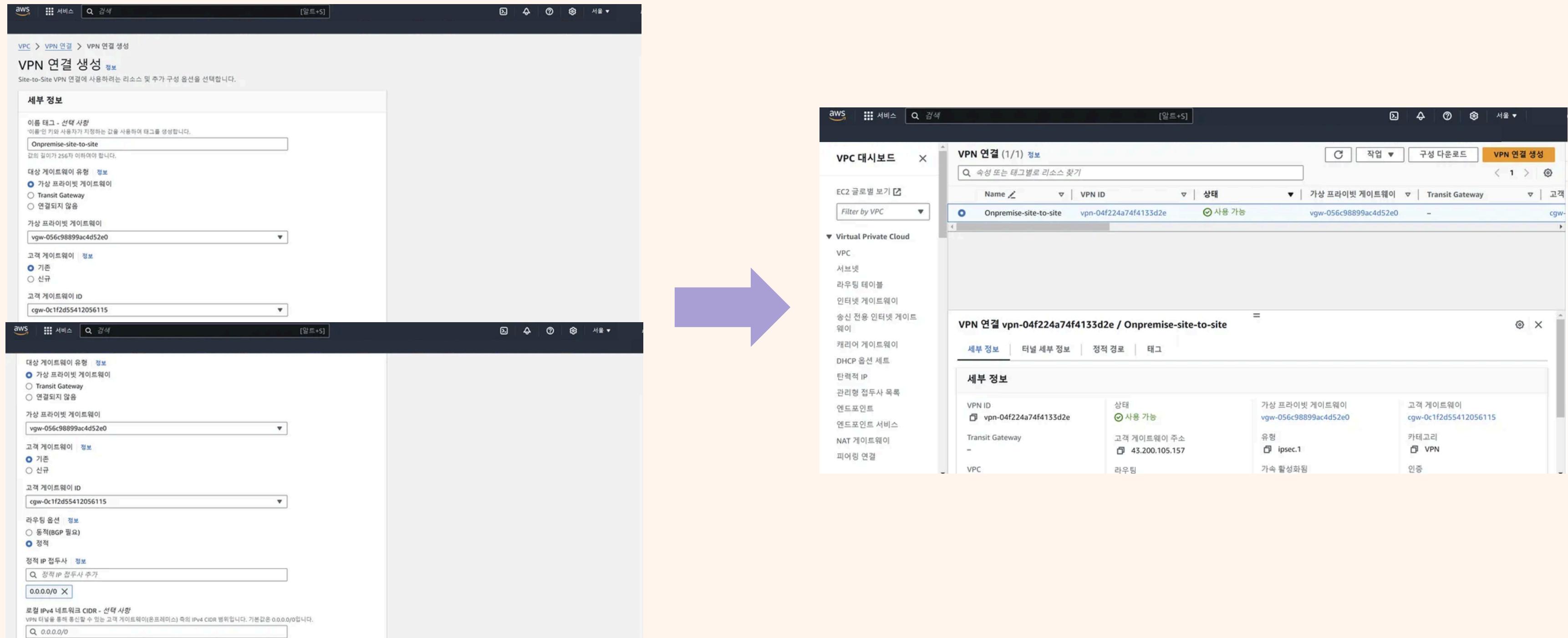
**Autonomous System Number(ASN)**  
 Amazon 기본 ASN  
 사용자 지정 ASN

**태그**  
태그는 사용자가 AWS 리소스에 할당하는 레이블입니다. 각 태그는 키와 선택적 값으로 구성됩니다. 태그를 사용하여 리소스를 검색 및 필터링하거나 AWS 비용을 추적할 수 있습니다. 이름 태그는 리소스를 쉽게 추적하도록 도와주므로 추가하는 것이 좋습니다.

키	값 - 선택 사항
<input type="text" value="Name"/>	<input type="text" value="Onpremise-vpg"/>

**새 태그 추가**  
최대 49개의 태그를 더 추가할 수 있습니다.

## 서울 리전 Site-to-Site VPN 생성 후 하이브리드 클라우드 환경 구축



## 각 리전 인증서 요청 및 발급

The screenshot shows the AWS Certificate Manager (ACM) console in the Seoul region. The main interface displays a table titled "인증서 (1)" (Certificates (1)). The table has columns for "인증서 ID" (Certificate ID), "도메인 이름" (Domain Name), "유형" (Type), and "상태" (Status). One row is listed:

인증서 ID	도메인 이름	유형	상태
b6401c82-4ef4-4f63-9bbb-5ca57e803a14	sol.cloud.space	Amazon 발급	발급됨

The screenshot shows the AWS Certificate Manager (ACM) console in the São Paulo region. The main interface displays a table titled "인증서 (1)" (Certificates (1)). The table has columns for "인증서 ID" (Certificate ID), "도메인 이름" (Domain Name), "유형" (Type), and "상태" (Status). One row is listed:

인증서 ID	도메인 이름	유형	상태
d71464b2-f066-4458-a2bd-c1a1e5d196f5	sol.cloud.space	Amazon 발급	발급됨

## 각 리전 RDS 생성

DB 식별자	상태	인스턴스	엔진	리전 및 AZ	크기	권장 사항
video-storage-db	사용 가능		MySQL Co...	ap-northe...	db.t3.micro	

DB 식별자	상태	인스턴스	엔진	리전 및 AZ	크기	권장 사항
video-storage-db	생성 중		MySQL Co...	-	db.t3.micro	
video-storage-db	사용 가능		MySQL Co...	ap-northe...	db.t3.micro	

## 각 리전 데이터베이스 스키마 생성

```
[ec2-user@ip-10-1-2-55 ~]$ mysql -h video-storage-db.cr08qwyi824z.ap-northeast-2.rds.amazonaws.com -P 3306 -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 30
Server version: 8.0.32 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> CREATE DATABASE video_storage;
Query OK, 1 row affected (0.02 sec)

MySQL [(none)]> USE video_storage;
Database changed
MySQL [video_storage]> CREATE TABLE users (
    ->     id VARCHAR(36) PRIMARY KEY,
    ->     username VARCHAR(255) NOT NULL UNIQUE,
    ->     email VARCHAR(255) NOT NULL UNIQUE,
    ->     password_hash VARCHAR(255) NOT NULL,
    ->     created_at DATETIME DEFAULT CURRENT_TIMESTAMP
    -> );
Query OK, 0 rows affected (0.16 sec)
```

```
[ec2-user@ip-10-3-2-82 ~]$ mysql -h video-storage-db.ch20wsuisaf8.sa-east-1.rds.amazonaws.com -P 3306 -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 57
Server version: 8.0.39 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> CREATE DATABASE video_storage;
Query OK, 1 row affected (0.03 sec)

MySQL [(none)]> USE video_storage;
Database changed
MySQL [video_storage]> CREATE TABLE users (
    ->     id VARCHAR(36) PRIMARY KEY,
    ->     username VARCHAR(255) NOT NULL UNIQUE,
    ->     email VARCHAR(255) NOT NULL UNIQUE,
    ->     password_hash VARCHAR(255) NOT NULL,
    ->     created_at DATETIME DEFAULT CURRENT_TIMESTAMP
    -> );
Query OK, 0 rows affected (0.07 sec)
```

## 멀티 리전 버킷 구성

The screenshot shows the AWS S3 console interface. The top navigation bar includes the AWS logo, a search bar, and various navigation icons. The main title is "Amazon S3 > 버킷". A banner at the top right says "계정 스냅샷 - 24시간마다 업데이트 [모든 AWS 리전]" and "Storage Lens 대시보드 보기". On the left, a sidebar lists "버킷", "Access Grants", "액세스 지점", "객체 Lambda 액세스 지점", "다중 리전 액세스 지점", "배치 작업", and "S3용 IAM Access Analyzer". Below this is a section for "Storage Lens" with "대시보드" and "Storage Lens 그룹" options. At the bottom of the sidebar, there's a link to "기능 스포트라이트" with a blue badge containing the number 7. The main content area shows a table titled "범용 버킷 (2) 정보 [모든 AWS 리전]". The table has columns for "이름", "AWS 리전", "IAM Access Analyzer", and "생성 날짜". It lists two buckets: "video-storage-bucket-2024-1" in "아시아 태평양(서울) ap-northeast-2" and "video-storage-bucket-2024-2" in "남아메리카(상파울루) sa-east-1". Each bucket row includes a "C" icon, an ARN link, and a "비어 있음" button.

이름	AWS 리전	IAM Access Analyzer	생성 날짜
<a href="#">video-storage-bucket-2024-1</a>	아시아 태평양(서울) ap-northeast-2	<a href="#">ap-northeast-2에 대한 분석기 보기</a>	2024. 11. 8. am 11:38:51 AM KST
<a href="#">video-storage-bucket-2024-2</a>	남아메리카(상파울루) sa-east-1	<a href="#">sa-east-1에 대한 분석기 보기</a>	2024. 11. 8. am 11:39:20 AM KST

## 각 리전 주요 권한 설정

**Amazon S3 > 버킷 > video-storage-bucket-2024-2 > 버킷 정책 편집**

**버킷 정책 편집** 정보

**버킷 정책**

JSON으로 작성된 버킷 정책은 버킷에 저장된 객체에 대한 액세스 권한을 제공합니다. 버킷 정책은 다른 계정이 소유한 객체에는 적용되지 않습니다. [자세히 알아보기](#)

**버킷 ARN**  
arn:aws:s3:::video-storage-bucket-2024-2

**정책**

```

1  {
2    "Version": "2012-10-17",
3    "Statement": [
4      {
5        "Sid": "PublicReadWriteAccess",
6        "Effect": "Allow",
7        "Principal": "*",
8        "Action": [
9          "s3:GetObject",
10         "s3:PutObject"
11       ],
12       "Resource": "arn:aws:s3:::video-storage-bucket-2024-1/*"
13     }
14   ]
  
```

**문 편집**

정책에서 기존 문을 선택하거나 새 문을 추가합니다.

**+ 새 문 추가**

**Amazon S3 > 버킷 > video-storage-bucket-2024-2 > CORS(cross-origin 리소스 공유) 편집**

**CORS(cross-origin 리소스 공유) 편집** 정보

**CORS(Cross-origin 리소스 공유)**

JSON으로 작성된 CORS 구성은 한 도메인에 로드되어 다른 도메인의 리소스와 상호 작용하는 클라이언트 헤더에 대한 방법을 정의합니다. [자세히 알아보기](#)

```

1  {
2    "AllowedHeaders": [
3      "*"
4    ],
5    "AllowedMethods": [
6      "GET",
7      "PUT",
8      "POST"
9    ],
10   "AllowedOrigins": [
11     "*"
12   ],
13   "ExposeHeaders": [
14     "ETag"
15   ],
16   "MaxAgeSeconds": 3600
17 }
18
19
  
```

**S3용 AWS Marketplace**

## 교차 리전 복제 구성

**Amazon S3 > 버킷 > video-storage-bucket-2024-1**

### video-storage-bucket-2024-1

관리 탭에서 복제 규칙을 설정합니다.

**수명 주기 규칙**

수명 주기 규칙을 사용하여 객체 수명 주기 동안 Amazon S3에서 수행할 작업을 정의하십시오(예: 객체를 다른 스토리지 클래스로 전환, 객체 아카이브 또는 지정된 기간이 지난 후 객체 삭제). [자세히 알아보기](#)

수명 주기 규칙 이름	상태	범위	현재 버전 작업	이전 버전 작업	만료된 객체 삭제...	완료되지 않은 멀티파트 업로드
수명 주기 규칙 없음						
이 버킷에 대한 수명 주기 규칙이 없습니다.						

[수명 주기 규칙 생성](#)

**복제 규칙 (1)**

복제 규칙을 사용하여 서버 측 암호화, 복제본 소유권, 복제본을 다른 스토리지 클래스로 전환 등과 같이 복제 중에 Amazon S3가 적용할 옵션을 정의합니다. [자세히 알아보기](#)

복제 규칙 이름	상태	대상 버킷	대상 리전	우선 순위	범위	스토리지 클래스	복제본 소유자	복제 시간 제어	KMS 암호화 객체(SSE-KMS 또는 DSSE-KMS)	복제본 수정 동기화
seoul-video-backup	활성화됨	s3://video-storage-bucket-2024-	남아메리카(상파울루) sa-	0	전체 버킷	원본과 동일	원본과 동일	비활성됨	복제 안 함	활성화됨

**Amazon S3 > 버킷 > video-storage-bucket-2024-2**

### video-storage-bucket-2024-2

관리 탭에서 복제 규칙을 설정합니다.

**수명 주기 규칙**

수명 주기 규칙을 사용하여 객체 수명 주기 동안 Amazon S3에서 수행할 작업을 정의하십시오(예: 객체를 다른 스토리지 클래스로 전환, 객체 아카이브 또는 지정된 기간이 지난 후 객체 삭제). [자세히 알아보기](#)

수명 주기 규칙 이름	상태	범위	현재 버전 작업	이전 버전 작업	만료된 객체 삭제 마커	완료되지 않은 멀티파...
수명 주기 규칙 없음						
이 버킷에 대한 수명 주기 규칙이 없습니다.						

