

CLOUD ARCHITECT PROJECT

AWS 클라우드 마이그레이션

Table of Contents

1. 프로젝트 개요	3
1.1. 도입	3
1.2. 역할 분담	3
1.3. WBS(WORK BREAKDOWN STRUCTURE)	4
2. HYBRID-BTS.....	4
2.1. 고객사 인터뷰	4
2.2. 고객 요구 분석.....	5
2.3. HYBRID-BTS 웹사이트.....	5
3. 마이그레이션 시나리오	8
3.1. HYBRID-BTS 의 기존 온프레미스 환경	8
3.2. AWS 클라우드 마이그레이션 이후 환경	10
4. 온프레미스 인프라스트럭쳐	12
5. AWS 인프라스트럭쳐	15
5.1. AMAZON VPC(VIRTUAL PRIVATE CLOUD)	15
5.2. AMAZON EC2(ELASTIC COMPUTE CLOUD).....	21
5.2.1. Bastion.....	21
5.2.2. AMI 및 Launch Template.....	24
5.2.3. ALB(Application Load Balancer).....	30
5.2.4. Auto Scaling	34
5.3. AMAZON RDS(AMAZON RELATIONAL DATABASES)	39
5.3.1. Amazon RDS(Amazon Relational Databases)	39
5.3.2. Read-Only Replica.....	48
5.4. AMAZON ELASTICACHE FOR REDIS	51
5.5. AMAZON ECS(ELASTIC CONTAINER SERVICE)	56
5.5.1. 컨테이너	56
5.5.2. Amazon ECR(Elastic Cluster Repository)	57
5.5.3. Amazon ECS(Elastic Container Service).....	57
5.6. AMAZON VPC PEERING	68
5.7. AWS CLOUDFORMATION	73

5.8. AWS GLOBAL ACCELERATOR	80
5.9. AMAZON ROUTE 53	88
5.10. AMAZON S3.....	90
5.10.1. Amazon S3	90
5.10.2. AWS Elemental MediaConvert.....	91
5.11. AMAZON CLOUDFRONT	95
5.12. ACM(AWS CERTIFICATE MANAGER)	100
5.13. AWS WAF(WEB APPLICATION FIREWALL).....	104
5.14. AMAZON EVENTBRIDGE.....	110
6. 결론	121

1. 프로젝트 개요

1.1. 도입

오늘날, 많은 기업들이 클라우드 전환 또는 하이브리드 클라우드의 도입을 고려하거나 진행하고 있다. MSP의 입장에서 쇼핑몰 'HYBRID-BTS'의 온프레미스 기반 IT 인프라를 AWS 클라우드로 마이그레이션 하였다. 고객사의 E-Commerce 중심 사업에 클라우드 혁신을 접목하여 비지니스의 성장을 도모하고자 한다.

1.2. 역할 분담

이재현(조장)

프로젝트 관리

On-premise

- NAT, DHCP, OSPF
- ASA, Failover, VRRP, ACL

AWS

- CloudFormation

TEST

- On-premise Network

보고서 작성

김영진

On-premise

- NAT, DHCP, OSPF
- ASA, Failover, VRRP, ACL

AWS

- CloudFront, Global Accelerator

TEST

- On-premise Network
- CloudFront, Global Accelerator

보고서 작성

박예슬

프로젝트 기획

아키텍쳐 설계

AWS

- VPC, ALB, Auto Scaling
- Container, ECS, ECR
- Redis, ACM, WAF

TEST

- Virginia Region, WAF

PPT, 보고서 작성(총괄)

최동환

아키텍쳐 설계

웹사이트 개발

On-premise

- On-premise 환경 서버 구축(DNS, WEB, DB)

AWS

- VPC, ALB, Auto Scaling
- RDS, Redis, Multi-AZ, Read-only replica
- Route53, GlobalAccelerator, CloudWatch
- Lambda, EventBridge, S3, MediaConvert

TEST

- Auto Scaling, Global Accelerator

PPT, 보고서 작성

황재성

On-premise

- On-premise 환경 서버 구축(DNS, WEB, DB)

AWS

- VPC, ALB, Auto Scaling
- RDS, Redis, Multi-AZ, Read-only replica
- VPC peering

TEST

- VPC peering, RDS replica
- Seoul Region

PPT, 보고서 작성

1.3. WBS(Work Breakdown Structure)

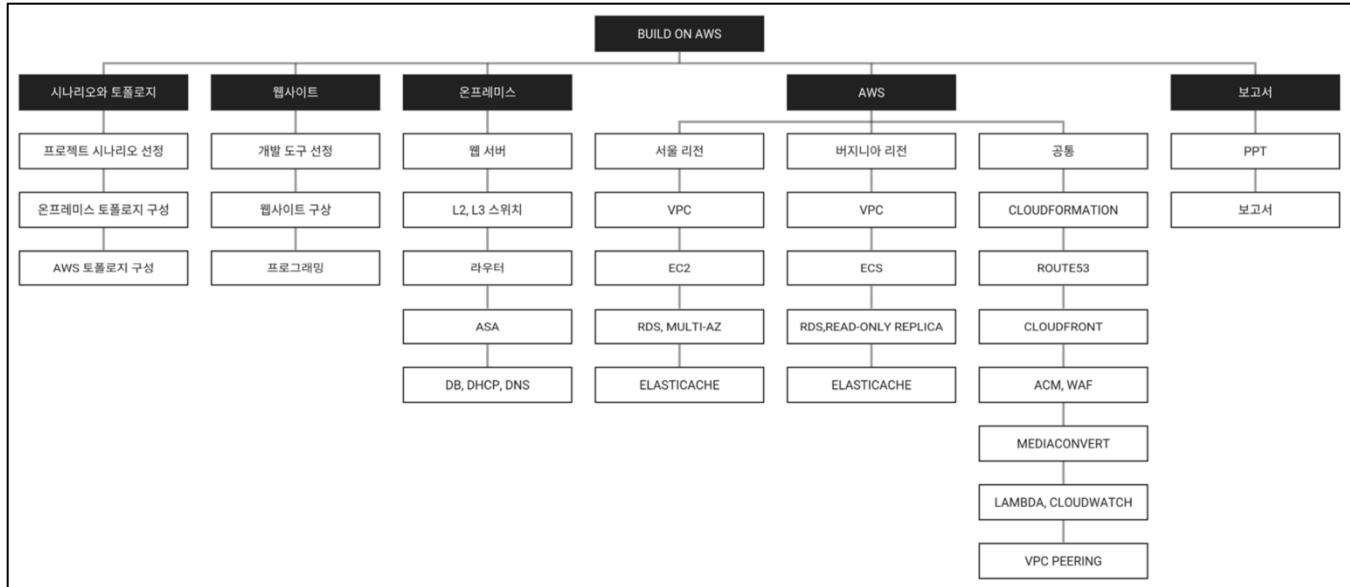


그림 1 프로젝트 WBS

2. HYBRID-BTS

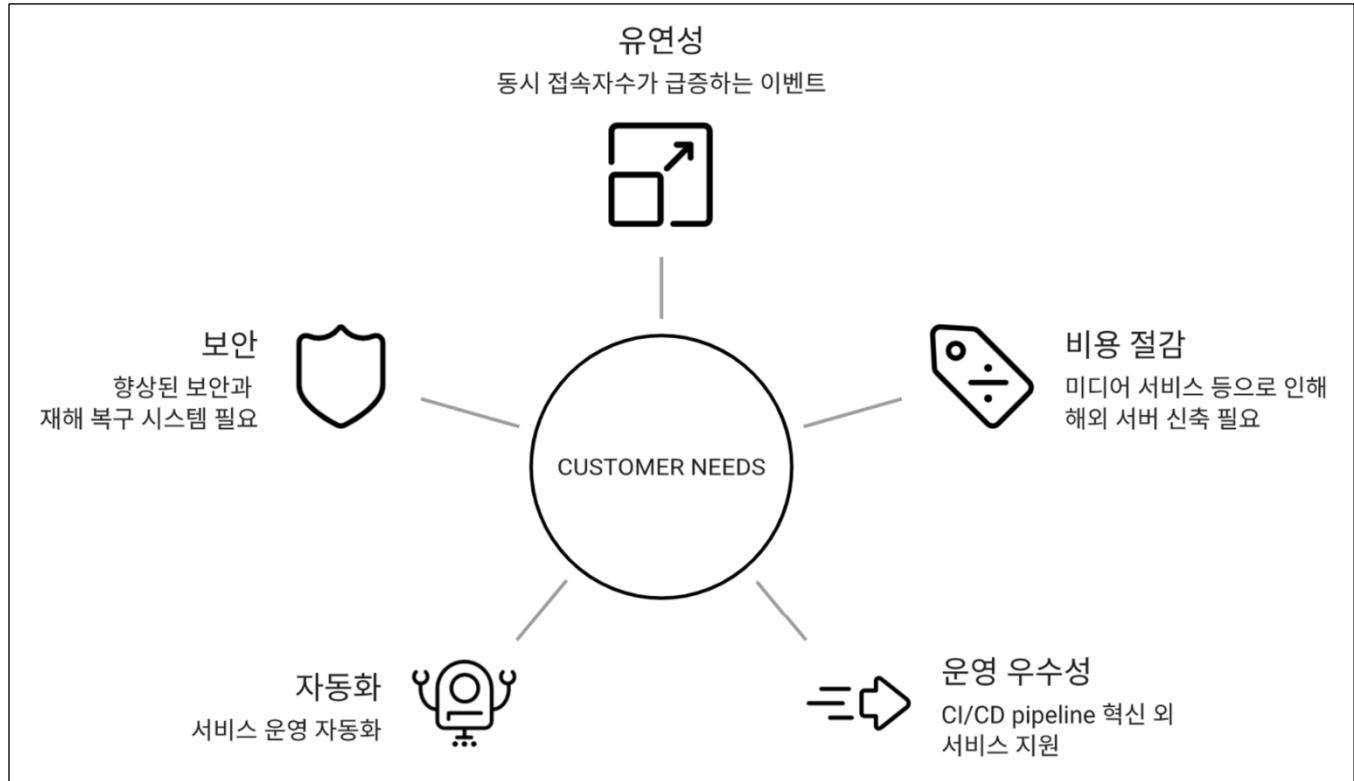
2.1. 고객사 인터뷰

“저희 HYBRID-BTS는 방탄소년단 팬을 위한 쇼핑몰입니다. 음반을 비롯해서 사진, 인형, 의류, 생활용품, 동영상 등 수천 가지의 방탄소년단 관련 상품을 판매하고 있습니다.

현재 IT 인프라는 온프레미스 환경입니다. 그런데 방탄소년단의 인기가 높아지고 글로벌화 됨에 따라 온프레미스 인프라의 한계를 경험하고 있습니다. 예를 들어 얼마 전 방탄소년단 친필 싸인 포토카드 1000 장 한정판 굿즈 판매 이벤트가 있었습니다. 전 세계에서 몇십만 명이 동시에 접속해서 서버가 다운되어 항의가 빗발치는 재앙같은 결과를 낳았습니다. 그뿐만 아니라, 저희 HYBRID-BTS는 방탄소년단 동영상 스트리밍 서비스를 제공하고 있는데 해외에서 이용하는 고객들이 잦은 버퍼링과 저화질의 경험을 호소하고 있습니다.

저희가 IT 인프라의 클라우드 환경 이전을 통해 이러한 문제를 해결할 수 있을까요?”

2.2. 고객 요구 분석



2.3. HYBRID-BTS 웹사이트

www.hybridbts.com 소개

BTS 의 굿즈 상품을 판매하는 쇼핑몰이다. 음반을 비롯해서 사진, 인형, 의류, 생활용품, 동영상 등 방탄소년단 관련 상품을 판매한다. 한국을 비롯한 전세계 곳곳의 방탄소년단 팬들이 주요 고객층이다.

개발 도구



- **Bitnami** 개발 스택용 소프트웨어 패키지 및 설치 라이브러리
- **Bitnami WAMP** 웹서버 구축에 사용하는 프로그램으로써 Window에 Apache(웹 서버), MySQL(데이터베이스), PHP(프로그래밍 언어)를 설치해준다.

웹사이트 구성

1. 메인 페이지

BTS Category ▾ Review Event Media Login Sign-up

MAP OF THE SOUL ON:E DVD/BLU-RAY

단독판매 구매하라 가기

위버스 카드

팬을 위한 모든 순간,
위버스 카드

아티스트 포토 플레이트와 다양한 혜택까지!

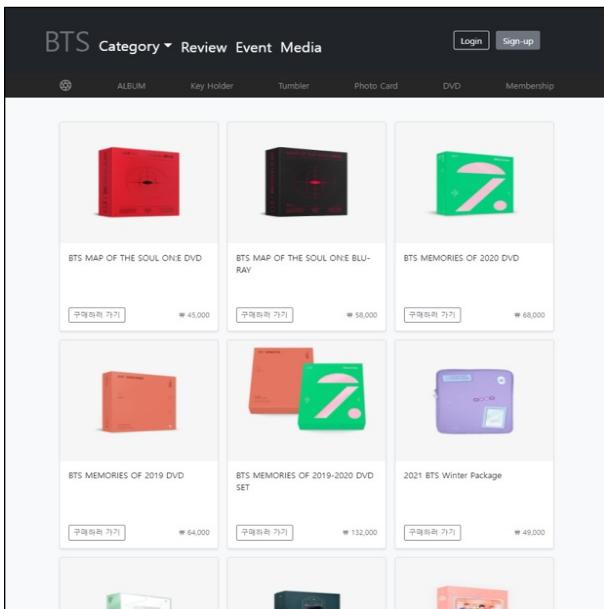
구매하라 가기

BTS GLOBAL OFFICIAL FANCLUB ARMY MEMBERSHIP

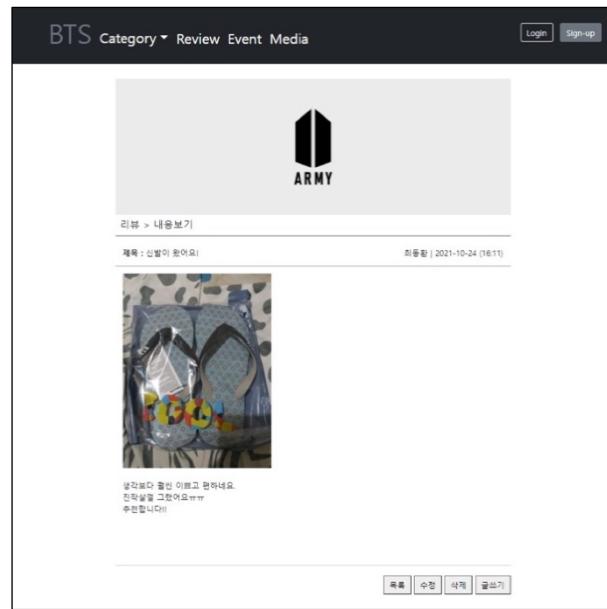
BTS 공식 멤버십 가입은 위버스샵에서!

구매하라 가기

2. 카탈로그 페이지



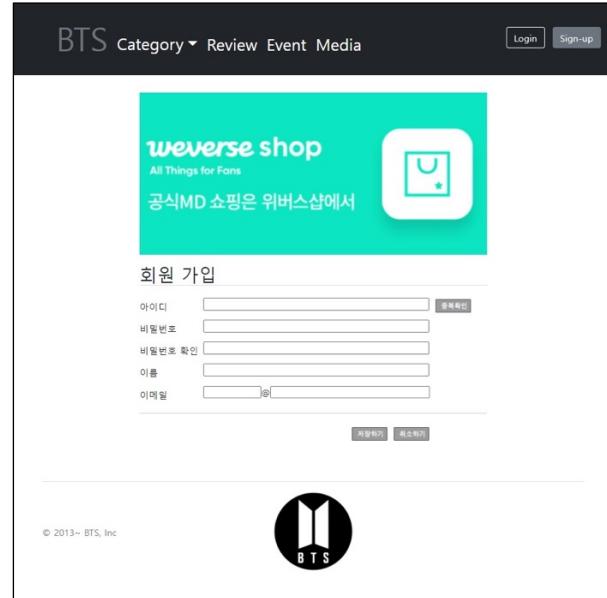
3. 리뷰 페이지



4. 미디어 스트리밍 페이지



5. 회원가입 페이지



3. 마이그레이션 시나리오

3.1. HYBRID-BTS 의 기존 온프레미스 환경

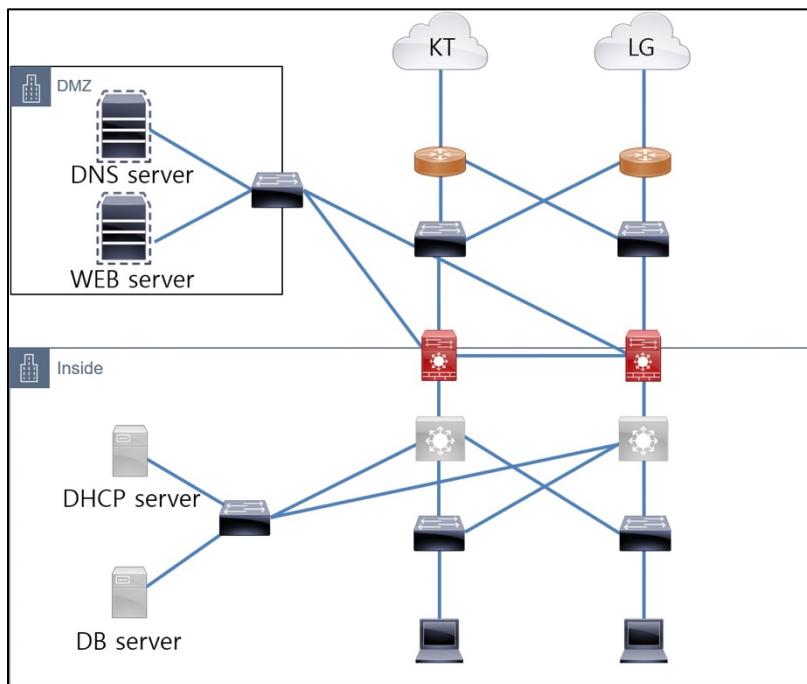


그림 3 HYBRID-BTS 의 기존 온프레미스 환경 토플로지

기존 On-premise IT 인프라는 DMZ 와 내부 서버로 이루어져 있다. 두 영역은 L3 스위치와 ASA 방화벽을 걸쳐 통신하고 있다.

DMZ 에는 DNS, WEB 서버 각각 한 대가 있는데 DNS 서버가 도메인 이름을 IP 주소로 변환한다. 내부 서버에서는 데이터베이스 서버, DHCP 서버를 각각 한 대씩 운영하고 있고 데이터베이스 서버에 웹 서버가 사용하는 데이터를 저장한다. DHCP 서버는 내부 사설 IP 를 자동 할당 및 관리한다. 게이트웨이로 사용되는 L3 스위치는 이중화 상태로 각각 2 대의 L2 스위치와 연결되어 Inbound/Outbound 트래픽을 라우팅 한다.

보안을 위해 ASA 이중화를 설정하였다. HYBRID-BTS의 내부 자원들은 2 대의 ASA 방화벽을 거쳐서 외부와 연결한다. 메인 ASA에 문제가 생길 경우 서브 ASA가 메인 ASA의 역할을 대체하게 하여 안정성을 확보했다. 방화벽은 2 대의 L2 스위치에 연결되어 2 대의 라우터를 거쳐 ISP에 연결된다.

ISP는 KT와 LG U+ 두 회사를 사용하여 멀티 호밍으로 구성하였다. 한쪽에 오류가 발생되었을 경우를 대비해 BGP 프로토콜을 이용한 이중화 상태이다.

기존 구조의 문제점

IT 인프라가 물리적인 장치들로 구성되어 있기에 구조가 복잡하고 관리자가 관리해야 할 대상이 많다. 하드웨어 유지/보수 비용이 높을 뿐 아니라 향후 쇼핑몰의 수요가 예측 불가능한 상황에서 인프라 확장에 제약이 크다. 요구에 맞춰 확장을 한 경우에도 유휴 자원이 낭비되는 측면이 존재한다. 내부적인 인프라 외에도 애플리케이션 운영에 필요한 여타 서비스들을 아웃소싱을 맡기거나 다양한 서드 파티의 도구를 사용하여 분산된 서비스 이용으로 관리가 용이하지 않다.

3.2. AWS 클라우드 마이그레이션 이후 환경

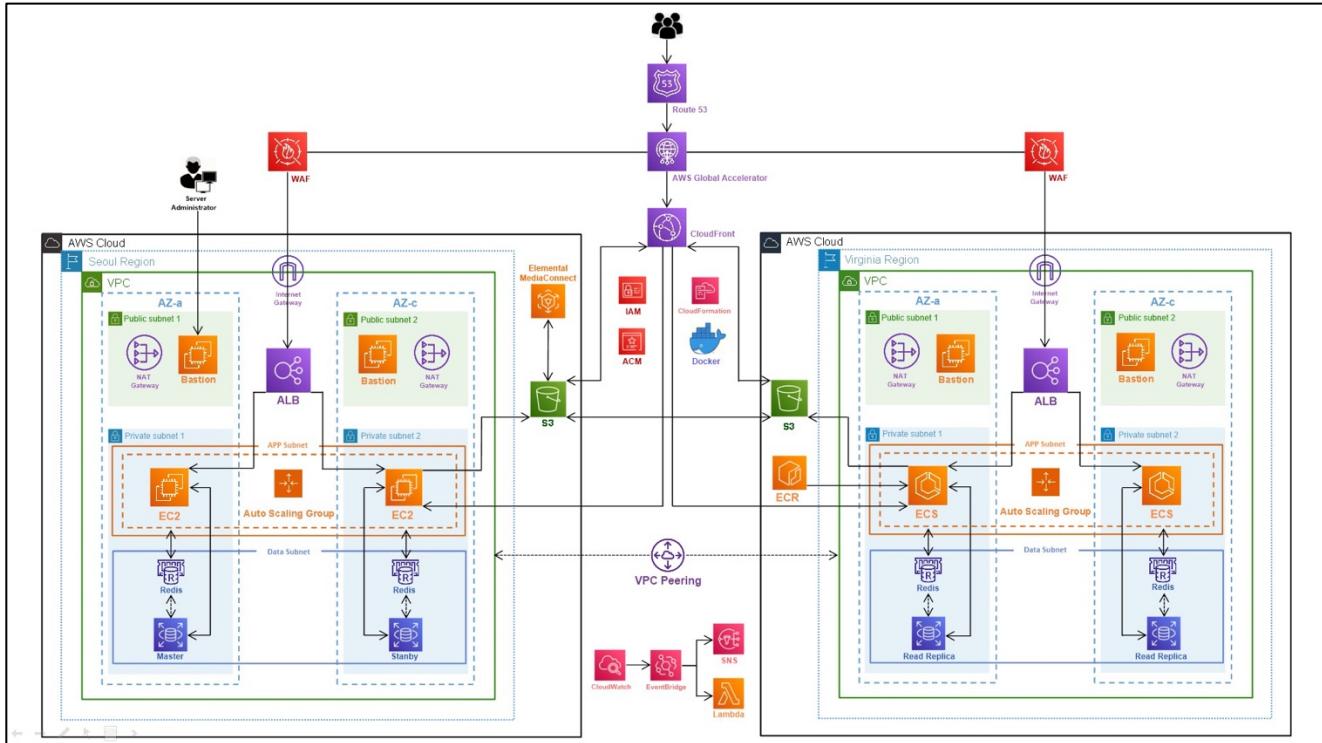


그림 4 HYBRID-BTS의 새로운 AWS 환경 토플로지

HYBRID-BTS의 모든 IT 인프라를 AWS 클라우드 환경 안에 배치하였다. 서울 리전에 10.0.0.0/16 대역 네트워크의 VPC를 생성한 뒤 퍼블릭 서브넷과 프라이빗 서브넷을 각각 가용 영역 a와 c에 분산 배치하여 가용성과 안정성을 높였다. 퍼블릭과 프라이빗 라우팅 테이블을 생성한 뒤 내부와 외부와 통신이 되도록 서브넷들을 각각 연결하였다. 방화벽은 각 서버에 적합한 보안 그룹 설정을 하여 특정 트래픽 소스를 허용하고 포트를 개방하였다. 이에 더해 WAF를 각 리전의 ALB와 CloudFront에 배치해서 위협과 공격의 필터링과 차단으로 보안을 강화하였다.

웹 서버는 기본적으로 각 가용 영역마다 한 대씩 운영되나, AWS의 Auto Scaling 기능을 사용하여 사용자의 요청에 의한 컴퓨팅 리소스의 요구 정도에 따라 서버 개수가 확장되고 축소 된다. 웹 서버와 동일한 가용 영역에 RDS 데이터베이스 서버와 Redis 서버를 배치하였다. DNS 서비스인 Route 53로 요청을 효과적으로 라우팅하도록 했다. CDN 서비스인 CloudFront를 정적 컨텐츠가 담겨 있는 S3 버킷에 연결하여 동영상 등 정적 컨텐츠 제공의 속도와 성능을 눈에 띄게 높였다. 이렇게 생성된 인프라를 IaC(Infrastructure as Code)

서비스인 CloudFormation 소스 코드로 생성하여 빠르고, 편리하고, 지속성 있는 재해 복구책을 마련하였다. 그 외 관리자의 각종 서비스 접근은 IAM 역할과 정책 부여를 사용하여 효율적으로 분리분배 하였다. SSL/TLS 인증서로는 ACM 을 사용하여 AWS로부터 발급받은 퍼블릭 인증서를 HTTPS 프로토콜 요청에 활용하고 있다.

이에 더해 서울 리전의 장애 발생시 재해 복구를 위해 버지니아 리전을 신설하였다. 서울 리전과 버지니아 리전에 Global Accelerator 를 배치하여 사용자가 지리적으로 가까운 리전의 서버로 라우팅되게 하였다. 버지니아 리전에는 웹 서버에 컨테이너 기술을 도입해 배포하였다. 서울 리전과 버지니아 리전을 VPC peering 하여 웹 서버, RDS 와 Redis 등 프라이빗 리소스에 서로 접근 가능하도록 연결했다.

4. 온프레미스 인프라스트럭쳐

ASA

```
* 1 ASA-1 * + DMZ
access-list DMZ_In extended permit udp object SVR2012-B object SVR2012-A eq domain
access-list DMZ_In extended permit tcp object SVR2012-B object SVR2012-A eq domain
access-list DMZ_In extended permit udp object SVR2012-B any eq domain
access-list DMZ_In extended deny ip any any log warnings interval 60

* 1 ASA-1 * + Outside
access-list Outside_In extended deny ip object-group Bogon-Prefix object DMZ_NET
access-list Outside_In extended permit udp any object SVR2012-B eq domain
access-list Outside_In extended permit object-group WEB any object SVR2000
access-list Outside_In extended deny ip any any log warnings interval 60


* 1 ASA-1 * + Inside
access-list Inside_In extended permit object-group WEB object-group HQ_NET object SVR2000
access-list Inside_In extended permit udp object-group HQ_NET object SVR2012-B eq domain
access-list Inside_In extended permit tcp object SVR2012-A object SVR2012-B eq domain
access-list Inside_In extended deny ip any object DMZ_NET log warnings interval 60
```

그림 5 ASA 이중화

회선의 각 이더넷 별로 연결된 영역(**Inside, Outside, DMZ**)에 다른 보안 레벨을 부여하고, 사용자가 원하는 특정 프로토콜만 허용하는 보안 레벨 시스템이다. 필요한 트래픽만 통신 가능하게 한다.

FHRP 이중화

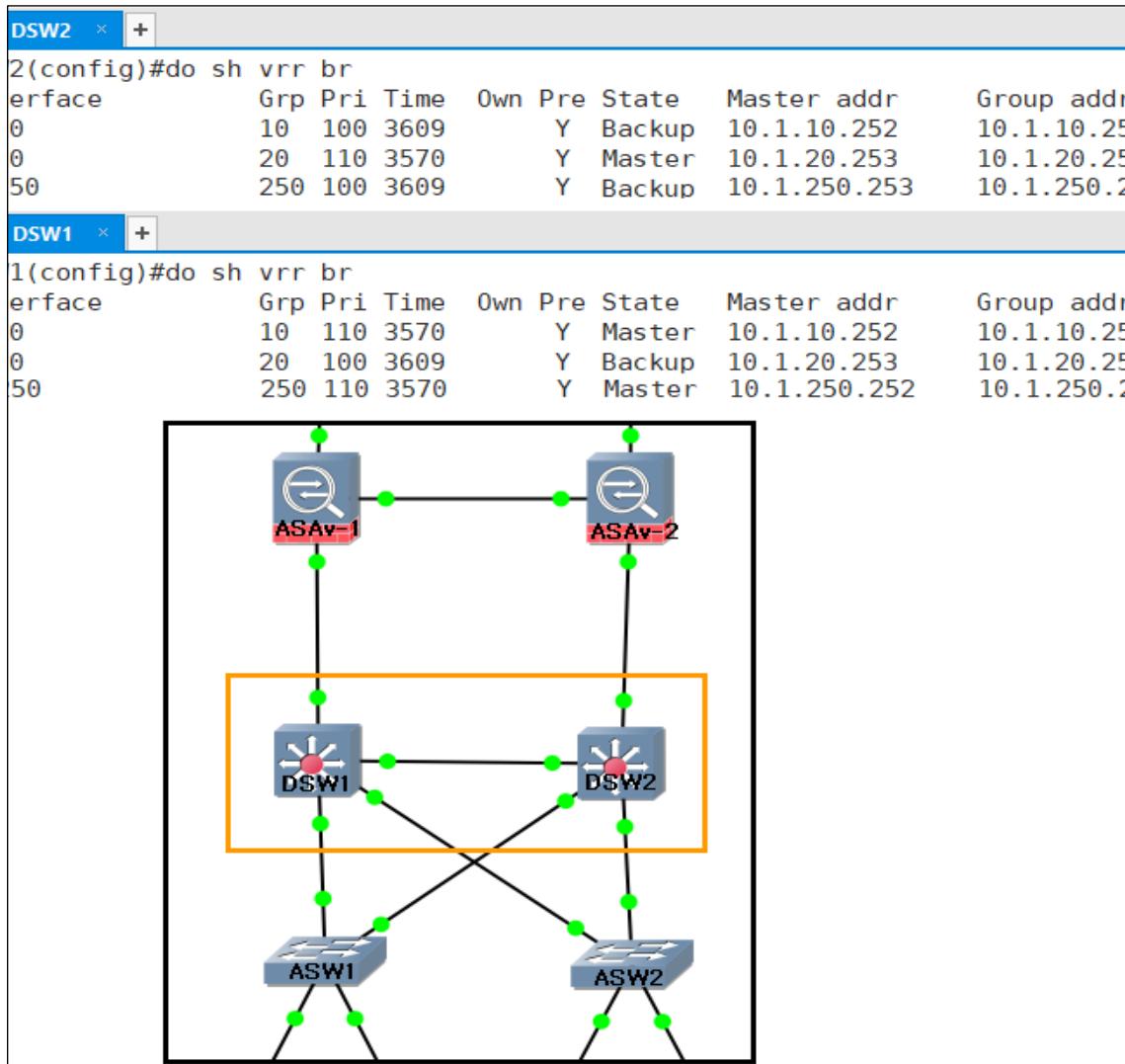


그림 6 L3 스위치 이중화

FHRP란 네트워크 환경에서 Gateway 역할을 하는 라우터나 L3 스위치 장비를 이중화 하는 Protocol이다.

