

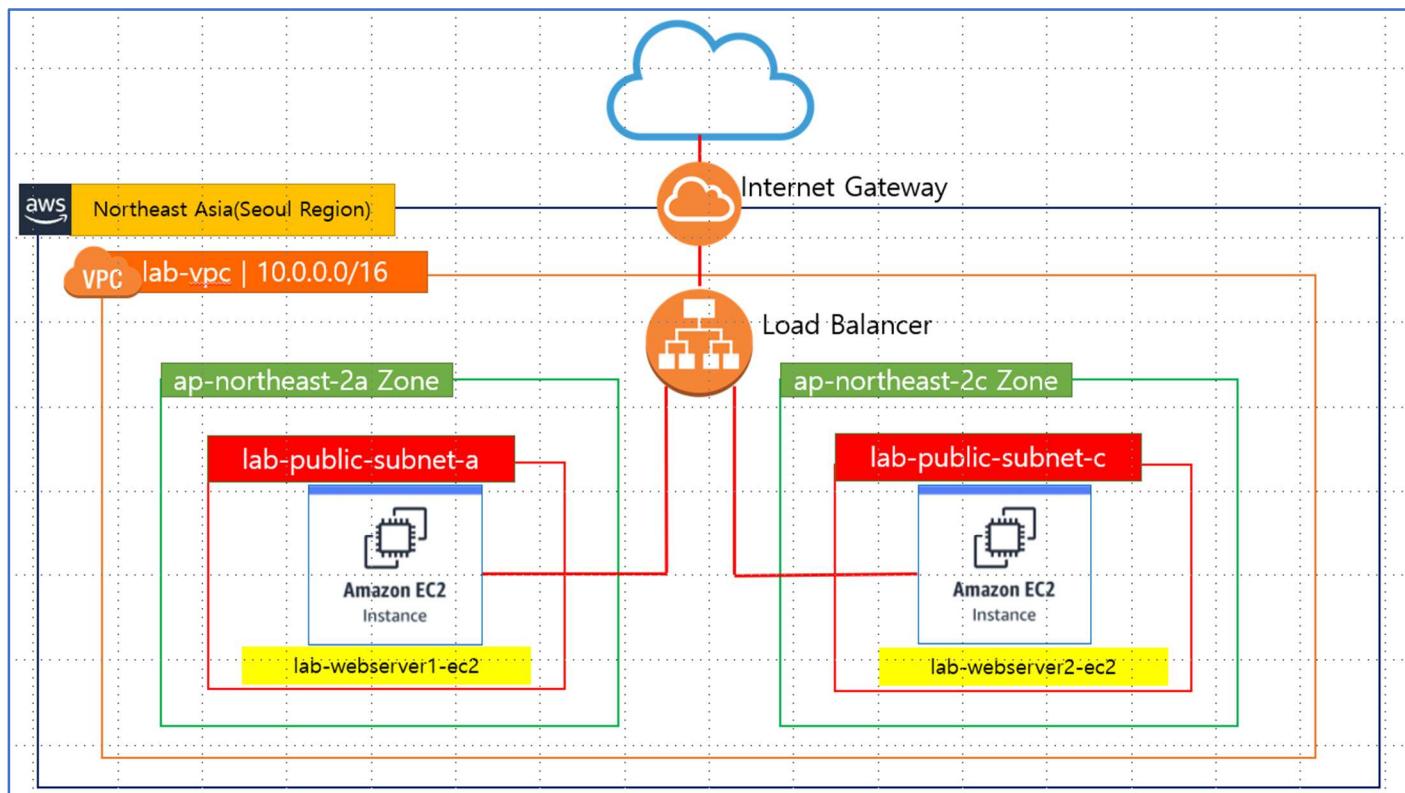
Lab. Create and Deploy Application Load Balancer

목적

Elastic Load Balancing은 단일 가용 영역 또는 여러 가용 영역에서 다양한 애플리케이션 부하를 처리할 수 있다. 이 학습은 AWS Free-Tier를 활용하여 진행한다. 서비스 중인 서버 한 대가 중지되었을 때 Application Load Balancing 서비스가 어떻게 동작하는지 확인한다. 마지막부분에서는 웹 사이트의 로그 인을 유지하기 위한 Sticky Session 활성화 방법을 알아보기로 한다.

사전 준비물

AWS Free-Tier 계정



네트워크 환경 구성하기

1. 다음 그림과 같이 VPC를 생성한다.
 - A. [생성할 리소스] : VPC 등
 - B. [이름 태그 자동 생성] : lab
 - C. [IPv4 CIDR] : 10.0.0.0/16
 - D. [IPv6 CIDR 블록] : IPv6 CIDR 블록 없음
 - E. [태넌시] : 기본값

VPC 설정

생성할 리소스 정보
VPC 리소스 또는 VPC 및 기타 네트워킹 리소스만 생성합니다.