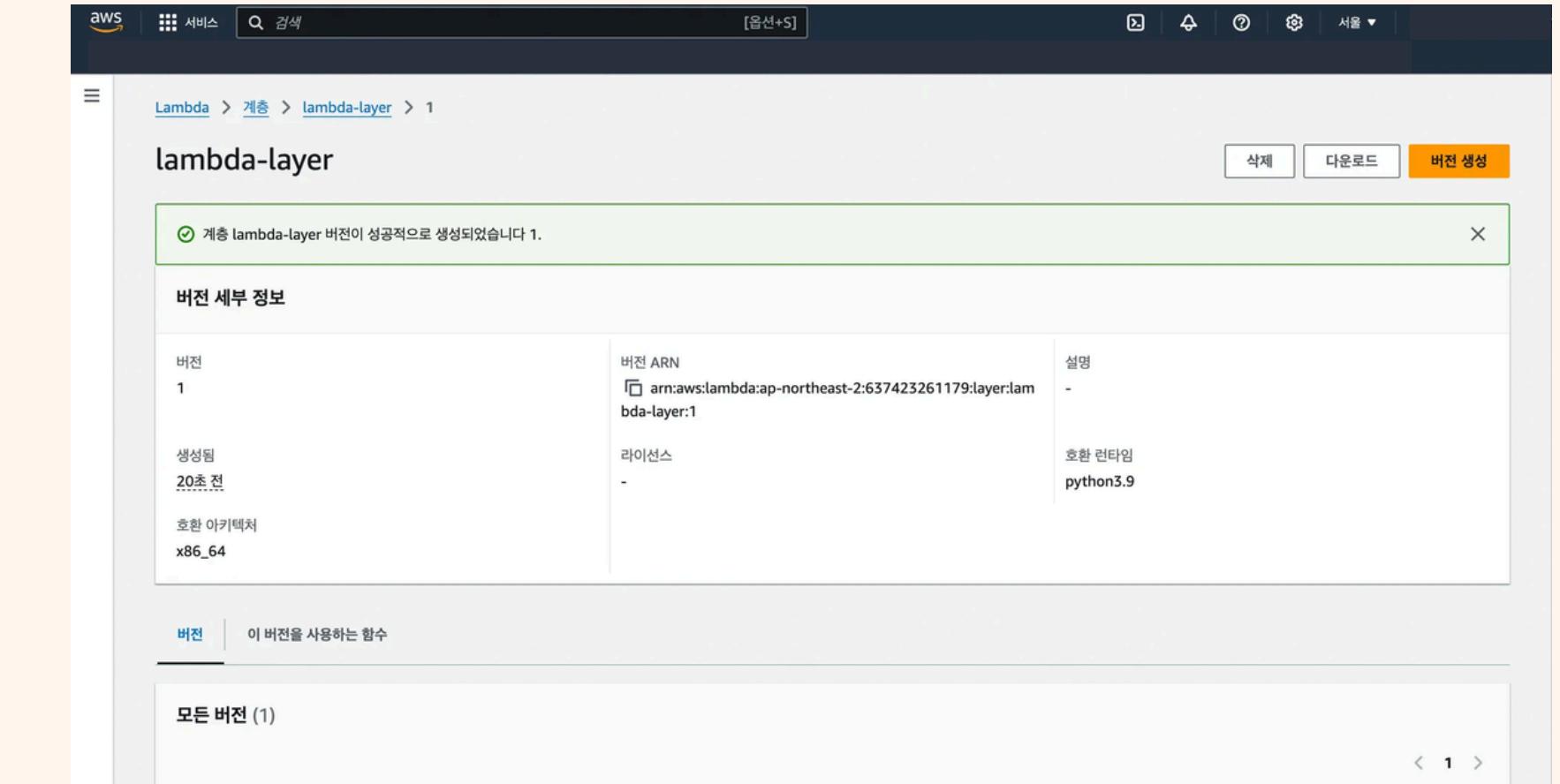
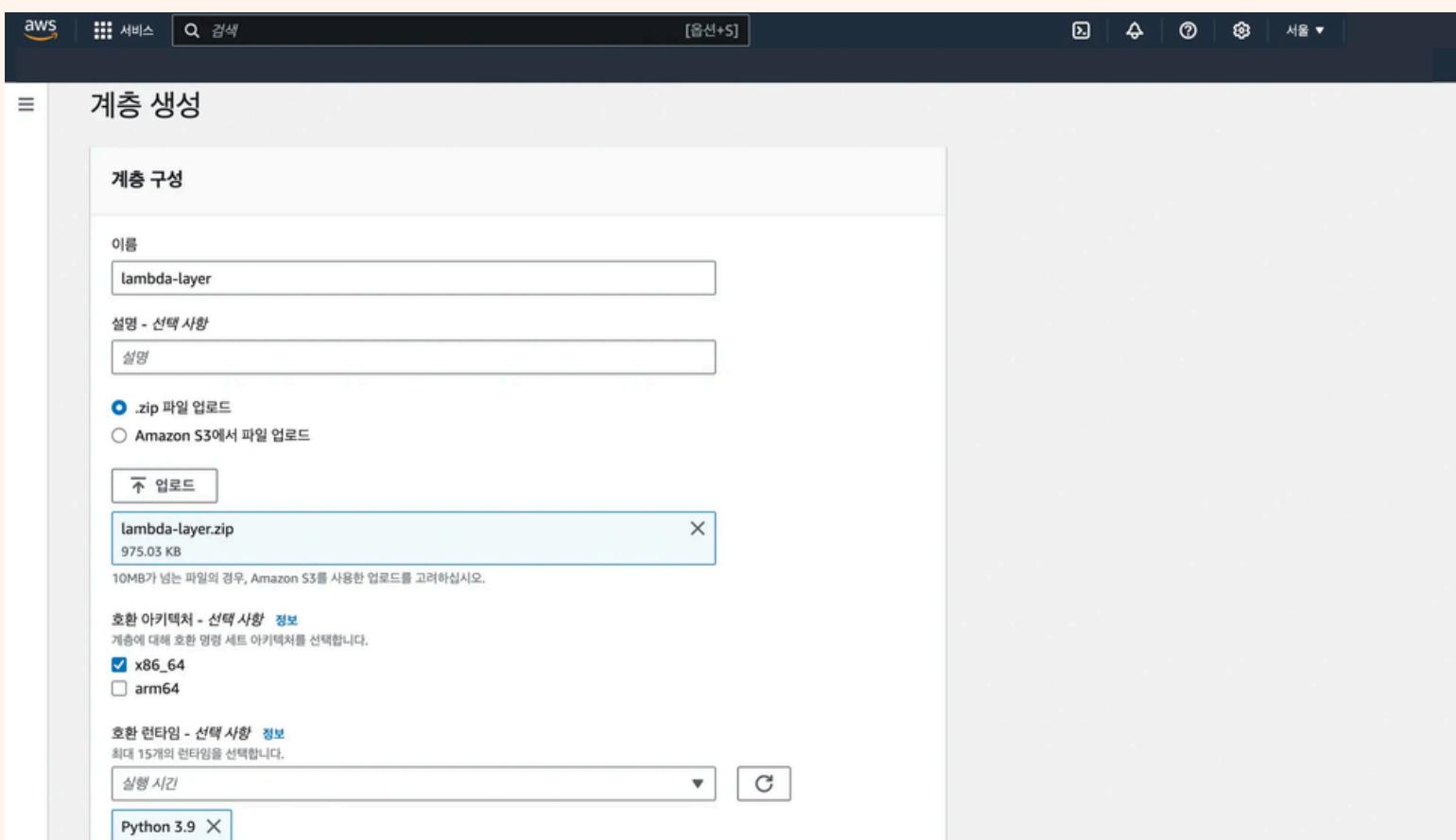
[수명 주기 규칙 생성](#)

**복제 규칙 (1)**

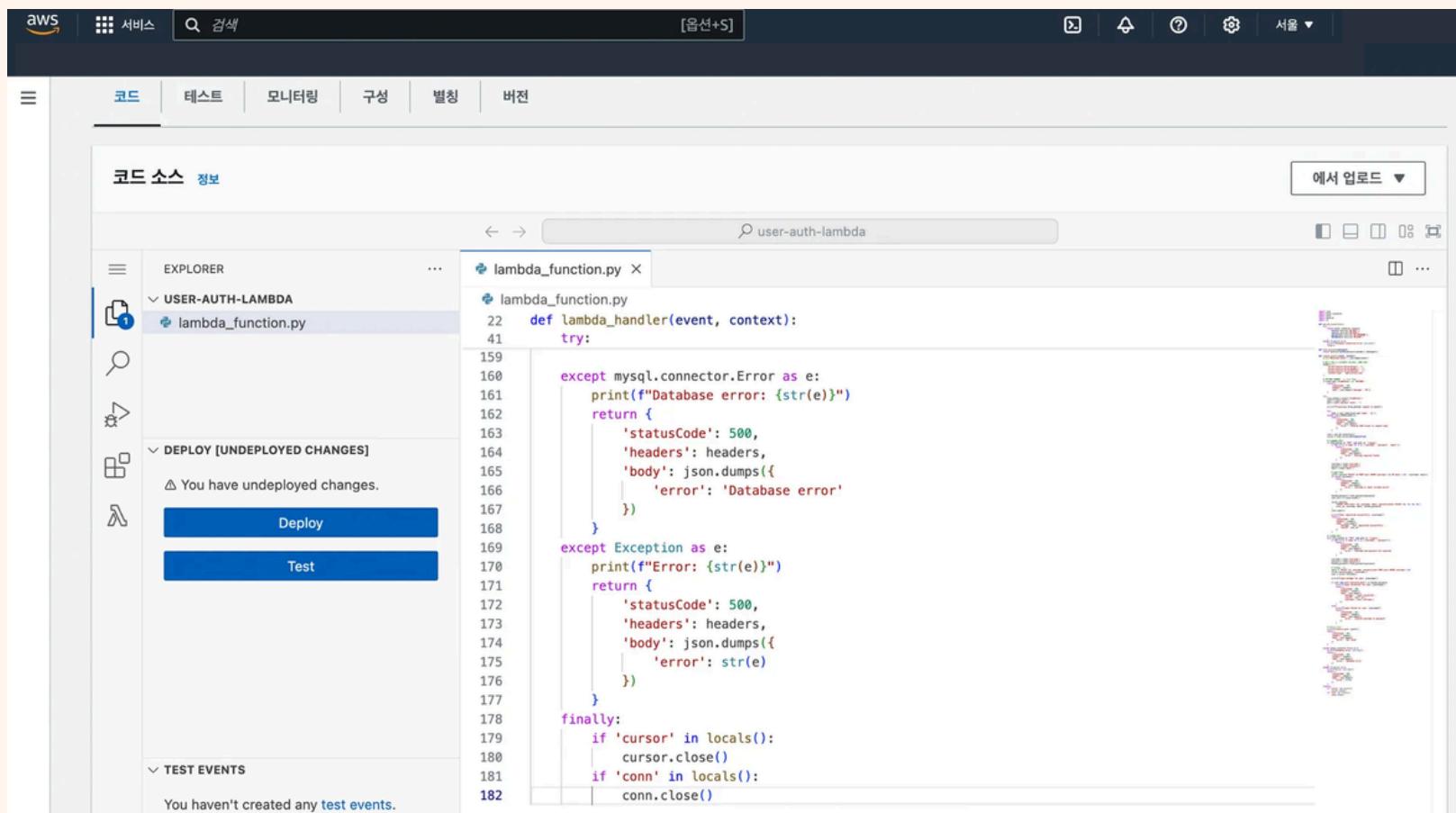
복제 규칙을 사용하여 서버 측 암호화, 복제본 소유권, 복제본을 다른 스토리지 클래스로 전환 등과 같이 복제 중에 Amazon S3가 적용할 옵션을 정의합니다. [자세히 알아보기](#)

복제 규칙 이름	상태	대상 버킷	대상 리전	우선 순위	범위	스토리지 클래스	복제본 소유자	복제 시간 제어	KMS 암호화 객체(SSE-KMS 또는 DSSE-KMS)	복제본 수정 동기화
saopaulo-video-backup	활성화됨	s3://video-storage-bucket-2024-	아시아 태평양 (서울) ap-	0	전체 버킷	원본과 동일	원본과 동일	비활성됨	복제 안 함	활성화됨

## 계층 생성



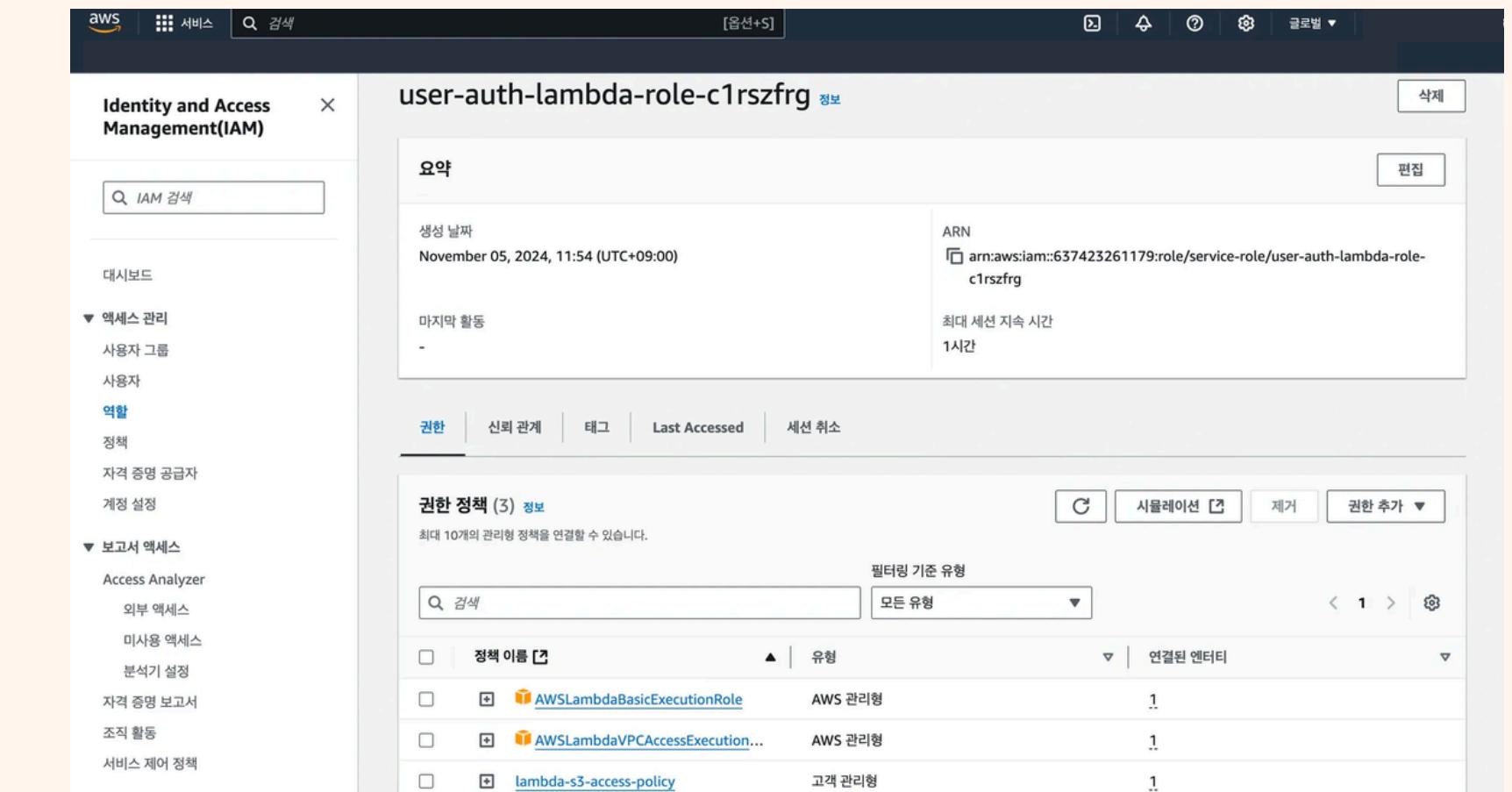
## Lambda 함수 (user-auth-lambda)



```

def lambda_handler(event, context):
    try:
        # Database connection logic
    except mysql.connector.Error as e:
        print(f"Database error: {str(e)}")
        return {
            'statusCode': 500,
            'headers': headers,
            'body': json.dumps({
                'error': 'Database error'
            })
        }
    except Exception as e:
        print(f"Error: {str(e)}")
        return {
            'statusCode': 500,
            'headers': headers,
            'body': json.dumps({
                'error': str(e)
            })
        }
    finally:
        if 'cursor' in locals():
            cursor.close()
        if 'conn' in locals():
            conn.close()