2 대의 Gateway 장치 중 하나가 다운되면 자동으로 다른 장비가 그 역할을 수행하게 된다.

IEEE 표준 프로토콜인 VRRP을 사용하였으며 여러 대의 라우터를 그룹으로 묶어 하나의 가상 IP 주소를 부여하고, 마스터로 지정된 라우터에 장애 발생 시 백업 라우터가 마스터로 전환되는 이중화 프로토콜이다. 이에 따라 마스터 라우터의 장애로 인한 네트워크 서비스의 중단을 예방할 수 있다.

DHCP 서버

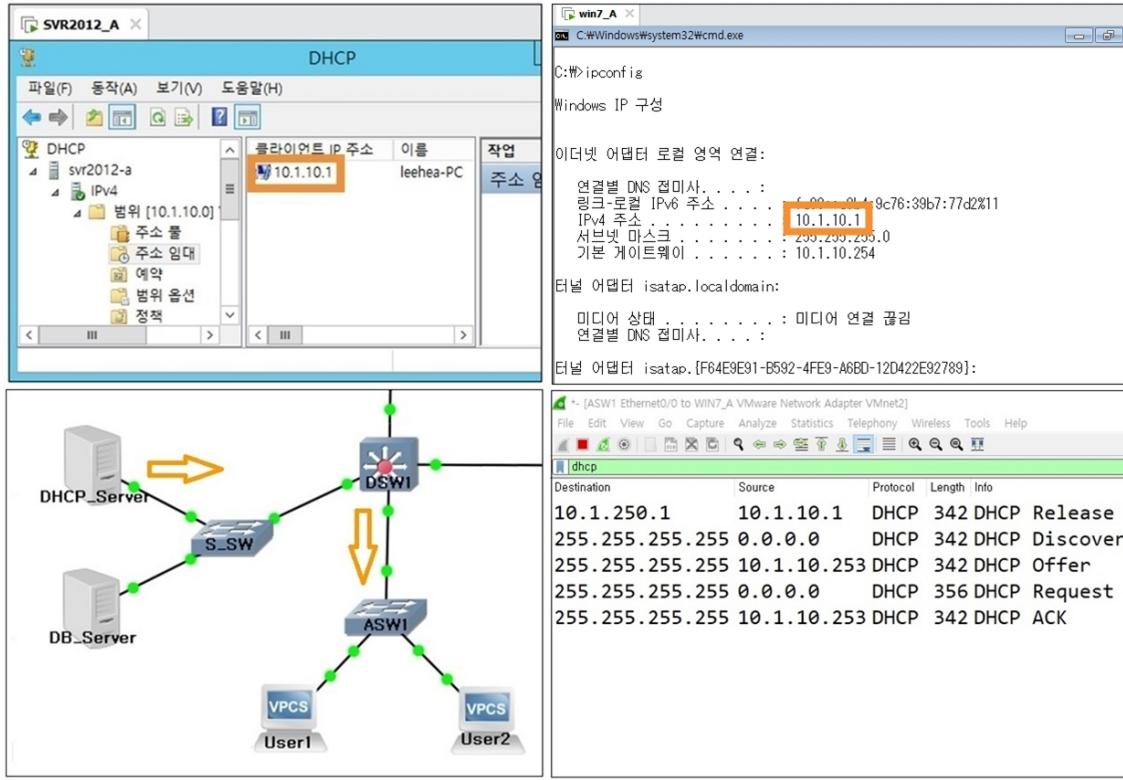


그림 7 DHCP 서버

DHCP는 호스트의 IP 주소를 클라이언트에게 자동적으로 할당해주는 프로토콜이다. LAN으로 연결된 컴퓨터 대수가 많고 IP 주소가 변경되어도 되는 기기에는 DHCP 서버를 통해 자동으로 네트워크 설정을 하는 것이 편리하다.

DHCP 서버는 배포하도록 설정한 IP 주소의 범위 내에서 미사용중인 IP 주소를 배포한다. 그 외에 네트워크의 설정 정보도 배포한다. 관리자가 배포할 IP 주소의 범위나 각종 설정 정보를 등록하면 IP 주소의 할당을 서버에게 맡길 수 있다.

5. AWS 인프라스트럭처

5.1. Amazon VPC(Virtual Private Cloud)

정의

AWS 의 자원을 실행할 수 있는 가상 네트워크 서비스

구성

- 서브넷: VPC 의 IP 주소 범위
 - 퍼블릭 서브넷 : 퍼블릭 인터넷에 대한 인바운드/아웃바운드 액세스를 지원하도록 인터넷 게이트웨이에 대한 라우팅 테이블 항목을 포함한다.
 - 프라이빗 서브넷 : 인터넷 게이트웨이에 대한 라우팅 항목이 없어 직접 액세스가 불가능하며 일반적으로 제한된 아웃 바운드 퍼블릭 인터넷 액세스를 지원하기 위해 NAT 게이트 웨이를 사용한다.
- 라우팅 테이블: 네트워크 트래픽을 전달할 위치를 결정하는데 사용하는 라우팅의 집합. AWS 에서는 VPC 를 생성할 경우 동일 VPC 에 있는 모든 서브넷이 기본적으로 라우팅 되어있다.
- 인터넷 게이트웨이: VPC 의 리소스와 인터넷 간의 통신을 활성화 하기 위해 VPC 의 퍼블릭 서브넷에 연결하는 게이트웨이
- NAT 게이트웨이: 프라이빗 서브넷에서 일반적으로 제한되어 있는 인터넷 액세스를 해야 할때 사용하는 게이트웨이
- CIDR 블록 : 인터넷 프로토콜 주소 할당 및 라우팅 블록 형태로 집계한 것

작동방식

VPC 에서 사용할 CIDR 범위를 설정하고 서브넷을 생성한 후 라우팅 테이블을 구성한다. 각 서브넷을 라우팅 테이블에 연결해서 서브넷에 대한 라우팅을 제어한다.

HYBRID-BTS

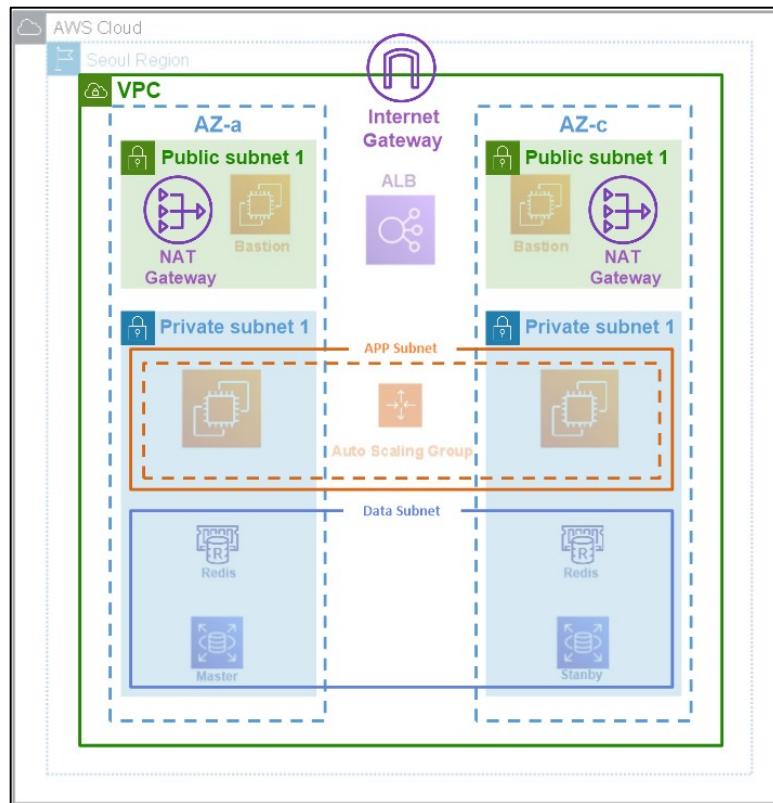


그림 8 HYBRID-BTS 의 VPC 다이어그램

Seoul Region에서는 10.0.0.0/16 대역의 VPC를 사용하고 있고 Virginia Region에서는 172.16.0.0/16 대역의 VPC를 사용하고 있다. 가용성을 위해 각각 Region의 두 가용 영역에 서브넷을 구성하였다. 라우팅 테이블에 서브넷을 연결하여 인터넷 게이트웨이와 연결되어 있는 퍼블릭 서브넷과 NAT 게이트웨이와 연결되어 있는 프라이빗 서브넷으로 구분하였다. 각 퍼블릭 서브넷마다 NAT 게이트웨이를 생성하였다.

구축 단계

개요

1. VPC 생성
2. 서브넷 생성
3. 인터넷 게이트웨이 생성 후 퍼블릭 라우팅 테이블 등록
4. NAT 게이트웨이 생성 후 프라이빗라우팅 테이블 등록

과정

1. VPC 생성

VPC의 이름을 정하고 사용할 IPv4 CIDR을 지정한다.

VPC 설정

이름 태그 - 선택 사양
'Name' 키와 사용자가 지정하는 값으로, 고유한 대역을 식별하는 데 사용됩니다.

생성할 VPC의 이름 입력

IPv4 CIDR 블록 정보
10.0.0.0/16 **사용할 VPC IP 대역 범위 지정**

IPv6 CIDR 블록 정보
 IPv6 CIDR 블록 없음
 Amazon 제공 IPv6 CIDR 블록

태년시 정보
기본값

태그
태그는 AWS 리소스에 할당하는 레이블입니다. 각 태그는 키와 선택적 값으로 구성됩니다. 태그를 사용하여 리소스를 검색 및 필터링하거나 AWS 비용을 추적할 수 있습니다.

키	값 - 선택 사양
<input type="text" value="Name"/>	<input type="text" value="HybridBTS-VPC"/> 검색 및 필터링 할 태그 입력

2. 서브넷 생성

Public subnet과 Private subnet을 구분하여 생성한다. Public subnet에 생성될 인터페이스들에 대해 퍼블릭 IPv4 주소가 자동으로 할당되게 한다.

선택	Name	서브넷 ID	상태	VPC
<input type="checkbox"/>	BTS_Public 1	subnet-06a5d0897321cf052	<input checked="" type="radio"/> Available	vpc-08e8ad0df69a8a5fe HybridBTS-VPC
<input type="checkbox"/>	BTS_Public 2	subnet-08df70c47d45e76c1	<input checked="" type="radio"/> Available	vpc-08e8ad0df69a8a5fe HybridBTS-VPC
<input type="checkbox"/>	BTS_Private 1	subnet-0876b6bdbf81870ee	<input checked="" type="radio"/> Available	vpc-08e8ad0df69a8a5fe HybridBTS-VPC
<input type="checkbox"/>	BTS_Private 2	subnet-04f8b8c4a96d7898c	<input checked="" type="radio"/> Available	vpc-08e8ad0df69a8a5fe HybridBTS-VPC

외부에서 접근 가능한 Public Subnet과 내부에서만 접근 가능한 Private Subnet 대역을 생성.
고가용성을 위해 AZ-a, AZ-c에 두 개씩 생성한다.

자동 할당 IP 설정 수정 정보

자동 할당 IP 주소 설정을 활성화하여 이 서브넷의 새 네트워크 인터페이스의 퍼블릭 IPv4 또는 IPv6 주소를 자동으로 요청합니다.

설정

서브넷 ID
subnet-08df70c47d45e76c1

자동 할당 IPv4 정보
 퍼블릭 IPv4 주소 자동 할당 활성화

Public Subnet 1,2 가 외부에 개방되도록 퍼블릭 IPv4 주소 자동 할당 설정

3. 인터넷 게이트웨이 생성 후 퍼블릭 라우팅 테이블 등록

인터넷 게이트웨이 설정

이름 태그
'Name' 키와 사용자가 지정하는 값을 조합하는 태그로 생성합니다.

BTS-igw 인터넷 게이트웨이 이름 지정

태그 - 선택 사항
태그는 AWS 리소스에 할당하는 레이블입니다. 각 태그는 키와 선택적 값으로 구성됩니다. 태그를 사용하여 리소스를 검색 및 필터링하거나 AWS 비용을 추적할 수 있습니다.

키 값 - 선택 사항
Name BTS-igw 검색 및 필터링 할 태그 입력

라우팅 편집

Public subnet 들이 인터넷으로 연결이 가능하게 라우팅 테이블을 생성한 뒤 Public subnet 과 연결

대상	대상
10.0.0.0/16	Q local
Q 0.0.0.0/0	Q igw-0c5abf689ae41a5d5

라우팅 추가 아니요 제거

The screenshot shows the AWS VPC service in the AWS Management Console. A specific route table, 'BTS-Public RT', is selected. It contains two entries under the '영시적 서브넷 연결' (Internet Gateway) section. The first entry connects the 'rtb-06c7e4bd047422c90 / BTS_Public 1' subnet to the internet gateway '2 서브넷'. The second entry connects the 'subnet-06a5d0897321cf052 / BTS_Public 1' and 'subnet-08df70c47d45e76c1 / BTS_Public 2' subnets to the same internet gateway.

4. NAT 게이트웨이 생성 후 프라이빗 라우팅 테이블 등록

Private subnet에서 인터넷에 아웃바운드 연결이 가능하도록 하기 위해 NAT 게이트웨이를 각 가용 영역마다 1개씩 생성한다. Private subnet 1,2의 라우팅 테이블을 생성한다. 서브넷을 연결하고 라우팅 테이블에 NAT 게이트웨이를 등록한다.

This screenshot shows the 'NAT 게이트웨이 설정' (NAT Gateway Configuration) step of the AWS NAT Gateway creation wizard. The 'Name' field is set to 'BTS_ngw 2'. The '서브넷' (Subnet) dropdown is set to 'subnet-08df70c47d45e76c1 (BTS_Public 2)'. The '연결 유형' (Connection Type) is set to '퍼블릭' (Public). The '탄력적 IP 할당 ID 정보' (Elastic IP Allocation ID Information) field is filled with 'eipalloc-03ef3fa7ff2c282a9'. The 'NAT 게이트웨이에 탄력적 IP 자동 할당' (Automatically Assign Elastic IP to NAT Gateway) checkbox is checked. The 'IP 할당' (IP Allocation) button is visible at the bottom right.

라우팅 테이블 생성 정보

라우팅 테이블은 VPC, 인터넷 및 VPN 연결 내 서브넷 간에 패킷이 전달되는 방법을 지정합니다.

라우팅 테이블 설정

이름 - 선택 사항

'Name' 키와 사용자가 지정하는 이름은 라우팅 테이블에 표시됩니다.

BTS_Private RT

프라이빗 라우팅 테이블 이름 지정

VPC

이 라우팅 테이블에 대해 사용할 VPC입니다.

vpc-08e8ad0df69a8a5fe (HybridBTS-VPC)

라우팅 테이블을 생성할 VPC 선택

라우팅 테이블 (4) 정보

C 작업 ▾ 라우팅 테이블 생성

Q 라우팅 테이블 필터링

search: vpc-08e8ad0df69a8a5fe X

필터 지우기

**퍼블릭 라우팅 테이블과 프라이빗 라우팅 테이블
생성 확인**

Name	라우팅 테이블 ID	명시적 확장	설정	설정	설정
BTS_Public RT	rtb-06c7e4bd047422c90	2 서브넷	-	아니요	vpc-08e8ad0df69a8a5fe HybridBTS-VPC 489851543453
BTS_Private RT 2	rtb-0bc8f07d3355d26ab	subnet-04ff8b8c4a96d7...	-	아니요	vpc-08e8ad0df69a8a5fe HybridBTS-VPC 489851543453
BTS_Private RT 1	rtb-0f048c6dc29dfa567	subnet-0876b6bdbfb818...	-	아니요	vpc-08e8ad0df69a8a5fe HybridBTS-VPC 489851543453

NAT 게이트웨이를 프라이빗 라우팅 테이블의 라우트로 등록

10.0.0.0/16	local	활성
0.0.0.0/0	nat-0581241d4ef672a25	활성

5.2. Amazon EC2(Elastic Compute Cloud)

5.2.1. Bastion

정의

관리자가 외부에서 내부 인스턴스로 접근하여 관리하게끔 만들어진 인스턴스

기능

Public subnet에 bastion 인스턴스를 관리자의 IP만 접근할 수 있게 만들어 외부에서 Private subnet 안의 웹 서버를 관리한다.

HYBRID-BTS

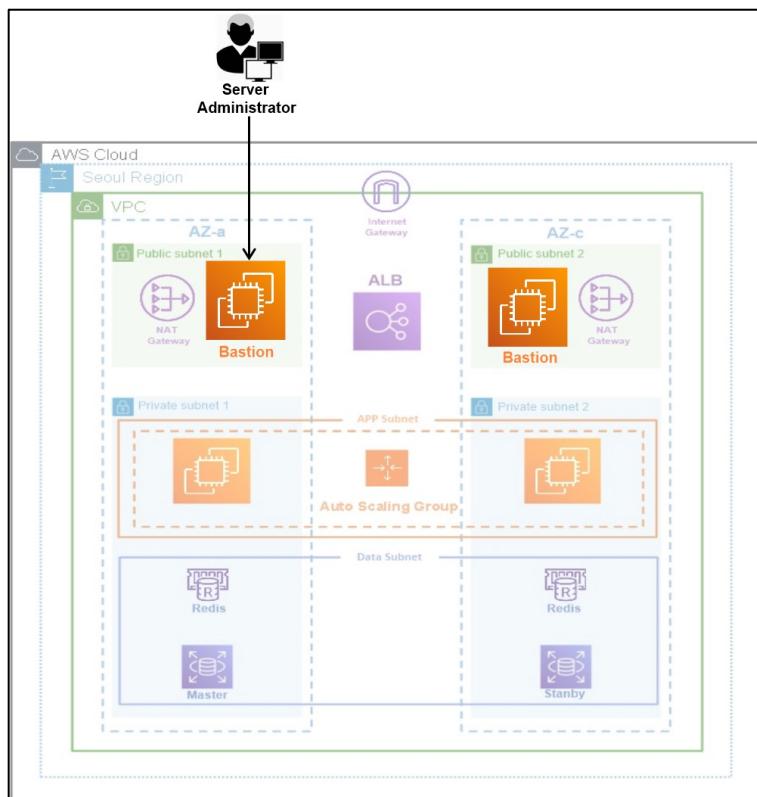


그림 9 HYBRID-BTS 의 Bastion host 디아이어그램

Public subnet1,2에 배치되어 HYBRID-BTS 웹 서버가 있는 Private subnet 1,2에 SSH 원격 접근하여 쇼핑몰 웹 서버를 관리한다.

구축 단계

과정

퍼블릭 서브넷에 EC2 인스턴스를 생성한 뒤 관리자 IP만 허용하는 보안 그룹을 부여한다. 웹 서버의 키 페어를 Bastion에 복사한다.



보안 그룹 생성 정보

보안 그룹은 인바운드 및 아웃바운드 트래픽을 관리하는 인스턴스의 가장 방화벽 역할을 합니다. 새 보안 그룹을 생성하려면 아래의 펌드를 작성하십시오.

기본 세부 정보

보안 그룹 이름: **BTS_Bastion_SG** Bastion 보안 그룹 이름 등록

설명: **BTS_Bastion_SG**

VPC: **vpc-08e8ad0df69a8a5fe**

인바운드 규칙 정보 인바운드 유형 SSH 선택 접근 가능 소스에 관리자 IP 등록

유형: SSH 소스: 내 IP: 58.237.166.233/32

TCP 22 SSH access

새 키 페어 생성

키 페어 유형: RSA

키 페어 이름: **BTS_Bastion_Key** Bastion 서버의 키페어 생성 키 페어 다운로드

<input type="checkbox"/> BTS_Bastion 1	i-0491e0ac0b060d56e	실행 중	t2.micro	2/2개 검사 통과...	경보 없음	+	ap-northeast-2a	-	3.36.112.112
<input type="checkbox"/> BTS_Bastion 2	i-0addc27e2367ac316	실행 중	t2.micro	2/2개 검사 통과...	경보 없음	+	ap-northeast-2c	-	54.180.91.232

가용영역-a 의 서버를 관리할 Bastion1, 가용영역-c 서버를 관리할 Bastion2 생성

```
[ec2-user@ip-10-0-0-74 ~]$ vi .BTS_WEB_KEY.pem
```

```
-----BEGIN RSA PRIVATE KEY-----  
MIIEpAIBAAKCAQEo8BEf4iVjkVdGRSkXmud7LKfhld9u6NTUeN33uJF8JzW71J  
0KKLAZ4y3MvKrMluYDNZfIIgfi91iLZUz405TCzu2YpTzbngbeAcfyMqyoxEFPT  
rBi8oBbBifF+Bu0tT0IM5cl1PPoUnefKozsPRxh931EE5KfE4jD9ywlvIYJh11Q4  
jGZqN{ <F  
U1POC{ r0  
Z2dPvv 3i  
qWlyuc )G  
S0t6bf Vu  
bNjq9e =D
```

Private subnet 내부 서버의 키페어를 Bastion 인스턴스 1,2로 복사

```
[ec2-user@ip-10-0-0-74 ~]$ ssh -i .BTS_WEB_KEY.pem ec2-user@10.0.1.54
The authenticity of host '10.0.1.54 (10.0.1.54)' can't be established.
ECDSA key fingerprint is SHA256:as0yr39bgSXiuDVlqujjgD7M0IQ6zPZ3S0zSYytchF0.
ECDSA key fingerprint is MD5:9f:45:c8:6a:a7:0d:5e:b6:bd:17:25:0a:cd:36:c8:be.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.1.54' (ECDSA) to the list of known hosts.

[ec2-user@ip-10-0-0-74 ~]$
```

5.2.2. AMI 및 Launch Template

정의

AMI는 소프트웨어 구성이 기재된 템플릿이다. Launch Template은 AMI를 활용하여 EC2 인스턴스를 생성할 때 각 단계를 일일이 지정할 필요 없이 각 파라미터 값을 정의한다. EC2 인스턴스 생성 과정을 간단히 할 수 있다.

HYBRID-BTS

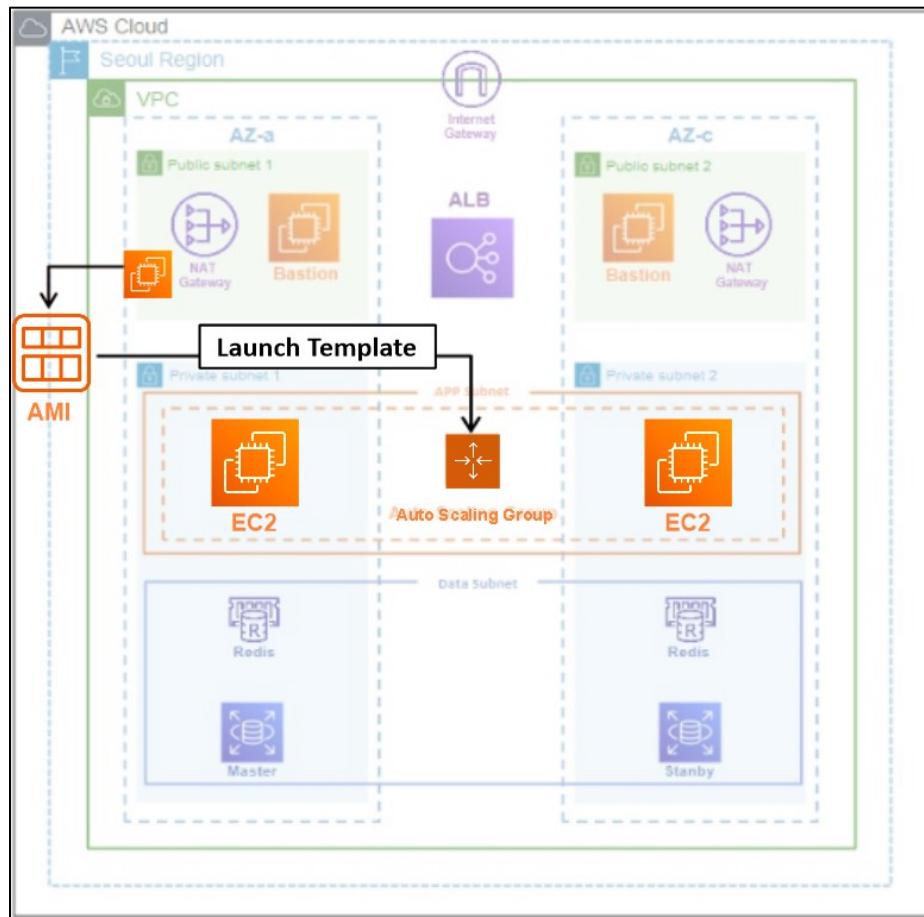


그림 10 HYBRID-BTS 의 AMI 및 Launch Template 디아이어그램

Auto scaling 으로 생성 될 HYBRID-BTS 웹 서버를 구성하는 파라미터 값(AMI,IAM,User data)이 정의된 템플릿을 생성하였다.

구축 단계

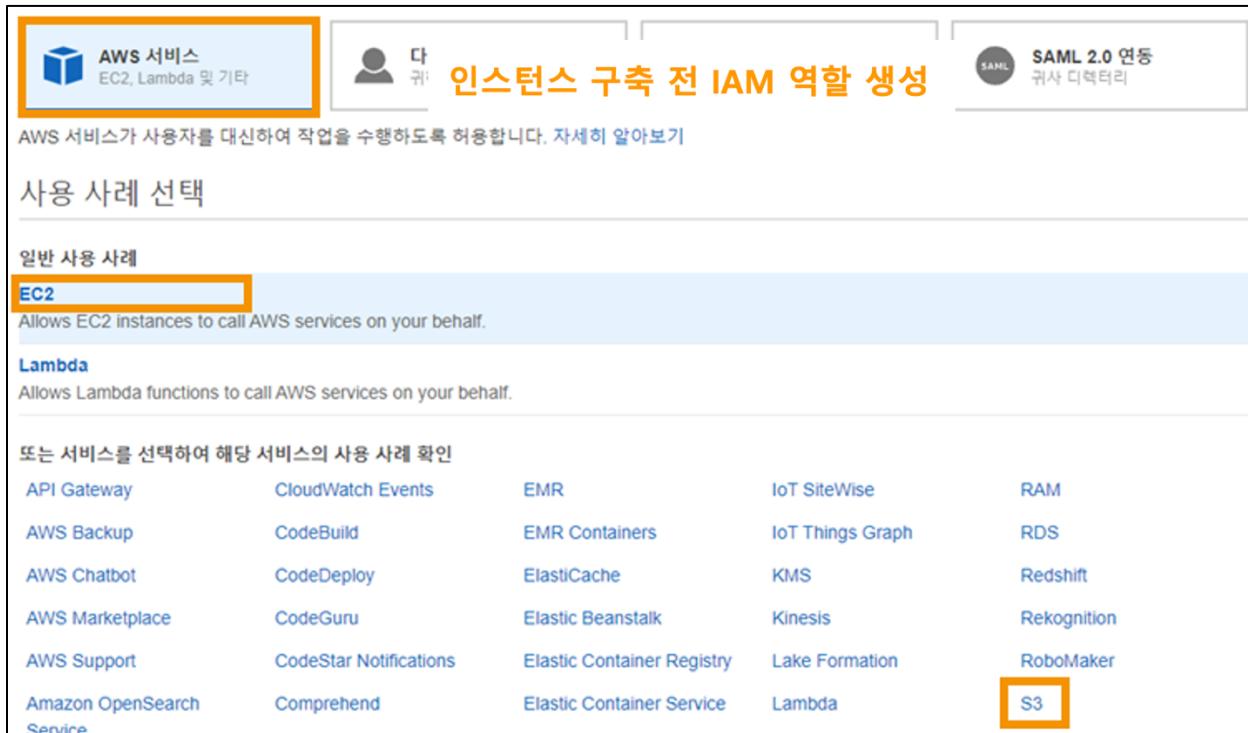
개요

1. 인스턴스 IAM 역할 생성
2. 템플릿으로 활용할 표본 인스턴스 생성
3. 생성된 인스턴스로 골든 이미지 생성
4. 시작 템플릿 생성

과정

1. 인스턴스 IAM 역할 생성

인스턴스 생성시 Userdata에 S3 버킷에서 소스코드를 불러오기 위해 S3ReadOnlyAccess 정책 선택한다.



AWS 서비스가 사용자를 대신하여 작업을 수행하도록 허용합니다. 자세히 알아보기

사용 사례 선택

일반 사용 사례

EC2 Allows EC2 instances to call AWS services on your behalf.

Lambda Allows Lambda functions to call AWS services on your behalf.

또는 서비스를 선택하여 해당 서비스의 사용 사례 확인

API Gateway	CloudWatch Events	EMR	IoT SiteWise	RAM
AWS Backup	CodeBuild	EMR Containers	IoT Things Graph	RDS
AWS Chatbot	CodeDeploy	ElastiCache	KMS	Redshift
AWS Marketplace	CodeGuru	Elastic Beanstalk	Kinesis	Rekognition
AWS Support	CodeStar Notifications	Elastic Container Registry	Lake Formation	RoboMaker
Amazon OpenSearch Service	Comprehend	Elastic Container Service	Lambda	S3



역할 만들기

검토

생성하기 전에 아래에 필요한 정보를 입력하고 이 역할을 검토하십시오.

역할 이름* **S3_ReadOnly_Role** **S3 를 허용한 IAM 역할 이름 등록**

영문자 및 '=,@-' 문자를 사용합니다. 최대 64자입니다.

역할 설명 Allows EC2 instances to call AWS services on your behalf.

최대 1000자입니다. 영문자 및 '=,@-' 문자를 사용합니다.

신뢰할 수 있는 **읽기전용 S3 허용 정책 선택**

정책 **AmazonS3ReadOnlyAccess**

2. 템플릿으로 활용할 표본 인스턴스 생성



Enclave i 활성화

메타데이터 액세스 가능
메타데이터 버전
메타데이터 토큰 응답 흡 제한

사용자 데이터 i 텍스트로 파일로 입력이 이미 base64로 인코딩됨

```
#!/bin/bash
yum install -y httpd mysql
amazon-linux-extras install -y php7.2
systemctl start httpd
systemctl enable httpd
```

사용자 데이터에 인스턴스 생성 과정에서 실행될 명령어를 입력한다. (httpd , mysql, php 패키지 설치)

단계 6: 보안 그룹 구성
보안 그룹은 인스턴스에 대한 트래픽을 제어하는 방화벽 규칙 세트입니다. 이 페이지에서는 특정 트래픽을 인스턴스에 도달하도록 허용할 규칙을 추가할 수 있습니다. 예를 들면 웹 서버를 설정하여 인터넷 트래픽을 인스턴스에 도달하도록 허용:에 대한 무제한 액세스를 적용하는 규칙을 추가합니다. 새 보안 그룹을 생성하거나 아래에 나와 있는 기존 보안 그룹 중에서 선택할 수 있습니다. Amazon EC2 보안 그룹에 대해 자세히 알아보기.