VPC만 VPC 등

이름 태그 자동 생성 정보
이름 태그의 값을 입력합니다. 이 값은 VPC의 모든 리소스에 대한 이름 태그를 자동으로 생성하는 데 사용됩니다.

자동 생성
lab

IPv4 CIDR 블록 정보
CIDR 표기법을 사용하여 VPC의 시작 IP와 크기를 결정합니다.

10.0.0.0/16 65,536 IPs

CIDR 블록 크기는 /16에서 /28 사이여야 합니다.

IPv6 CIDR 블록 정보

IPv6 CIDR 블록 없음 Amazon 제공 IPv6 CIDR 블록

테넌시 정보

기본값 ▾

- F. [가용 영역(AZ) 수] : 1
 - i. [첫 번째 가용 영역] : ap-northeast-2a
 - ii. [두 번째 가용 영역] : ap-northeast-2c
- G. [퍼블릭 서브넷 수] : 2
- H. [프라이빗 서브넷 수] : 2
 - I. [ap-northeast-2a 퍼블릭 서브넷 CIDR 블록] : 10.0.10.0/24
 - J. [ap-northeast-2c 퍼블릭 서브넷 CIDR 블록] : 10.0.20.0/24
 - K. [ap-northeast-2a 프라이빗 서브넷 CIDR 블록] : 10.0.30.0/24
 - L. [ap-northeast-2c 프라이빗 서브넷 CIDR 블록] : 10.0.40.0/24

가용 영역(AZ) 수 정보
서브넷을 프로비저닝할 AZ 수를 선택합니다. 고가용성을 위해서는 최소 2개 이상의 AZ를 사용하는 것이 좋습니다.

1 2 3

▼ AZ 사용자 지정

첫 번째 가용 영역

ap-northeast-2a ▾

두 번째 가용 영역

ap-northeast-2c ▾

퍼블릭 서브넷 수 정보

VPC에 추가할 퍼블릭 서브넷 수입니다. 인터넷을 통해 공개적으로 액세스할 수 있어야 하는 웹 애플리케이션에는 퍼블릭 서브넷을 사용합니다.

0 2

프라이빗 서브넷 수 정보

VPC에 추가할 프라이빗 서브넷 수입니다. 프라이빗 서브넷을 사용하여 퍼블릭 액세스가 필요 없는 백엔드 리소스를 보호합니다.

0 2 4

▼ 서브넷 CIDR 블록 사용자 지정

ap-northeast-2a 퍼블릭 서브넷 CIDR 블록

10.0.10.0/24 256 IPs

ap-northeast-2c 퍼블릭 서브넷 CIDR 블록

10.0.20.0/24 256 IPs

ap-northeast-2a 프라이빗 서브넷 CIDR 블록

10.0.30.0/24 256 IPs

ap-northeast-2c 프라이빗 서브넷 CIDR 블록

10.0.40.0/24 256 IPs

M. [NAT 게이트웨이] : AZ당 1개

N. [VPC 엔드포인트] : 없음

O. [DNS 호스트 이름 활성화] 체크

P. [DNS 확인 활성화] 체크

NAT 게이트웨이(\$) 정보

NAT 게이트웨이를 생성할 가용 영역(AZ) 수를 선택합니다. 각 NAT 게이트웨이마다 요금이 부과됩니다.

없음 1개의 AZ에서 AZ당 1개

VPC 엔드포인트 정보

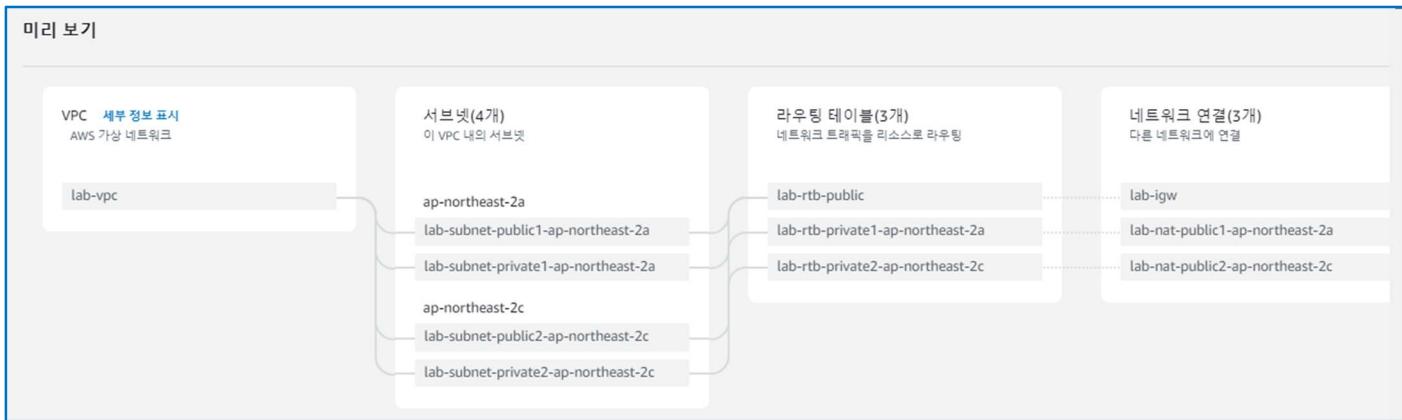
엔드포인트는 VPC에서 S3에 직접 액세스하여 NAT 게이트웨이 요금을 줄이고 보안을 강화할 수 있습니다. 기본적으로 모든 액세스 정책이 사용됩니다. 언제든지 이 정책을 사용자 지정할 수 있습니다.