```

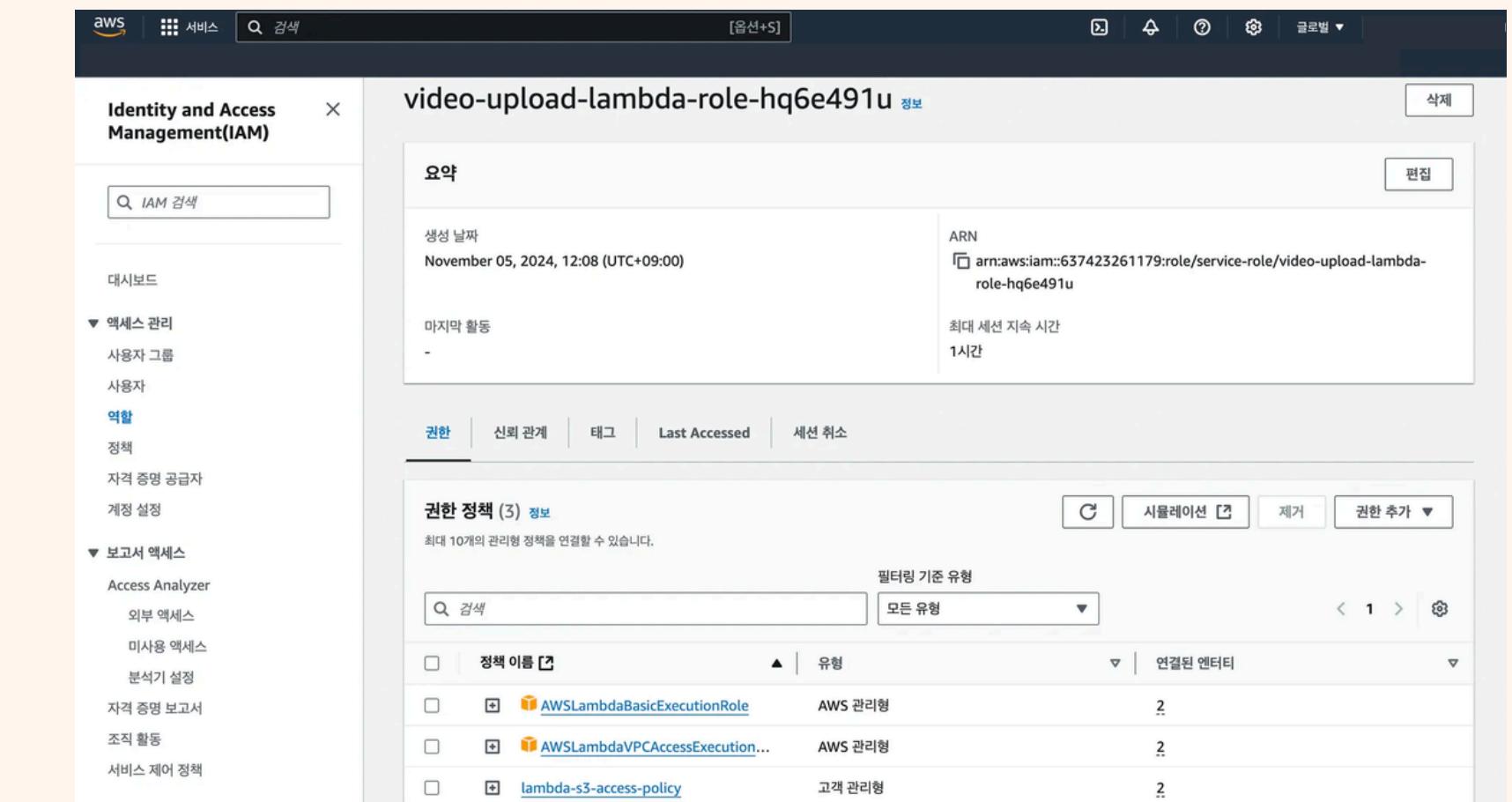


## Lambda 함수 (video-upload-lambda)

```

def lambda_handler(event, context):
    try:
        if event['httpMethod'] == 'POST' and path.endswith('/complete'):
            try:
                return {
                    'statusCode': 200,
                    'headers': headers,
                    'body': json.dumps({'message': 'Upload completed successfully'})
                }
            except Exception as e:
                print(f"Error in POST /complete: {str(e)}")
                return {
                    'statusCode': 500,
                    'headers': headers,
                    'body': json.dumps({'error': f'Internal server error: {str(e)}'})
                }
        print(f"No matching route for path: {path}")
        return {
            'statusCode': 404,
            'headers': headers,
            'body': json.dumps({'error': 'Not found'})
        }
    except Exception as e:
        print(f"General error: {str(e)}")
        return {
            'statusCode': 500,
            'headers': headers,
            'body': json.dumps({'error': str(e)})
        }

```



## API Gateway 구성

The screenshot shows the 'Stage details' section for the 'prod' stage. On the left, a tree view lists API resources: / (auth, signin, signup, videos, complete, upload-url). Under each resource, methods like OPTIONS, POST, GET, etc., are listed with their respective status (e.g., Inactive). The main area displays the following configuration:

- Stage name:** prod
- Rate Info:** -
- Web ACL:** -
- Burst Info:** -
- Client certificate:** -
- Invoke URL:** https://d4om1z1oxg.execute-api.ap-northeast-2.amazonaws.com/prod
- Active deployment:** deszow on November 04, 2024, 11:25 (UTC+09:00)

Below this is the 'Logs and tracing' section with options for CloudWatch logs, Detailed metrics, and Data tracing, all set to Inactive.

The screenshot shows the 'CORS 활성화' (Enable CORS) configuration page for the 'video-storage-api' resource. It includes the following sections:

- CORS 설정 정보:** A note explaining CORS is used for cross-origin requests and how to enable it.
- 게이트웨이 응답:** Options for enabling CORS on gateway responses.
  - 기본 4XX
  - 기본 5XX
- Access-Control-Allow-Methods:** A checkbox for enabling the 'OPTIONS' method.
- Access-Control-Allow-Headers:** A list of headers: Content-Type, X-Amz-Date, Authorization, X-Api-Key, X-Amz-Security-Token.
- Access-Control-Allow-Origin:** A field where '\*' is entered as the allowed origin.

## Visual Studio Code로 작성한 소스코드를 Github에 업로드 후 Amplify로 웹 서비스 배포

서울 리전

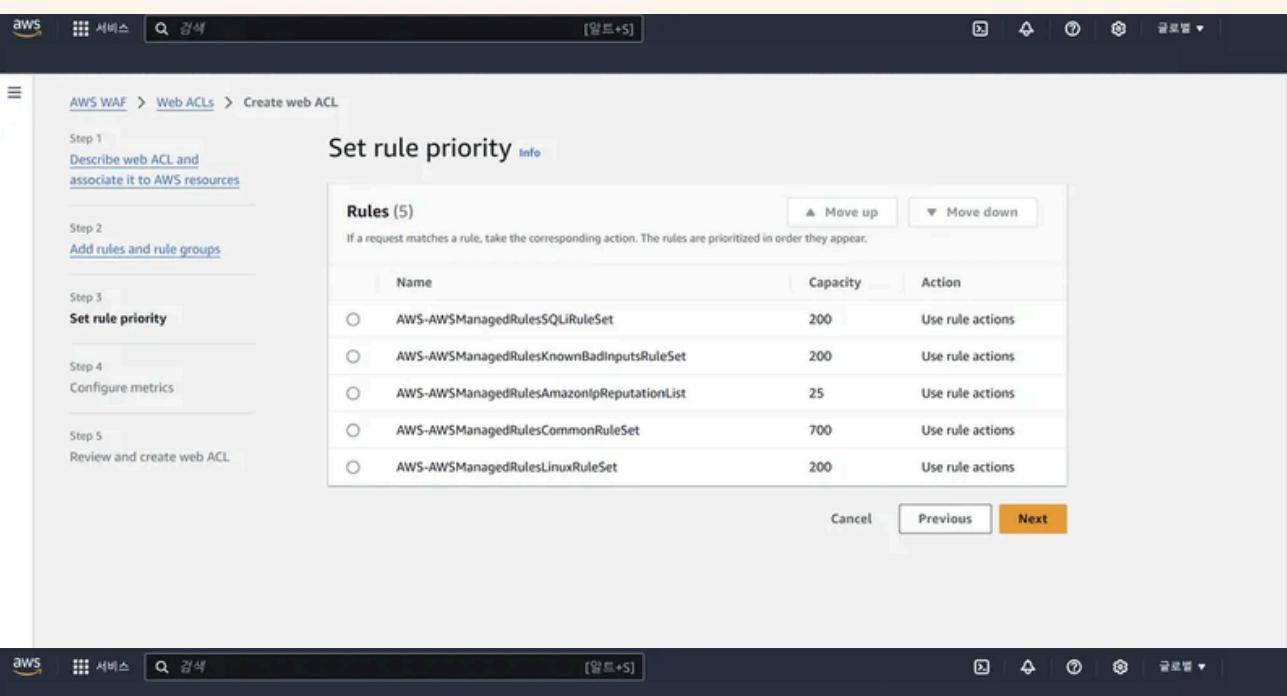
The screenshot shows the GitHub repository page for 'freeV.Sotrage\_kr'. The repository is private and contains 1 branch and 0 tags. It has 2 commits from 'lucy0920'. The codebase includes files like 'public', 'server', 'src', '.env', '.gitignore', 'README.md', 'eslint.config.js', 'index.html', 'package-lock.json', 'package.json', 'postcss.config.js', 'tailwind.config.js', and 'vite.config.js'. The repository has 0 stars, 1 watching, and 0 forks. The 'Languages' section shows JavaScript at 96.5%, CSS at 2.3%, and HTML at 1.2%. Below the repository details, there's a 'Suggested workflows' section.

상파울루 리전

The screenshot shows the GitHub repository page for 'freeV.Storage'. The repository is private and contains 1 branch and 0 tags. It has 4 commits from 'lucy0920'. The codebase includes files like 'public', 'server', 'src', '.env', '.gitignore', 'README.md', 'eslint.config.js', 'index.html', 'package-lock.json', 'package.json', 'postcss.config.js', 'tailwind.config.js', and 'vite.config.js'. The repository has 0 stars, 1 watching, and 0 forks. The 'Languages' section shows JavaScript at 96.4%, CSS at 2.3%, and HTML at 1.3%. Below the repository details, there's a 'Suggested workflows' section.

## 각 리전 WAF 설정

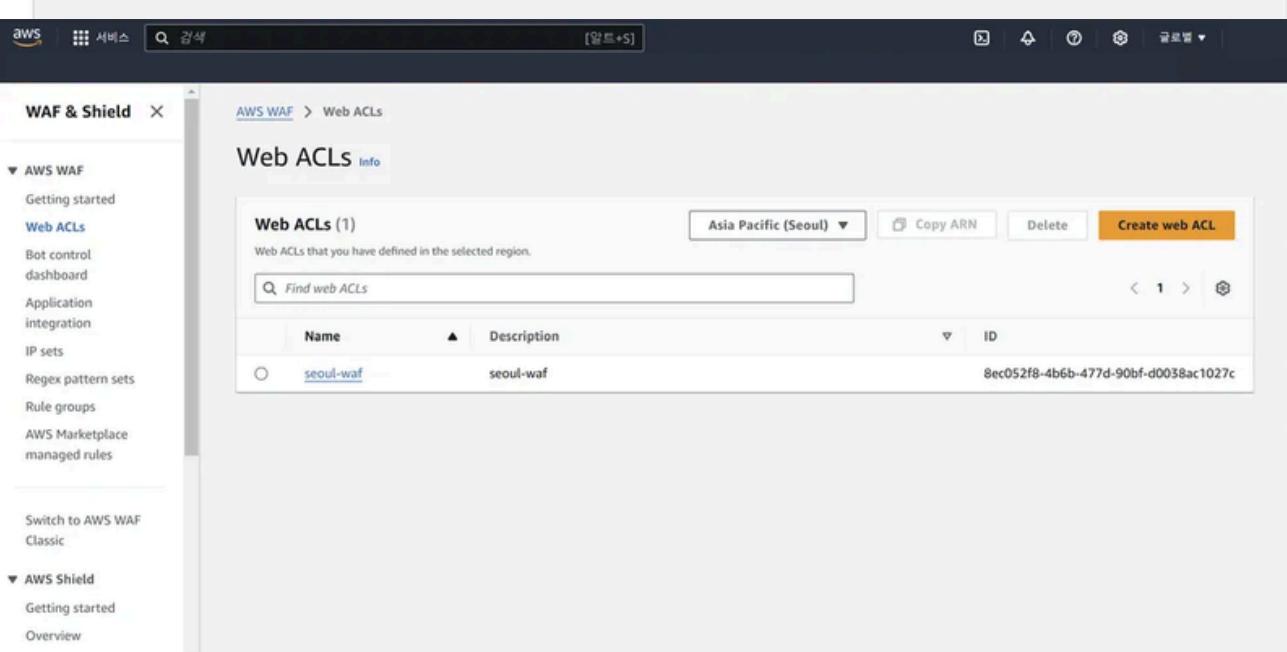
서울 리전



The screenshot shows the 'Set rule priority' step of creating a web ACL. It displays a table of rules with their names, capacities, and actions. The rules listed are:

Name	Capacity	Action
AWS-AWSManagedRulesSQLiRuleSet	200	Use rule actions
AWS-AWSManagedRulesKnownBadInputsRuleSet	200	Use rule actions
AWS-AWSManagedRulesAmazonIpReputationList	25	Use rule actions
AWS-AWSManagedRulesCommonRuleSet	700	Use rule actions
AWS-AWSManagedRulesLinuxRuleSet	200	Use rule actions

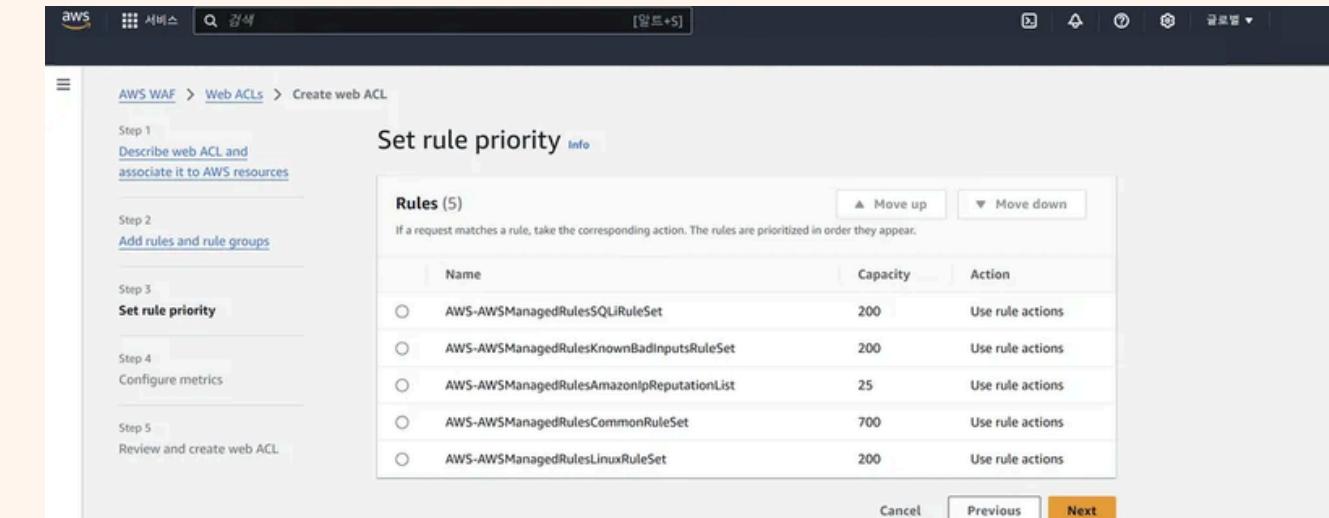
Buttons at the bottom include 'Cancel', 'Previous', and 'Next'.