기존 보안 그룹 선택

보안 그룹 ID	이름	설명
sg-0025581abdb8729a2	BTS Bastion SG	BTS Bastion SG
sg-0941c469189961c66	BTS_WEB_sg	BTS_WEB_sg
sg-0701ccee20c07562f	default	default VPC security group

HTTP, HTTPS, SSH 접근 허용하는 보안그룹 선택

설정하는 것이 좋습니다.

유형	프로토콜	포트 범위	소스	설명
HTTP	TCP	80	0.0.0.0/0	http access
SSH	TCP	22	0.0.0.0/0	ssh access
HTTPS	TCP	443	0.0.0.0/0	https access

```
[ec2-user@ip-10-0-0-97 html]$ systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Drop-In: /usr/lib/systemd/system/httpd.service.d
             └─php-fpm.conf
Active: active (running)
          Docs: man:httpd.service
         Main PID: 3410 (httpd)
        Status: "Total requests: 3; Idle/Busy workers 100/0;Requests/sec: 0.00128; Bytes served/sec: 9 B/sec"
       CGroup: /system.slice/httpd.service
                 ├─3410 /usr/sbin/httpd -DFOREGROUND
```

인스턴스 내부에 들어가서 Apache 작동 확인

3. 생성된 인스턴스로 골든 이미지를 생성

생성된 인스턴스로 이미지 생성

이미지 생성 정보
이미지(AMI)라고도 할)는 EC2 인스턴스를 시작할 때 적용되는 프로그램 및 설정을 정의합니다. 기존 인스턴스의 구성에서 이미지를 생성할 수 있습니다.

인스턴스 ID
i-0cebd5bf2cd43c1be (BTS_GoldenInstance)

이미지 이름
BTS_AMI

구성된 웹 서버의 이미지 이름 지정

4. 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
BTS-template-0 파라미터를 저장할 템플릿 이름 지정

이 계정에 대해 고유해야 합니다. 최대 128자입니다. &lt;/p>
 이 템플릿은 다른 계정에서 사용할 수 없습니다.

템플릿 버전 설명
version 1 버전 관리를 위한 버전 입력

최대 255자

시작 템플릿 콘텐츠

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

▼ Amazon Machine Image(AMI) 정보

AMI

생성한 골든 이미지를 시작 템플릿의 AMI로 선택

BTS-AMI

ami-096aed87866051d3d
카탈로그: 내 AMI 아키텍처: 64비트(x86) 가상화: hvm

▼ 키 페어(로그인) 정보

키 페어를 사용하여 인스턴스에 안전하게 연결할 수 있습니다. 인스턴스를 시작하기 전에 선택한 키 페어에 대한 액세스 권한이 있는지 확인하세요.

키 페어 이름

생성될 인스턴스에 사용될 키페어 등록

BTS_WEB_KEY

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 시작합니다.

보안 그룹

보안 그룹 선택

BTS_WEB_sg sg-0941c469189961c66 X
VPC: vpc-08e8ad0df69a8a5fe

네트워크와 보안그룹 설정

▼ 고급 세부 정보 정보

구매 옵션 정보

S3에서 웹 소스코드를 받아 사용하기 위해
S3 IAM 역할 부여

S3_ReadOnly_Role
arn:aws:iam::489851543453:instance-profile/S3_ReadOnly_Role

새 IAM 프로파일 생성

사용자 데이터 정보

```
#!/bin/bash
aws s3 cp s3://btsbucket-123/htdocs.zip .
unzip htdocs -d /var/www/html
```

S3 버킷에서 웹 소스코드 다운 후 압축 해제

5.2.3. ALB(Application Load Balancer)

정의

Application Load Balancer는 7 계층인 애플리케이션 계층에서 작동하는 로드 밸런서. 트래픽을 대상들에 분배한다.

기능

클라이언트로부터 오는 HTTP/HTTPS 트래픽을 허용하고, 하나 이상의 가용 영역에서 등록된 대상으로 요청을 라우팅한다.

작동 방식

ALB는 요청을 받으면 우선 순위에 따라 리스너 규칙을 평가하여 적용할 규칙을 결정한 다음, 규칙 작업의 대상 그룹에서 대상을 선택한다. 애플리케이션 트래픽의 콘텐츠를 기반으로 다른 대상 그룹에 요청을 라우팅하도록 리스너 규칙을 구성할 수 있다. 대상이 여러 개의 대상 그룹에 등록이 된 경우에도 각 대상 그룹에 대해 독립적으로 라우팅을 수행한다. 대상 그룹 레벨에서 사용되는 라우팅 알고리즘(라운드 로빈)을 구성하여 대상그룹으로 트래픽을 분배하여 서버 인스턴스에 부하를 줄인다.

구축 단계

개요

1. 보안 그룹 및 대상 그룹 생성
2. ALB 생성

과정

1. 보안 그룹 및 대상 그룹 생성

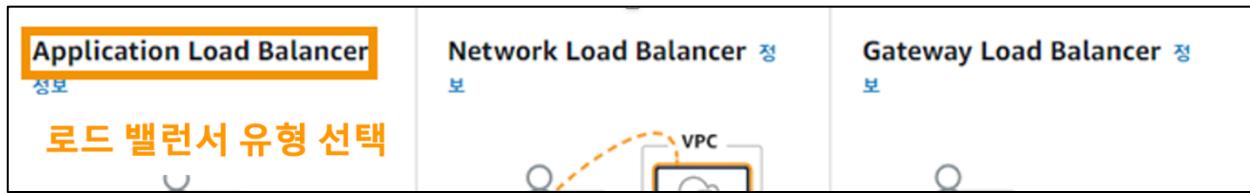
ALB에 적용될 보안 그룹을 생성하고 요청을 분산 처리할 대상 그룹을 생성한다.

인바운드 규칙 (2)			
외부에서 들어오는 요청을 받아 분배하도록 HTTP, HTTPS 를 허용한다.			
	Name	보안 그룹 규칙 ID	IP 버전
<input type="checkbox"/>	-	sgr-0e3b2ae4f714a8a0c	IPv4
<input type="checkbox"/>	-	sgr-09d14bd8bde05a...	IPv4

2. ALB 생성

Load Balance 유형, 이름, VPC, 부하분산 시킬 가용 영역을 선택한다.

기본 구성	
로드 밸런서 이름 <small>이름은 AWS 계정 내에서 고유해야 하며 로드 밸런서 생성 후에는 변경할 수 없습니다.</small>	BTS_ALB ALB 의 이름 지정
<small>하이픈을 포함하여 최대 32자리의 영문자·숫자를 사용할 수 있지만 이름이 하이픈으로 시작하거나 끝나지 않아야 합니다.</small>	



기본 구성

대상 그룹이 생성된 후에는 이 섹션의 설정을 변경할 수 없습니다.

대상 유형 선택

인스턴스

- 특정 VPC 내의 인스턴스에 대한 로드 밸런싱을 지원합니다.

대상 그룹 이름

BTS_tg **대상 그룹 이름 지정**

하이픈을 포함하여 최대 32자리 깊은 서브넷을 포함한 두 짧은 이름이 하이픈으로 시작하거나 끝나지 않아야 합니다.

프로토콜 **포트**

TCP : 80

VPC

대상 그룹에 포함할 인스턴스가 있는 VPC를 선택합니다.

HybridBTS-VPC
vpc-08e8ad0df69a8a5fe
IPv4: 10.0.0.0/16

대상 그룹에 포함할 타겟이 있는 VPC 선택

네트워크 매핑 정보
로드 벨런서는 IP 주소 설정에 따라 선택한 서브넷의 대상으로 트래픽을 라우팅합니다.

VPC 정보
대상에 대한 Virtual Private Cloud(VPC)를 선택합니다. 인터넷 게이트웨이가 있는 VPC만 선택할 수 있습니다. 로드 벨런서 생성 후에는 선택한 VPC 레이어 대상 그룹 를 참조하세요.

HybridBTS-VPC
vpc-08e8ad0df69a8a5fe
IPv4: 10.0.0.0/16

ALB 를 배치할 VPC 선택

매핑 정보
가용 영역을 2개 이상 선택하고 영역당 하나의 서브넷을 선택합니다. 로드 벨런서는 이러한 가용 영역의 대상으로만 트래픽을 라우팅합니다. 로드 벨런서는 이전에 선택한 대상 그룹을 제거할 수 없습니다. 로드 벨런서가 생성된 후에는 서브넷을 제거할 수 있지만 서브넷을 추가할 수 있습니다.

ap-northeast-2a

서브넷
트래픽을 부하분산 받는
가용영역과 퍼블릭 서브넷 선택

subnet-06a5d0897321cf052

보안 그룹 정보
보안 그룹은 로드 벨런서에 대한 트래픽을 제어하는 방화벽 규칙 세트입니다.

보안 그룹
보안 그룹 선택
 새 보안 그룹 생성

BTS_ALB_sg sg-05b661d67c0d7c169 X
VPC: vpc-08e8ad0df69a8a5fe

HTTP/HTTPS 를 허용한 보안 그룹 선택

리스너 및 라우팅 정보
리스너는 구성한 프로토콜 및 포트를 사용하여 연결 요청을 확인하는 프로세스입니다. 리스너가 수신한 트래픽은 해당 사양에 따라 라우팅됩니다. 로드 벨런서가 생성된 후 리스너당 여러 규칙과 여러 인증서를 지정할 수 있습니다.

리스너 HTTP:80

연결 요청을 확인하는 리스너의 대상 그룹 선택

프로토콜	포트
HTTP	: 80 1-65535
다음으로 전달:	
BTS-tg 대상 유형: 인스턴스, IPv4	
<input checked="" type="checkbox"/> 대상 그룹 생성	

5.2.4. Auto Scaling

정의

클라우드의 유연성을 돌보이게 하는 핵심기술로 CPU, 메모리, 디스크, 네트워크 트래픽과 같은 시스템 자원들의 메트릭(Metric) 값을 모니터링하여 서버 사이즈를 자동으로 조절한다.

작동 방식

CloudWatch에서 Auto Scaling 그룹 지표를 모니터링하여 Auto Scaling에 경보를 보내고 인스턴스 개수를 확장하거나 축소한다.

장점

서버의 트래픽 요구 사항에 맞춰 유연하게 서버 사이즈를 조정하므로 사용량이 많고 불 특정하게 트래픽이 몰리는 애플리케이션에 적합하다.

구축 단계

개요

1. Auto Scaling 그룹 생성
2. 테스트

과정

1. Auto Scaling 그룹 생성

시작 템플릿, 네트워크, 로드 밸런서를 지정하여 오토 스케일링 그룹을 생성한다.

시작 템플릿 또는 구성 선택 [정보](#)

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

이름

Auto Scaling 그룹 이름
그룹을 식별할 이름은 아래와 같이

BTS AS **Auto Scaling 그룹 이름 지정**

현재 리전에서 이 계정에 대해 고유해야 하며 255자 이상 3자로 줄여야 합니다.

시작 템플릿 정보 [시작 구성으로 전환](#)

시작 템플릿
Amazon Machine Image 시작 템플릿을 선택합니다.

BTS-template-0 **ASG 생성에 사용할 시작 템플릿 선택** [▼](#) [C](#)

네트워크 정보

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-08e8ad0df69a8a5fe (HybridBTS-VPC)
10.0.0.0/16 [▼](#) [C](#)

VPC 생성 [\[+\]](#)

서브넷

확장/축소할 인스턴스가 있는 VPC와 Subnet 선택

서브넷 선택 [▼](#) [C](#)

ap-northeast-2a subnet-0876b6bdbf81870ee (BTS_Private 1) 10.0.1.0/24	X
ap-northeast-2c subnet-04f8b8c4a96d7898c (BTS_Private 2) 10.0.3.0/24	X

로드 밸런싱 - 선택 사항 정보

아래 옵션을 사용하여 Auto Scaling 그룹을 기준 로드 밸런서 또는 사용자가 정의한 새 로드 밸런서에 연결합니다.

- 로드 밸런서 없음**
Auto Scaling 그룹에 대한 트래픽은 로드 밸런서가 앞에 있지 않습니다.
- 기준 로드 밸런서에 연결**
기준 로드 밸런서 중에서 선택합니다.
- 새 로드 밸런서에 연결**
Auto Scaling 그룹에 연결할 기본 로드 밸런서를 빠르게 생성합니다.

오토스케일링으로 확장되는 인스턴스가 부하분산 처리를 받게하기 위해 ALB 선택

Auto Scaling 그룹에 연결할 로드 밸

작업 기록 (14)						
상태	설명	원인	시작 시간	종료 시간		
Successful	Launching a new EC2 instance: i-02fa5ec56947c2d01	At 2021-10-25T12:03:57Z a user request update of AutoScalingGroup constraints to min: 2, max: 2, desired: 2 changing the desired capacity from 0 to 2. At 2021-10-25T12:04:05Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 2.	2021 10월 25, 09:04:07 오후 +09:00	2021 10월 25, 09:04:39 오후 +09:00		
Successful	Launching a new EC2 instance: i-0cae5cccd9c752f5a	At 2021-10-25T12:03:57Z a user request update of AutoScalingGroup constraints to min: 2, max: 2, desired: 2 changing the desired capacity from 0 to 2. At 2021-10-25T12:04:05Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 2.	2021 10월 25, 09:04:07 오후 +09:00	2021 10월 25, 09:05:09 오후 +09:00		
<input type="checkbox"/>	BTS-AS	i-0cae5cccd9c752f5a	 실행 중	 2/2개 검사 통과...	경보 없음	+
<input type="checkbox"/>	BTS-AS	i-02fa5ec56947c2d01	 실행 중	 2/2개 검사 통과...	경보 없음	+

2. 테스트

Auto Scaling 으로 생성된 인스턴스들의 CPU 사용량을 과부하 시켜 확장시킨다. CloudWatch 로 모니터링하여 정상적으로 작동되는 모습을 확인한다.

CPU 를 과부화 시키는 테스트 도구인 stress 를 설치한다.

```
[root@ip-10-0-0-74 ~]# amazon-linux-extras install -y epel
[root@ip-10-0-0-74 ~]# yum install stress -y
```

Auto Scaling에 적용된 조정 정책은 평균 CPU 사용률 75%이다.

Target Tracking Policy

정책 유형:

대상 추적 조정

활성화 또는 비활성화?

활성

다음 경우에 정책 실행:

평균 CPU 사용률을(를) 75

조정 정책이 실행되는 CPU 수치 설정

작업 수행:

필요에 따라 용량 단위 추가 또는 제거

인스턴스 요구 사항:

300초(지표에 포함하기 전 워밍업 시간)

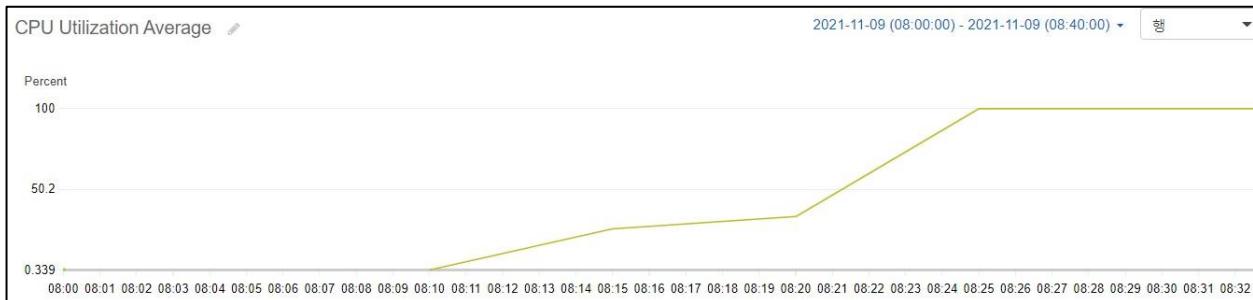
축소:

활성

Stress tool을 사용하여 인스턴스 CPU 과부하 시킨다.

```
[ec2-user@ip-10-0-3-138 ~]$ stress --cpu 1 CPU 과부하 실행 [ec2-user@ip-10-0-1-16 ~]$ stress --cpu 1
stress: info: [3555] dispatching hogs: 1 cpu, 0 io, 0 vm, 0 hdd
stress: info: [3623] dispatching hogs: 1 cpu, 0 io, 0 vm, 0 hdd
```

CloudWatch에서 Auto scaling의 조정 정책 임계치 값을 넘어가는 것을 확인한다.

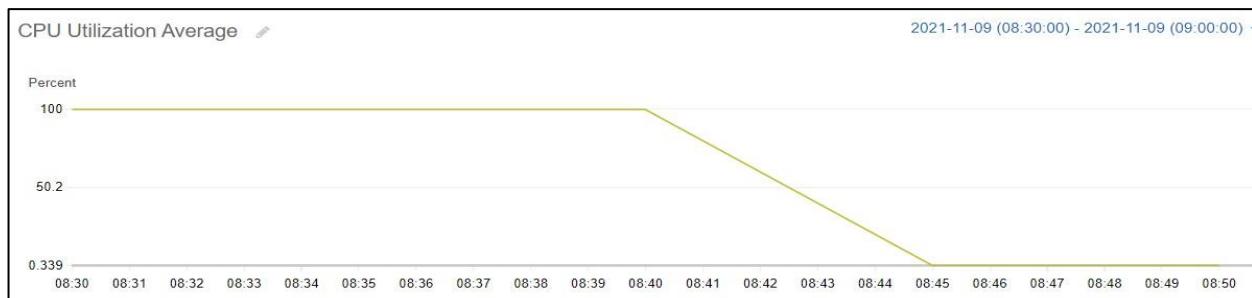


Auto scaling 용량 조정 정책에 의해 인스턴스 개수가 확장되었다.

인스턴스 (13) 정보								
	Name	인스턴스 ID	인스턴스 상태	인스턴스 유형	상태 검사	경보 상태	가용 영역	연결
□	BTS_Bastion 1	i-0491e0ac0b060d56e	실행 중	t2.micro	2/2개 검사 통과	경보 없음	+ ap-northeast-2a	
□	BTS-AS	i-001b3848f466de97c	실행 중	t2.micro	2/2개 검사 통과	경보 없음	+ ap-northeast-2a	
□	BTS-AS	i-00980b355b22b3be0	실행 중	t2.micro	2/2개 검사 통과	경보 없음	+ ap-northeast-2a	
□	BTS-AS	i-00a017c2e25c59e88	실행 중	t2.micro	2/2개 검사 통과	경보 없음	+ ap-northeast-2a	
□	BTS-AS	i-01ecc273913a6b8dd	실행 중	t2.micro	2/2개 검사 통과	경보 없음	+ ap-northeast-2a	

조정 정책에 의해 서버 확장된 모습

Stress tool을 멈추면 CPU가 정상으로 돌아간다.

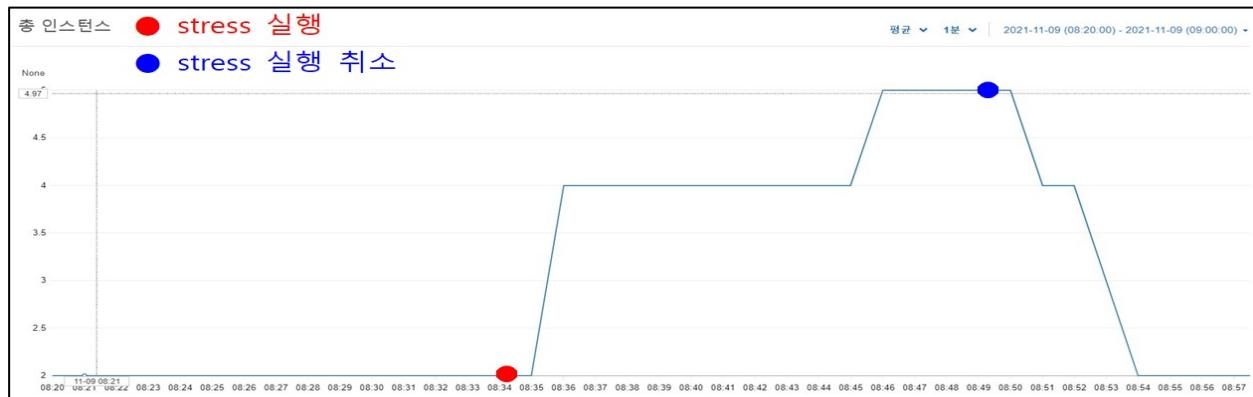


CPU가 정상 수치로 돌아가 인스턴스가 축소되었다.

인스턴스 (12) 정보								
	Name	인스턴스 ID	인스턴스 상태	인스턴스 유형	상태 검사			연결
□	BTS_Bastion 1	i-0491e0ac0b060d56e	실행 중	t2.micro	C			
□	BTS-AS	i-0f26ad2214ff9ae16	실행 중	t2.micro	C			
□	BTS-AS	i-00980b355b22b3be0	실행 중	t2.micro	C			
□	BTS-AS	i-01ecc273913a6b8dd	실행 중	t2.micro	C			
□	BTS-AS	i-00a017c2e25c59e88	종료 중					

CPU가 정상 수치로 돌아가자 축소 되는 모습

Auto Scaling 이 정상적으로 작동하는 것을 확인하였다.



5.3. Amazon RDS(Amazon Relational Databases)

5.3.1. Amazon RDS(Amazon Relational Databases)

정의

AWS에서 관계형 데이터베이스를 쉽게 설치, 운영 및 확장할 수 있는 관리형 서비스

기능

RDS는 MySQL, MariaDB, Oracle 등 데이터베이스의 전체 기능에 액세스할 수 있다. MySQL용 Amazon RDS는 다중 AZ 배포와 읽기 전용 복제본이라는 서로 다르지만 상호 보완적인 두 가지 복제 기능을 제공한다. 이 기능들을 함께 사용하면 데이터베이스 가용성이 향상되고, 예기치 않은 장애에 대비해 데이터베이스의 최신 변경 사항을 스냅샷을 이용해 보호할 수 있다.

HYBRID-BTS

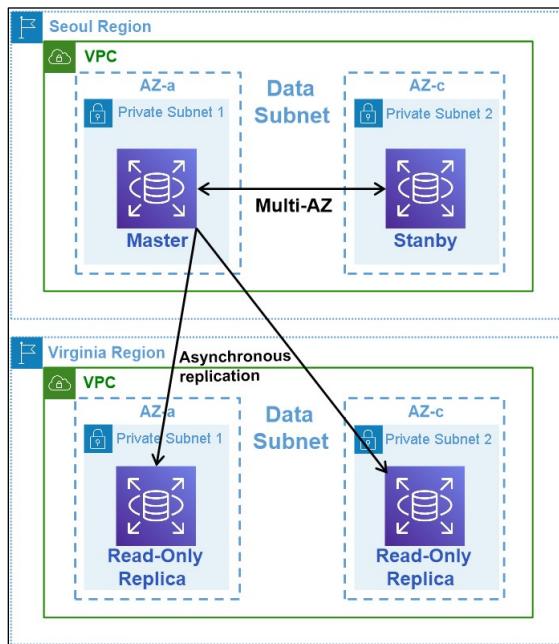


그림 11 HYBRID-BTS 의 RDS 다이어그램

HYBRID-BTS 는 명확하게 정의된 스키마인 회원 데이터를 저장하고 있기에 MySQL 엔진을 탑재한 Amazon RDS 를 선택하였다. 회원정보는 가용 영역 a 에 있는 마스터 데이터베이스에 있고 장애 대비를 위해 가용영역 c 에 대기 상태인 Standby 데이터베이스(Multi-AZ)를 배치하였다. 또한 마스터 데이터베이스의 성능과 내구성을 강화하기 위해 읽기 쿼리만을 라우팅하여 마스터 데이터베이스의 부하를 줄여주는 읽기전용 복제본을 Virginia Region 에 배치하였다.

구축 단계

개요

1. 데이터베이스 보안 그룹 생성
2. DB 서브넷 그룹 지정
3. 데이터베이스 생성
4. 데이터 생성
5. 테스트

과정

1. 데이터베이스 보안 그룹 생성

2. 데이터베이스의 고가용성을 위해 DB 서브넷 그룹 지정

서브넷 추가

가용 영역

추가할 서브넷이 포함된 가용 영역을 선택합니다.

가용 영역 선택

ap-northeast-2a X

ap-northeast-2c X

고가용성을 위한 가용영역 선택

서브넷

추가할 서브넷을 선택합니다. 목록에는 선택한 가용 영역의 서브넷이 포함됩니다.

서브넷 선택

subnet-0876b6bdbf81870ee (10.0.1.0/24) X

DB 가 배치 될 Subnet 선택

subnet-04f8b8c4a96d7898c (10.0.3.0/24) X

3. 데이터베이스 생성

엔진 옵션

엔진 유형 정보

데이터베이스 유형 선택

Amazon Aurora



MySQL



MariaDB



템플릿

해당 사용 사례를 충족하는 샘플 템플릿을 선

데이터베이스 유형을 프로젝트 용도에 맞게 선택

프로덕션

고가용성 및 빠르고 일관된 성능을 위해 기본값을 사용하세요.

개발/테스트

이 인스턴스는 프로덕션 환경 외부에서 개발 용도로 마련되었습니다.

프리 티어

RDS 프리 티어를 사용하여 새로운 애플리케이션을 개발하거나, 기존 애플리케이션을 테스트하거나 Amazon RDS에서 실무 경험을 쌓을 수 있습니다. [정보](#)

가용성 및 내구성

다중 AZ 배포

데이터베이스 장애 대비 다중 AZ 선택

- 대기 인스턴스 생성(생산 사용량에 권장)

데이터 중복을 제공하고, I/O 중지를 없애고, 시스템 백업 중에 지연 시간 스파이크를 최소화하기 위해 다른 가용 영역(AZ)에 대기 인스턴스를 생성합니다.

- 대기 인스턴스를 생성하지 마세요.

설정

DB 인스턴스 식별자 정보

DB 인스턴스 이름을 알려주세요. 이로써 현재 AWS 리전에서 AWS 계정이 소유하는 모든 DB 인스턴스에 대해 고유해야 합니다.

BTS_DB

사용할 DB 인스턴스의 이름 등록

DB 인스턴스 식별자는 대소문자를 구분하지 않지만 'mydbinstance'와 같이 모두 소문자로 저장됩니다. 제약: 1자~60자의 영숫자 또는 이一个问题으로 구성되어야 합니다. 첫 번째 문자는 글자이어야 합니다. 하이픈 2개가 연속될 수 없습니다. 끝에 하이픈이 올 수 없습니다.

▼ 자격 증명 설정

마스터 사용자 이름 정보

DB 인스턴스의 마스터 사용자에 대한 인증 정보를 알려주세요.

BTS_admin

DB 관리자 이름 등록

1~16자의 영숫자. 첫 번째 문자는 글자이어야 합니다.

암호 자동 생성

Amazon RDS에서 사용자를 대신하여 암호를 생성하거나 사용자가 직접 암호를 지정할 수 있습니다.

마스터 암호 정보

.....

DB 관리자 패스워드 등록

제약 조건: 8자 이상의 길이의 영문 ASCII 문자. 특수문자는 포함할 수 없습니다. /(슬래시), '(작은따옴표)', "(큰따옴표)" 및 @(액 기호).

암호 확인 정보

.....

연결

Virtual Private Cloud(VPC) 정보
이 DB 인스턴스의 가상 네트워킹 환경을 정의하는 VPC.

HybridBTS-VPC (vpc-08e8ad0df69a8a5fe) **DB 인스턴스가 배치 될 VPC 선택**

해당 DB 서브넷 그룹이 있는 VPC만 나열됩니다.

ⓘ 데이터베이스를 생성한 후에는 VPC를 변경할 수 없습니다.

서브넷 그룹 정보
선택한 VPC에서 DB 인스턴스가 어떤 서브네트와 IP 범위를 사용할 수 있는지를 정의하는 DB 서브넷 그룹.

bts_db_sg **DB 가 배포 될 Private subnet 그룹 선택**

퍼블릭 액세스 정보

예
VPC 외부의 Amazon EC2 인스턴스 및 디바이스는 데이터베이스에 연결할 수 있습니다. 데이터베이스에 연결할 수 있는 VPC 내부의 EC2 인스턴스 및 디바이스는 데이터베이스에 연결할 수 없습니다.

아니요 **DB 가 내부 자원과의 연결만 가능하도록 퍼블릭 액세스 아니요 선택**
RDS는 데이터베이스만 데이터베이스에 연결할 수 있습니다.

VPC 보안 그룹
데이터베이스에 대한 액세스를 허용할 VPC 보안 그룹을 선택합니다. 보안 그룹 규칙이 적절한 수신 트래픽을 허용하는지 확인합니다.

기존 항목 선택
기존 VPC 보안 그룹 선택

새로 생성
새 VPC 보안 그룹 생성

기존 VPC 보안 그룹
VPC 보안 그룹 선택

BTS_DB_sg X **데이터베이스 보안 그룹 선택**

가용 영역 정보
ap-northeast-2a **데이터베이스 인스턴스가 생성될 가용 영역 선택**

▼ 추가 구성

데이터베이스 옵션, 백업 활성화됨, 역추적 비활성화됨, 향상된 모니터링 비활성화됨, 유지 관리, CloudWatch Logs, 삭제 보호 비활성화됨

데이터베이스 옵션

초기 데이터베이스 이름 정보

BTS

웹 사이트에서 사용할 DB 이름 등록

데이터베이스 이름을 등록합니다.

4. 생성된 데이터베이스에 데이터 생성

```
[ec2-user@ip-10-0-0-97 ~]$ mysql -u BTS_admin -p -h bts-db.c3avcavkhqtj.ap-northeast-2.rds.amazonaws.com
Enter password:
Welcome to the MariaDB
Your MySQL connection
Server version: 8.0.23
Copyright (c) 2000, 20 DB 관리자 이름, RDS 엔드 포인트 입력하여
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
MySQL [(none)]> █
```

```
MySQL [BTS]> desc board;
+-----+-----+-----+-----+-----+
| Field | Type  | Null | Key | Default | Extra       |
+-----+-----+-----+-----+-----+
| num   | int   | NO  | PRI | NULL    | auto_increment
| id    | char(15)| NO |     | NULL    |
| name  | char(10) | NO |     | NULL    |
| subject | char(200)| NO |     | NULL    |
| content | text   | NO  |     | NULL    |
| regist_day | char(20)| NO |     | NULL    |
| hit   | int   | NO  |     | NULL    |
| file_name | char(40) | YES |     | NULL    |
| file_type | char(40) | YES |     | NULL    |
| file_copied | char(40) | YES |     | NULL    |
+-----+-----+-----+-----+-----+
10 rows in set (0.01 sec)
```

```
MySQL [BTS]> desc members;
+-----+-----+-----+-----+-----+
| Field | Type  | Null | Key | Default | Extra       |
+-----+-----+-----+-----+-----+
| num   | int   | NO  | PRI | NULL    | auto_increment
| id    | char(15)| NO |     | NULL    |
| pass  | char(15) | NO |     | NULL    |
| name  | char(10) | NO |     | NULL    |
| email | char(80) | YES |     | NULL    |
| regist_day | char(20)| YES |     | NULL    |
| level | int   | YES |     | NULL    |
| point | int   | YES |     | NULL    |
+-----+-----+-----+-----+-----+
8 rows in set (0.00 sec)
```

데이터베이스 BTS 에 생성한
board, members 테이블

```
MySQL [(none)]> use BTS;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
```

Database changed

```
MySQL [BTS]> select * from members;
```

num	id	pass	name	email	regist_day	level	point
1	hjs6558	password	황재성	hjs6558@naver.com	2021-10-25 (12:09)	9	0
1 row in set (0.00 sec)							

웹 사이트에서 회원정보를 저장하고 테이블에 정보가 들어가는지 확인

6. 멀티 가용 영역 테스트

마스터 데이터베이스가 가용 영역 c에 위치한 상태에서 가용 영역 c에 장애가 발생했을 경우, 가용 영역 a에서 대기하고 있던 스탠바이 데이터베이스 인스턴스가 마스터로 활성화 되는 것을 확인한다.