없음 S3 게이트웨이

DNS 옵션 정보

- DNS 호스트 이름 활성화
- DNS 확인 활성화

2. 지금 생성하고 있는 VPC는 다음과 같다.



3. [VPC 생성] 버튼 클릭한다.

VPC 워크플로 생성

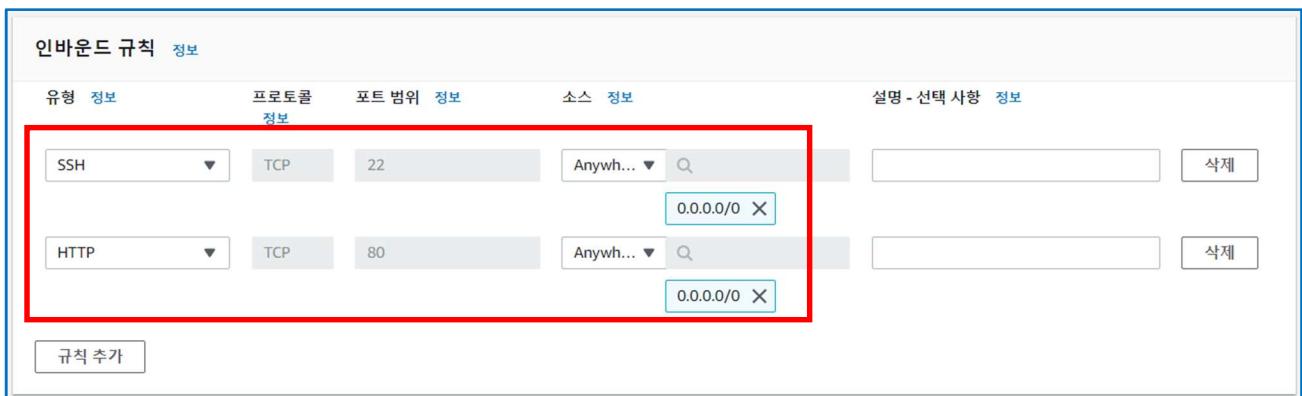
성공

▼ 세부 정보

- ✓ VPC 생성: [vpc-02ff6d53ce26bb1c6](#)
- ✓ DNS 호스트 이름 활성화
- ✓ DNS 확인 활성화
- ✓ VPC 생성 확인: [vpc-02ff6d53ce26bb1c6](#)
- ✓ 서브넷 생성: [subnet-00741cb2ca85f0f48](#)
- ✓ 서브넷 생성: [subnet-0416a1c4941baa1a3](#)
- ✓ 서브넷 생성: [subnet-0f00080b35d16fa5c](#)
- ✓ 서브넷 생성: [subnet-0d767c045b043f26d](#)
- ✓ 인터넷 게이트웨이 생성: [igw-015a77a5892874e3f](#)
- ✓ VPC에 인터넷 게이트웨이 연결
- ✓ 라우팅 테이블 생성: [rtb-00725fbc452ba75ad](#)
- ✓ 경로 생성
- ✓ 라우팅 테이블 연결
- ✓ 라우팅 테이블 연결
- ✓ 탄력적 IP 할당: [eipalloc-0d5708053f3f9e851](#)
- ✓ 탄력적 IP 할당: [eipalloc-002b4c2d104113963](#)
- ✓ NAT 게이트웨이 생성: [nat-0b2014a948182c40b](#)
- ✓ NAT 게이트웨이 생성: [nat-01e3036ea2aa9ef6b](#)
- ✓ NAT 게이트웨이가 활성화될 때까지 대기
- ✓ 라우팅 테이블 생성: [rtb-067b066025de82aaa](#)
- ✓ 경로 생성
- ✓ 라우팅 테이블 연결
- ✓ 라우팅 테이블 생성: [rtb-01dd83e0b514f48a6](#)
- ✓ 경로 생성
- ✓ 라우팅 테이블 연결
- ✓ 라우팅 테이블 생성 확인