The screenshot shows the 'Web ACLs' list page for the Asia Pacific (Seoul) region. It displays one entry named 'seoul-waf' with the description 'seoul-waf'. The table columns are Name, Description, and ID.

Name	Description	ID
seoul-waf	seoul-waf	8ec052f8-4b6b-477d-90bf-d0038ac1027c

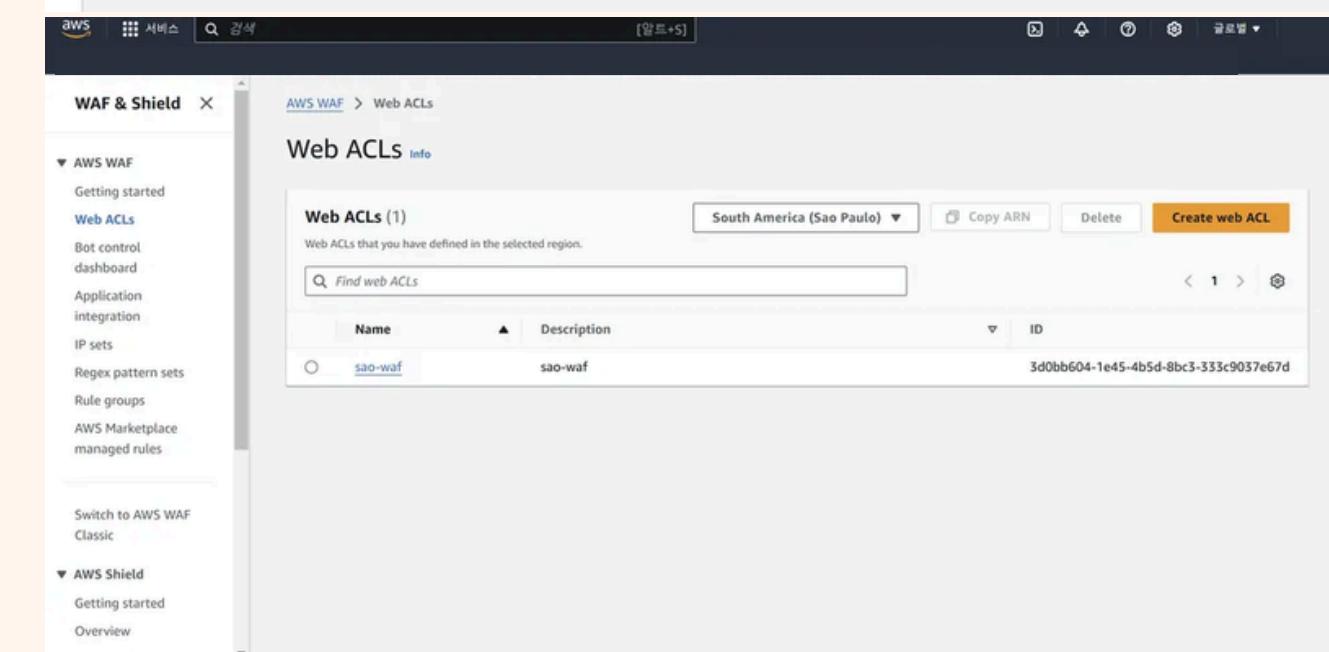
상파울루 리전



The screenshot shows the 'Set rule priority' step of creating a web ACL for the South America (Sao Paulo) region. It displays a table of rules with their names, capacities, and actions. The rules listed are identical to the Seoul region:

Name	Capacity	Action
AWS-AWSManagedRulesSQLiRuleSet	200	Use rule actions
AWS-AWSManagedRulesKnownBadInputsRuleSet	200	Use rule actions
AWS-AWSManagedRulesAmazonIpReputationList	25	Use rule actions
AWS-AWSManagedRulesCommonRuleSet	700	Use rule actions
AWS-AWSManagedRulesLinuxRuleSet	200	Use rule actions

Buttons at the bottom include 'Cancel', 'Previous', and 'Next'.



The screenshot shows the 'Web ACLs' list page for the South America (Sao Paulo) region. It displays one entry named 'sao-waf' with the description 'sao-waf'. The table columns are Name, Description, and ID.

Name	Description	ID
sao-waf	sao-waf	3d0bb604-1e45-4b5d-8bc3-333c9037e67d

## 각 리전 ALB/WAF 연동 설정

AWS CloudWatch Metrics

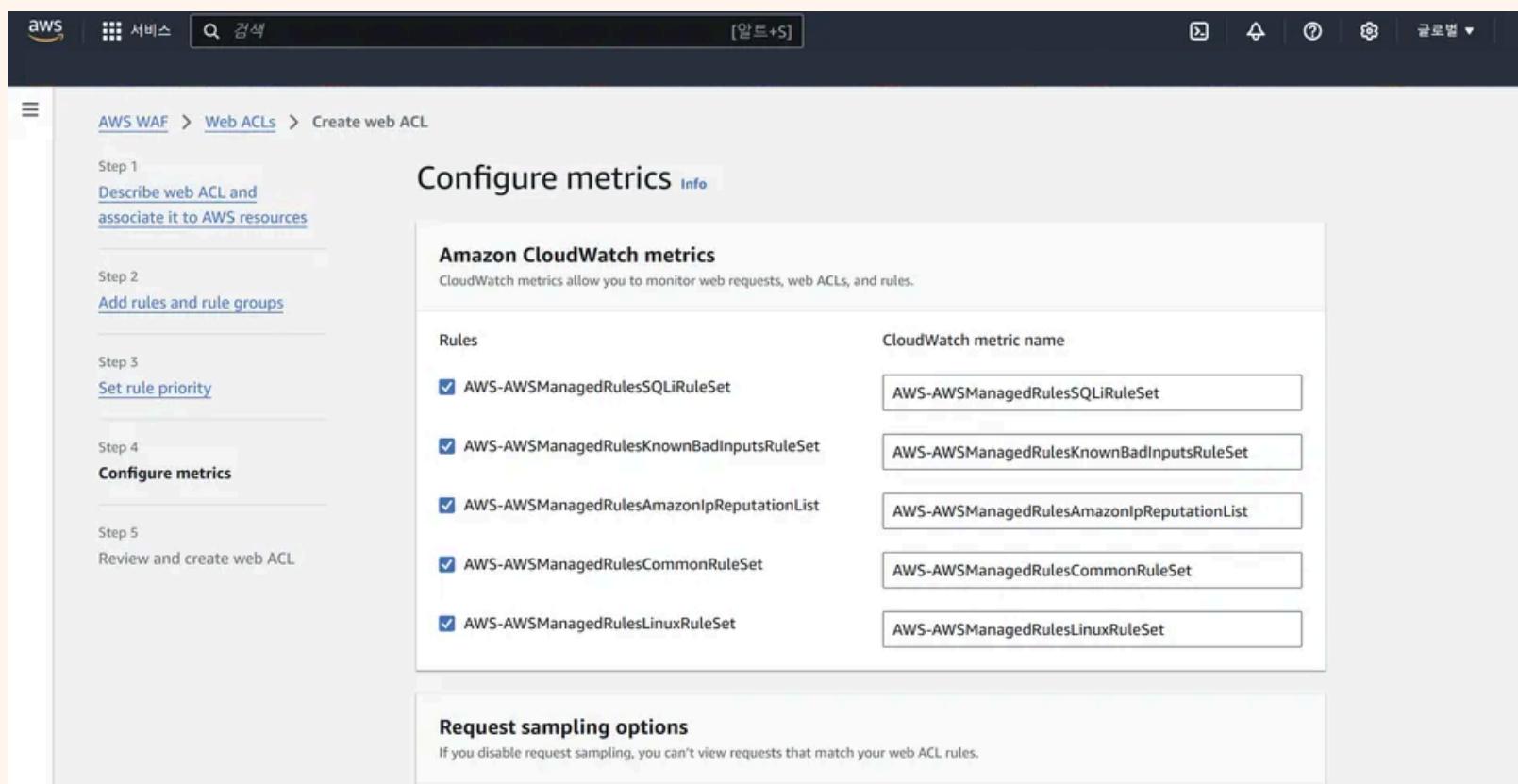
HelloWorld

Metric Name	Value	Unit
HelloWorld	1	Count

HelloWorld2

Metric Name	Value	Unit
HelloWorld2	1	Count

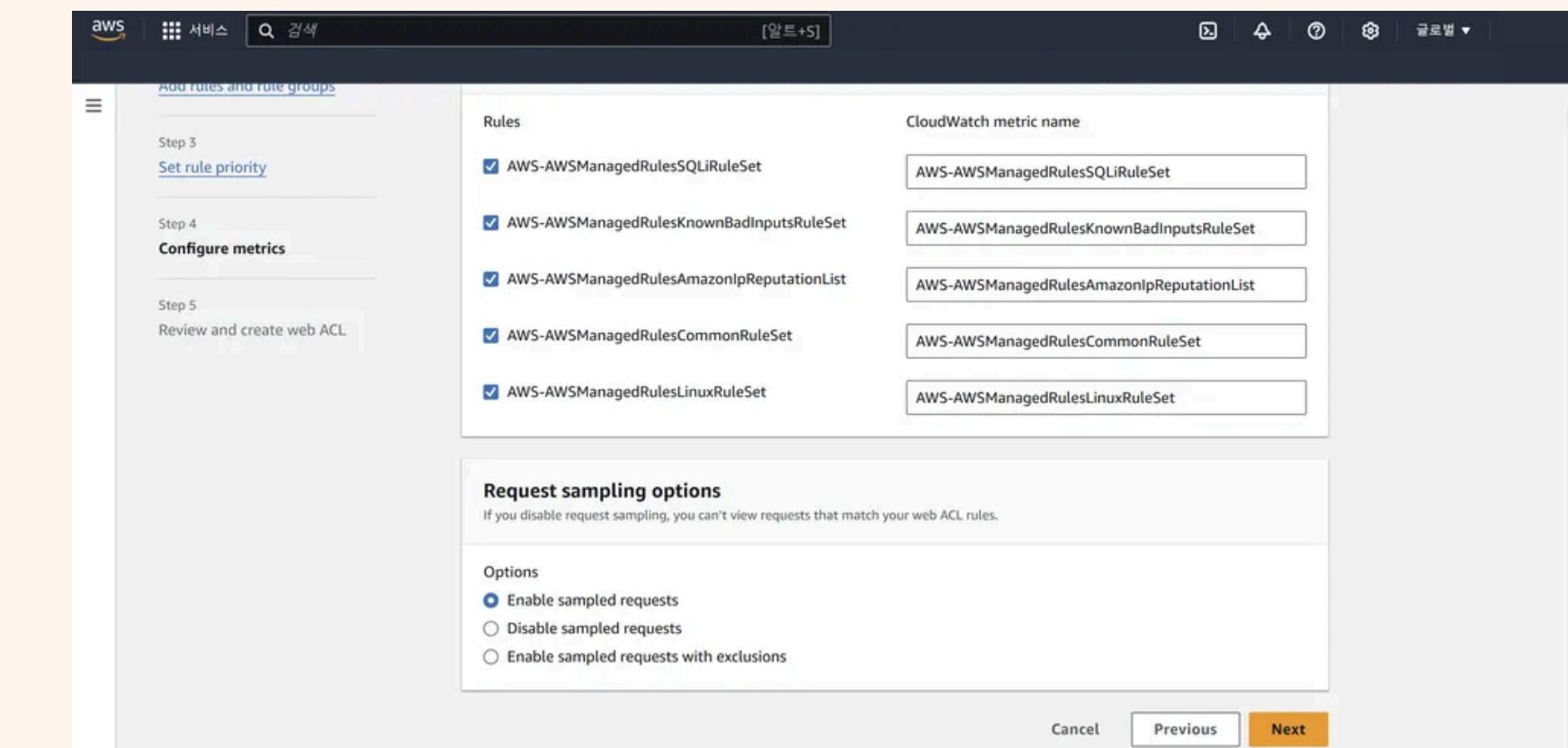
## WAF/CloudWatch 모니터링 및 로깅 설정



The screenshot shows the AWS WAF CloudWatch Metrics configuration step. It lists five AWS-Managed Rule Sets under 'Rules' and their corresponding 'CloudWatch metric name' fields. All checkboxes are checked.

Rules	CloudWatch metric name
AWS-AWSManagedRulesSQLiRuleSet	AWS-AWSManagedRulesSQLiRuleSet
AWS-AWSManagedRulesKnownBadInputsRuleSet	AWS-AWSManagedRulesKnownBadInputsRuleSet
AWS-AWSManagedRulesAmazonIpReputationList	AWS-AWSManagedRulesAmazonIpReputationList
AWS-AWSManagedRulesCommonRuleSet	AWS-AWSManagedRulesCommonRuleSet
AWS-AWSManagedRulesLinuxRuleSet	AWS-AWSManagedRulesLinuxRuleSet

**Request sampling options**  
If you disable request sampling, you can't view requests that match your web ACL rules.



The screenshot shows the final review step of the WAF CloudWatch Metrics configuration. It lists the selected rules and their corresponding CloudWatch metric names. It also includes a 'Request sampling options' section with a radio button for 'Enable sampled requests'.