마스터 데이터베이스는 가용 영역 c에 위치해 있다.

bts-rds-db

요약

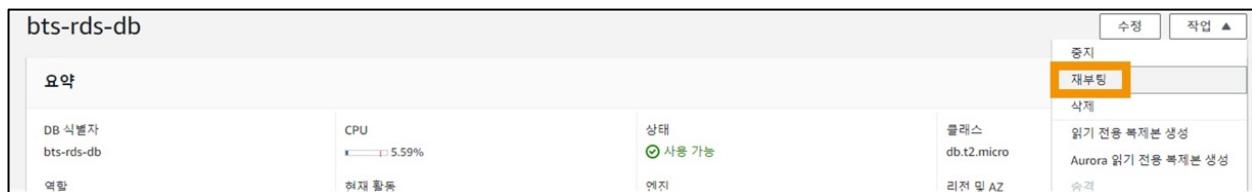
DB 식별자 bts-rds-db	CPU 5.59%	상태 사용 가능	클래스 db.t2.micro
역할 인스턴스	현재 활동 1 연결	엔진 MySQL Community	리전 및 AZ ap-northeast-2c

연결 & 보안 | 모니터링 | 로그 및 이벤트 | 구성 | 유지 관리 및 백업 | 태그

연결 & 보안

엔드포인트 및 포트 엔드포인트 bts-rds-db.c3avcavkhqt.ap-northeast-2.rds.amazonaws.com	네트워킹 네트워킹 가용 영역 ap-northeast-2c	보안 Master DB가 위치한 가용 영역
포트 3306	서브넷 그룹 bts_db_sg	퍼블릭 액세스 가능 아니오
	서브넷 subnet-04f8b8c4a96d7898c subnet-0876b6bdbf81870ee	인증 기관 rds-ca-2019
		인증 기관 날짜 August 23, 2024, 02:08 (UTC±2:08)

장애 발생을 가정하고 마스터 데이터베이스를 장애 조치로 재부팅한다.



DB 인스턴스 재부팅

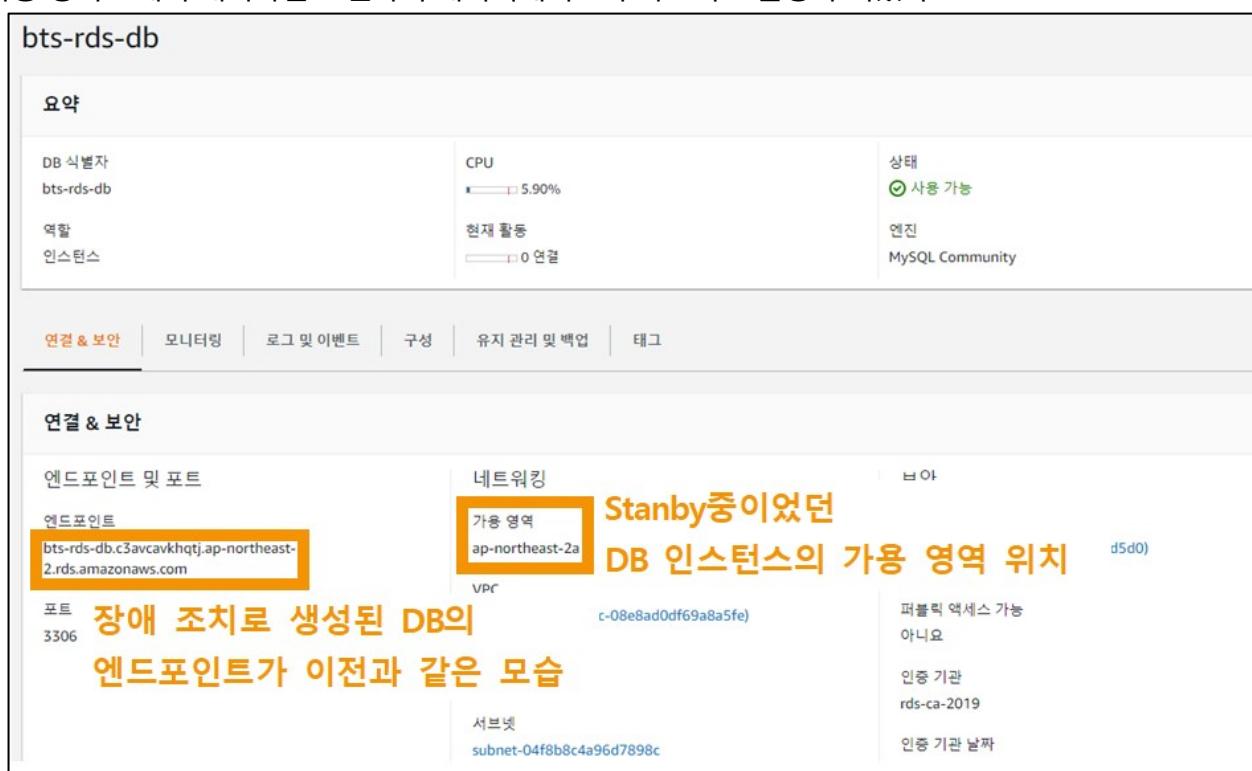
DB 인스턴스

이 DB 인스턴스를 재부팅하시겠습니까?

- bts-rds-db

장애 조치로 재부팅하시겠습니까? 장애 조치 확인을 위한 재부팅

가용 영역 a에서 대기하던 스탠바이 데이터베이스가 마스터로 활성화 되었다.



5.3.2. Read-Only Replica

정의

RDS 데이터베이스의 성능과 내구성을 높여주는 서비스

기능

읽기 전용 복제본으로 읽기 쿼리만을 라우팅하여 원본 DB의 부하를 줄여준다. 읽기 쿼리를 도와 전체 읽기 처리량을 향상 시킨다.

작동 방식

Read-only replica는 비동기식 복제 기능을 사용해 마스터 데이터베이스가 변경될 때마다 업데이트를 한다. CloudWatch에서 마스터 데이터베이스를 모니터링하다 임계치가 초과되거나 마스터 데이터베이스에 문제가 생기면 문제발생 시점 전까지 백업하고 승격된다. 승격된 Replica는 독립 실행형 DB로 읽기, 쓰기 사용이 가능하다.

구축 단계

개요

1. Read-only replica 생성
2. 테스트

구축 단계

과정

1. Read-only replica 생성

AWS 리전

대상 리전
복제본이 시작될 리전입니다.

US East (N. Virginia) 읽기전용 복제본이 시작될 리전 선택

연결

C

서브넷 그룹 정보
선택한 VPC에서 DB 인스턴스가 어떤 서브넷과 IP 범위를 사용할 수 있는지를 정의하는 DB 서브넷 그룹.

vg-bts-db-subnetgroup 복제본을 배포할 서브넷 그룹 선택

퍼블릭 액세스

- 퍼블릭 액세스 가능
VPC 외부의 EC2 인스턴스 및 디바이스가 인스턴스에 연결할 수 있습니다. 지원되는 디바이스 및 인스턴스에 대한 보안 그룹을 정의해야 합니다.
- 퍼블릭 액세스 불가능
DB 인스턴스에 할당된 IP 주소가 없습니다. VPC 외부의 EC2 인스턴스 및 디바이스는 연결할 수 없습니다.

기존 VPC 보안 그룹

VPC 보안 그룹 선택

VG-BTS-DB-sg X DB 보안 그룹 선택

가용 영역 정보
데이터베이스가 생성될 EC2 가용 영역입니다.

us-east-1a 가용 영역 선택

복제 (2) 마스터 데이터베이스의 읽기전용 복제본 생성 완료			
<input type="text"/> 복제본(들) 기준으로 필터링			
DB 인스턴스	역할	리전 및 AZ	복제 원본
bts-rds-db (Seoul)	기본	ap-northeast-2a	-
bts-rds-db-replica	복제본	us-east-1a	bts-rds-db

2. 테스트

서울 리전에 있는 마스터 데이터베이스에 문제가 발생되고 대기 인스턴스 마저 사용 할 수 없다고 가정했을 때, 버지니아 리전에 있는 Read-only Replica 가 승격되어 DB instance 로 사용이 되는지 확인한다.

버지니아 리전 데이터베이스가 replica 상태이다.

The screenshot shows the AWS RDS console under the 'Virginia' region. The database instance 'bts-rds-replica' is listed in the table. The 'Status' column indicates it is a 'Replica' (복제본). The 'Performance' column shows usage at 4.14% and 0 connections. The 'VPC' column shows the VPC ID as vpc-039cc68acd30a053b.

승격 테스트를 위해 서울 리전의 데이터베이스를 삭제한다.

The screenshot shows the AWS RDS console under the 'Seoul' region. The database instance 'rds-db' is selected and the 'Delete' button is highlighted. The status bar at the bottom right shows 'Deleting' (삭제 중).

버지니아 리전 데이터베이스가 인스턴스로 승격되었다.

The screenshot shows the AWS RDS console under the 'Virginia' region. The database instance 'bts-rds-replica' is listed in the table. The 'Status' column indicates it is now a 'Promoted' instance ('인스턴스'). The 'Performance' column shows 0 connections. The 'VPC' column shows the VPC ID as vpc-039cc68acd30a053b.

5.4. Amazon ElastiCache for Redis

정의

클라우드에서 인 메모리 캐시를 배포, 운영, 조정하는데 사용되는 웹 서비스

기능

- 비교적 느린 디스크 기반 데이터베이스에 전적으로 의존하기보다는 빠른 관리형 인 메모리 데이터스토어에서 정보를 검색할 수 있도록 지원함으로써 웹 애플리케이션의 성능을 향상한다.
- 자동 장애 조치가 적용된 Redis 다중 AZ를 통해 고가용성을 지원할 수 있다.
- 애플리케이션은 ElastiCache에서 데이터를 검색하고 캐시에서 데이터를 찾을 수 없을 때에는 데이터베이스를 참조한다.

장점

- 완전 관리형 서비스이기 때문에 사용하지 않는 캐시 노드를 계속 실행할 필요가 없다.
- 많은 용량이 필요할 경우 클러스터를 확장하여 수용할 수 있게 한다.
- AWS 서비스로써 다른 서비스들과 같은 고안정성 인프라에서 실행된다.

HYBRID-BTS

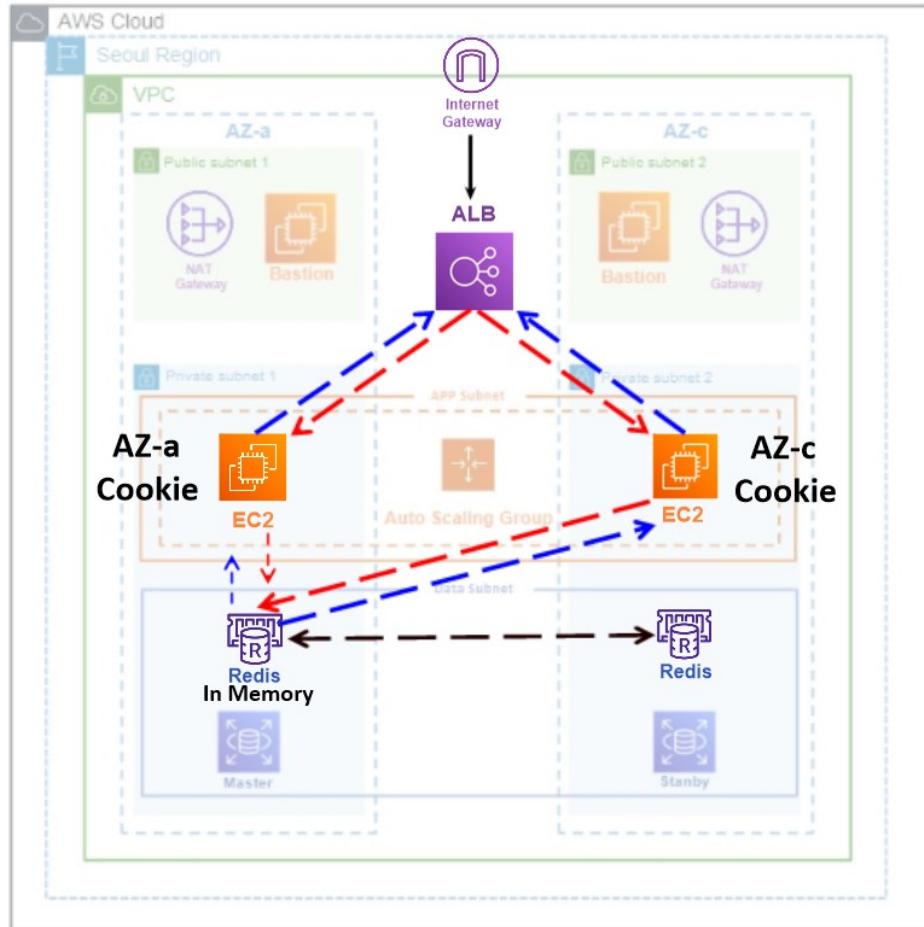


그림 12 HYBRID-BTS 의 ElastiCache 디아이어그램

Application Load Balancer 사용으로 인한 분산 처리로 인해 캐시(쿠키) 또한 분산되어 세션이 끊기는 문제를 해결하기 위하여 Redis 를 고정 세션으로 사용하였다.

구축 단계

개요

1. Redis 클러스터 엔진 선택
2. 기본 설정과 가용 영역 선택
3. 웹 서버 내부에서 Redis 를 세션 핸들러로 하도록 패키지 설치 및 파일 수정
4. 작동 확인

과정

1. Redis 클러스터 엔진 선택

Amazon ElastiCache 클러스터 만들기

클러스터 엔진 **Redis** 데이터베이스
인메모리 데이터 구조 저장소입니다. Redis는 ElastiCache는 Multi-AZ 및 자동 장애 조치 기능을 제공하여 기능이 더욱 강력해졌습니다.

클러스터 모드 활성화

Memcached
고성능, 분산 메모리 객체 캐싱 시스템, 동적 웹 애플리케이션 속도 향상 용도입니다.

위치

위치 선택 **Amazon 클라우드** ElastiCache 인스턴스에 **클러스터를 배치할 위치로 AWS 선택**

온프레미스 AWS Outposts에서 ElastiCache 인스턴스를 생성

2. 기본 설정과 가용 영역 선택

VPC ID vpc-08e8ad0df69a8a5fe

서브넷

서브넷 ID	가용 영역	CIDR 블록
subnet-0876b6bdbf81870ee	ap-northeast-2a	10.0.1.0/24
subnet-04ff8b8c4a96d7898c	ap-northeast-2c	10.0.3.0/24

Redis 가 배치될 서브넷 선택

가용 영역 배치 기본 설정 가용 영역 선택

기본	ap-northeast-2a
복제본1	ap-northeast-2c

3. 웹 서버 내부에서 Redis 를 세션 핸들러로 하도록 패키지 설치 및 파일 수정

php-redis 익스텐션 설치

```
[root@ip-10-0-0-211 ~]# yum install -y php-redis
[root@ip-10-0-0-211 ~]# vi /etc/php-fpm.d/www.conf
```

fpm 설정 파일에서 redis 를 세션 핸들러로 수정

```
425 php_value[session.save_handler] = redis
426 php_value[session.save_path] = tcp://bts-redis.lwiw6a.ng.0001.apn2.cache.amazonaws.com:6379
```

```
[root@ip-10-0-0-211 ~]# systemctl restart httpd
[root@ip-10-0-0-211 ~]# systemctl restart php-fpm
```

변경 된 코드가 적용되도록 httpd, php-fpm 재가동

4. 테스트

Redis-cli 를 사용해서 웹사이트에서 생성되는 쿠키 값들이 정상적으로 저장되는지 확인하고, 세션 핸들러로써 가용 영역이 변경되어도 세션이 끊기지 않고 공동 세션을 유지시켜주는지 확인한다.

연동 확인을 위해 웹 서버에 Redis 를 설치한다.

Redis 설치 위해 gcc 컴파일러 설치

```
[root@ip-10-0-0-211 ~]# yum install -y gcc
wget http://download.redis.io/redis-stable.tar.gz
tar xvzf redis-stable.tar.gz
cd redis-stable
```

make

웹 서버에 소스 코드로 Redis 설치

설치한 Redis 패키지 빌드

가용 영역 a 와 c 에서 웹사이트에 접속해 본 뒤 세션이 끊기지 않는 것을 확인한다.

The image contains two screenshots of the BTS website. Both screenshots show a group of people sitting on beach chairs under orange umbrellas. A callout box in the top right corner of each screenshot says "웹 사이트에 로그인". In the bottom right corner of the top screenshot, there is a callout box with the text "가용 영역 a 로 접속" and "Availability Zone: ap-northeast-2a". In the bottom right corner of the bottom screenshot, there is a callout box with the text "가용 영역 c 로 접속해도 세션 유지 확인" and "Availability Zone: ap-northeast-2c". The website header includes "Category", "Review", "Event", and "Media" tabs, and a "test님 환영합니다" button.

웹사이트와 연동된 Redis 서버에 접속해서 세션 핸들러로써 작동하고 있는 것을 확인한다.

The image shows a terminal session with several highlighted command outputs and annotations:

- redis-stable 폴더로 이동**: The first command shown is [root@ip-10-0-1-27 redis-stable]# pwd /root/redis-stable
- 세션 핸들러로 지정한 마스터 Redis node 에 원격 접속**: The second command shown is [root@ip-10-0-1-27 redis-stable]# src/redis-cli -c -h bts-redis.lwiw6a.ng.0001.apn2.cache.amazonaws.com -p 6379
- Redis 서버에 입력된 명령어들을 출력하는 명령어**: The third command shown is bts-redis.lwiw6a.ng.0001.apn2.cache.amazonaws.com:6379> monitor
- Redis 가 세션 핸들러로 작동하고 있는 모습 확인**: The final command shown is 1635257745.246155 [0 10.0.3.28:47592] "get" "PHPREDIS_SESSION:kmc1p7cl72ncp6flqiv6ts6luc" 1635257745.252097 [0 10.0.3.28:47592] "setex" "PHPREDIS_SESSION:kmc1p7cl72ncp6flqiv6ts6luc" "1440" "" 1635257748.119041 [0 10.0.1.27:35100] "get" "PHPREDIS_SESSION:kmc1p7cl72ncp6flqiv6ts6luc" 1635257748.119536 [0 10.0.1.27:35100] "setex" "PHPREDIS_SESSION:kmc1p7cl72ncp6flqiv6ts6luc" "1440" "userid|s:7:\"hjs6558\";username|s:9:\"\\xed\\x99\\xec\\xe9\\xac\\ec\\x84\\xb1\";userlevel|s:1:\"9\";userpoint|s:1:\"0\";"

5.5. Amazon ECS(Elastic Container Service)

5.5.1. 컨테이너

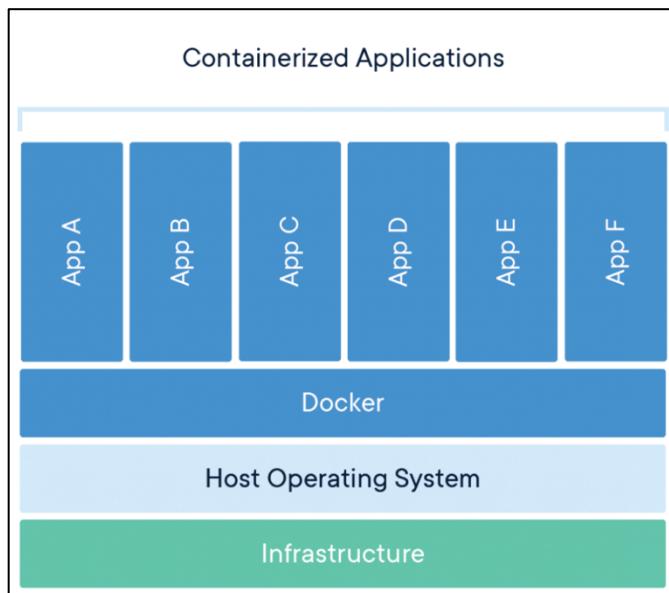


그림 13 컨테이너 구조 (이미지 출처: www.docker.com)

정의

관련 코드, 런타임, 시스템 도구 및 라이브러리 등 소프트웨어 애플리케이션이 필요한 모든 것을 포함하는 소프트웨어 개발의 표준화된 단위. 기존 애플리케이션 배포 방식은 서버에 하이퍼바이저를 이용하여 가상 머신을 배치하고, 각 가상 머신마다 독립 운영체제 위에 애플리케이션을 운용하는 방식이었다. 컨테이너는 그 방식에서 가상 머신과 독립 운영체제 부분을 삭제하고, 호스트 서버의 운영체제 위에 애플리케이션을 컨테이너에 담아 운용한다.

장점

- 애플리케이션 별 독립된 환경이 제공되고 어디에서 실행되는 내용이 유지된다.
- 이미지의 용량이 작기 때문에 빠른 배포가 가능하다.
- 운영체제나 서버 유형 같은 실행 환경의 제한이 줄어들므로 이식성이 좋다.
- Kubernetes, ECS 같은 오케스트레이션 서비스를 통해 복제가 쉽고 자동화가 가능하다.

5.5.2. Amazon ECR(Elastic Cluster Repository)

정의

ECR은 AWS가 제공하는 컨테이너 이미지 레지스트리이다. 사용자가 생성한 ECR 리포지토리에 빌드한 컨테이너 이미지를 푸시하면 ECR이 이미지를 저장한다. ECS와 연동되어 저장된 컨테이너 이미지를 애플리케이션으로 배포할 수 있다.

5.5.3. Amazon ECS(Elastic Container Service)

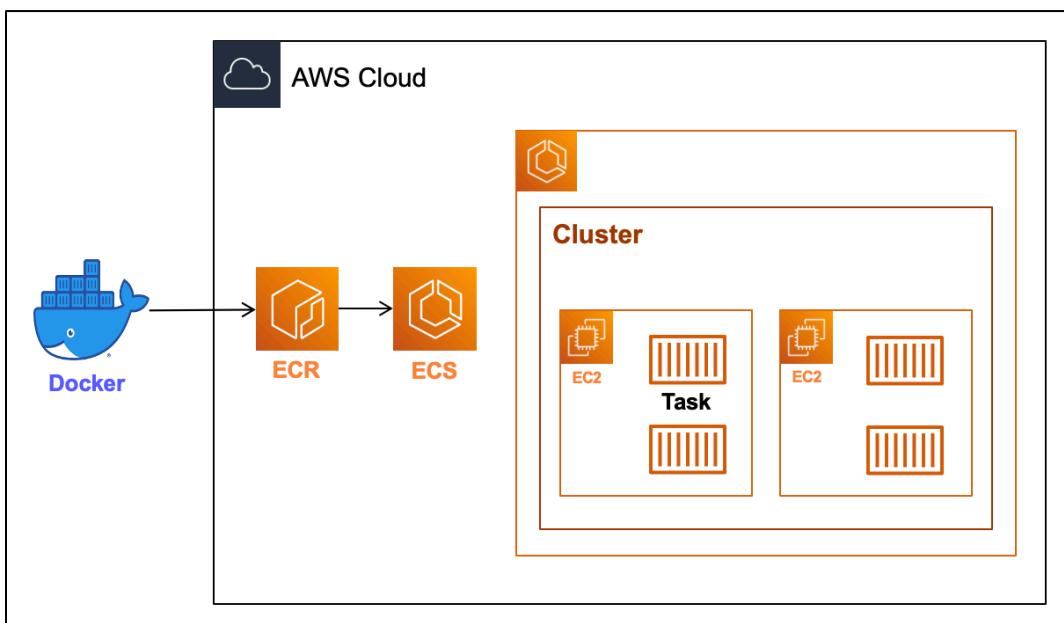


그림 14 AWS 클라우드 내 컨테이너 배포 다이어그램

정의

AWS의 컨테이너 관리 서비스. 클러스터에서 컨테이너를 손쉽게 실행, 중지 및 관리할 수 있게 해준다. ECS는 클러스터, 서비스, 작업 정의, 작업, 컨테이너 에이전트 등의 서비스를 제공한다. Docker Hub, ECR 같은 컨테이너 레지스트리에서 컨테이너 이미지를 불러와 클러스터를 생성하고, 작업 정의와 클러스터 서비스를 정의하면 ECS가 VPC에 컨테이너를 배포하는 전반 과정을 담당한다.

기능

- 클러스터 서비스에서 리소스 요구 사항, 격리 정책, 가용성 요구 사항 등을 지정하면 그에 따라 클러스터 전체에 컨테이너 배치가 가능하다.
- 리전 내의 여러 가용 영역에서 고가용성 방식으로 컨테이너를 실행하는 과정을 간소화한다.
- 클러스터가 실행 중인 경우 클러스터에서 실행할 컨테이너 이미지를 정의하는 작업 정의를 생성한다.

장점

- ECS가 클러스터 관리 및 구성 관리 시스템을 운영하므로 자체적으로 관리 또는 인프라 조정 할 필요가 없다.
- 일관된 빌드 및 배포 환경의 생성이 가능하다.
- マイ크로 서비스 모델에 정교한 애플리케이션 아키텍처를 구축할 수 있다.

요소

- 작업 정의** 애플리케이션을 구성하는 컨테이너의 설정이다. 컨테이너 이미지, 작업의 용량, 볼륨, 포트 개방, 배포할 작업의 개수, 네트워크 모드 등 컨테이너 단위에 대한 정보가 담겨 있다.
- 작업** 작업 정의를 인스턴스화 한 각각의 컨테이너를 지칭한다.
- 클러스터** 하나의 서비스를 공유하는 대상으로, 컨테이너가 배포 된 EC2 인스턴스로 이루어진 논리적 그룹이다. 클러스터의 서비스 설정에 따라 호스트 EC2 인스턴스의 유형, 네트워크, 로드 밸런싱, 오토 스케일링이 결정되고 배포된다.
- 컨테이너 에이전트** 클러스터 내의 각각의 호스트 EC2 안에서 작동한다. 현재 실행중인 작업과 리소스 사용에 대한 정보를 ECS에 전송한다.

HYBRID-BTS

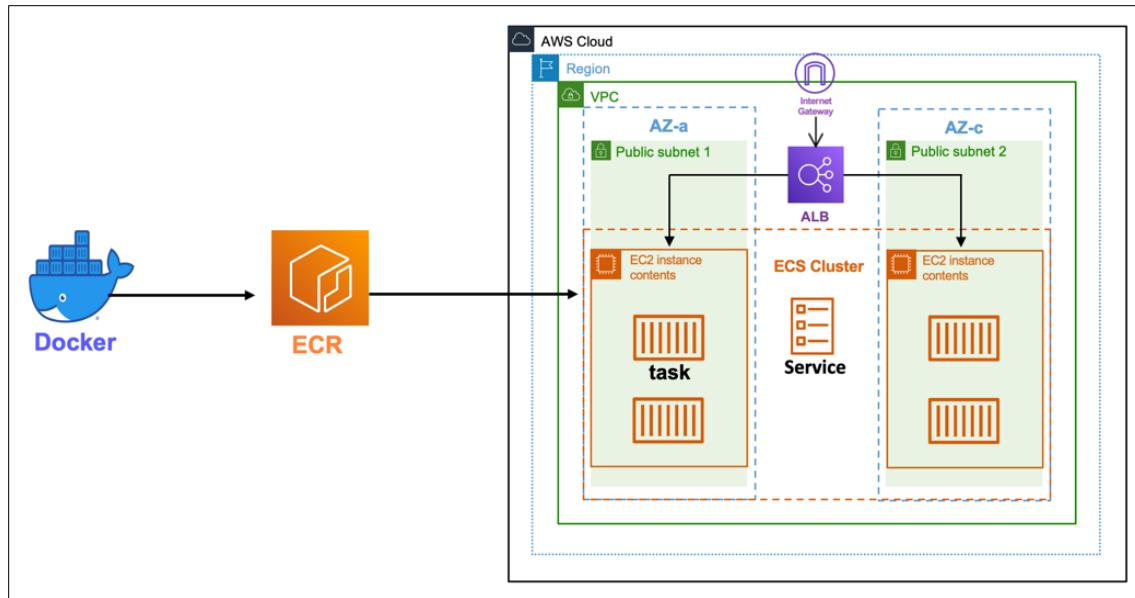


그림 15 HYBRID-BTS 의 ECS 다이어그램

서울 리전의 장애 발생시를 대비하여 버지니아 리전에도 웹 서버를 배포하였다. 컨테이너 호스트 서버인 EC2는 기본적으로 인스턴스 2대, 웹 애플리케이션인 작업은 4개로 설정했다. 서비스 유형을 DAEMON으로 선택하여 Auto Scaling 을 자동화하였다.