Rules	CloudWatch metric name
AWS-AWSManagedRulesSQLiRuleSet	AWS-AWSManagedRulesSQLiRuleSet
AWS-AWSManagedRulesKnownBadInputsRuleSet	AWS-AWSManagedRulesKnownBadInputsRuleSet
AWS-AWSManagedRulesAmazonIpReputationList	AWS-AWSManagedRulesAmazonIpReputationList
AWS-AWSManagedRulesCommonRuleSet	AWS-AWSManagedRulesCommonRuleSet
AWS-AWSManagedRulesLinuxRuleSet	AWS-AWSManagedRulesLinuxRuleSet

**Request sampling options**  
If you disable request sampling, you can't view requests that match your web ACL rules.

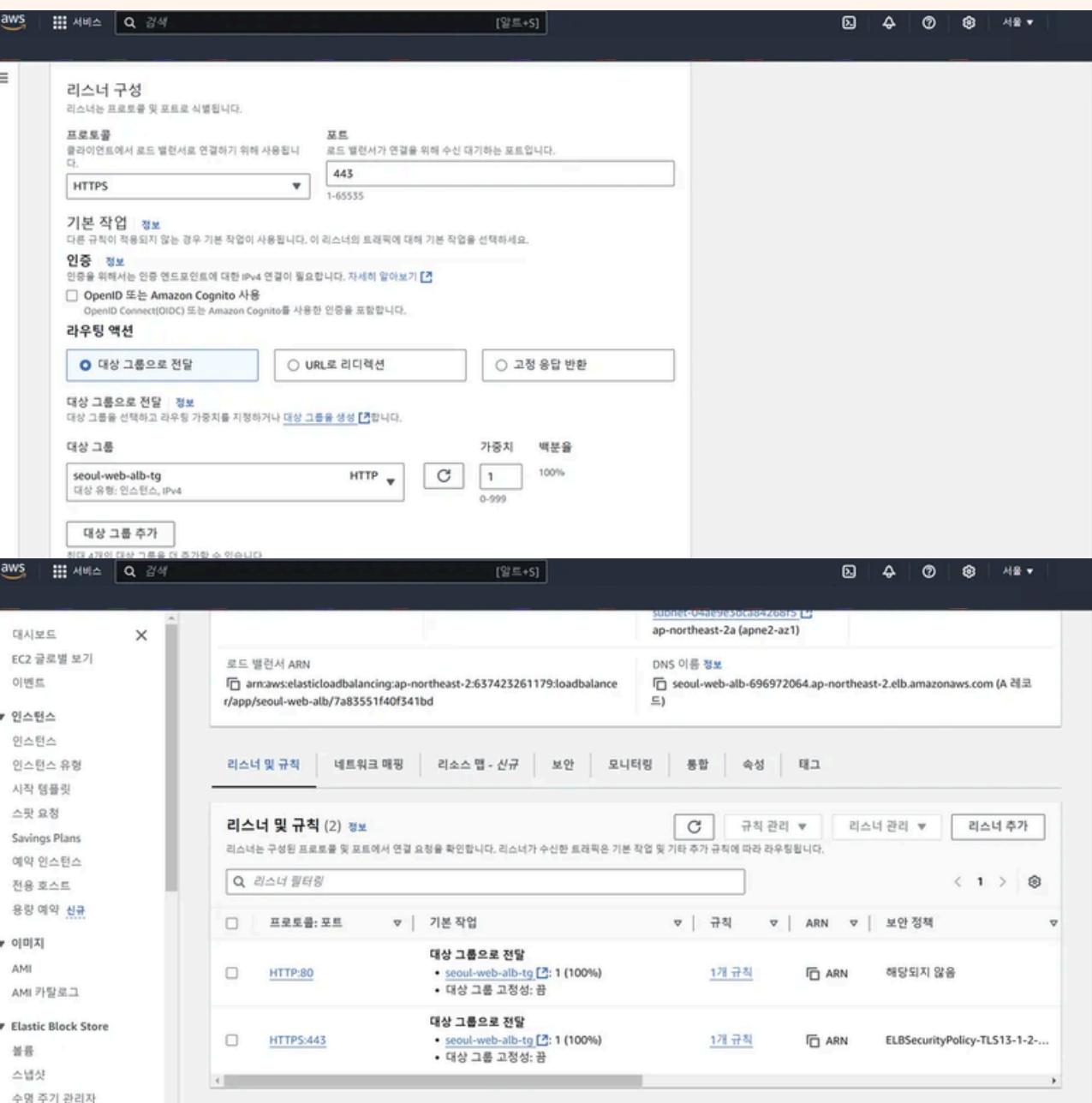
**Options**

Enable sampled requests  
 Disable sampled requests  
 Enable sampled requests with exclusions

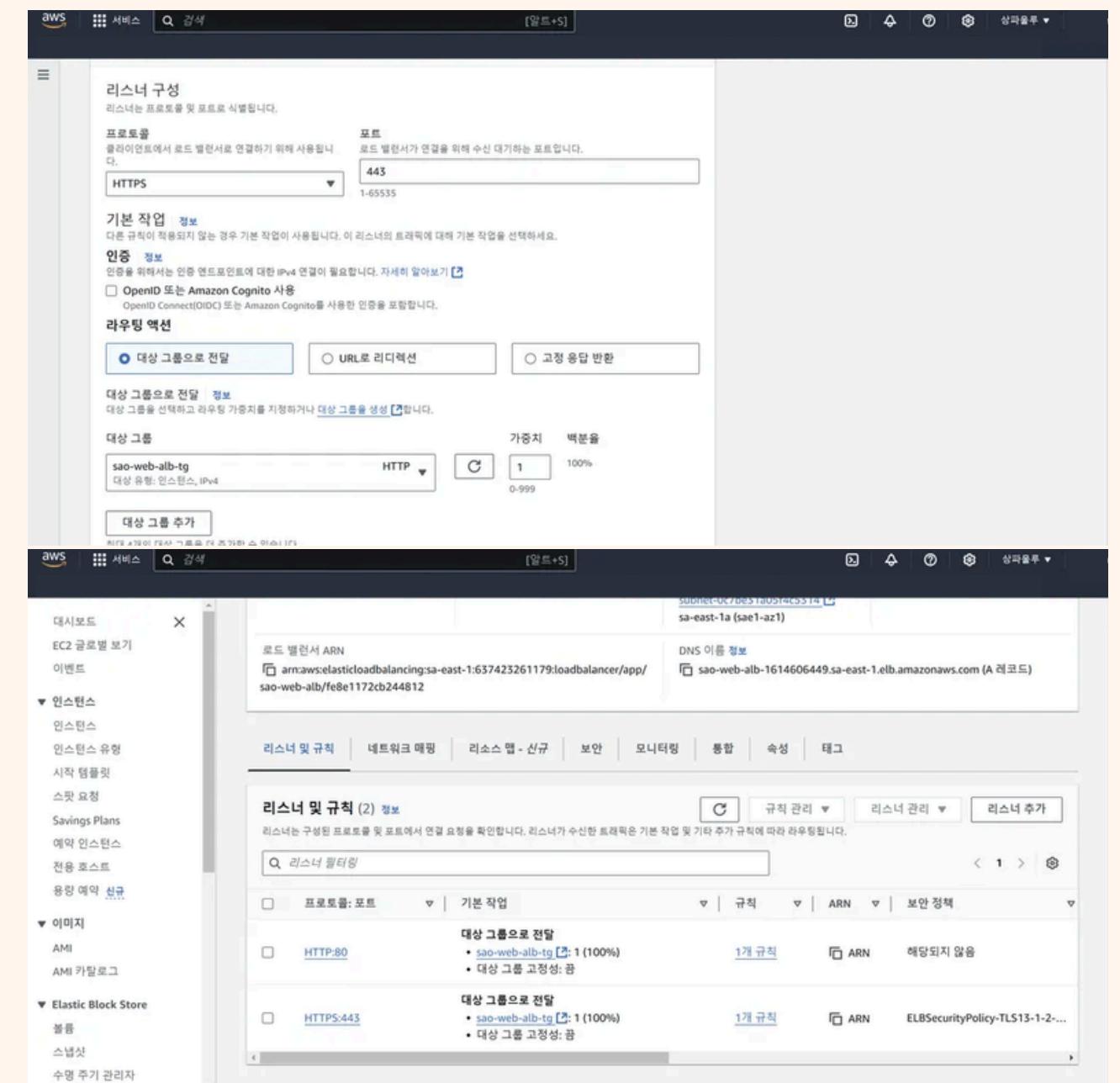
Cancel Previous Next

# 각 리전 ALB에 리스너(443) 추가 생성

서울 리전



상파울루 리전



## 리스너 및 엔드포인트 그룹 추가

The image consists of three screenshots from the AWS Global Accelerator console, illustrating the step-by-step process of adding a Listener and Endpoint Groups.

**Screenshot 1: Listener Configuration (Step 3)**

This screenshot shows the "Listener" configuration section. It includes two listeners:

- Listener 1:** Port 80, Protocol TCP, Client Port Selection None.
- Listener 2:** Port 443, Protocol TCP, Client Port Selection None.

A "리스너 추가" (Add Listener) button is visible at the bottom.

**Screenshot 2: Listener Summary (Step 4)**

This screenshot shows the "리스너: 80 TCP" summary. It lists two endpoint groups:

- 엔드포인트 그룹 1:** Region ap-northeast-2, Traffic Type 100.
- 엔드포인트 그룹 2:** Region sa-east-1, Traffic Type 100.

Links for "Port Configuration", "State Confirmation", and "Endpoint Group Configuration" are provided.

**Screenshot 3: Listener Summary (Step 4)**

This screenshot shows the "리스너: 443 TCP" summary, which is identical to the one above, listing the same two endpoint groups and traffic types.

## 엔드포인트(ALB) 추가 및 트래픽 라우팅

**Step 1**  
이름 입력

**Step 2**  
리스너 추가

**Step 3**  
엔드포인트 그룹 추가

**Step 4**  
엔드포인트 추가

**리스너: 80 TCP**

Global Accelerator는 이러한 포트에 도착하는 트래픽을 리전 엔드포인트 그룹의 엔드포인트로 라우팅합니다. 엔드포인트 그룹의 모든 엔드포인트는 동일한 리전에 있어야 합니다.

▼ 엔드포인트 그룹: ap-northeast-2  
트래픽 다이얼: 100%

**엔드포인트 1**

크로스 계정 연결에 지정된 리소스 추가  
크로스 계정 연결 소유자의 계정 ID 선택

엔드포인트 유형: Application Load Balancer  
엔드포인트: arn:aws:elasticloadbalanc...  
가중치: 128

클라이언트 IP 주소 보존  
Global Accelerator는 이 기능을 사용하지 않도록 설정하기 위해 확인란의 선택을 취소하지 않는 한 인터넷 연결 Application Load Balancer의 클라이언트 IP 주소를 보존합니다. 모든 내부 Application Load Balancer(은)는 클라이언트 IP 주소를 자동으로 보존합니다. 엔드포인트가 보존된 클라이언트 IP 주소의 트래픽을 수락하도록 구성되어 있는지 확인합니다.

**리스너: 443 TCP**

Global Accelerator는 이러한 포트에 도착하는 트래픽을 리전 엔드포인트 그룹의 엔드포인트로 라우팅합니다. 엔드포인트 그룹의 모든 엔드포인트는 동일한 리전에 있어야 합니다.

▼ 엔드포인트 그룹: ap-northeast-2  
트래픽 다이얼: 100%

**엔드포인트 1**

크로스 계정 연결에 지정된 리소스 추가  
크로스 계정 연결 소유자의 계정 ID 선택

엔드포인트 유형: Application Load Balancer  
엔드포인트: arn:aws:elasticloadbalanc...  
가중치: 128

클라이언트 IP 주소 보존  
Global Accelerator는 이 기능을 사용하지 않도록 설정하기 위해 확인란의 선택을 취소하지 않는 한 인터넷 연결 Application Load Balancer의 클라이언트 IP 주소를 보존합니다. 모든 내부 Application Load Balancer(은)는 클라이언트 IP 주소를 자동으로 보존합니다. 엔드포인트가 보존된 클라이언트 IP 주소의 트래픽을 수락하도록 구성되어 있는지 확인합니다.

**엔드포인트 추가**

▼ 엔드포인트 그룹: sa-east-1  
트래픽 다이얼: 100%

**엔드포인트 1**

크로스 계정 연결에 지정된 리소스 추가  
크로스 계정 연결 소유자의 계정 ID 선택

엔드포인트 유형: Application Load Balancer  
엔드포인트: arn:aws:elasticloadbalanc...  
가중치: 128

클라이언트 IP 주소 보존  
Global Accelerator는 이 기능을 사용하지 않도록 설정하기 위해 확인란의 선택을 취소하지 않는 한 인터넷 연결 Application Load Balancer의 클라이언트 IP 주소를 보존합니다. 모든 내부 Application Load Balancer(은)는 클라이언트 IP 주소를 자동으로 보존합니다. 엔드포인트가 보존된 클라이언트 IP 주소의 트래픽을 수락하도록 구성되어 있는지 확인합니다.

**엔드포인트 추가**

▼ 엔드포인트 그룹: sa-east-1  
트래픽 다이얼: 100%

**엔드포인트 1**

크로스 계정 연결에 지정된 리소스 추가  
크로스 계정 연결 소유자의 계정 ID 선택

엔드포인트 유형: Application Load Balancer  
엔드포인트: arn:aws:elasticloadbalanc...  
가중치: 128

클라이언트 IP 주소 보존  
Global Accelerator는 이 기능을 사용하지 않도록 설정하기 위해 확인란의 선택을 취소하지 않는 한 인터넷 연결 Application Load Balancer의 클라이언트 IP 주소를 보존합니다. 모든 내부 Application Load Balancer(은)는 클라이언트 IP 주소를 자동으로 보존합니다. 엔드포인트가 보존된 클라이언트 IP 주소의 트래픽을 수락하도록 구성되어 있는지 확인합니다.

## Global Accelerator 생성 완료

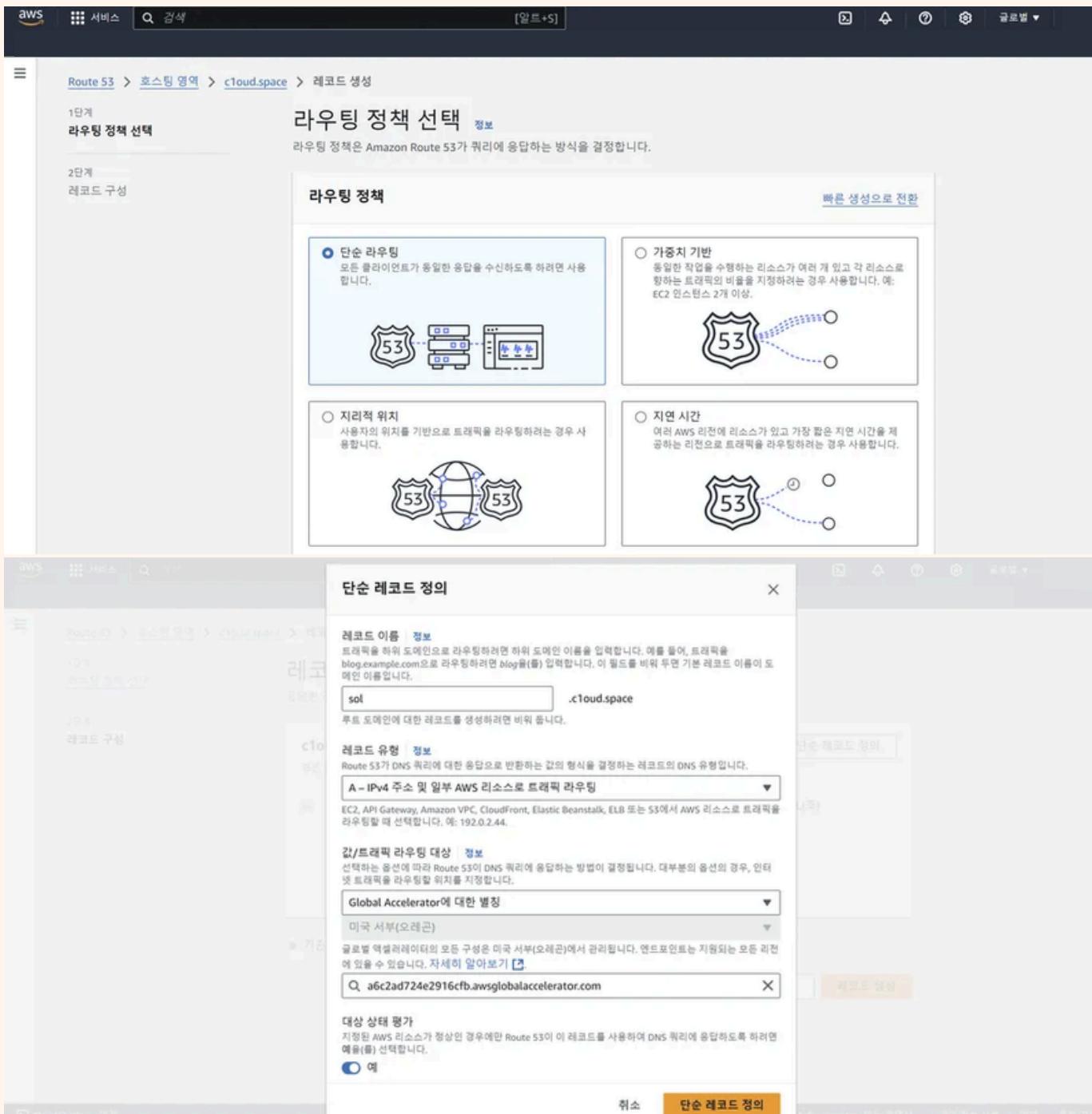
The screenshot shows the AWS Global Accelerator console interface. At the top, there's a navigation bar with the AWS logo, a '서비스' (Services) button, a search bar, and a '글로벌' (Global) dropdown. Below the navigation is a breadcrumb trail: 'AWS Global Accelerator > 액셀러레이터'. The main area is titled '액셀러레이터 (1)' and contains a table with one row of data. The table columns are: 이름 (Name), 유형 (Type), IPv4, IPv6, 활성화됨 (Enabled), DNS 이름 (DNS Name), 이중 스택 DNS 이름 (Dual Stack DNS Name), 상태 (Status), 생성됨 (Created), and 편집됨 (Last Modified). The data in the table is as follows:

이름	유형	IPv4	IPv6	활성화됨	DNS 이름	이중 스택 DNS 이름	상태	생성됨	편집됨
sol.cloud.space	스탠다드	75.2.32.28 76.223.120.66		활성화됨	a592a0c7db36ef6ff.awsglobalaccelerator.com	-	진행 중	2024년 11월 4 일 월요일 오전 6:38 GMT	2024년 11월 4 일 월요일 오전 6:38 GMT

03-13

# Route 53

c1oud.space(A 레코드) 생성



Record Type	Name	Type	Value	TTL (초)
NS	ns-1092.awsdns-08.org.	NS		172800
NS	ns-1801.awsdns-33.co.uk.	NS		-
NS	ns-989.awsdns-59.net.	NS		-
NS	ns-273.awsdns-34.com.	NS		-
SOA	c1oud.space	SOA		-
A	sol.c1oud....	A	a6c2ad724e2916cfb.awsglo...	-

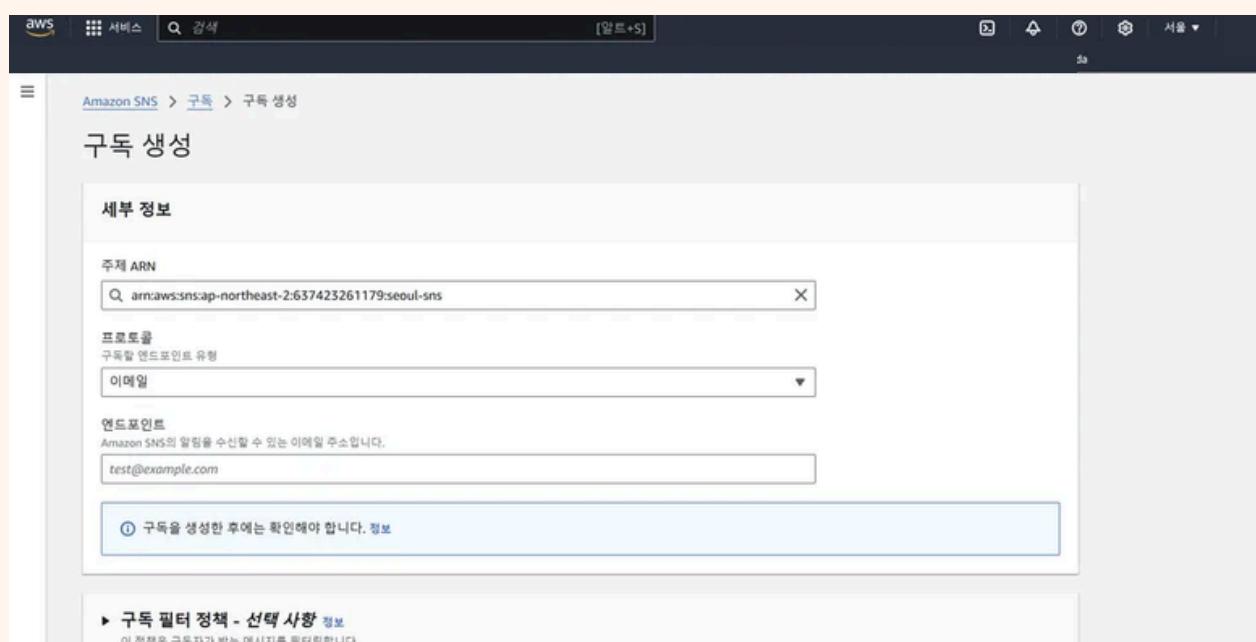
## 각 리전 SNS 주제 생성

The screenshot shows the AWS CloudWatch & SNS service interface in the Seoul region. The user is creating a new SNS topic named "seoul-sns". The "Type" is set to "Standard". The "Message Delivery Order" dropdown is set to "Standard (FIFO)". The "Topic Name" field contains "seoul-sns". The "Display Name" field contains "내 주제". The "Description" field is empty.

The screenshot shows the AWS CloudWatch & SNS service interface in the São Paulo region. The user is creating a new SNS topic named "sao-sns". The "Type" is set to "Standard". The "Message Delivery Order" dropdown is set to "Standard (FIFO)". The "Topic Name" field contains "sao-sns". The "Display Name" field contains "내 주제". The "Description" field is empty.

## 각 리전 SNS 구독 생성 및 이메일 확인

서울 리전



Simple Notification Service

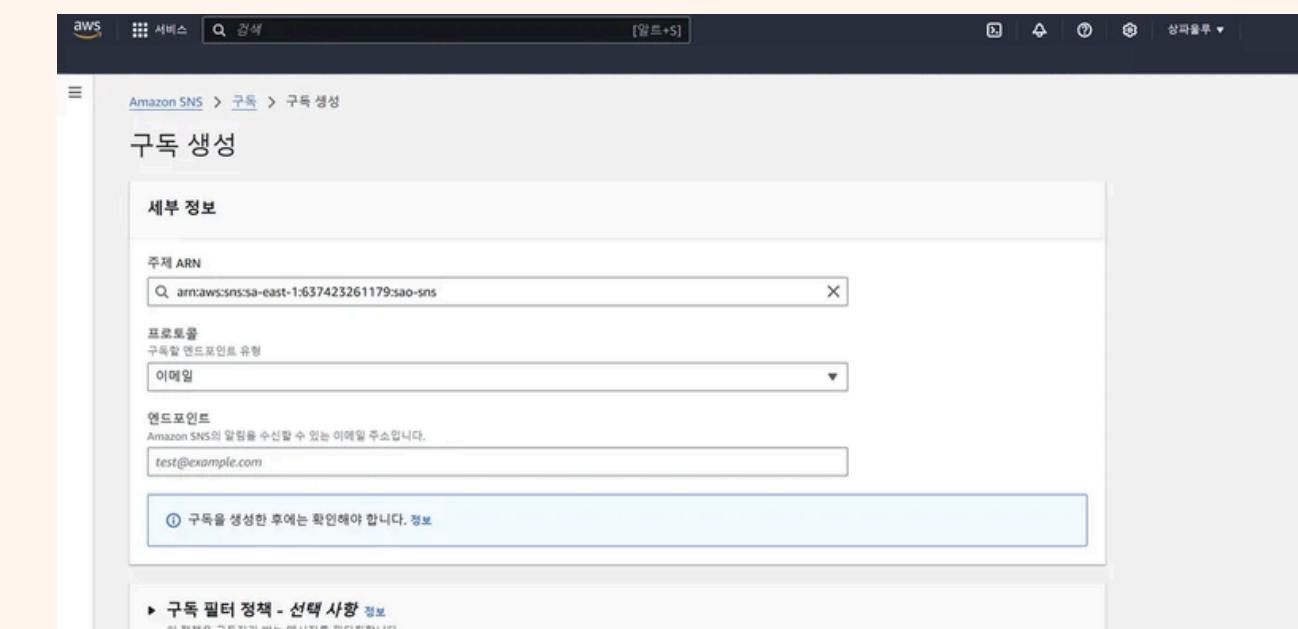
### Subscription confirmed!

You have successfully subscribed.

Your subscription's id is:  
arn:aws:sns:ap-northeast-2:63d3e11ab585a7

If it was not your intention to subscribe, [click here to unsubscribe](#).

상파울루 리전



Simple Notification Service

### Subscription confirmed!

You have successfully subscribed.

Your subscription's id is:  
arn:aws:sns:sa-east-1:637423

If it was not your intention to subscribe, [click here to unsubscribe](#).

## CloudWatch 경보 지정 및 환경 설정

서울 리전

**CloudWatch > 경보 > 경보 생성**

**지표 및 조건 지정**

**지표**

이 경보는 5 분 내 1개의 데이터 포인트에 대해 파란색 줄이 빨간색 줄을 초과할 때 트리거됩니다.

Count: 1  
지표 이름: Errors  
FunctionName: user-auth-lambda  
Resource: user-auth-lambda  
통계: 평균

**작업 구성**

**알림**

경보 상태 드리거: 경보 상태 (선택)

다음 SNS 주제에 알림을 보냅니다. 알림을 수신할 SNS(Simple Notification Service) 주제를 정의합니다.

기존 SNS 주제 선택: seoul-sns

상파울루 리전

**CloudWatch > 경보 > 경보 생성**

**지표 및 조건 지정**

**지표**

이 경보는 5 분 내 1개의 데이터 포인트에 대해 파란색 줄이 빨간색 줄을 초과할 때 트리거됩니다.

Count: 1  
지표 이름: Errors  
FunctionName: user-auth-lambda  
Resource: user-auth-lambda  
통계: 평균

**작업 구성**

**알림**

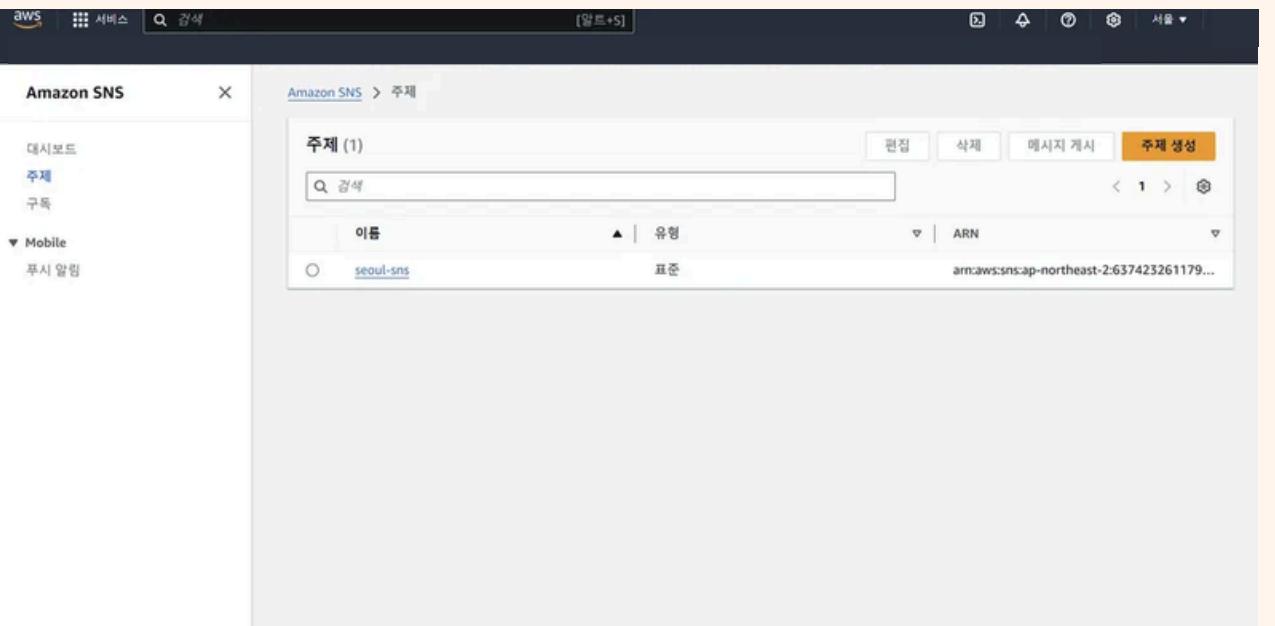
경보 상태 드리거: 경보 상태 (선택)

다음 SNS 주제에 알림을 보냅니다. 알림을 수신할 SNS(Simple Notification Service) 주제를 정의합니다.