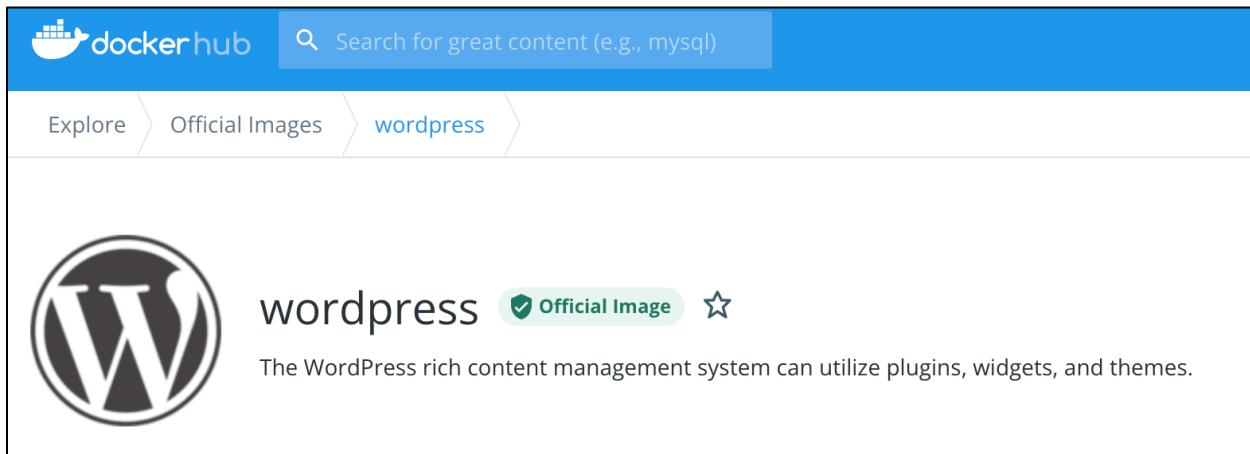
구축 단계

개요

1. 웹 애플리케이션 컨테이너 이미지 생성
2. ECR 에 컨테이너 이미지 푸시
3. 작업 정의
4. ECS 클러스터 생성
5. 클러스터에서 서비스 생성

과정

1. Docker 사용하여 웹 애플리케이션의 컨테이너 이미지를 생성



```
[project@teamOne ~ % docker pull wordpress
```

컨테이너의 베이스 이미지로 사용할 wordpress 이미지를 다운로드한다.

```
[project@teamOne ~ % cd bts_web  
project@teamOne bts_web % vi dockerfile
```

웹사이트 소스코드에 들어가 dockerfile 을 생성한다.

```
FROM wordpress:php7.4-apache wordpress 의 php7.4-apache 버전을 베이스 이미지로 선택  
  
COPY . /var/www/html/ 웹 소스코드를 컨테이너의 /var/www/html/ 폴더로 복사  
  
RUN apt update && apt upgrade -y  
RUN apt install vim systemctl -y 이미지 실행시 입력될 명령어들  
RUN rm /etc/apt/preferences.d/no-debian-php  
  
EXPOSE 80 컨테이너의 80 번 포트 개방
```

dockerfile 의 내용이다.

```
[project@teamOne bts_web % docker build -t btsweb:latest .]
```

현재 작업 디렉토리가 소스코드 폴더 내부인 상태에서 btsweb 이름과 latest 태그로 컨테이너 이미지를 빌드한다.

```
[project@teamOne bts_web % docker run --name bts_config -p 80:80 btsweb:latest
[project@teamOne ~ % docker exec -it btsconfig bash
```

빌드한 이미지를 컨테이너로 실행하고, 설정을 추가하기 위해서 터미널 모드 접속한다.

```
root@dd6f0f0a23e9:/# apt install wget
wget https://download.redis.io/releases/redis-6.2.6.tar.gz
tar xzf redis-6.2.6.tar.gz
cd redis-6.2.6
make
```

소스 코드로 redis-server 패키지 설치한다.

```
[root@dd6f0f0a23e9:/# apt install -y php-redis
```

PHP 세션 핸들러를 Redis 로 지정하기 위해 php-redis 패키지를 설치한다.

```
[root@dd6f0f0a23e9:/# vi /etc/php/7.4/fpm/php.ini
```

php-fpm 기본 설정 파일을 편집한다.

```
extension = redis.so
session.save_handler = redis
session.save_path = "tcp://bts-redis-001.lwiw6a.0001.apn2.cache.amazonaws.com:6379"
```

웹사이트의 세션 핸들러를 Redis 로 하고, 세션 저장소를 ElastiCache for Redis 로 지정하도록 코드 수정한다.

```
root@dd6f0f0a23e9:/# apt install sudo
sudo apt install ufw
sudo service ufw start
sudo ufw default allow outgoing
sudo ufw allow 3306/tcp
sudo ufw allow 6379/tcp

service ufw status
```

root 권한으로 컨테이너 내부에서 Redis/6379, MySQL/3306, HTTP/80 포트를 개방한다.

To	Action	From
--	-----	----
6379/tcp	ALLOW	Anywhere
3306/tcp	ALLOW	Anywhere
80/tcp	ALLOW	Anywhere

포트가 개방된 것을 확인한다.

```
root@dd6f0f0a23e9:/# systemctl enable apache2
systemctl enable php7.4-fpm
systemctl enable redis-server
```

시스템이 실행될 때 자동으로 apache2, php7.4-fpm, redis-server 가 실행되도록 설정한다.

```
root@dd6f0f0a23e9:/# docker inspect -f "{{ .Config.Env }}" btsconfig
```

설정을 추가한 실행 중의 컨테이너를 검사한다.

```
root@dd6f0f0a23e9:/# docker commit --change "ENV DEBUG=true" btsconfig btsweb:latest
```

추가 설정을 마친 컨테이너를 최종 이미지로 빌드한다.

2. ECR에서 bts-web 리포지토리 생성 후 CLI 사용하여 컨테이너 이미지를 푸시

생성한 이미지로 ECS에 컨테이너를 배포하기 위해 ECR에 빌드한 이미지의 리포지토리를 생성한다.



```
[root@dd6f0f0a23e9:/# aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin 48980003453.dkr.ecr.us-east-1.amazonaws.com]
```

AWS us-east-1 리전의 ECR에 로그인한다.

```
[root@dd6f0f0a23e9:/# docker tag btsweb:latest 48980003453.dkr.ecr.us-east-1.amazonaws.com/btsweb:latest]
```

ECR의 btsweb 리포지토리에 생성한 이미지를 태그한다.

```
[root@dd6f0f0a23e9:/# docker push 48980003453.dkr.ecr.us-east-1.amazonaws.com/btsweb:latest]
```

태그한 이미지를 푸시하여 AWS에 도커로 빌드한 이미지를 전달한다.

3. ECS 작업 정의

작업 및 컨테이너 정의 구성

작업 정의는 작업에 포함할 컨테이너와 그 컨테이너들이 삽입 후 작용하는 방식을 지정합니다. 컨테이너가 사용할 데이터 볼륨을 지정할 수도 있습니다. [자세히 알아보기](#)

작업 정의 이름 지정

태스크 정의 이름* BTS-TASK-DEFINITION i

호환성 요구 사항* EC2

태스크 역할 taskRole i

컨테이너에 ElastiCacheReadOnlyAccess, S3ReadOnlyAccess 정책의 역할 부여

네트워크 모드 호스트 i

ElastiCache 와 연동하기 위해 컨테이너에 호스트 EC2 의 IP 부여

컨테이너 추가

▼ 표준

컨테이너 이름* BTS-WEB i 컨테이너 이름 지정 i

이미지* 489851543453.dkr.ecr.us-east-1.amazonaws.com/bts-web i 컨테이너 생성할 이미지의 리포지토리 URI 입력

프라이빗 레지스트리 인증* i

메모리 제한(MiB)* 하드 제한 i 128 i 컨테이너의 하드 용량 지정

+ 소프트 제한 추가

4. ECS 클러스터 생성

클러스터 구성

클러스터 이름* WEB-BTS-CLUSTER WEB-BTS-CLUSTER i

클러스터의 이름 지정

인스턴스 구성

프로비저닝 모델 온디맨드 인스턴스 i

온디맨드 모델 선택

있습니다.

스팟

Amazon EC2 스팟 인스턴스를 사용하면 AWS 클라우드에서 미사용 EC2 용량을 활용할 수 있습니다. 스팟 인스턴스는 온디맨드 요금보다 최대 90% 할인된 가격으로 사용할 수 있습니다. 자세히 알아보기

호스트 EC2 인스턴스의 유형 선택

EC2 인스턴스 유형* t2.micro t2.micro i

클러스터 안에 배치 될 인스턴스 개수 선택

인스턴스 개수* 2 2 i

원격 접속에 사용할 키 페어 선택

키 페어 BTS-ECS-KEY BTS-ECS-KEY i

키 페어가 없으면 EC2 인스턴스에 SSH(으)로 접속할 수 없습니다.

네트워킹

컨테이너 인스턴스가 사용할 VPC를 구성합니다. VPC는 AWS 네트워크의 기본 구성 요소로 Amazon EC2 인스턴스와 같은 AWS 객체로 워집니다. 기존 VPC를 선택하거나 이 마법사로 **클러스터를 배치할 VPC 선택**

VPC: **vpc-08094e508d8dc37...**

Amazon EC2 콘솔에서 **vpc-08094e508d8dc3700**의 구조를 확인합니다.

호스트 EC2 인스턴스를 배포할 프라이빗 서브넷 선택

서브넷:

- subnet-09edae2feaaa49baea (10.0.3.0/24) | Private Subnet 2 - us-east-1c 생성 시 ipv6 할당: Disabled
- subnet-04433186ddd6f8053 (10.0.2.0/24) | Private Subnet 1 - us-east-1a 생성 시 ipv6 할당: Disabled

ALB로부터 HTTP, HTTPS
Bastion으로부터 SSH
0.0.0.0/0으로부터 MySQL, Redis 허용 보안 그룹 지정

퍼블릭 IP 자동 할당

보안 그룹: **sg-06c7a61ef067013b1...**

컨테이너 인스턴스 IAM 역할

Amazon ECS 컨테이너 에이전트는 사용자를 대신하여 Amazon ECS API 작업을 호출하므로 에이전트가 사용자에게 속한다는 것을 서비스가 알기 위해서는 에이전트를 실행하는 컨테이너 인스턴스에 ecsInstanceRole IAM 정책과 역할이 필요합니다. ecsInstanceRole이 아직 없는 경우, AWS가 생성합니다.

컨테이너 인스턴스에 IAM 역할 부여

컨테이너 인스턴스 IAM 역할: **ecsInstanceRole**

5. 클러스터 서비스 생성

서비스 구성

서비스를 통해 클러스터에서 실행하고 유지 관리할 작업 정의의 사본 개수를 지정할 수 있습니다. Elastic Load Balancing 로드 밸런서를 옵션으로 사용하여 들어오는 트래픽을 서비스 내 컨테이너에 분산할 수 있습니다. Amazon ECS는 로드 밸런서를 통해 작업의 개수를 유지하고 작업 일정을 조정합니다. 서비스 Auto Scaling을 옵션으로 사용하여 서비스 내 작업의 개수를 조정할 수도 있습니다.

작업 유형 FARGATE EC2 EXTERNAL

용량 공급자 전략으로 전환 ●

서비스가 배포할 컨테이너의 작업 정의 선택

작업 정의 BTS-TASK-DEFINITION 값 입력

개정 3 (latest)

클러스터 WEB-BTS-CLUSTER ●

서비스 이름 service **서비스 이름 지정** ●

작업 개수 REPLICA DAEMON ●

**ECS 가 메트릭에 따라 작업 개수를 오토 스케일링
하는 DAEMON 서비스 유형 선택**

로드 밸런서 유형*

클러스터 내부에서 웹 트래픽을 로드 밸런싱 할 ALB 유형 선택

Application Load Balancer

컨테이너가 동적 호스트 포트 매핑을 사용하도록 허용합니다(컨테이너 인스턴스마다 다중 작업이 허용됨). 여러 서비스가 규칙 기반 라우팅 및 경로를 사용하여 단일 로드 밸런서에서 동일한 리스너 포트를 사용할 수 있습니다.

Network Load Balancer

Network Load Balancer는 OSI(개방형 시스템 상호 연결) 모델의 4번째 계층에서 작동합니다. 로드 밸런서가 요청을 수신하면, 흐름 해시 라우팅 알고리즘을 사용하여 기본 규칙의 대상 그룹에서 대상을 선택합니다.

Classic Load Balancer

정적 호스트 포트 매핑을 필요로 합니다(컨테이너 인스턴스당 1개의 작업만 허용됨). 규칙 기반 라우팅 및 경로가 정적입니다.

서비스에 클러스터 관리 역할 부여

서비스의 IAM 역할 선택 AWSServiceRoleForECS ●

로드 밸런서 이름 WEB-ALB **퍼블릭 서브넷 1,2 에 배치된 로드 밸런서 선택**



5.6. Amazon VPC peering

정의

두 VPC 간에 비공개로 트래픽을 라우팅할 수 있도록 하는 네트워킹 연결

작동방식

피어링 연결 된 VPC 의 인스턴스들은 동일한 네트워크 안에 있는 것처럼 서로 통신할 수 있다. 자체 VPC, 다른 AWS 리전의 VPC 사이에서 VPC 피어링 연결을 생성할 수 있다. VPC Peering 은 AWS 자체 네트워크를 사용한다. 별도의 물리적 하드웨어에 의존하지 않아 통신에 대한 단일 장애 지점이나 대역폭 병목 현상이 없다.

HYBRID-BTS

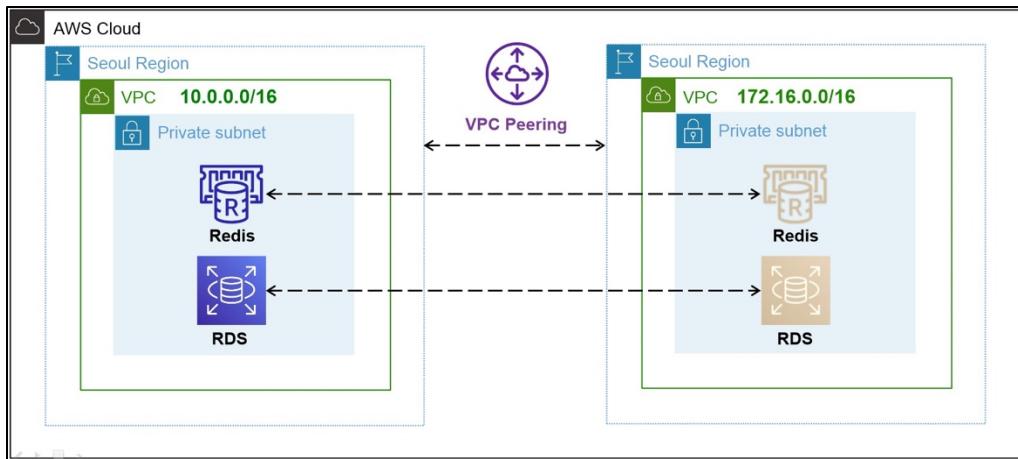


그림 17 HYBRID-BTS 의 VPC peering 디아이어그램

Seoul region 과 Virginia region 간에 VPC peering 을 생성하여 Virginia region 에서도 Seoul region 의 데이터베이스와 Redis 를 사용할 수 있게 했다.

구축 단계

개요

1. 피어링 연결 생성
2. 라우팅 테이블 수정
3. 데이터베이스의 보안 그룹 변경
4. 피어링 테스트

과정

1. 피어링 연결 생성

Seoul region에서 Virginia region으로 요청하는 피어링 연결을 생성한다.

피어링 연결 생성

VPC 피어링 연결은 두 VPC 간에 트래픽을 비공개로 라우팅할 수 있게 하는 두 VPC 간의 네트워킹 연결입니다. 정보

피어링 연결 설정

이름 - 선택 사항

'Name' 키와 사용자가 지정하는 값을 포함하는 태그를 생성합니다.

BTS-vpc-peering

사용할 VPC 피어링 이름 지정

피어링할 로컬 VPC 선택

VPC ID(요청자)

vpc-08e8ad0df69a8a5fe (HybridBTS-VPC)

피어링 연결을 요청하는 VPC (Seoul region) 선택

피어링할 다른 VPC 선택

계정

내 계정 피어링 수락 VPC의 계정 유형 선택

다른 계정

리전

현재 리전(ap-northeast-2)

다른 리전

Virginia region의 VPC 아이디 입력

미국 동부 (버지니아 북부) (us-east-1)

VPC ID(수락자)

vpc-039cc68acd30a053b

피어링 연결 (1/1) 정보

Q 피어링 연결 필터링

Seoul region의 VPC 피어링 연결 상태

Name	피어링 연결 ID	상태	요청자 VPC	수락자 VPC	요청자 CIDR	수락자 CIDR
<input checked="" type="radio"/> BTS-VP	pcx-0eb9986239b5fbe50	<input checked="" type="radio"/> 활성	vpc-08e8ad0df69a8a5fe / Hy...	vpc-039cc68acd30a053b	10.0.0.0/16	172.16.0.0/16

Virginia region 의 VPC 피어링 연결 상태						
Name	피어링 연결 ID	상태	부모 VPC	부모 VPC	수락자 CIDR	요청자 소유자 ID
bts-vp	pcx-0eb9986239b5fbe50	활성	vpc-08e8ad0df69a8a5fe	vpc-039cc68acd30a053b / BT...	10.0.0.0/16	172.16.0.0/16

2. 라우팅 테이블 설정

VPC 간 Private subnet에 접근이 가능하도록 라우팅 테이블을 수정한다.

라우팅 (4)	
Seoul region 의 프라이빗 라우팅 테이블에	
대상	Virginia region 의 프라이빗 서브넷 등록하고 대상으로 피어링 지정
172.16.2.0/24	pcx-0eb9986239b5fbe50
172.16.3.0/24	pcx-0eb9986239b5fbe50

라우팅 (4)	
Virginia region 의 프라이빗 라우팅 테이블에	
대상	Seoul region 의 프라이빗 서브넷 등록하고 대상으로 피어링 지정
10.0.1.0/24	pcx-0eb9986239b5fbe50
10.0.3.0/24	pcx-0eb9986239b5fbe50
172.16.0.0/16	local
0.0.0.0/0	nat-07e923e1b6c9fdb0b

3. Seoul region 의 데이터베이스 보안 그룹 설정

Virginia region에서 데이터베이스를 사용할 수 있도록 설정한다.

Seoul region 의 데이터베이스 보안 그룹							
인바운드 규칙 (3)		Virginia region 으로부터 3306 포트 접근 허용					
sg-096a6cd6d7311d5d0 - BTS_DB_sg							Reachability Analyzer 실행
Name	보안 그룹 규칙 ID	IP 버전	유형	프로토콜	포트 범위	소스	설명
-	sgr-045facc76849b502e	IPv4	MySQL/Aurora	TCP	3306	172.16.0.0/16	-
-	sgr-0918d2d1c851bbf1	IPv4	MySQL/Aurora	TCP	2300	0.0.0.0/0	-
-	sgr-0e0b7ef88hd4ca5rf6d	-	MySQL/Aurora	TCP	3306	sg-0941c469189961c...	-

4. 테스트

VPC peering 으로 연결된 리전 간에 프라이빗 서브넷 안 웹 서버와의 통신을 테스트한다.

서울 리전에서 버지니아 리전으로 ping 테스트 한다.

The screenshot shows the AWS CloudWatch Metrics interface for an EC2 instance. The instance is named 'ECS Instance - EC2ContainerService-WEB-BTS-CLUSTER'. It has an ID of i-0b6fd01c34fafafe2a3 and is in a 'Running' state. The instance type is t2.micro and it has 2/2 successful health checks. Below the table, a message says 'Now you can use the Reachability Analyzer to check network connectivity.' A tab labeled 'Networking' is selected. Under the 'Networking' tab, there is a section titled 'Private IPv4 Address' which displays the IP address 172.16.2.196.

Seoul region 서버에서 Virginia region 서버로 ping 성공

```
[ec2-user@ip-10-0-1-232 ~]$ ping 172.16.2.196
PING 172.16.2.196 (172.16.2.196) 56(84) bytes of data.
64 bytes from 172.16.2.196: icmp_seq=1 ttl=255 time=171 ms
64 bytes from 172.16.2.196: icmp_seq=2 ttl=255 time=171 ms
64 bytes from 172.16.2.196: icmp_seq=3 ttl=255 time=171 ms
64 bytes from 172.16.2.196: icmp_seq=4 ttl=255 time=171 ms
```

버지니아 리전에서 웹사이트 접근하여 서울 리전에 배치한 데이터베이스와의 연결을 확인한다.



회원 가입
Seoul region 의 데이터베이스와의 연동을 테스트하기 위해 회원 가입

아이디	<input type="text" value="jaesunc"/>
비밀번호	<input type="password" value="*****"/>
비밀번호 확인	<input type="password" value="*****"/>
이름	<input type="text" value="jaesung"/>
이메일	<input type="text" value="jaesung"/> @ <input type="text" value="naver.com"/>



서버의 위치는 Virginia region

Instance-ID: i-0b6fd01c34fafef2 Availability Zone : us-east-1a

5.7. AWS CloudFormation

정의

예상 가능한 작업을 반복적으로 수행하게 리소스 프로비저닝을 자동화 시키는 AWS의 IaC(Infrastructure as Code) 서비스

기능

인프라를 코드로 처리함으로써 손쉬운 방법으로 관련된 AWS 및 서드 파티 리소스 모음을 모델링하고, 일관된 방식으로 간단히 프로비저닝하고, 수명 주기 전반에 걸쳐 관리할 수 있다. CloudFormation 템플릿에는 원하는 리소스와 종속성이 설명되어 있으므로 이를 모두 하나의 스택으로 구성하고 시작할 수 있다. 리소스를 개별적으로 관리하는 대신 템플릿을 통해 전체 스택을 단일 단위로 처리하여 필요한 만큼 자주 생성 및 업데이트하고 삭제할 수 있다. 스택은 여러 AWS 계정 및 AWS 리전에 걸쳐 관리 및 프로비저닝 할 수 있다.

요소

- **Template** 스택을 구성하는 AWS 리소스를 JSON 혹은 YAML 형식으로 선언한 텍스트 파일
- **Parameter** CloudFormation 을 생성 및 업데이트할 때 Template 에 전달하는 커스텀 값
- **Stack** 하나의 단위로 관리할 수 있는 AWS 리소스들의 모음

HYBRID-BTS

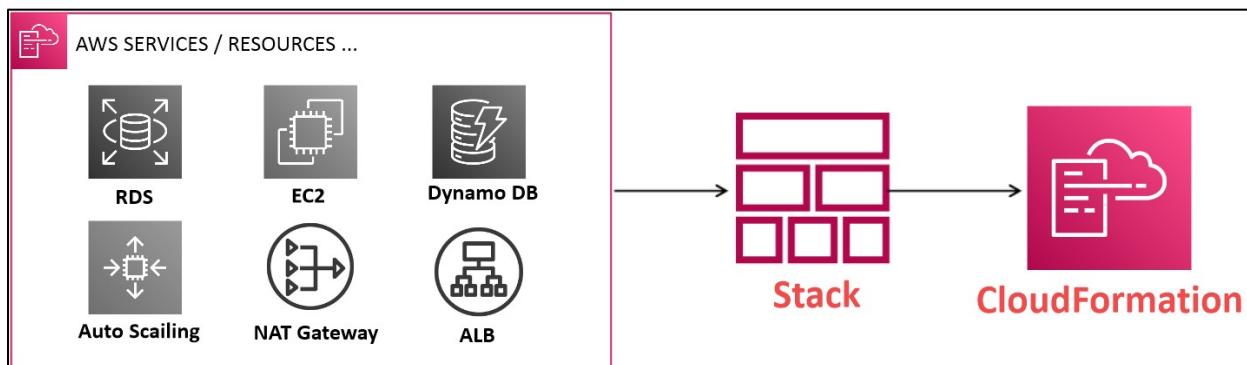


그림 18 HYBRID-BTS 의 CloudFormation 디어그램

서울 리전에 구성된 인프라를 CloudFormation 스택으로 생성하여 이후의 프로비저닝을 할 때와 재해 발생시를 대비하였다.

CloudFormation-Functions

AWS::Region

<pre> Mappings: RegionMap: ap-northeast-2: 서울 NAT: "ami-00550ccc38e992b78" us-east-1: 오레곤 NAT: "ami-00a36856283d67c39" us-west-2: 버지니아 NAT: "ami-0032ea5ae08aa27a2" </pre>	<pre> NATInstance1: Type: AWS::EC2::Instance Properties: ImageId: !FindInMap - RegionMap - !Ref 'AWS::Region' - NAT KeyName: !Ref NatKey InstanceType: t2.micro SecurityGroupIds: - !Ref NATSG SourceDestCheck: false SubnetId: !Ref BTSPublic1 Tags: - Key: Name </pre>
--	--

AWS::Region 은 포괄 리소스를 생성하는 AWS 리전을 나타내는 “문자열”로 반환 해주는 가상 파라미터이다. 예를 들면 서울 리전에 스택을 올리게 되면 서울 리전 ID 인 “ap-northeast-2” 가 반환이 되어 RegionMap 를 선택 하여 리전의 NAT 이미지를 가져 오게 되는 것이다. 서울, 오레곤, 버지니아 중에 하나를 선택하여 스택을 올리면 맵핑을 읽어 스택을 생성 할 수 있다.

!Ref

```

NATSG:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: Allow All Traffics to Internet
    VpcId: !Ref BTSVPC
    SecurityGroupIngress:
      - IpProtocol: All
        CidrIp: 0.0.0.0/0
    Tags:
      - Key: Name
        Value: BTS-NAT-SG

NATInstance1:
  Type: AWS::EC2::Instance
  Properties:
    ImageId: !FindInMap
      - RegionMap
      - !Ref 'AWS::Region'
      - NAT
    KeyName: !Ref NatKey
    InstanceType: t2.micro
    SecurityGroupIds:
      - !Ref NATSG
    SourceDestCheck: false
    SubnetId: !Ref BTSPublic1
    Tags:
      - Key: Name
        Value: BTS-NAT-Instance1

```

지정된 파라미터 또는 리소스에 대한 정보를 반환한다. BTS-NAT-SG 는 보안 그룹 이름을 뜻 하며, BTS-NAT-SG 가 생성될 ID 값은 “NATSG” 라는 것을 의미한다. 보안 그룹을 사전에 NATSG 를 정의 하고, 스택이 생성될 때 NATSG 값을 참조해서 NATInstance1 생성시 보안 그룹은 NATSG 가 된다.

!Select

<p>BTSPublic1</p> <pre>Type: AWS::EC2::Subnet Properties: VpcId: !Ref BTSVPC CidrBlock: !Ref Public1Cidr AvailabilityZone: !Select - 0 - !GetAZs Ref: 'AWS::Region' MapPublicIpOnLaunch: true Tags: - Key: Name Value: BTS-Public1</pre>	<p>BTSPublic1</p> <pre>Type: AWS::EC2::Subnet Properties: VpcId: !Ref BTSVPC CidrBlock: !Ref Private1Cidr AvailabilityZone: !Select - 0 - !GetAZs Ref: 'AWS::Region' Tags: - Key: Name Value: BTS-Private1</pre>
<p>BTSPublic2</p> <pre>Type: AWS::EC2::Subnet Properties: VpcId: !Ref BTSVPC CidrBlock: !Ref Public2Cidr AvailabilityZone: !Select - 2 - !GetAZs Ref: 'AWS::Region' MapPublicIpOnLaunch: true Tags: - Key: Name Value: BTS-Public2</pre>	<p>BTSPublic2</p> <pre>Type: AWS::EC2::Subnet Properties: VpcId: !Ref BTSVPC CidrBlock: !Ref Private2Cidr AvailabilityZone: !Select - 2 - !GetAZs Ref: 'AWS::Region' Tags: - Key: Name Value: BTS-Private2</pre>

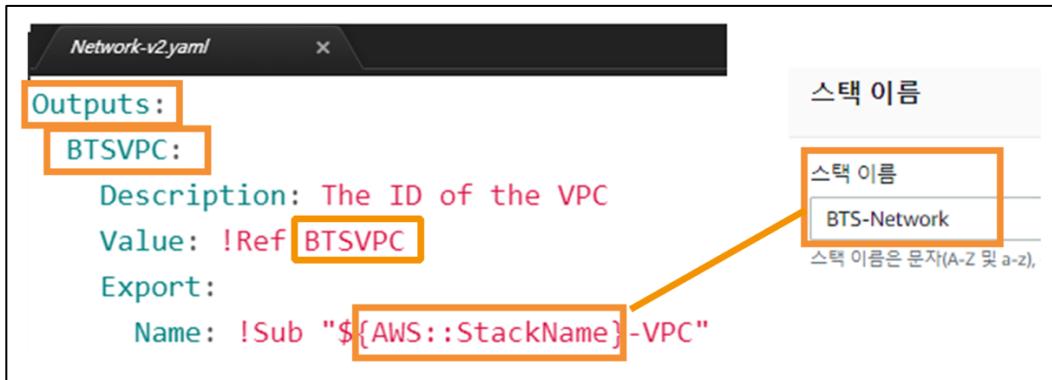
숫자 0 과 1, 2, 3 등으로 표현을 하는데 가용 영역을 표시하여 차례로 A 와 B, C 로의 문자로 변환 해주는 함수이다. 숫자 "0" 는 A, 숫자 2 는 "C"를 뜻한다. 같이 쓰는 함수로 **!GetAZs** 가 있는데 이는 스택 생성시 선택 가능한 가용 영역을 표시해준다. Private1 과 Public1 은 가용 영역 A, Private2 과 Public2 은 가용 영역 C 로 생성된다.

Fn::Base64

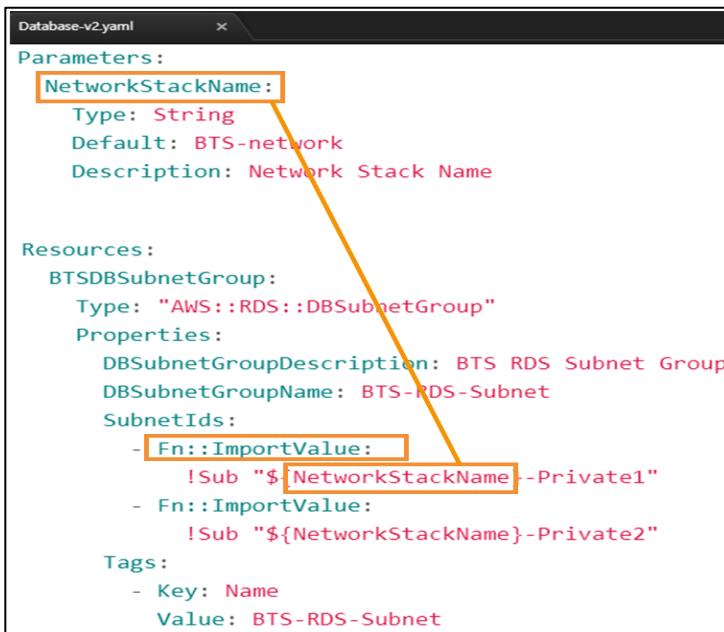
```
UserData:  
Fn::Base64: !Sub |  
#!/bin/bash  
yum install -y httpd mysql  
amazon-linux-extras install -y php7.2  
aws s3 cp s3://btsbucket-123/htdocs.zip .  
unzip htdocs -d /var/www/html S3버킷의 Web 파일 url 주소  
systemctl start httpd  
systemctl enable httpd
```

Launch Template 을 생성할 때 UserData 의 명령어를 그냥 보내면 코드를 읽지 못한다. Fn::Base64 내장 함수로 UserData 속성을 통해 인코딩해야 한다. 인코딩 된 코드를 EC2 인스턴스에 전달해 처리 할 수 있다.