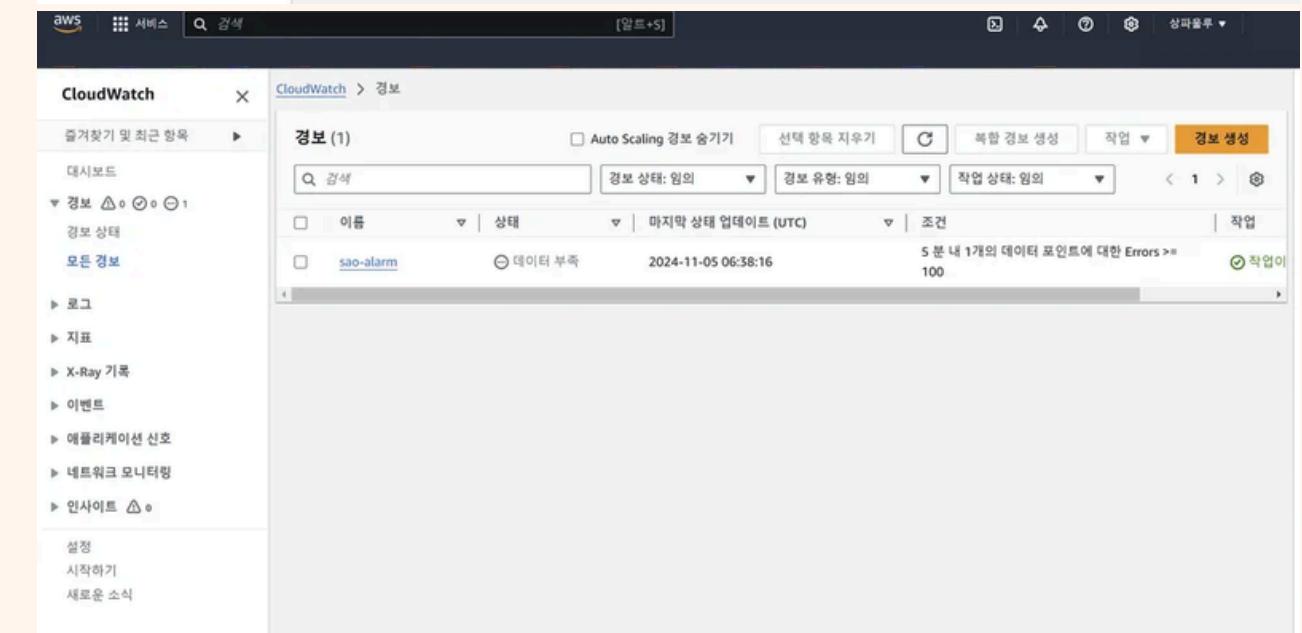
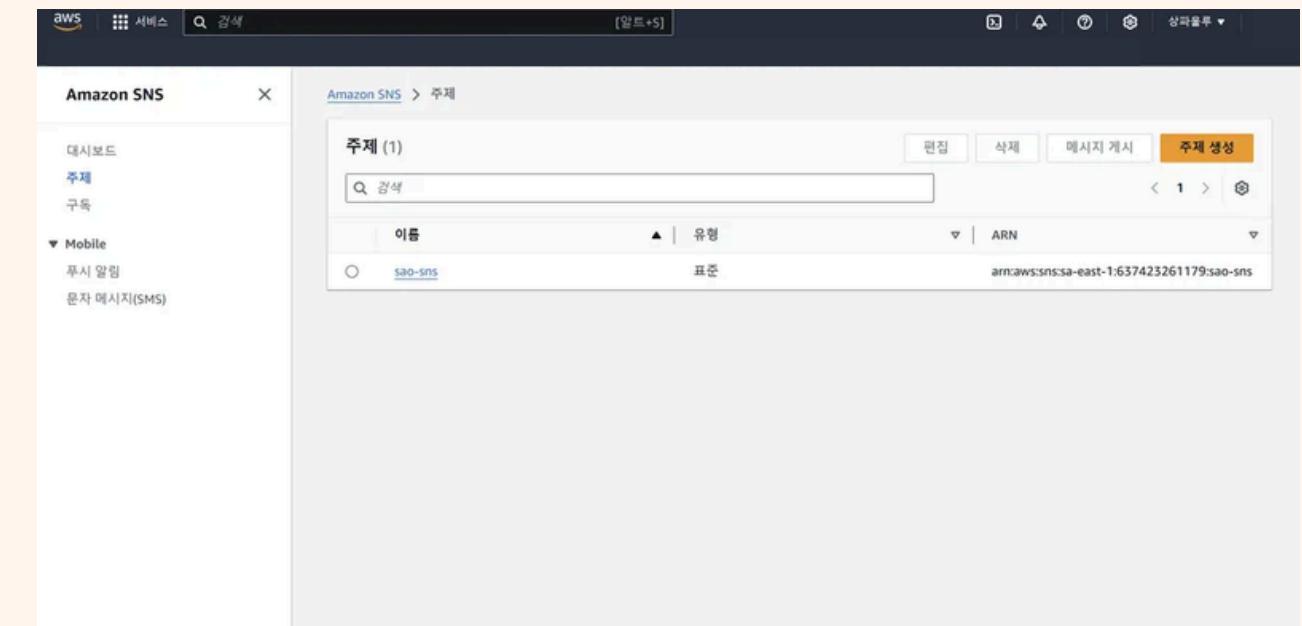
기존 SNS 주제 선택: sao-sns