Fn::ImportValue

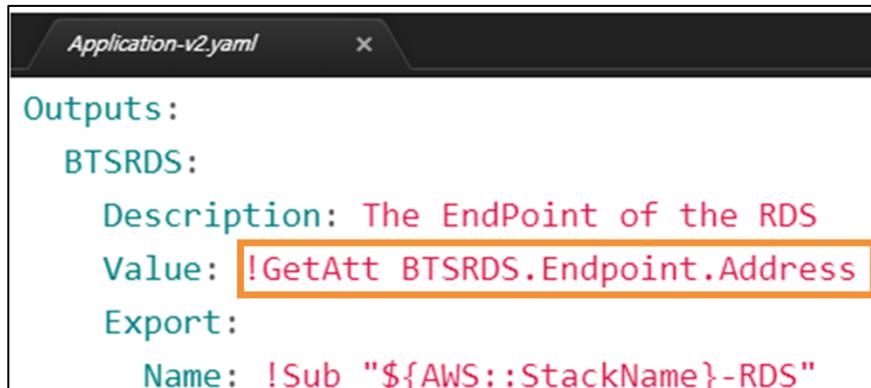


Template 을 네트워크, 데이터베이스, 스토리지 등으로 세분화해서 생성할 때, 다른 템플릿에 있는 값을 받아와야 하게 된다. 이때 사용하는 함수가 Fn::ImportValue: 이다. 먼저 natwork-Template에서 BTSVPC의 값을 "\${AWS::StackName}-VPC"의 값으로 내보낸다. AWS::StackName은 예를 들면 BTS-network라는 스택을 생성하게 되면 BTS-network 이름이 AWS::StackName 안으로 대입이 되어 BTS-network-VPCID는 "VPC-xxxxx"라고 알려 주게 되는 것이다.



나중 작업 시 다른 스택 이름을 써도 적용이 가능하게 Database-Template에서 NetworkStackName 파라미터를 생성한다. 지정된 스택의 이름을 선택해서, NetworkStackName의 값으로 입력된다. 스택을 생성할 때 \${NetworkStackName} 변수로 참조하면 추후 스택의 이름이 변동이 되어도 알아서 변경 처리가 가능하다.

!GetAtt



엔드 포인트 값을 참조할 때 쓰이는 함수이다. Value 에 !GetAtt 함수로 BTSRDS 의 엔드 포인트 값을 넣으면 BTSRDS 는 참조될 때 데이터베이스의 엔드 포인트를 반환한다.

5.8. AWS Global Accelerator

정의

Amazon Web Service 의 글로벌 네트워크 인프라를 사용하여 트래픽의 성능을 올려주는 네트워킹 서비스

기능

- AWS 글로벌 네트워크를 통한 트래픽을 클라이언트와 가장 가까운 리전의 엔드 포인트로 보낸다.
- 로컬 및 글로벌 사용자들이 이용하는 인터넷 애플리케이션의 가용성을 향상 시킨다.

Hybrid-BTS

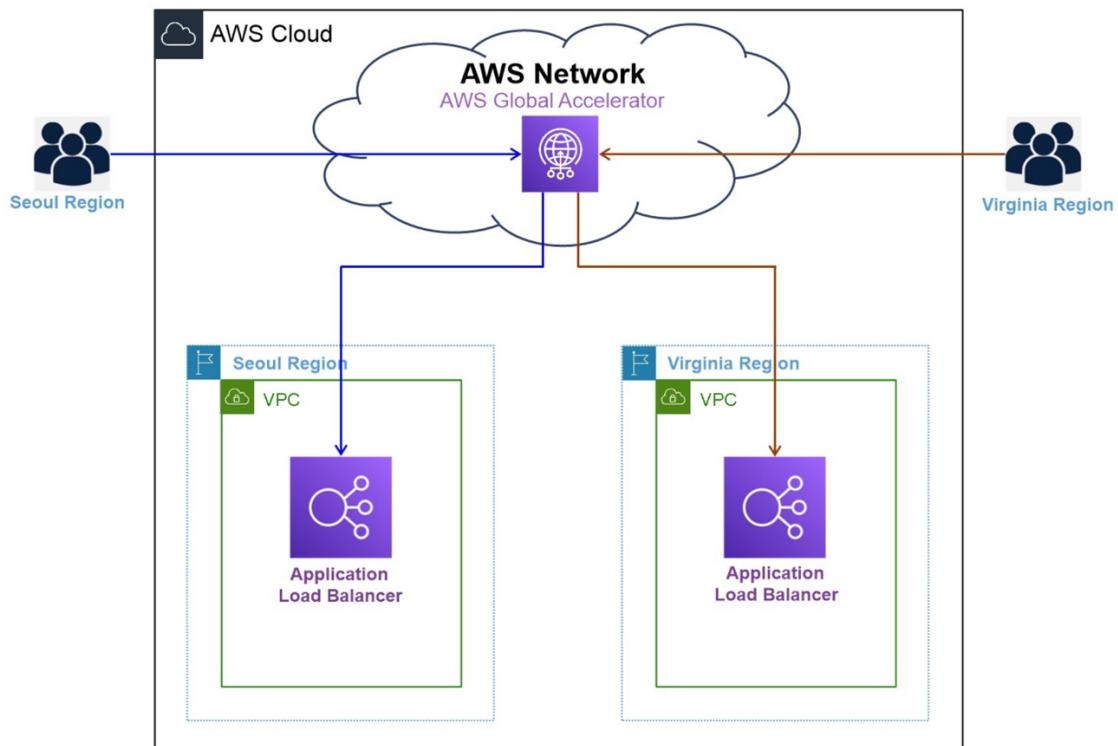


그림 19 HYBRID-BTS 의 Global Accelerator 디아이어그램

현재 웹 서버가 서울 리전과 버지니아 북부에 위치해 멀티 리전 구조이다. Route 53 와 각 리전 사이에 Global Accelerator 를 배치하여 사용자가 www.hybridbts.com 도메인을 입력하여 접속하면 사용자의 지역에 가까운 쪽의 리전으로 라우팅 되도록 하였다. 한쪽 리전에 이상이 생길 경우 다른 리전으로 자동 우회된다.

구축 단계

개요

1. Accelerator 기본 구성
2. 리스너 설정
3. Route 53 레코드 생성
4. 작동 확인

과정

1. Accelerator 기본 구성

기본 구성

가속기의 이름을 입력하고 다른 옵션을 선택하십시오.

가속기 이름
액셀러레이터와 연결할 이름을 제공합니다.

BTS-AGA Accelerator 이름 입력

공백 없이 문자, 숫자 또는 하이픈(-)만 입력하십시오.

가속기 유형
생성하려는 액셀러레이터 유형을 선택합니다.

기준
트래픽을 사용자에게 가장 가까운 정상 엔드포인트로 자동 라우팅

커스텀 라우팅
여러 사용자를 VPC 서브넷의 특정 EC2 인스턴스 대상으로 라우팅

IP 주소 유형

IPv4 커스텀으로 할 경우 선택 가능

2. 리스너 설정

개방할 포트, 접속을 가속화 할 리전, 서버의 엔드 포인트를 지정한다.

청취자

수신할 특정 포트 또는 포트 범위를 선택하여 수신기를 지정합니다.

포트 정보 443 HTTPS	프로토콜 정보 TCP	클라이언트 선호도 정보 없음
--------------------------------------	---------------------------	-------------------------------

쉼표를 사용하여 포트 번호 또는 범위를 구분합니다.

리스너 추가

리스너: 443 TCP

각 수신기에는 여러 끝점 그룹이 있을 수 있습니다. 각 엔드포인트 그룹에는 하나의 리전에 있는 엔드포인트만 포함될 수 있습니다. 엔드포인트에 대한 트래픽은 엔드포인트에 도달하지 않습니다.

지역 정보 ap-북동쪽-2	서울 리전	트래픽 다이얼 정보 100 기본값, 0은 트래픽 차단
▶ 포트 재정의 구성		
▶ 상태 확인 구성		
us-east-1	버지니아 리전	100 0에서 100 사이의 숫자입니다.

리스너: 443 TCP

Global Accelerator는 이러한 포트에 도착하는 트래픽을 지역 엔드포인트 그룹의 엔드포인트로 라우팅합니다. 엔드포인트 그룹의 모든 엔드포인트는 동일한 트래픽 다이얼을 사용합니다.

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

엔드포인트 유형 정보 애플리케이션 로드 밸런서	엔드포인트 정보 arn:aws:elasticloadbalancing:ap-nort...	무게 정보 128 가중치 설정 가능, 기본값으로 같은값 입력 0에서 25
서울 리전의 ALB 엔드포인트 지정		
클라이언트 IP 주소 정보 유지 Global Accelerator는 확인란을 선택 취소하여 기능을 비활성화하지 않는 한 인터넷 연결 Application Load Balancer에 대한 클라이언트 IP 주소를 유지 및 EC2 인스턴스는 클라이언트 IP 주소를 자동으로 보존합니다. 엔드포인트가 보존된 클라이언트 IP 주소의 트래픽을 수락하도록 구성되었는지 확인합니다. <input checked="" type="checkbox"/> 클라이언트 IP 주소 유지		
엔드포인트 추가		
▼ 엔드포인트 그룹: us-east-1 트래픽 다이얼: 100 %		
엔드포인트 유형 정보 애플리케이션 로드 밸런서	엔드포인트 정보 arn:aws:elasticloadbalancing:us-east-...	무게 정보 128 0에서 255 사이의 숫자입니다.
버지니아 리전의 ALB 엔드포인트 지정		
클라이언트 IP 주소 정보 유지 Global Accelerator는 확인란을 선택 취소하여 기능을 비활성화하지 않는 한 인터넷 연결 Application Load Balancer에 대한 클라이언트 IP 주소를 유지 및 EC2 인스턴스는 클라이언트 IP 주소를 자동으로 보존합니다. 엔드포인트가 보존된 클라이언트 IP 주소의 트래픽을 수락하도록 구성되었는지 확인합니다. <input checked="" type="checkbox"/> 클라이언트 IP 주소 유지		

생성된 Accelerator 를 확인한다.

BTS-AGA 구성

이름: **BTS-AGA Accelerator 이름**

고정 IP 주소 설정

IP 주소	주소 풀
15.197.247.24	아마존
3.33.245.194	아마존

프로비저닝 상태: 배포됨

DNS 이름 정보: a097a8c9419edff57.awsglobalaccelerator.com

Amazon에서 부여하는 고정된 진입점 역할을 하는 정적 퍼블릭 IP 2개

만들어진: 2021년 11월 11일 목요일 오전 8:48 GMT

GMT

3. Route 53 A 레코드 생성

빠른 레코드 생성 정보

레코드 1

레코드 이름: **www** .hybridbts.com

레코드 유형: **A – IPv4 주소 및 일부 AWS 리소스로 트래픽 ...**

트래픽 라우팅 대상: **Global Accelerator에 대한 별칭**

대상 상태 평가: 예

라우팅 정책: 단순 라우팅

마법사로 전환

생성한 A 레코드를 확인한다.

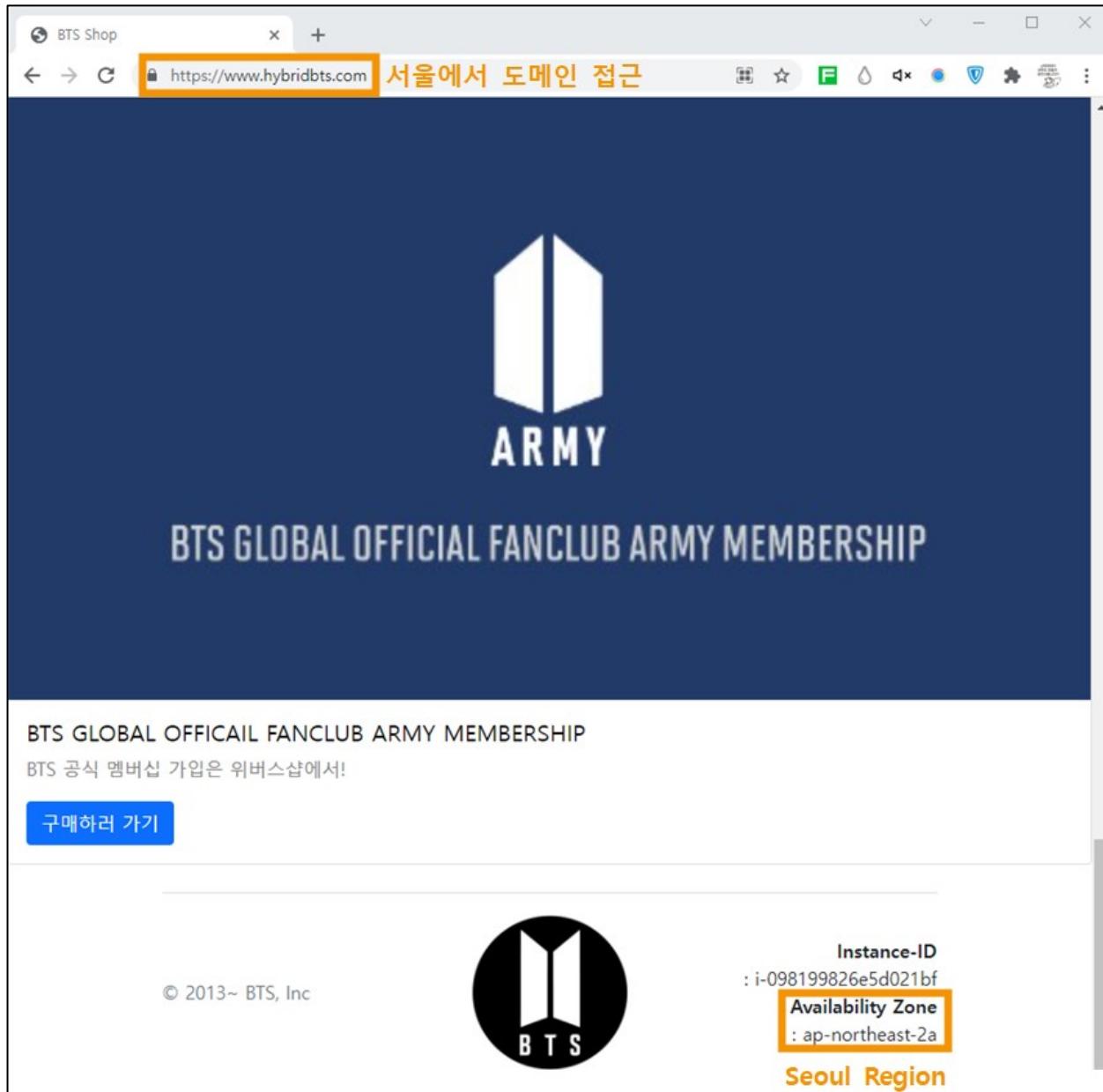
레코드 이름	유형	라우팅 ...	차별...	값/트래픽 라우팅 대상
www.hybridbts.com	A	단순	-	a097a8c9419edff57.awsglobalaccelerator.com.

Accelerator로 라우팅 대상 설정 확인

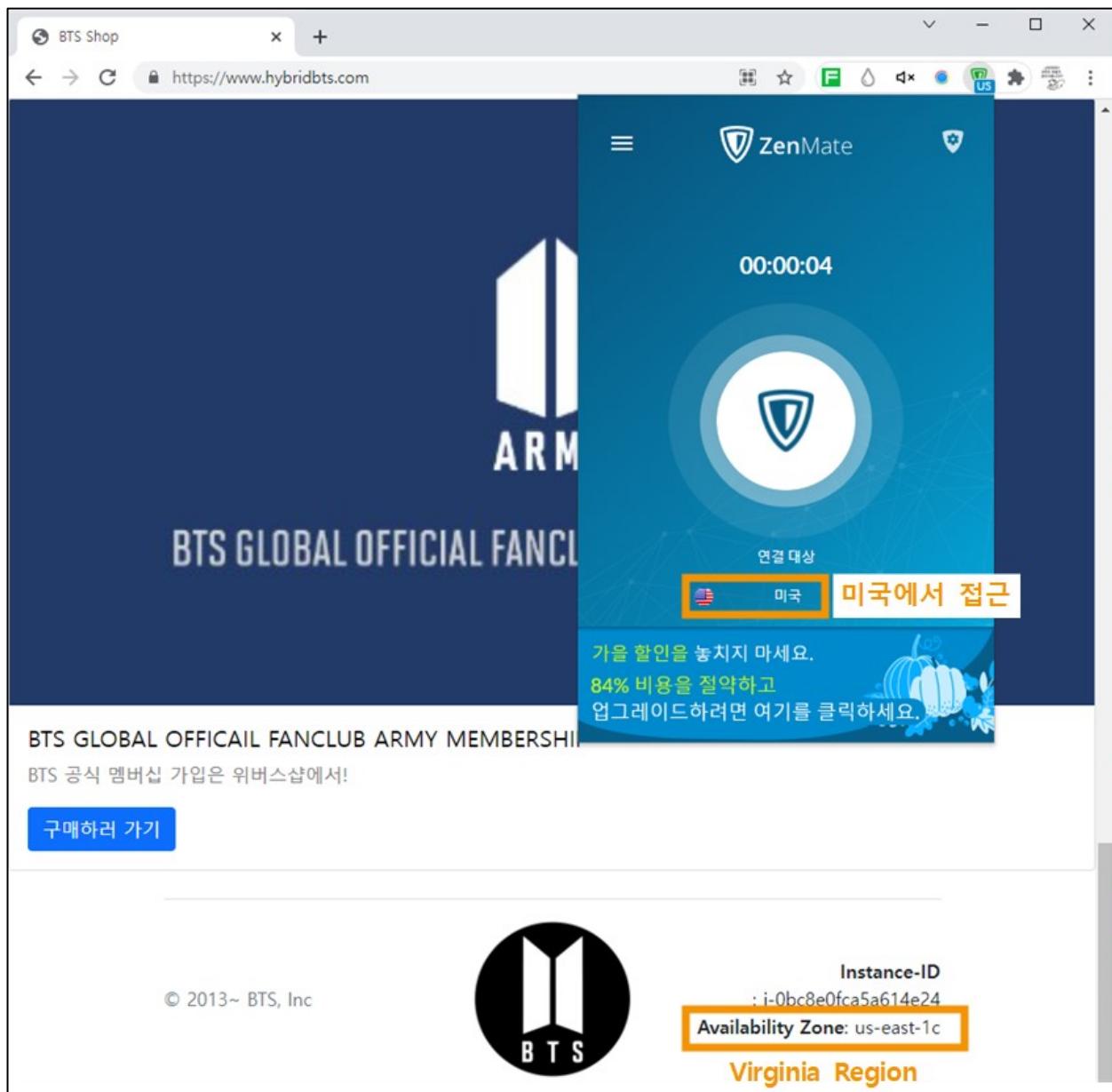
4. 작동 확인

AGA를 사용하여 웹사이트에 접속하는 사용자의 지역 위치에 따라 가까운 리전의 웹 서버로 라우팅 해주는지 테스트 한다.

서울에서 웹사이트에 접속한 경우 서울 리전으로 라우팅 된다.

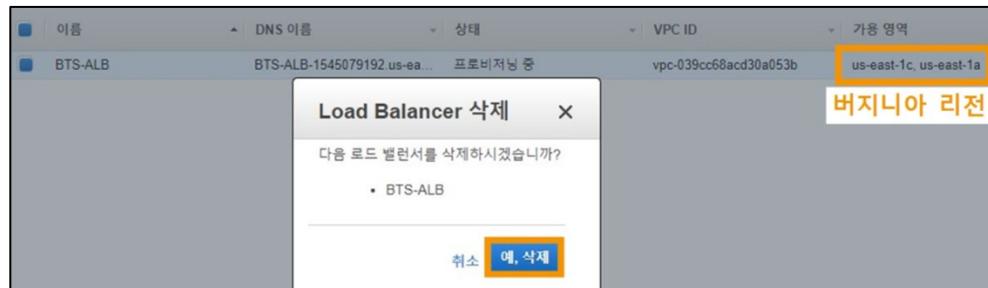


미국에서 접속한 경우 버지니아 리전으로 라우팅 된다.



다른 한쪽 리전의 서버에 장애가 생길 경우 다른 리전으로의 장애 조치 라우팅을 해주는지 테스트 한다.

버지니아 리전에 장애가 난 경우 Route 53 가 이상을 감지하고 버지니아 리전 ALB 를 삭제한다. 미국에서 접속해도 서울 리전으로 라우팅 된다.



이름	상태	설명
hybrid-Check	하루 전 2분 전 비정상	http://ip-10-0-0-46.us-west-2.compute.internal:80/index.php

버지니아 웹서버 접속 불량

BTS Shop

hybridbts.com

ZenMate

00:00:09

연결 대상

미국 미국으로 접근

가을 할인을 놓치지 마세요.
84% 비용을 절약하고
업그레이드하려면 여기를 클릭하세요.

BTS GLOBAL OFFICIAL FANCLUB ARMY MEMBERSHIP

BTS 공식 멤버십 가입은 위버스샵에서!

구매하러 가기

© 2013~ BTS, Inc

Instance-ID : i-098199826e5d021bf

Availability Zone : ap-northeast-2a

Seoul Region으로 우회

5.9. Amazon Route 53

정의

가용성이 높고 확장성이 뛰어난 AWS 의 DNS 서비스

기능

- 사용자의 요청을 EC2 인스턴스, Load Balancer, S3 버킷 등 AWS 에서 실행되는 인프라에 효과적으로 라우팅한다.
- 도메인 이름의 구매 및 등록 서비스를 제공한다.

장점

- 트래픽 흐름을 사용하여 지연 시간 기반, 지역 근접성, 가중치 기반 등 다양한 라우팅 유형을 통해 전역적으로 트래픽을 관리 할 수 있다.
- AWS 의 가용성이 높고 신뢰할 수 있는 인프라를 사용하여 개발하였다.
- 전 세계 DNS 서버의 글로벌 애니캐스트 네트워크를 사용하여 네트워크 상황에 따른 최적의 위치로 자동 라우팅을 한다.

HYBRID-BTS

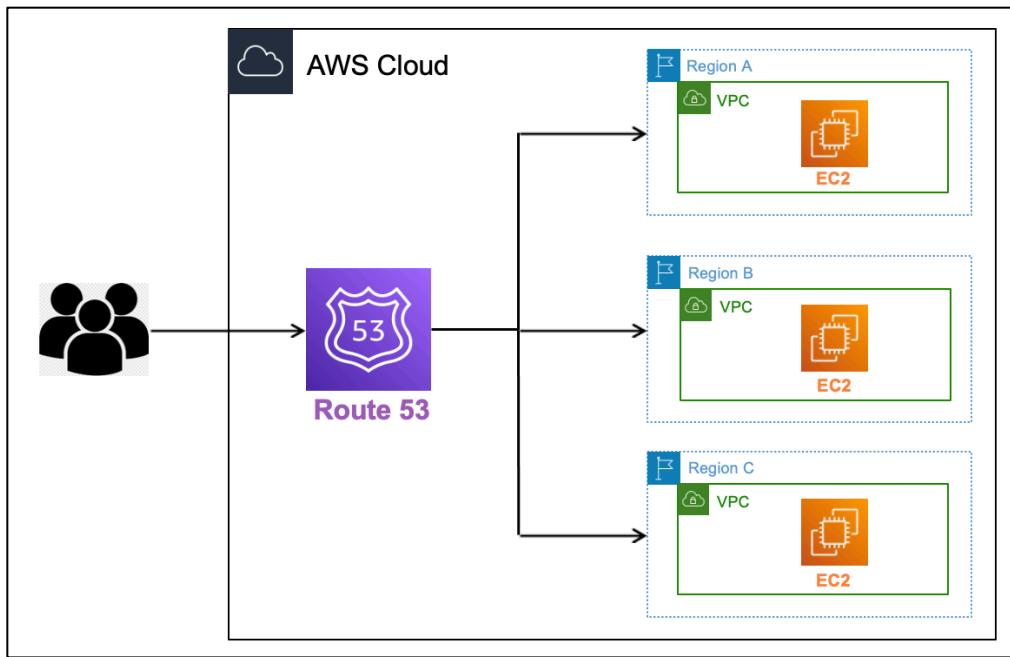


그림 20 HYBRID-BTS 의 Route 53 다이어그램

Route53에서 hybridbts.com 도메인을 구매하고 웹 서버의 도메인 이름으로 등록하였다. 호스트 이름 www를 비롯하여 kr, Apex 도메인, *. 도메일 대체 도메인으로 등록하였다.

구축 단계

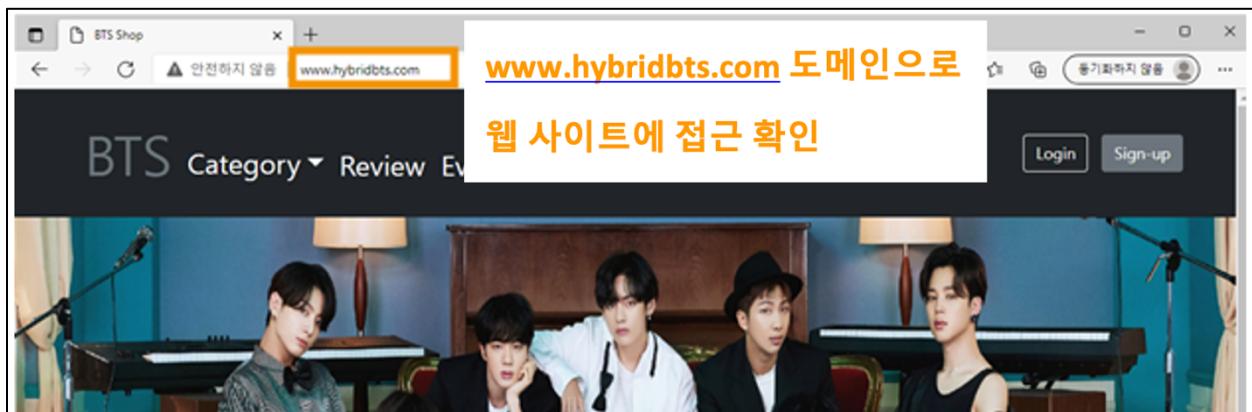
과정

1. 도메인 구매



2. A 레코드를 생성하여 Global Accelerator 에 연결

The screenshot shows the AWS Global Accelerator console interface. A new A record is being created for the domain `www.hybridbts.com`. The record type is set to `A - IPv4 주소 및 일부 AWS 리소스로 트래픽...`. The target is set to `Global Accelerator에 대한 별칭` (alias of the Global Accelerator). The geographical location is set to `미국 서부(오레곤)`. The status is `예` (Yes).



5.10. Amazon S3

5.10.1. Amazon S3

정의

Amazon에서 제공하는 업계 최고의 확장성, 데이터 가용성 및 보안성과 성능을 제공하는 객체 스토리지 서비스

기능

- 언제 어디서나 인터넷상에서 용량과 관계없이 데이터를 저장하고 검색하는 데 사용할 수 있는 웹 서비스 인터페이스를 제공한다.
- Amazon에서 글로벌 웹 사이트 네트워크를 운영할 때와 같은 동일한 수준의 높은 확장성, 안정성, 보안성 및 속도를 갖춘 비용 효율적인 인프라로 활용 가능하다.

장점

- 버킷에 저장할 수 있는 객체의 수에는 제한이 없고 객체 하나의 크기는 최대 5TB 까지 지원한다.
- 저장된 객체의 복사본을 자동으로 생성하고 저장하여 저장된 객체에 대해 99.999999999%의 내구성과 99.99%의 가용성을 제공하도록 설계되어 있다.
- HTTP/S 엔드 포인트를 사용하여 웹에서 언제 어디서든 원하는 데이터를 저장하고 검색할 수 있다.
- 초기 투자나 리소스 구매 과정 없이, 요구 사항에 따라 스토리지 리소스를 확장 및 축소할 수 있다.

Hybrid-BTS

EC2에서 사용하는 웹 소스파일을 S3 버킷에 저장하고 로드하여 웹 애플리케이션이 생성되게 했다. 웹사이트를 운영하는데 필요한 모든 정적 콘텐츠(비디오 파일 및 이미지 파일)를 S3 버킷에 연동하고, CloudFront를 사용하여 배포하였다.

5.10.2. AWS Elemental MediaConvert

정의

파일 기반 콘텐츠를 브로드캐스팅 및 멀티 스크린 송출에 적합하도록 변환하는 서비스

기능

- 비디오를 모든 유형의 전송용 브로드캐스트급으로 편집 가능하다.
- 선행 투자 비용없이 고품질의 완벽한 비디오 처리 워크플로우를 생성 할 수 있다.
- 유지 관리를 하지 않고 원하는 비디오만 처리 설정으로 작업이 가능하다.

장점

- 작업이 간단하며, 종량 과금제 방식으로 제공하여 추가 비용이 생기지 않는다.
- S3 와 CloudFront 를 사용하여 AWS 서비스 내에서 배포가 간편하다.
- 작업에 대한 워크로드, 복구, 모니터링 등의 요소들을 자동화한다.

HYBRID-BTS

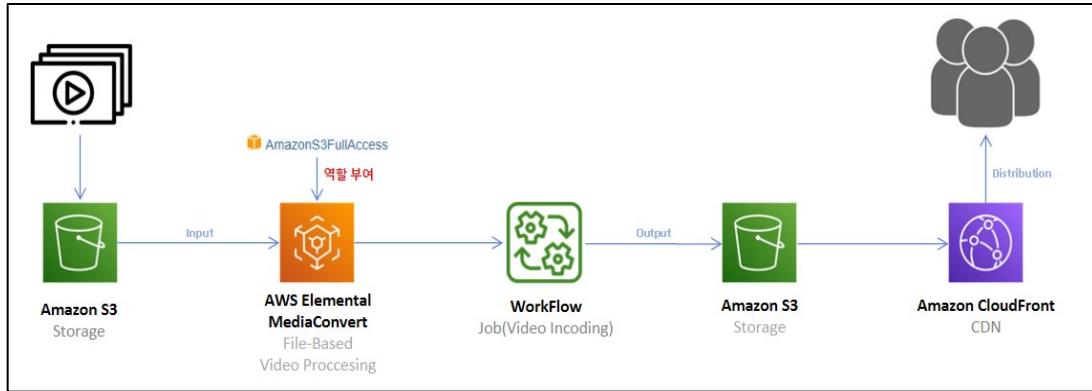


그림 21 HYBRID-BTS 의 MediaConvert 데이터그램

동영상 스트리밍 서비스를 이용하는 해외 사용자들을 위해서 비디오에 자막(영어)을 삽입하였다.

구축 단계

개요

- MediaConvert 에 IAM 역할 부여
- S3 에서 MediaConvert 로 작업할 동영상 파일 불러 오기
- 동영상 편집 후 S3 에 저장

과정

1. MediaConvert에 IAM 역할 부여

S3에 접근할 수 있는 역할을 부여한다.

서비스 액세스

리소스에 액세스하여 트랜스코딩 작업을 실행할 수 있는 권한을 MediaConvert에 부여합니다.

서비스 역할 제어 [Info](#)
이 작업에 기존 IAM 역할을 사용할지 아니면 새로 생성할지를 선택합니다.

새 서비스 역할 생성, 권한 구성

새 역할 이름 [Info](#)
계정에 아직 역할로 존재하지 않는 새 이름
MediaConvert_Default_Role **역할 이름 지정**

유효한 문자: A-Z, a-z, 0-9 및 + = , . @ _ -

입력 S3 위치 [Info](#)
MediaConvert에서 읽기 액세스를 허용하는 Amazon S3 위치를 지정합니다.

s3://btssbucket-123/ **동영상을 가져 올 S3 버킷 선택**

출력 S3 위치 [Info](#)
MediaConvert에서 쓰기 액세스를 허용하는 Amazon S3 위치를 지정합니다.

s3://btssbucket-123/ **변환한 동영상을 저장할 S3 버킷 선택**

API Gateway 엔드포인트 호출 [Info](#)
MediaConvert는 SPEKE DRM 및 Nielsen 비선형 워터마킹과 같은 다른 API와 통신하기 위해 API Gateway를 사용하는 기능에 대해 API Gateway 엔드포인트를 호출해야 합니다.

허용 안 함

2. S3에서 MediaConvert로 작업할 동영상 파일 불러 오기

입력 1 [Info](#)

입력 파일 URL [Info](#)

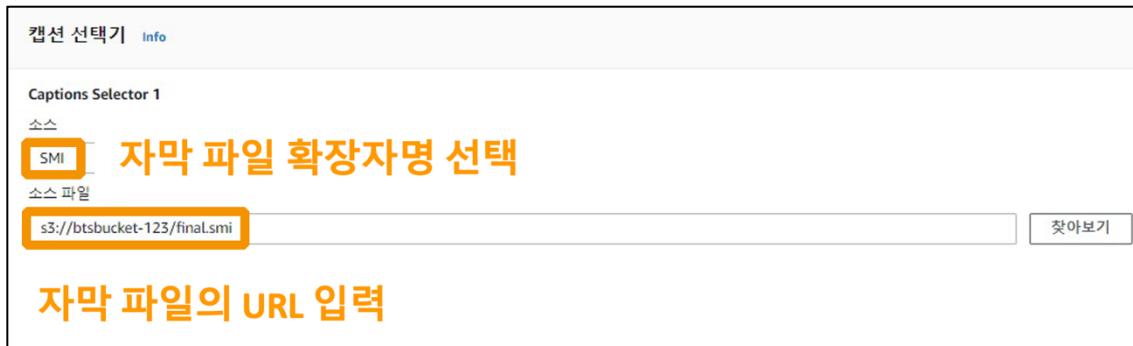
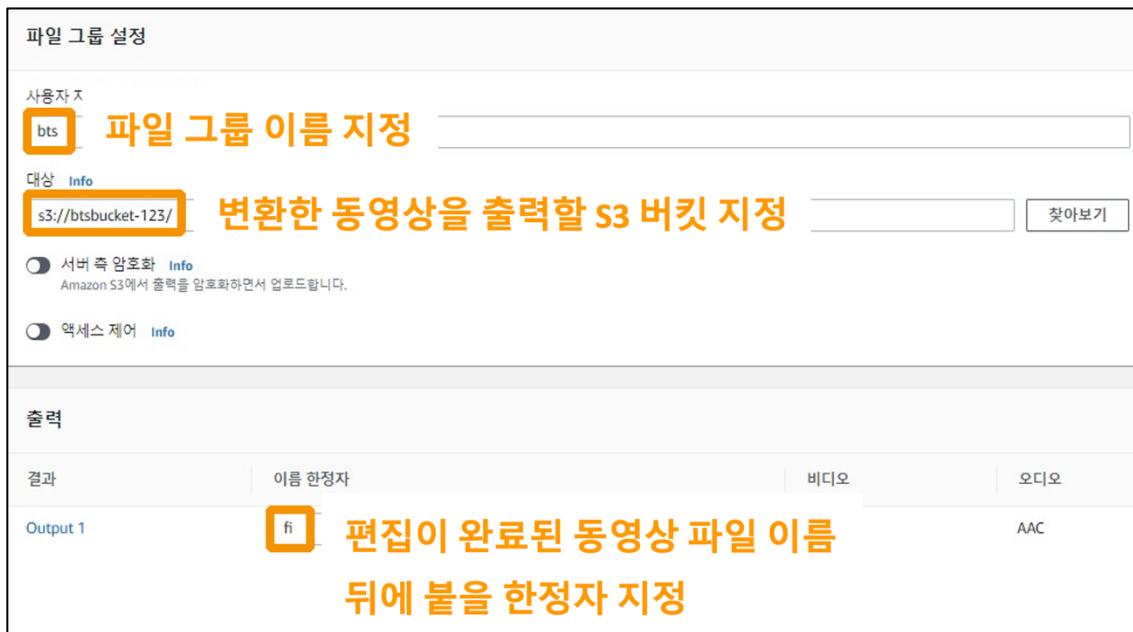
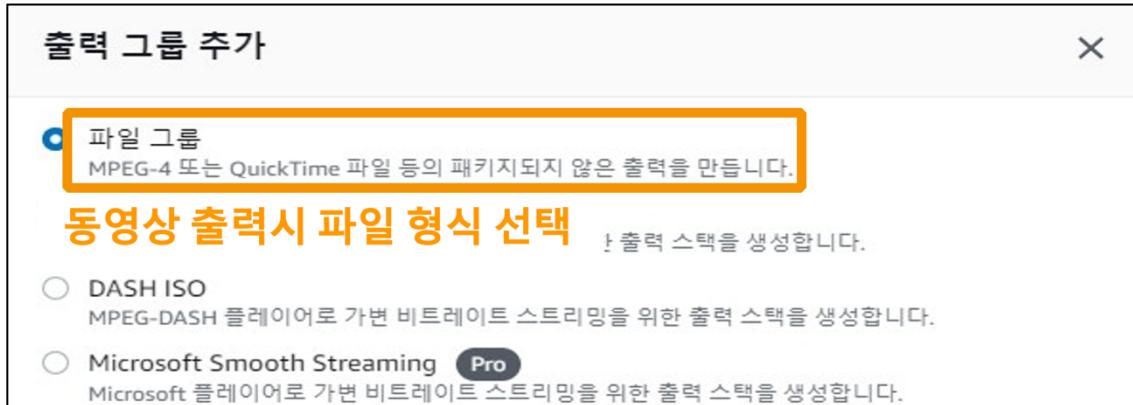
s3://btssbucket-123/inthesoop_short.mp4

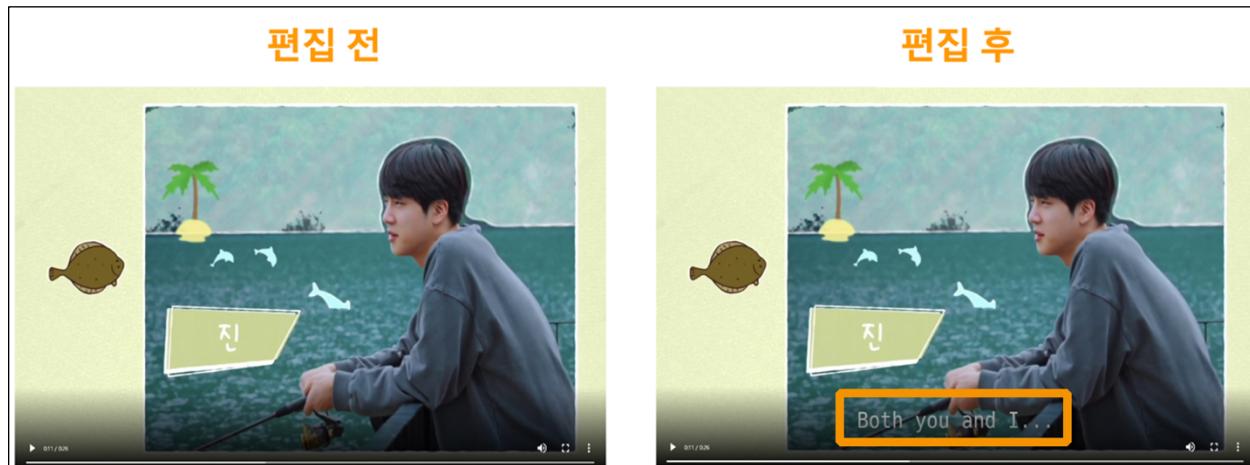
작업할 동영상 파일의 URL 입력

노동 IMI [Info](#)

3. 동영상 편집 후 S3 버킷에 저장

동영상에 자막을 삽입한 뒤 출력 대상 S3 버킷에 저장한다.





5.11. Amazon CloudFront

정의

정적/동적인 웹 콘텐츠를 사용자에게 빠른 속도로 배포하기 위한 서비스

작동 방식

사이트에 로딩되는 웹 콘텐츠를 엣지 로케이션이라는 AWS에서 운영하는 외부 캐시 서버에 저장한다. 사용자가 해당 콘텐츠에 접근하려고 하면 해당 사용자의 위치에 가장 빠르게 콘텐츠를 제공할 수 있는 위치의 엣지 로케이션에서 콘텐츠를 제공한다.

장점

서버가 위치한 리전과 물리적으로 멀리 떨어진 사용자도 자신과 가까운 엣지 로케이션에서 콘텐츠를 제공받으므로, 어디에서 접속하더라도 항상 빠르게 서비스를 제공받는 것은 물론 콘텐츠가 로케이션에 저장되는 것은 캐싱의 구조와 동일하기에, 캐싱을 통한 안정성 확보와 서버 부하 감소, 비용 부담 절감 역시 가능하다.

HYBRID-BTS

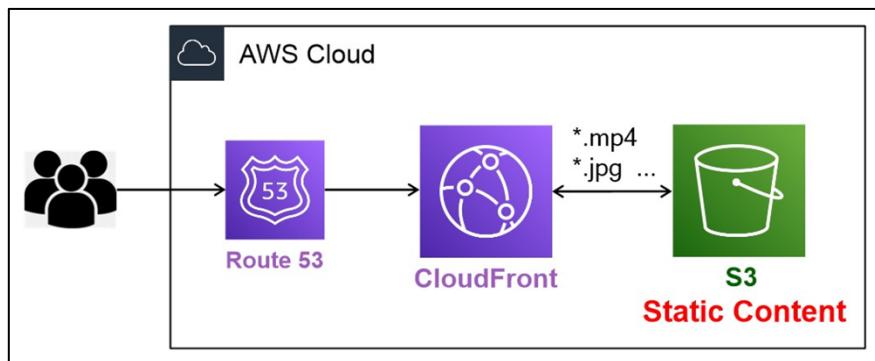


그림 22 HYBRID-BTS 의 CloudFront 디아그램

HYBRID-BTS 는 사용자가 전 세계에 고루 분포하여 있고, 용량이 큰 동영상 스트리밍 서비스를 제공한다. 때문에 기존 온프레미스 환경에서는 서버와 물리적으로 떨어진 사용자들은 낮은 퀄리티의 서비스를 경험해야 했다. 그러나 클라우드 이전을 통해서 CloudFront 를 S3 버킷에 연결하여 사용자들의 지리적 제한이 주는 문제를 해결하였다.

구축 단계

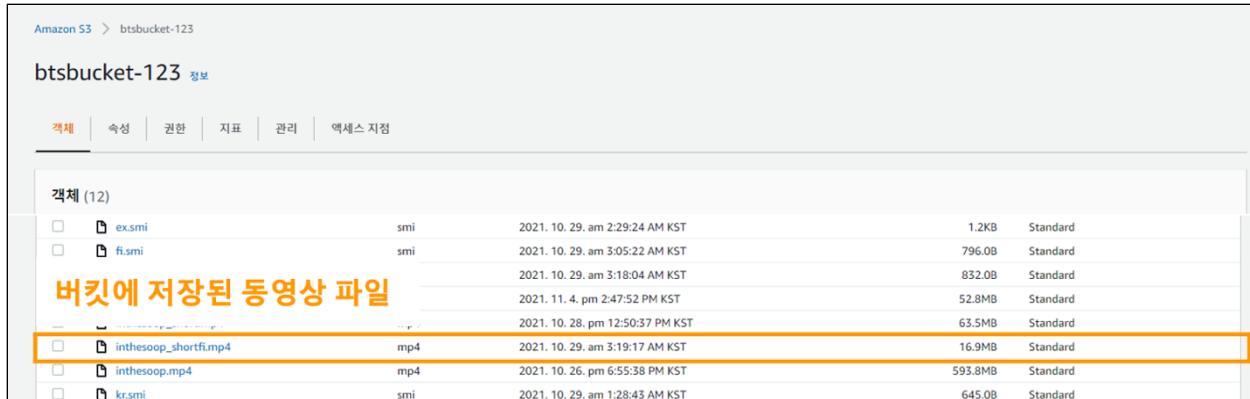
개요

1. S3 버킷에 동영상 파일 업로드
2. CloudFront 배포 생성
3. 웹 소스 코드 파일 경로 수정 후 버킷에 재업로드
4. 테스트

과정

1. 버킷에 동영상 파일 업로드

웹사이트에서 스트리밍 서비스로 제공할 동영상을 S3 버킷에 업로드 한다.

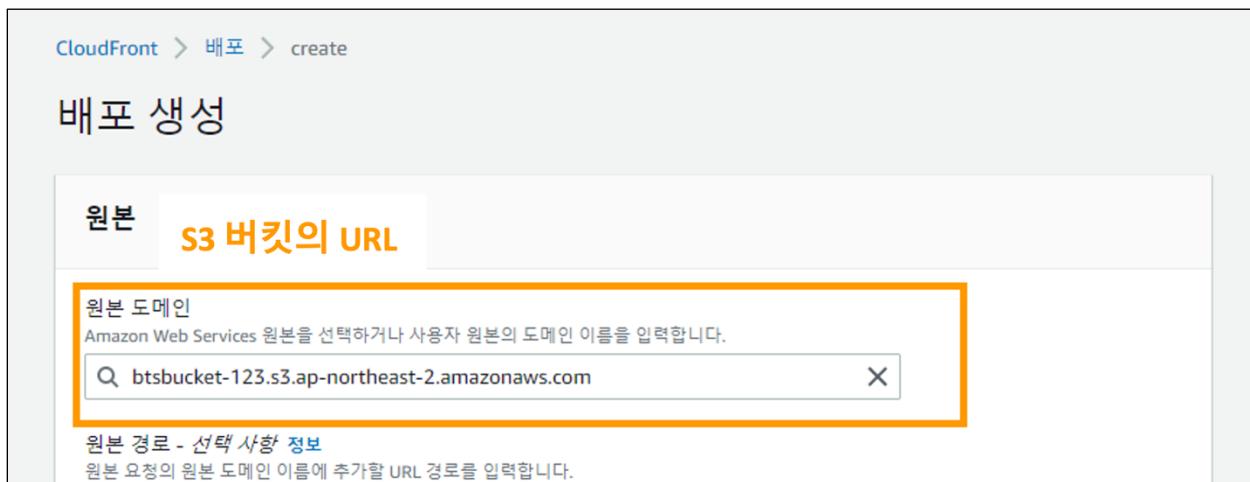


The screenshot shows the Amazon S3 console interface. In the top left, it says "Amazon S3 > btsbucket-123". Below that is the bucket name "btsbucket-123" with a "정보" (Information) link. A navigation bar at the top has tabs for "객체" (Objects), "속성" (Properties), "권한" (Permissions), "자료" (Data), "관리" (Management), and "액세스 지점" (Access Point). The main area is titled "객체 (12)". It lists several files: "ex.smi", "fi.smi", "kr.smi", "inthesoop_shortfli.mp4", "inthesoop.mp4", and "kr.smi". The file "inthesoop_shortfli.mp4" is highlighted with an orange border. Below the table, the text "버킷에 저장된 동영상 파일" (Videos stored in the bucket) is displayed.

파일명	타입	작성일	크기	Storage Class
ex.smi	smi	2021. 10. 29. am 2:29:24 AM KST	1.2KB	Standard
fi.smi	smi	2021. 10. 29. am 3:05:22 AM KST	796.0B	Standard
		2021. 10. 29. am 3:18:04 AM KST	832.0B	Standard
		2021. 11. 4. pm 2:47:52 PM KST	52.8MB	Standard
		2021. 10. 28. pm 12:50:37 PM KST	63.5MB	Standard
inthesoop_shortfli.mp4	mp4	2021. 10. 29. am 3:19:17 AM KST	16.9MB	Standard
inthesoop.mp4	mp4	2021. 10. 26. pm 6:55:38 PM KST	593.8MB	Standard
kr.smi	smi	2021. 10. 29. am 1:28:43 AM KST	645.0B	Standard

2. CloudFront 배포 생성

캐싱 할 정적 컨텐츠가 있는 S3 버킷을 대상으로 CloudFront 배포 생성한다.



The screenshot shows the CloudFront distribution creation process. It starts with a breadcrumb trail: "CloudFront > 배포 > create". The main title is "배포 생성". There are two tabs: "원본" (Origin) and "S3 버킷의 URL" (URL of the S3 bucket). The "S3 버킷의 URL" tab is active and highlighted with an orange border. Under the "원본" tab, there is a section for "원본 도메인" (Origin Domain) with a search input field containing "btsbucket-123.s3.ap-northeast-2.amazonaws.com". Below this, there is a note about "원본 경로 - 선택 사항" (Optional Origin Path) and a note about "원본 요청의 원본 도메인 이름에 추가할 URL 경로를 입력합니다." (Enter the URL path to add to the origin domain name in the original request).

3. 웹 소스 코드 파일 경로 수정 후 버킷에 재업로드

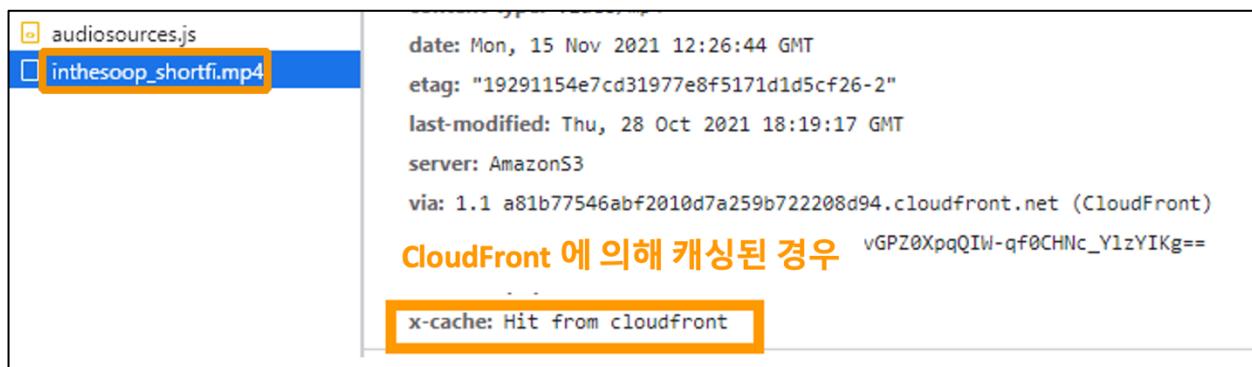
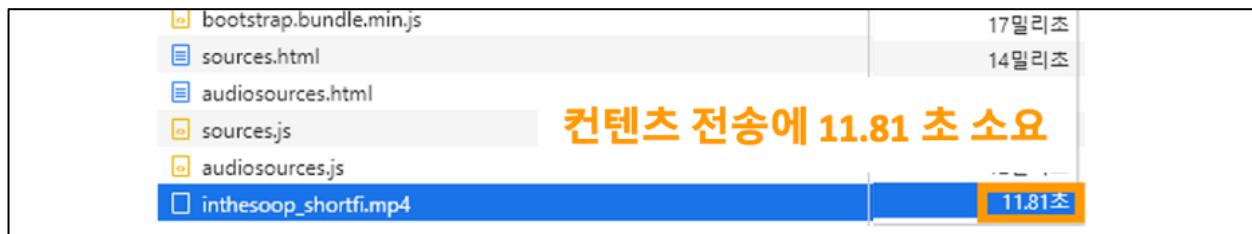
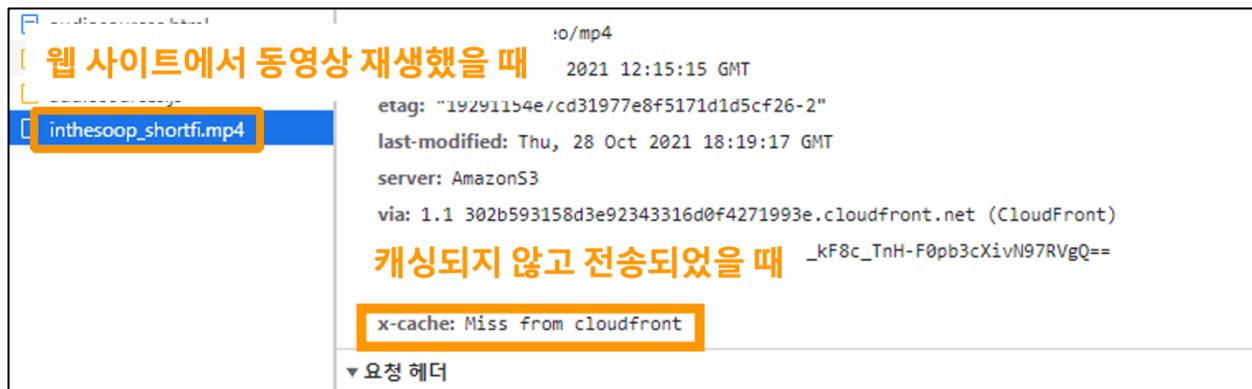
웹사이트의 소스 코드의 미디어 파일에서 동영상 섹션을 찾는다. 파일의 경로로 'CloudFront 배포 도메인/동영상 파일명'을 입력한다.

수정한 코드로 애플리케이션을 재배포하기 위해서, 웹 서버 인스턴스로부터 참조되고 있는 S3 버킷에 프로젝트 코드를 재업로드 한다. 이후 변경을 적용하기 위해 웹 서버 인스턴스 중지 후 재가동한다.



4. 테스트

정적 콘텐츠 전송에 CloudFront를 사용했을 때와 안 했을 때의 전송 속도 차이를 비교한다.



5.12. ACM(AWS Certificate Manager)

정의

AWS 의 SSL/TLS 인증서 관리 서비스. 인증서와 키를 만들고, 저장하고, 갱신하는 복잡한 과정을 단순화하고 비용의 부담을 완화한다. SSL/TLS 인증서는 네트워크 통신을 보호하고 인터넷상에서 웹 사이트의 자격 증명과 프라이빗 네트워크상에서 리소스의 자격 증명을 설정하는 데 사용된다. ELB, CloudFront, API Gateway 등의 서비스와 연결하여 쓰인다.

인증서 유형

인증서 발급 기관은 Public CA(Certificate Authority)와 Private CA 의 두 종류이다. Public CA 가 발급한 퍼블릭 인증서는 클라이언트와 웹서버 간에 통신할 때 사용된다. Private CA 가 발급한 프라이빗 인증서는 내부 서버와 애플리케이션, 서버, 기기, 서비스 간에 통신할 때 사용된다.

장점

- 인증서의 보안
- 요청, 배포, 관리가 빠르고 쉽다.

HYBRID-BTS

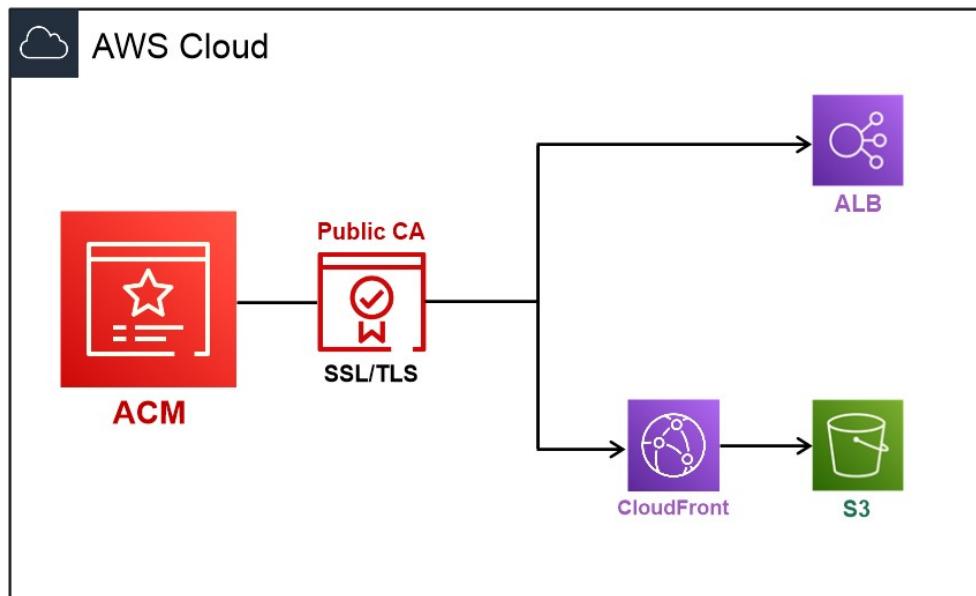


그림 23 HYBRID-BTS 의 ACM 다이어그램

구축 단계

개요

1. AWS 에 퍼블릭 인증서 요청
2. ALB 에 HTTPS 리스너 추가할 때 퍼블릭 인증서 사용

과정

1. AWS 에 퍼블릭 인증서 요청

인증서 요청

인증서 유형 정보

ACM 인증서는 인터넷 또는 내부 네트워크 내에서 안전한 통신 액세스를 설정하는 데 사용할 수 있습니다. acm이 제공할 인증서 유형은 퍼블릭 인증서와 프라이빗 인증서입니다.

요청할 인증서의 종류 선택

퍼블릭 인증서 요청 Amazon으로부터 퍼블릭 SSL/TLS 인증서를 요청합니다. 기본적으로 브라우저 및 운영 체제는 퍼블릭 인증서를 신뢰합니다.

프라이빗 인증서 요청 발급할 수 있는 프라이빗 CA가 없습니다.

프라이빗 인증서를 요청하려면 Private Certificate Authority(CA)를 생성해야 합니다. Private CA를 생성하는 방법은?

퍼블릭 인증서 요청

도메인 이름

완전히 정규화된 도메인 이름 정보

www.hybridbts.com	인증서가 보호할 FQDN 도메인 이름 등록
hybridbts.com	Apex 도메인 이름 등록
*.hybridbts.com	도메인 내 여러 사이트 보호를 위해 와일드 카드 도메인 이름 등록

이 인증서에 다른 이름 추가

이 인증서에 이름을 추가할 수 있습니다. 예를 들어, 'www.example.com'에 대한 인증서를 요청하는 경우 고객이 두 이름 중 하나를 선택할 수 있습니다.

AWS Certificate Manager > 인증서 > c4782b9e-0149-4278-a451-1f298cd9420f > Amazon Route 53에서 DNS 레코드 생성

Amazon Route 53에서 DNS 레코드 생성 (3/3)

도메인 검색		3 일치 항목			
검증 상태: 검증 대기 중	검증 상태: 실패	도메인이 Route 53에 있습니까?: 예	필터 지우기		
<input checked="" type="checkbox"/> 도메인	검증 상태	유형	CNAME 이름	CNAME 값	도메인이 Route 53에 있습니까?
<input checked="" type="checkbox"/> www.hybridbts.com	① 검증 대기 중	CNAME	도메인의 소유권/제어권 DNS 검증을 위해 Route53에서 DNS 레코드 생성	_9434028dba519 6c4ab7c0287911 b7e19.lggsghvb mf.acm- validations.aws.	예
<input checked="" type="checkbox"/> hybridbts.com	① 검증 대기 중	CNAME	_24f03602b724f07 779256ee4b993554 6.hybridbts.com.	_9434028dba519 6c4ab7c0287911 b7e19.lggsghvb mf.acm- validations.aws.	예
<input checked="" type="checkbox"/> *.hybridbts.com	① 검증 대기 중	CNAME	_24f03602b724f07 779256ee4b993554 6.hybridbts.com.	_9434028dba519 6c4ab7c0287911 b7e19.lggsghvb mf.acm- validations.aws.	예

2. ALB에 HTTPS 리스너 추가할 때 퍼블릭 인증서 사용

BTS-ALB | 리스너 추가 **HTTPS 접근 허용하기 위해 ALB에 리스너 추가**

Application Load Balancer에 속한 리스너는 구성한 프로토콜 및 포트를 사용하여 연결 요청을 확인합니다. 각 리스너는 모드와 라우팅 규칙을 만들고 관리할 수 있습니다. [자세히 알아보기](#)

프로토콜 : 포트
클라이언트에서 로드 밸런서로의 연결을 위한 프로토콜을 선택하고 트래픽을 수신할 포트 번호를 입력합니다.

HTTPS ▾ : 443 **프로토콜 HTTPS 과 포트 443 번 선택**

기본 작업
이 리스너가 다른 규칙에 의해 라우팅되지 않은 트래픽을 라우팅하는 방법을 나타냅니다. Only target groups with a compatible protocol will be listed.

+ 작업 추가 ▾

보안 정책
ELBSecurityPolicy-2016-08 ▾

기본 SSL 인증서
ACM (권장) ▾ **www.hybridbts.com - c4782b9e-0149-4278-a451-1f298cd9420f** ▾

생성한 Public CA 를 기본 SSL 인증서로 지정

기본 작업
이 리스너가 다른 규칙에 의해 라우팅되지 않은 트래픽을 라우팅하는 방법을 나타냅니다. Only target groups with a compatible protocol will be listed.

1. 전달 대상... 刪除

대상 그룹: 가중치(0-999)

BTS-tg **HTTPS 가 트래픽을 전달할 대상 그룹 지정** 1 ×

트래픽 배포 100%

5.13. AWS WAF(Web Application Firewall)

정의

웹 보안 위협에 대응하는 AWS의 클라우드 네이티브 웹 애플리케이션 방화벽. 웹 취약점 혹은 봇 공격을 필터링하고 차단한다.

작동 방식

WAF를 CloudFront, ALB, Amazon API Gateway 등에 적용한다. 콘솔, JSON 코드, 혹은 Managed Rules를 이용하여 정책을 생성한다. 생성된 정책을 바탕으로 위협이나 원하지 않는 트래픽을 차단하고 필터링할 수 있다. CloudWatch를 사용해서 유입되는 트래픽 메트릭을 모니터링하고, Kinesis Firehose로 웹 요청 상세 로그를 확인하여 WAF 정책을 조정한다.

AWS Managed Rules

웹 애플리케이션에 적용할 수 있도록 규정된 규칙의 모음. 알려진 공격과 위협 요소들을 차단한다. OWASP Top 10 Web Application Security Risks를 바탕으로 하였고, 보안 전문가들이 작성하였다.

- **Amazon IP reputation list** Amazon Threat Intelligence를 기반으로 작성된 규칙들을 포함한다.
- **Bot Control** 자원 고갈, 다운타임 초래, 악의적인 활동 등을 하는 봇들에게서 보호한다.
- **Core rule set** 웹 애플리케이션에 적용되는 일반적인 규칙들의 모음으로 넓은 범위의 취약점으로부터 보호한다. OWASP Top 10에 포함된 취약점들을 포함한다.
- **Known bad inputs** 취약점 공격으로 알려진 패턴을 포함한 리퀘스트를 차단한다. 애플리케이션의 취약점 노출로부터 보호한다.
- **PHP application** PHP function injection과 같은 PHP 사용의 취약점과 관련된 리퀘스트 패턴을 차단한다. 공격자가 원격으로 PHP 코드를 실행하는 것으로부터 보호한다.
- **SQL database** SQL injection attacks와 같은 SQL 데이터베이스의 취약점과 관련된 리퀘스트 패턴을 차단한다. 공격자가 원격으로 불법의 쿼리하는 것으로부터 보호한다.
- **WordPress application** WordPress에 특화된 취약점과 관련된 리퀘스트 패턴을 차단한다.