# CloudWatch & SNS 활성화

# 서울 리전



상파울루 리전

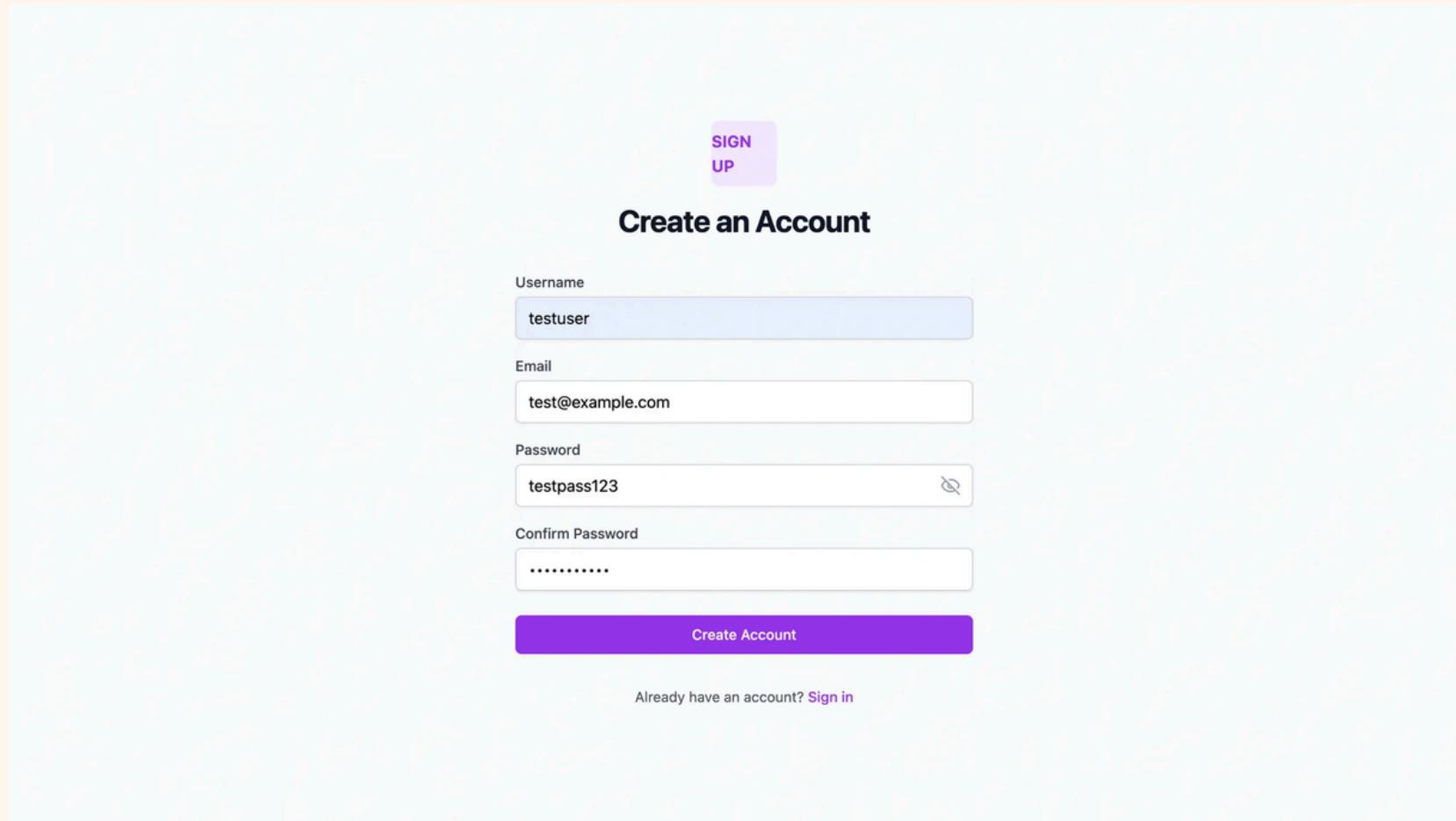


CHAPTER 03

# 서비스 구현

# 사용자 인증 및 저장소 연동 테스트

> 회원가입 시 사용자 정보 RDS에 데이터 저장



```
MySQL [video_storage]> SELECT * FROM users;
+-----+-----+-----+-----+
| id   | username | email      | password_hash          | created_at |
+-----+-----+-----+-----+
| 1cda9ffb-3e1e-4271-9ac8-da3ee5cb6c4b | testuser  | test@example.com | 7e6e0c3079a08c5cc6036789b57e951f65f82383913ba1a49ae992544f1b4b6e | 2024-10-06 14:06 |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

# 사용자 인증 및 저장소 연동 테스트

> 영상 업로드 시 사용자 ID 기반 S3 디렉토리 자동 생성 및 파일 저장

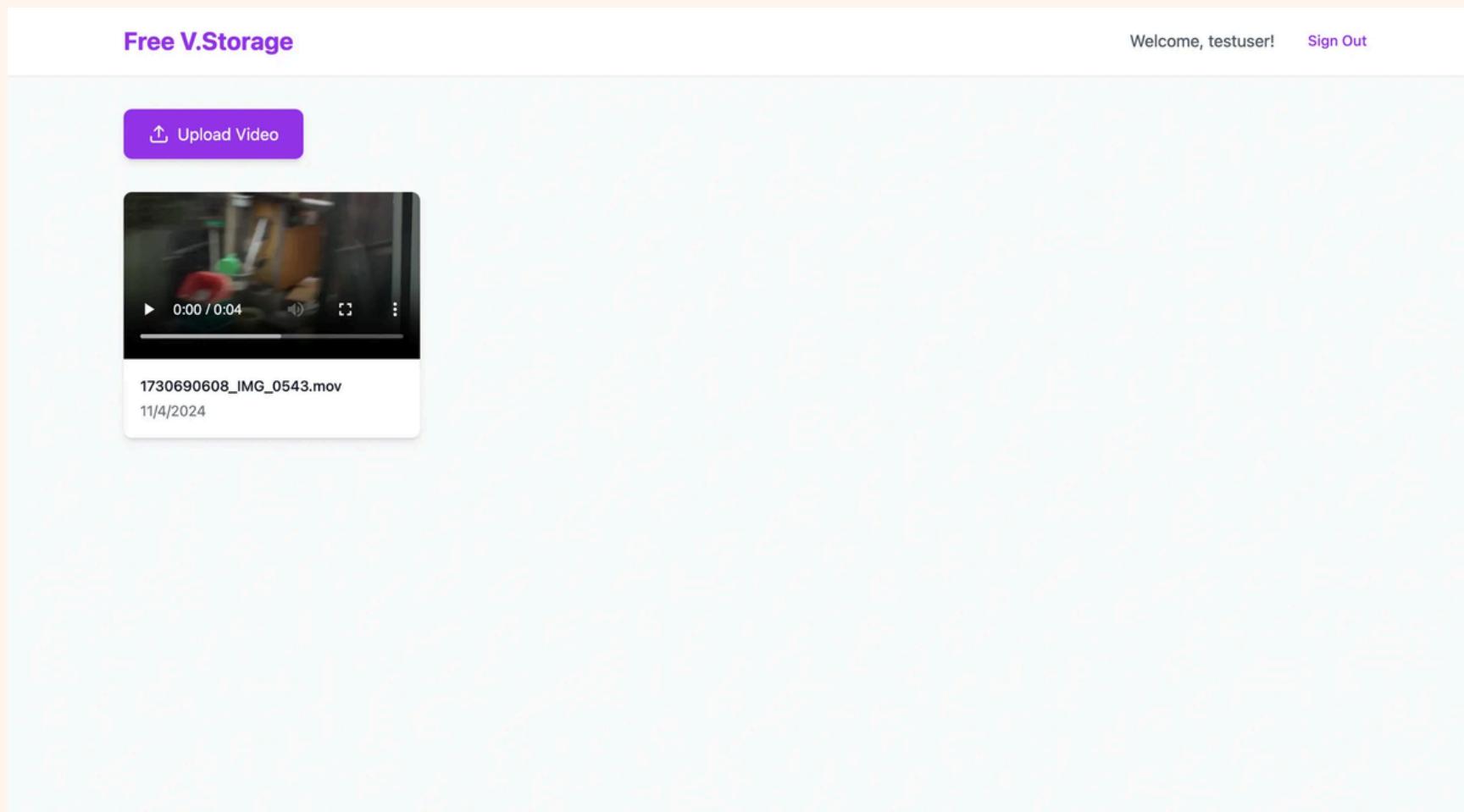
Free V.Storage

Welcome, testuser! Sign Out

Upload Video

0:00 / 0:04

1730690608\_IMG\_0543.mov  
11/4/2024



Amazon S3 > Buckets > video-storage-bucket-2024-1 > videos/ > 1cda9ffb-3e1e-4271-9ac8-da3ee5cb6c4b/

1cda9ffb-3e1e-4271-9ac8-da3ee5cb6c4b/

Copy S3 URI

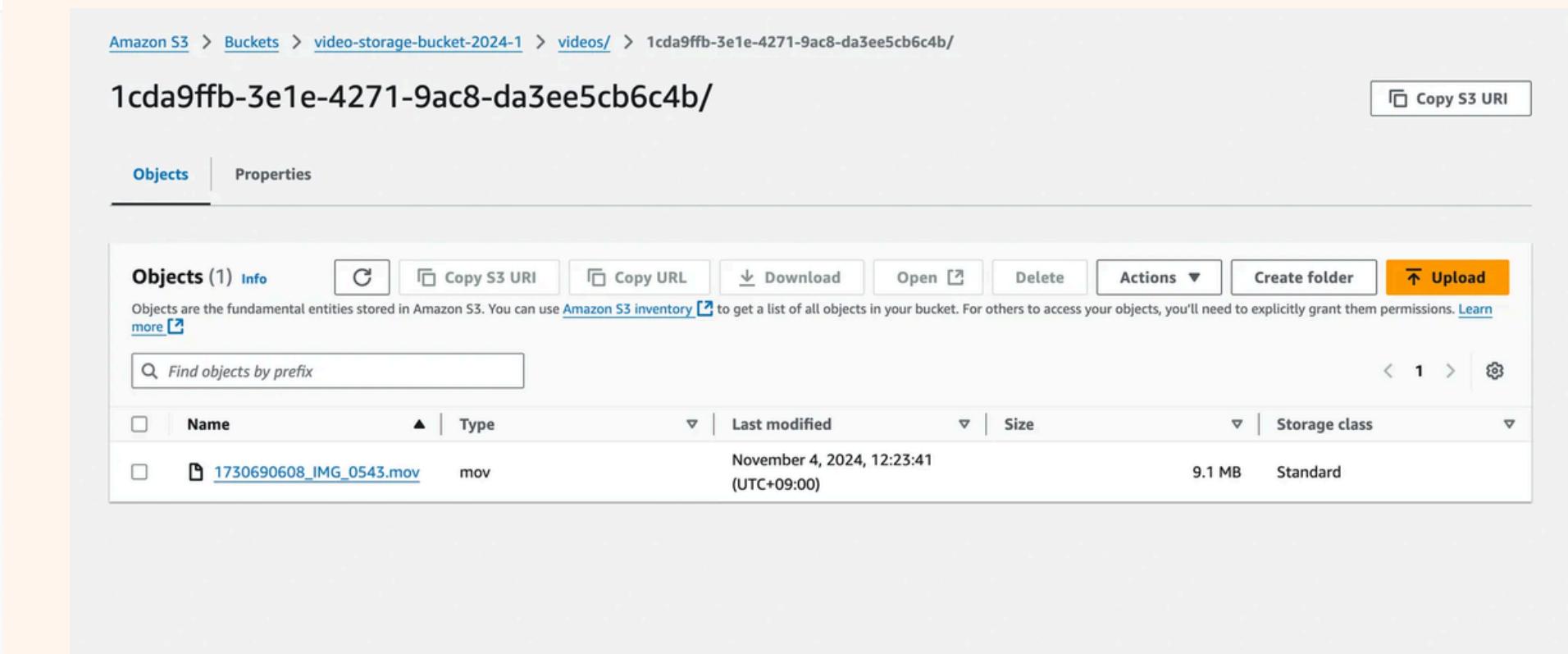
Objects Properties

Objects (1) Info C Copy S3 URI Copy URL Download Open Actions Create folder Upload

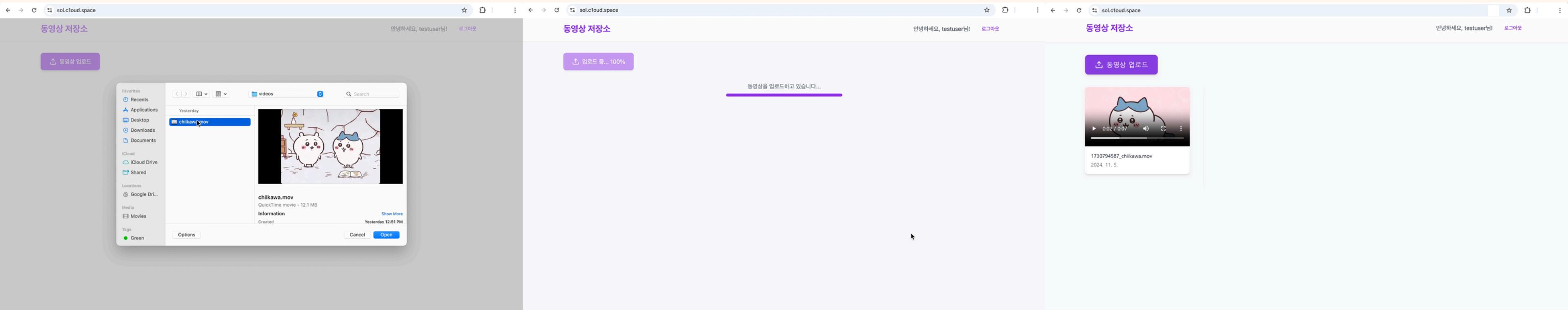
Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Find objects by prefix

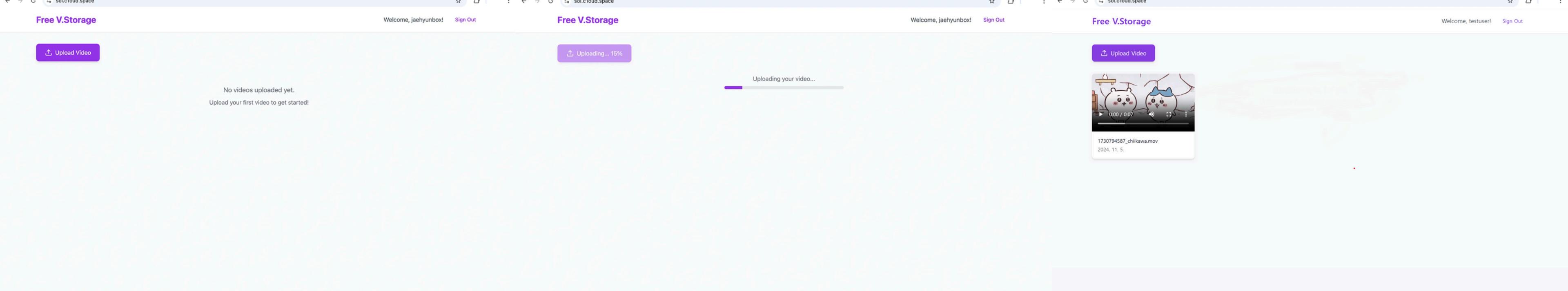
Name	Type	Last modified	Size	Storage class
1730690608_IMG_0543.mov	mov	November 4, 2024, 12:23:41 (UTC+09:00)	9.1 MB	Standard



# 웹 서비스 배포 (서울 리전 / 한국어 버전 구현 화면)



# 웹 서비스 배포 (상파울루 리전 / 영어 버전 구현 화면)



# 서울과 상파울루 리전 S3 버킷 간 양방향 복제 동작 확인

- > 서울 리전에 업로드된 영상이 서울 리전 버킷에 저장
- > 상파울루 리전 버킷으로 자동 복제 완료

The screenshot shows the AWS S3 console interface in the Seoul region. The left sidebar includes sections for Bucket, Access Grants, Lambda, IAM Access Analyzer, and Storage Lens. The main content area displays a single object named "236c68ed-27d4-4120-99f6-2afda7371fc6/" under the "Videos" folder. The object details show it was uploaded on November 5, 2024, at 5:16:33 PM KST, is 11.6MB in size, and has a Standard storage class. The "Actions" bar at the top right contains buttons for "S3 URI 복사" (Copy S3 URI), "URL 복사" (Copy URL), "다운로드" (Download), "열기" (Open), "삭제" (Delete), "작업" (Actions), and "폴더 만들기" (Create Folder). A note below the object details states: "객체는 Amazon S3에 저장되어 있는 기본 엔터티입니다. Amazon S3 인벤토리 [?]를 사용하여 버킷에 있는 모든 객체의 목록을 얻을 수 있습니다. 다른 사용자가 객체에 액세스할 수 있게 하려면 명시적으로 권한을 부여해야 합니다. 자세히 알아보기 [?]".

The screenshot shows the AWS S3 console interface in the São Paulo region. The left sidebar includes sections for Bucket, Access Grants, Lambda, IAM Access Analyzer, and Storage Lens. The main content area displays a single object named "236c68ed-27d4-4120-99f6-2afda7371fc6/" under the "Videos" folder. The object details show it was uploaded on November 5, 2024, at 5:16:33 PM KST, is 11.6MB in size, and has a Standard storage class. The "Actions" bar at the top right contains buttons for "S3 URI 복사" (Copy S3 URI), "URL 복사" (Copy URL), "다운로드" (Download), "열기" (Open), "삭제" (Delete), "작업" (Actions), and "폴더 만들기" (Create Folder). A note below the object details states: "객체는 Amazon S3에 저장되어 있는 기본 엔터티입니다. Amazon S3 인벤토리 [?]를 사용하여 버킷에 있는 모든 객체의 목록을 얻을 수 있습니다. 다른 사용자가 객체에 액세스할 수 있게 하려면 명시적으로 권한을 부여해야 합니다. 자세히 알아보기 [?]".

CHAPTER 04

# 프로젝트 결과

- 04-1 프로젝트 성과
- 04-2 보완점 및 향후 계획
- 04-3 결론



### 하이브리드 클라우드 아키텍처 구현

온프레미스와 클라우드 리소스의 안전한 통합을 위한 Site-to-Site VPN 구축 및 하이브리드 환경 최적화



### 글로벌 서비스 가용성 향상

Global Accelerator를 활용한 지능적인 트래픽 라우팅으로 지역별 최적화된 성능 및 가용성 확보



### 확장성 및 비용 효율성 달성

Auto Scaling을 활용한 수요 기반 자동 확장으로 리소스 최적화 및 운영 비용 절감



### 보안 체계 강화

WAF, IAM, Certificate Manager를 통한 다층적 보안 아키텍처 구현 및 인증서 기반 통신 보안 확보



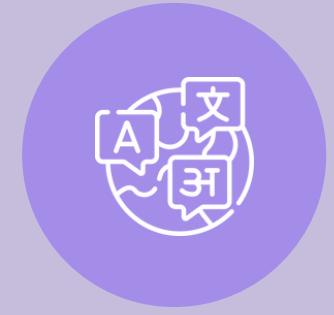
## 데이터 전송 및 관리 효율성 향상

S3, Lambda, 그리고 API Gateway를 활용한  
효율적인 데이터 관리 시스템 구축  
> S3 버킷 간 데이터 복제를 통한 가용성 확보  
> 서비스 아키텍처를 통한 유연한 확장성 제공



## 실시간 모니터링 및 장애 대응력 향상

CloudWatch와 SNS 알림을 활용한 포괄적  
모니터링 체계 구축  
> 시스템 상태 실시간 모니터링 구현  
> 즉각적인 알림 시스템으로 장애 대응 시간 단축



## 다국어 콘텐츠 제공

Global Accelerator와 지역별 최적화를 통한  
현지화된 서비스 제공  
> 리전별 맞춤형 콘텐츠 전송 체계 구축  
> 지역 특성을 고려한 서비스 최적화



## 운영 효율성 및 자동화 개선

CloudFormation을 통한 인프라의 자동화된  
관리 체계 구현  
> 인프라 프로비저닝 자동화  
> 운영 관리 효율성 최적화



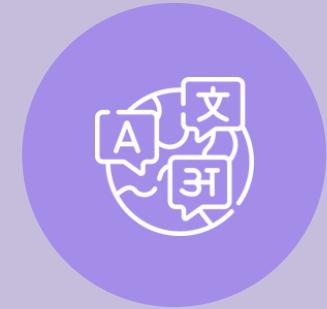
### 온프레미스와 클라우드 간 데이터 전송 최적화

현재 전송 속도와 효율성 개선이 필요한 상황  
> AWS Direct Connect 도입을 통한 네트워크 성능 향상



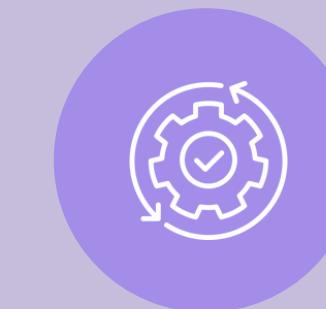
### 비용 최적화 필요성

클라우드 리소스 사용의 효율적 관리 필요  
> 비용 관리 도구와 예약 인스턴스를 활용 계획 수립



### 다국어 콘텐츠 관리 시스템 개선

콘텐츠 관리 프로세스의 효율화 필요  
> 자동화 시스템 도입으로 관리 효율성 극대화



### 보안 및 인증 시스템 강화 필요

보안시스템의 지속적인 개선 필요  
> AWS KMS 도입으로 데이터 암호화 강화



글로벌 미디어 기업 A사는 콘텐츠 전송 지연, 보안 취약성, 확장성 제한 등 다양한 운영상의 문제에 직면해 있었습니다. 이를 해결하기 위해 하이브리드 클라우드 환경을 성공적으로 구축하였고, 다음과 같은 주요 성과를 달성하였습니다:

### 1. 글로벌 서비스 품질 향상

- 지역별 콘텐츠 전송 지연시간 대폭 감소
- 서비스 안정성 및 가용성 크게 개선

### 2. 보안 및 규정 준수 강화

- WAF 및 다층 보안 아키텍처를 통한 보안 체계 고도화
- 글로벌 데이터 보호 규정 준수 체계 확립

### 3. 운영 효율성 및 확장성 확보

- Auto Scaling을 통한 탄력적인 리소스 관리 구현
- 클라우드 기반 자동화로 운영 효율성 향상
- 수요 변화에 따른 유연한 확장 가능

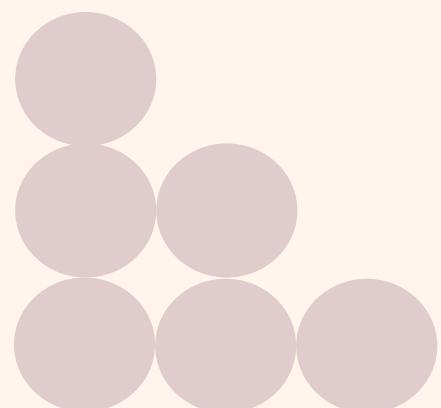
본 프로젝트를 통해 구축된 하이브리드 클라우드 환경은 A사의 글로벌 미디어 서비스 제공을 위한 안정적이고 확장 가능한 기반을 마련하였으며, 향후 비즈니스 성장에 따른 새로운 요구사항에도 유연하게 대응할 수 있는 인프라를 확보하였습니다.

# Q & A

---

감사합니다

---



이수민 | lucyi030920@gmail.com