장점

- CloudFront, ALB 등의 AWS 의 서비스와 연동되어 배치와 유지보수가 쉽다.
- 정의된 Managed Rules 를 사용하면 방화벽 개발 비용을 절감할 수 있다.

HYBRID-BTS

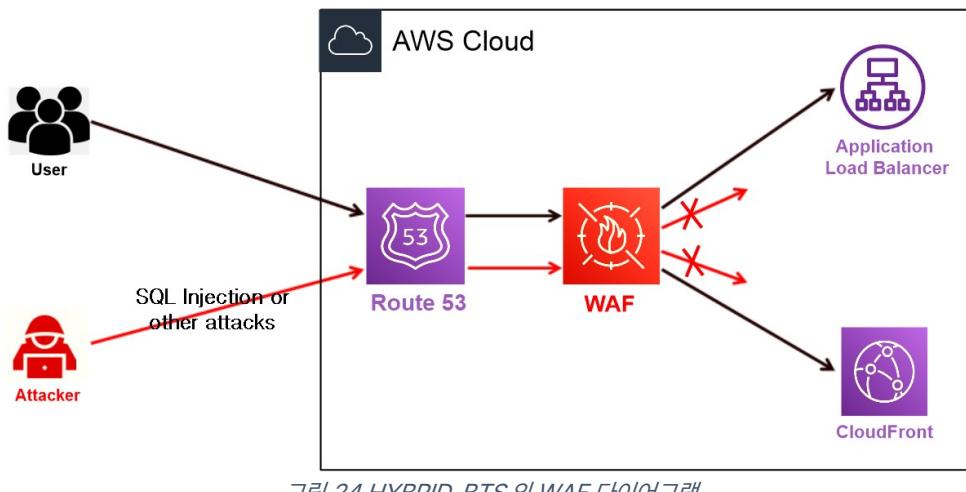


그림 24 HYBRID-BTS 의 WAF 디아이어그램

AWS Managed Rules 를 서울 리전과 버지니아 리전 웹 서버 대상 ALB 와, S3 버킷 대상 CloudFront 에 적용하였다.

구축 단계

개요

1. WebACL 생성
2. Managed Rules 추가
3. 작동 확인

과정

1. WebACL 생성

Web ACL details

Name **BTS-WebACL-Virginia-ALB** **WebACL 의 이름 지정**

The name must have 1-128 characters. Valid characters: A-Z, a-z, 0-9, - (hyphen), and _ (underscore).

Description - optional
Build firewall on Virginia web servers ALB.

The description can have 1-256 characters.

CloudWatch metric name **BTS-WebACL-Virginia-ALB** **동일한 메트릭 이름 지정**

The name must have 1-128 characters. Valid characters: A-Z, a-z, 0-9, - (hyphen), and _ (underscore).

Resource type
Choose the type of resource to associate with this Web ACL. **WebACL 을 적용할 리소스의 유형 선택**

CloudFront distributions

Regional resources (Application Load Balancer, API Gateway, AWS AppSync)

Region
Choose the AWS region to create this Web ACL in.
US East (N. Virginia) **WebACL 을 적용할 리전 선택**

2. Managed rules 추가

생성한 WebACL 에 AWS Managed rules 의 차단 규칙들을 추가한다.

Add rules and rule groups [Info](#)

A rule defines attack patterns to look for in web requests and the action to take when a request matches the patterns. Rule groups are reusable collections of rules. You can use managed rule groups offered by AWS and AWS Marketplace sellers. You can also write your own rules and use your own rule groups.

**WebACL 에 추가할 규칙 유형으로
managed rule groups 선택**

Rules	Action
If a request matches a rule, take:	
Edit	Delete
Add rules	
Add managed rule groups	
Add my own rules and rule groups	

No rules.
You don't have any rules added.

Rules			
If a request matches a rule, take the corresponding action. The rules are prioritized in order they appear.			
	Edit	Delete	Add rules ▾
<input type="checkbox"/>	생성중인 WebACL 에 포함될 규칙들	Capacity	Action
<input type="checkbox"/>	AWS-AWSManagedRulesAmazonIpReputationList	25	Use rule actions
<input type="checkbox"/>	AWS-AWSManagedRulesCommonRuleSet	700	Use rule actions
<input type="checkbox"/>	AWS-AWSManagedRulesKnownBadInputsRuleSet	200	Use rule actions
<input type="checkbox"/>	AWS-AWSManagedRulesPHPRuleSet	100	Use rule actions
<input type="checkbox"/>	AWS-AWSManagedRulesSQLiRuleSet	200	Use rule actions
<input type="checkbox"/>	AWS-AWSManagedRulesWordPressRuleSet	100	Use rule actions

Set rule priority Info			
Rules			
If a request matches a rule, take the corresponding action. The rules are prioritized in order they appear.			
<input type="radio"/>	Move up	Move down	패킷을 평가할 때 규칙 간의 우선순위 지정
			웹 앱의 위협 트래픽 패턴에 맞게 설정한다
Name		Capacity	Action
<input type="radio"/>	AWS-AWSManagedRulesAmazonIpReputationList	25	Use rule actions
<input type="radio"/>	AWS-AWSManagedRulesCommonRuleSet	700	Use rule actions
<input type="radio"/>	AWS-AWSManagedRulesKnownBadInputsRuleSet	200	Use rule actions
<input type="radio"/>	AWS-AWSManagedRulesPHPRuleSet	100	Use rule actions
<input type="radio"/>	AWS-AWSManagedRulesSQLiRuleSet	200	Use rule actions
<input type="radio"/>	AWS-AWSManagedRulesWordPressRuleSet	100	Use rule actions

Amazon CloudWatch metrics

CloudWatch metrics allow you to monitor web requests, web ACLs, and rules.

Rules	CloudWatch metric name
<input checked="" type="checkbox"/> AWS-AWSManagedRulesAmazonIpReputationList	BTS AMAZONIPREPUTATIONLIST
<input checked="" type="checkbox"/> AWS-AW 개별 규칙마다 CloudWatch 메트릭 이름 지정	BTS COMMONRULESET
<input checked="" type="checkbox"/> AWS-AWSManagedRulesKnownBadInputsRuleSet	BTS KNOWNBADINPUTSRULESET
<input checked="" type="checkbox"/> AWS-AWSManagedRulesPHPRuleSet	BTS PHPRULESET
<input checked="" type="checkbox"/> AWS-AWSManagedRulesSQLiRuleSet	BTS SQLIRULESET
<input checked="" type="checkbox"/> AWS-AWSManagedRulesWordPressRuleSet	BTS WORDPRESSRULESET

3. 작동 확인

CloudWatch에서 WAF의 로그를 확인하여 작동하는 것을 확인하고, 웹 서버에 negative/positive 공격을 시행해 false-positive(공격을 허용하는 것)나 false-negative(정상 트래픽을 차단하는 것)가 발생하지 않는 것을 확인한다.

CloudWatch에서 WAF의 로그를 모니터링 할 수 있다.

```
newmac:~ $ aws cloudwatch list-metrics --namespace "AWS/WAFV2"
aws-cli 사용하여 CloudWatch에서 WAF 로그 불러 오기
{
    "Metrics": [
        {
            "Namespace": "AWS/WAFV2",
            "MetricName": "AllowedRequests" 허용된 리퀘스트의 정보
            "Dimensions": [
                {
                    "Name": "WebACL",
                    "Value": "BTS-CloudFront-WebACL"
                },
                생성한 WebACL이 적용된 모습
                {
                    "Name": "Rule",
                    "Value": "ALL"
                }
            ]
        }
    ]
}
```

XSS, XXE, SQL injection 등 OWASP 보안 위험을 비롯한 취약점 공격이 모두 차단되는 것을 확인하였다.

웹 서버에 공격 테스트 결과					
Negative Tests					
TEST SET	TEST CASE	PERCENTAGE, %	BLOCKED	BYPASSED	
	FAILED				
community	community-xss	100.00	107	0	
304	197				
community	community-xxe	100.00	2	0	
2	0				
owasp	path-traversal	100.00	10	0	
66	54				
owasp	rce	100.00	3	0	
^o	^e				
owasp	sql-injection	100.00	10	0	
48	38				
owasp	ss-include	0.00	0	0	
20	20				
owasp	sst-injection	0.00	0	0	
32	32				
owasp	xml-injection	100.00	13	0	
13	0				
owasp	xss-scripting	100.00	20	0	
68	48				
owasp-api	graphql	0.00	0	0	
2	2				
owasp-api	graphql-post	100.00	1	0	
3	2				
owasp-api	grpc	0.00	0	0	
2	0				
owasp-api	rest	0.00	0	0	
2	2				
owasp-api	soap	100.00	2	0	

정상적인 접근시 False-negative 가 발생하지 않았다.

정상적인 접근 테스트 결과					
Positive Tests				BLOCKED	BYPASSED
TEST SET	TEST CASE	PERCENTAGE, %			
	FAILED				
false-pos	texts	0.00	0	0	0
17	17				
ENT:	DATE: 2021-11-10	WAF NAME: GENERIC	WAF POSITIVE SCORE: 0.00%	BLOCKED (RESOLVED): 0/0 (0.00%)	BYPASSED (RESOLVED): 0/0 (0.00%)
	17/17 (100.00%)				

5.14. Amazon EventBridge

정의

자체 애플리케이션 및 AWS 서비스에서 생성된 이벤트를 사용하여 이벤트 기반 애플리케이션을 구축할 수 있는 서비스 이벤트 버스

구축 단계

시나리오

Auto Scaling 으로 인해 최대 인스턴스 도달 시 관리자에게 SNS 발송 및 Bastion 인스턴스(관리자)를 자동 시작한다.

개요

1. 경고 SNS 를 받기 위해 SNS 생성 및 구독
2. Auto Scaling 시 트리거 될 Lambda 함수 생성
3. CloudWatch 를 통한 경보 생성
4. EventBridge 에 경보의 EventBridge 값 가져온 후 Lambda 함수 지정
5. 작동 확인

과정

1. 경고 SNS 를 받기 위해 SNS 생성 및 구독

SNS 주제를 생성한다.

The screenshot shows the 'Create Topic' wizard step 1. It has the following fields:

- Topic name:** admin_alert
- Type:** Standard (selected)
- Delivery methods:** SMS, Lambda, HTTP, SMS, Email, Mobile, AppSync Endpoints
- Delivery rules:** None
- Tags:** None
- ARN:** arn:aws:sns:ap-northeast-2:123456789012:admin_alert

생성한 주제에 대하여 구독을 생성한다.

★ AWS Notification – Subscription Confirmation

보낸 사람 **VIP** Auto Scaling alert <no-reply@sns.amazonaws.com>
 받는 사람 <choidh95@naver.com>

You have chosen to subscribe to the topic:
arn:aws:sns:ap-northeast-2:489851543453:admin_alert

To confirm this subscription, click or visit the link below (If this was in error no action is necessary):
[Confirm subscription](#)

Please do not reply directly to this email. If you wish to remove yourself from receiving all future SNS subscription confirmation emails, click [here](#).

구독 생성

세부 정보

주제 ARN
 arn:aws:sns:ap-northeast-2:489851543453:admin_alert **주제 설정**

프로토콜
 구독할 엔드포인트 유형
 이메일 **구독할 프로토콜 선택**

엔드포인트
 Amazon SNS의 알림을 수신할 수 있는 이메일 주소입니다.
 choidh95@naver.com **구독 할 이메일 입력**

ⓘ 구독을 생성한 후에는 확인해야 합니다. 정보

세부 정보	
ARN	상태  확인됨
arn:aws:sns:ap-northeast-2:489851543453:admin_alert:5b857441-e758-4a3d-b4a9-eb96a287d785	프로토콜 EMAIL
엔드포인트 choidh95@naver.com	주제 admin_alert

2. Auto Scaling 시 트리거 될 Lambda 함수 생성

Lambda 가 함수를 실행할 수 있도록 EC2 시작 및 중지 권한 부여.



```

1  {
2      "Version": "2012-10-17",
3      "Statement": [
4          {
5              "Effect": "Allow",
6              "Action": [
7                  "logs:CreateLogGroup",
8                  "logs:CreateLogStream",
9                  "logs:PutLogEvents"
10             ],
11             "Resource": "arn:aws:logs:*:*:*"
12         },
13         {
14             "Effect": "Allow",
15             "Action": [
16                 "ec2:Start*",
17                 "ec2:Stop*"
18             ],
19             "Resource": "*"
20         }
21     ]
22 }

```

EC2 권한 정책(시작 및 중지) 연결

Lambda 함수를 생성하고 역할을 부여한다.



함수 이름
함수의 용도를 설명하는 이름을 입력합니다.

EC2-start Lambda 함수 이름 입력

공백 없이 문자, 숫자, 하이픈 또는 밑줄만 사용합니다.

런타임 [Info](#)
함수를 작성하는데 사용할 언어를 선택합니다. 콘솔 코드 편집기는 Node.js, Python 및 Ruby

Python 3.8 Lambda 함수 작성할 언어 선택

▼ 기본 실행 역할 변경

실행 역할

함수에 대한 권한을 정의하는 역할을 선택합니다. 사용자 지정 역할을 생성하려면 [IAM 콘솔](#)로 이동합니다.

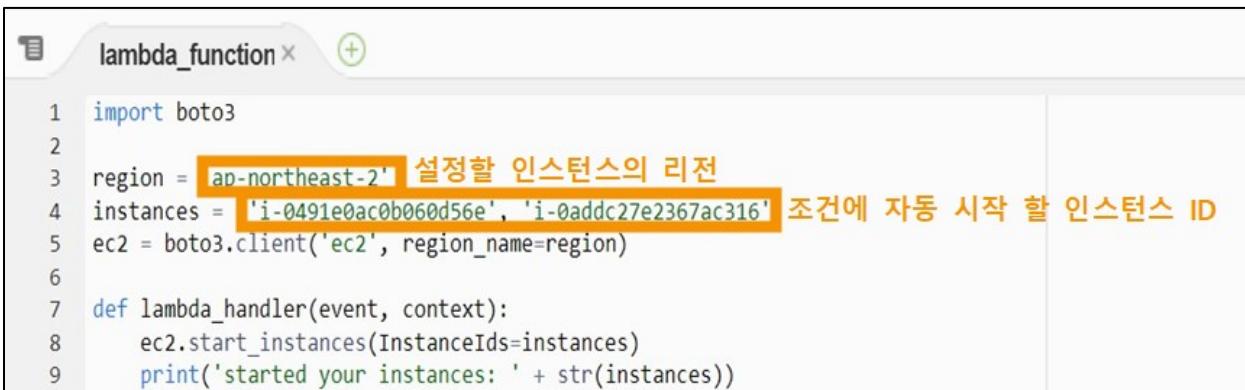
- 기본 Lambda 권한을 가진 새 역할 생성
- 기존 역할 사용
- AWS 정책 템플릿에서 새 역할 생성

기존 역할

생성한 기존 역할 중에 이 Lambda 함수와 함께 사용할 역할을 선택합니다. 이 역할에는 Amazon CloudWatch Log

Lambda-Role 사전에 생성한 EC2에 대한 권한 정책 선택

[IAM 콘솔](#)에서 Lambda-Role 역할을 확인합니다.

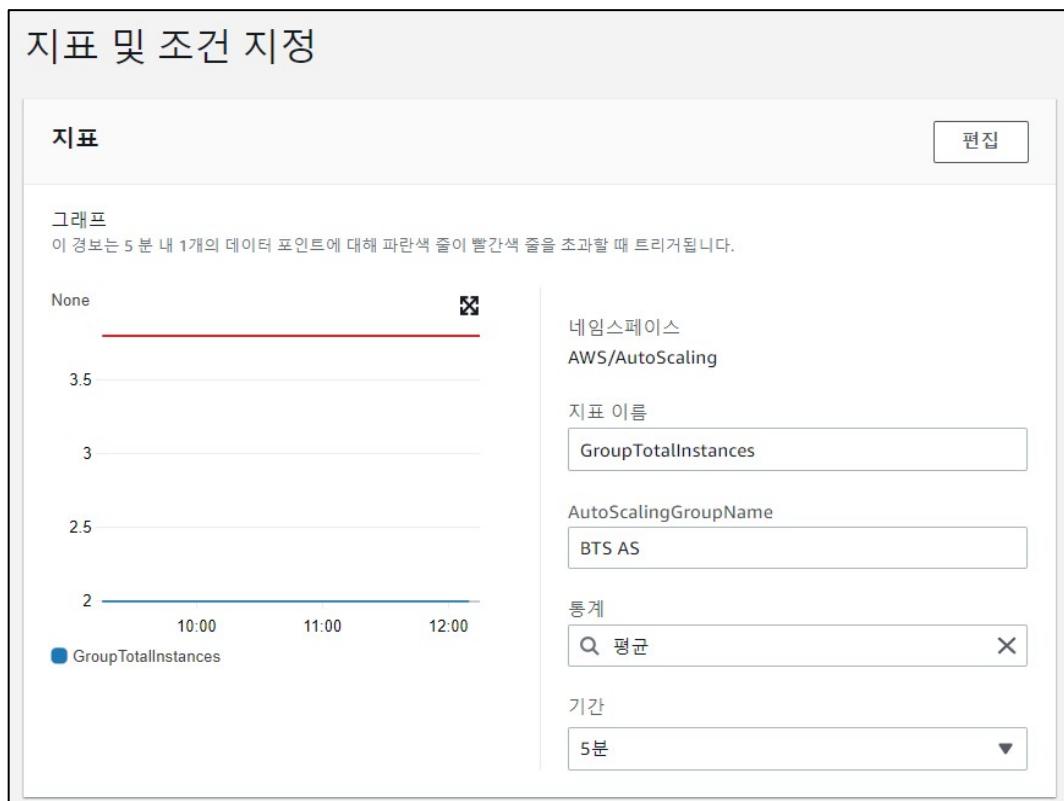


```

lambda_function x +
1 import boto3
2
3 region = 'ap-northeast-2' 설정할 인스턴스의 리전
4 instances = ['i-0491e0ac0b060d56e', 'i-0addc27e2367ac316'] 조건에 자동 시작 할 인스턴스 ID
5 ec2 = boto3.client('ec2', region_name=region)
6
7 def lambda_handler(event, context):
8     ec2.start_instances(InstanceIds=instances)
9     print('started your instances: ' + str(instances))

```

3. CloudWatch 를 통한 경보 생성



알림

경보 상태 트리거
이 작업을 트리거하는 경보 상태를 정의합니다.

경보 상태
지표 또는 표현식이 정의된 임계 값을 벗어났습니다.

정상
지표 또는 표현식이 정의된 임계 값 범위에 있습니다.

데이터 부족
경보가 방금 시작되었거나 사용 가능한 데이터가 부족합니다.

SNS 주제 선택
알림을 수신할 SNS(Simple Notification Service) 주제를 정의합니다.

기존 SNS 주제 선택

새 주제 생성

주제 ARN 사용

다음으로 알림 전송...

admin_alert **SNS 선택**

이 계정의 이메일 목록만 사용할 수 있습니다.

이메일(엔드포인트)
choidh95@naver.com - SNS 콘솔에서 보기

알림 추가

제거

CloudWatch에서 설정된 임계값 초과 시 SNS 발송 설정한다.

알림

경보 상태 트리거
이 작업을 트리거하는 경보 상태를 정의합니다.

경보 상태
지표 또는 표현식이 정의된 임계값을 벗어났습니다.

정상
지표 또는 표현식이 정의된 임계값 범위에 있습니다.

데이터 부족
경보가 방금 시작되었거나 사용 가능한 데이터가 부족합니다.

SNS 주제 선택
알림을 수신할 SNS(Simple Notification Service) 주제를 정의합니다.

기존 SNS 주제 선택

새 주제 생성

주제 ARN 사용

다음으로 알림 전송...

admin_alert **SNS 선택**

이 계정의 이메일 목록만 사용할 수 있습니다.

이메일(엔드포인트)
choidh95@naver.com - SNS 콘솔에서 보기

알림 추가

4. EventBridge에 CloudWatch에 있는 경보의 EventBridge 값 가져온 후, 해당 패턴의 이벤트 발생시 트리거 될 Lambda 함수 지정

CloudWatch에서 EventBridge 규칙을 확인한다.

▼ EventBridge 규칙 보기

경보 상태가 변경될 때 EventBridge를 사용하여 응답합니다. 이 사용자 정의 이벤트 패턴을 복사하여 이를 고급 이벤트 패턴의 시작점으로 사용할 수 있습니다. [정보](#)

```
{
  "source": [
    "aws.cloudwatch"
  ],
  "detail-type": [
    "CloudWatch Alarm State Change"
  ],
  "resources": [
    "arn:aws:cloudwatch:ap-northeast-2:489851543453:alarm:alert"
  ]
}
```

EventBridge 규칙을 생성한다.

규칙 생성

규칙은 특정 이벤트를 주시한 후 사용자가 선택한 AWS 대상으로 라우팅합니다. 다른 AWS 작업이 수행될 때 자동으로 AWS 작업을 수행하는 규칙 또는 설정된 일정에 따라 정기적으로 AWS 작업을 수행하는 규칙을 생성할 수 있습니다.

이름 및 설명

이름

AutoScaling-alert

대/소문자, 마침표(.), 대시(-), 밑줄(_)을 사용하여 최대 64자로 지정합니다.

설명 - 선택 사항

AutoScaling-alert

패턴 정의

이벤트 패턴을 작성 또는 사용자 지정하거나 대상을 호출할 일정을 설정합니다.

이벤트 패턴 정보

이벤트와 매칭할 패턴을 작성합니다

일정 정보

일정에 따라 대상 호출

이벤트 패턴과 매칭 시 규칙 작용

이벤트 매칭 패턴

서비스에서 제공하는 사전 정의된 패턴을 사용하거나 사용자 지정 패턴을 생성할 수 있습니다.

서비스에서 제공하는 사전 정의된 패턴

사용자 지정 패턴

저장

취소

이벤트 패턴

```

1 {
2   "source": [
3     "aws.cloudwatch"
4   ],
5   "detail-type": [
6     "CloudWatch Alarm State Change"
7   ],
8   "resources": [
9     "arn:aws:cloudwatch:ap-northeast-2:4898515434"
10  ]
11 }
12

```

CloudWatch에서 가져온 EventBridge 규칙

EventBridge에서 사용할 이벤트 패턴을 작성한다.

대상 선택

이벤트가 이벤트 패턴과 일치하거나 일정이 트리거될 때 호출할 대상을 선택합니다(규칙당 5개의 대상으로 제한).

대상 제거

이벤트가 이벤트 패턴과 일치하거나 일정이 트리거될 때 호출할 대상을 선택합니다(규칙당 5개의 대상으로 제한).

Lambda 함수

함수

이벤트 발생 시 트리거 될 함수 설정

- ▶ 버전/별칭 구성
- ▶ 입력 구성
- ▶ 재시도 정책 및 배달 못한 편지 대기열

대상 추가

규칙 (2/2)

규칙 찾기 모든 상태 ▾

이름	상태	유형
AutoScaling-alert	<input checked="" type="checkbox"/> Enabled	표준

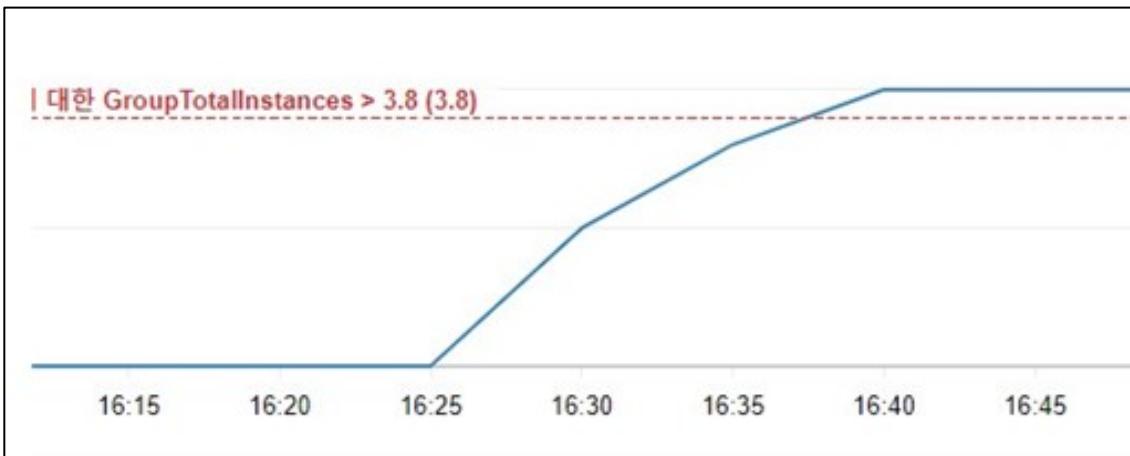
5. 작동 확인

이벤트 소스인 EC2 인스턴스를 Auto Scaling으로 확장시킨다. EventBridge가 작동하여 함수가 트리거 되고, 관리자에게 SNS 알람이 가고 Bastion 인스턴스가 자동 시작이 되는지 테스트한다.

테스트하기 위해 Auto Scaling 된 인스턴스를 과부화 시킨다.

```
[ec2-user@ip-10-0-3-41 ~]$ stress --cpu 1
stress: info: [30848] dispatching hogs: 1 cpu, 0 io, 0 vm, 0 hdd
```

EventBridge에서 설정한 이벤트에 도달하였다.



이벤트가 발생하여 SNS 알람을 수신하였다.

★ ALARM: "alert" in Asia Pacific (Seoul) [\[보기\]](#)

보낸 사람 **VIP** Auto Scaling alert-no-reply@sns.amazonaws.com>
받는 사람 <choidh95@naver.com>

You are receiving this email because your Amazon CloudWatch Alarm "alert" in the Asia Pacific (Seoul) region has entered datapoint for OK → ALARM transition." at "Tuesday 09 November, 2021 16:42:48 UTC".

View this alarm in the AWS Management Console:
<https://ap-northeast-2.console.aws.amazon.com/cloudwatch/deeplink.js?region=ap-northeast-2#alarmsV2:alarm/alert>

Alarm Details:

- Name: alert
- Description: alert
- State Change: OK → ALARM
- Reason for State Change: Threshold Crossed: 1 out of the last 1 datapoints [4.0 (09/11/21 16:37:00)] was greater than
- Timestamp: Tuesday 09 November, 2021 16:42:48 UTC
- AWS Account: 489851543453
- Alarm Arn: arn:aws:cloudwatch:ap-northeast-2:489851543453:alarm:alert

Threshold:

- The alarm is in the ALARM state when the metric is GreaterThanThreshold 3.8 for 300 seconds.

이벤트가 발생하여 Lambda 함수가 트리거 되었다.

Name	인스턴스 ID	인스턴스 상태	인스턴스 유형	상태 검사
BTS_Bastion 1	i-0491e0ac0b060d56e			
GoldenInstanc...	i-04c9627c7581e63bd			
BTS_Bastion 2	i-0addc27e2367ac316			

이벤트 매칭으로 Lambda 함수가 트리거 되어 Bastion 인스턴스가 켜진 모습

6. 결론

고객사 'HYBRID-BTS'는 AWS 클라우드 환경 이전을 통해 **비용 절감, 운영 우수성, 직원 생산성, 향상된 보안**을 달성하였다.

비용 절감

AWS 클라우드의 버지니아 리전에 전체 인프라를 서울 리전과 동일한 조건으로 신설하여 재해 복구 시스템을 마련하였다. 그러나 초기 투입되는 자본 비용(CapEx)이 없었고, 사용자의 요청에 따라 오토 스케일링으로 자원의 양을 조절하기 때문에 마치 한 리전을 운영하는 것과 같은 비용이 지출되고 있다. 이와 같이 확장에 대한 부담을 제거해주는 클라우드의 장점 덕분에, 한치 앞을 알 수 없는 엔터테인먼트 비지니스의 특성에도 불구하고 방탄소년단이 인기의 최정상 기도를 달리고 있는 현재 시점의 요구를 모두 충족하며 HYBRID-BTS는 비지니스가 기하급수적 성장하는 모멘텀(momentum)을 얻을 수 있었다.

운영 우수성

기존에는 각 서버마다 하나의 웹 애플리케이션을 운영하고 있었다. 그러나 멀티 리전이 주는 탄탄한 장애 대비 복구 시스템과, AWS 가 각 서버의 Health Check 를 통해 비정상 서버는 자동으로 라우팅 대상에서 제외되는 기능을 바탕으로 버지니아 리전에 컨테이너라는 새로운 기술을 실험 도입할 수 있었다.

이에 더해 IT 인프라의 모든 요소를 AWS 내의 연동된 서비스로 제공하고 있는 이점을 활용하였다. 네트워크나 각종 서버를 비롯해 방화벽, CDN, 관리자 권한 분배, 인증서, IaC 등 관리해야 하는 모든 서비스를 AWS 환경 내에 집적시킴으로써 운영의 효율성을 높였다. 기본 인프라 외에도 AWS 가 제공하는 다양한 서비스들을 브라우징하고 도입 비용의 부담 없이 실험해 봄으로써, Elemental MediaConvert, Lambda, SNS 등 새로운 서비스를 도입하고 비지니스가 진화하는 계기가 되었다.

직원 생산성

IT 부서의 직원들은 기존 손수 네트워크와 각종 서버 인프라를 구축, 관리, 업데이트하던 수동적인 방식에서 벗어났다. 기존 직원들에게 요구되던 작업의 상당수가 AWS 의 책임으로 넘어 가고 자동화 되면서, 이제는 테크니컬한 디테일을 일일이 설정하기보다는 요구되는 작업을 빠르게 명시하는 방식으로 업무 방식이 전환되었다. 그렇기 때문에 애플리케이션이 제공하는 서비스 저변에서 소요되던 시간을 대폭 줄이고, 사용자가 직접적으로 경험하는 서비스를 향상시키는데 소중한 인력 자원을 집중시킬 수 있었다.

향상된 보안

HYBRID-BTS 가 제공하는 대표적인 상품은 독점 제작한 방탄소년단의 동영상이다. 다큐멘터리, 영화, 브이로그, 콘서트 실황중계 등 다양한 미디어를 전세계 수 백만명의 고객들에 판매한다. 이와 같은 특성 때문에 콘텐츠의 보안이 다른 미디어 스트리밍 서비스 기업만큼이나 중요하다. 기존 온프레미스 방식에서는 보안에 있어 규모의 경제를 실현하기가 어려워 불완전했다. 하지만 AWS 로 이주하면서 AWS 의 탄탄한 인프라를 적용 받으면서 보안을 향상시킬 수 있었다.