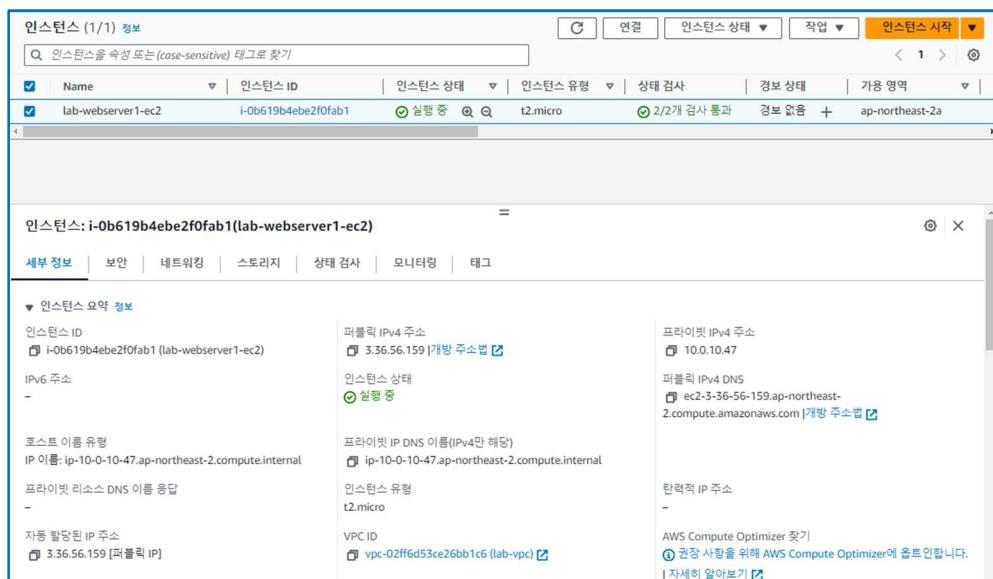
VPC 보기

4. 새 보안 그룹을 생성한다. VPC 페이지 좌측 메뉴에서 [보안] > [보안 그룹] > [보안 그룹 생성] 버튼 클릭
 5. [보안 그룹 생성] 페이지에서
 - A. [보안 그룹 이름] : lab-sg
 - B. [설명] : Security group for Application Load Balancer
 - C. [VPC] : lab-vpc
6. 다음 그림과 같이 [규칙 추가]를 클릭하여 SSH와 HTTP를 추가한다. 각각 [소스]는 Anywhere-IPv4로 설정한다. 그리고 페이지 하단의 [보안 그룹 생성]을 클릭한다.

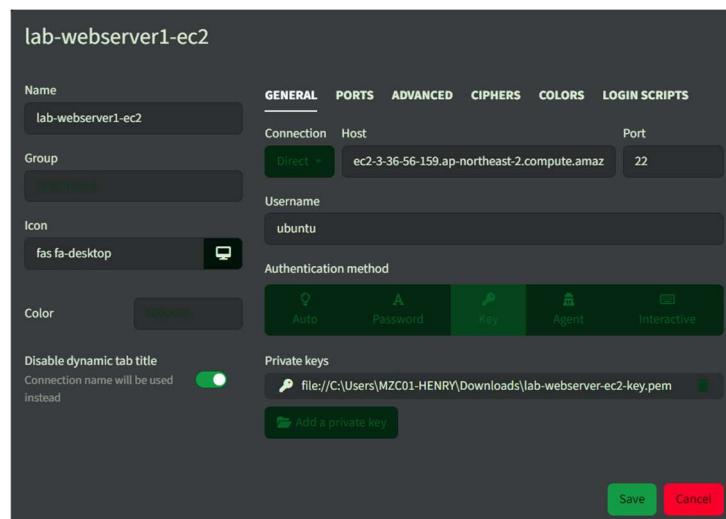


Apache Web Server AMI 백업을 통해 EC2 Instance 2개 생성하기

- ALB를 실습하기 위한 Web Server를 생성한다. 다음과 같이 lab-webserver1-ec2를 생성했다.
 - [이름] : lab-webserver1-ec2
 - [AMI] : Ubuntu Server 22.04 LTS (HVM), SSD Volume Type, 64비트(x86)
 - [인스턴스 유형] : t2.micro
 - [키 페어(로그인)] > [새 키 페어 생성] : lab-webserver-ec2-key
 - [네트워크 설정] > [편집] : lab-vpc, lab-subnet-public1-ap-northeast-2a, [퍼블릭 IP 자동 할당] 활성화
 - [보안 그룹] > [기존 보안 그룹 선택] : lab-sg
 - [스토리지 구성] : 8 GiB, gp2
 - [인스턴스 시작] 버튼 클릭



- macOS 사용자는 터미널을 이용해서, Windows 사용자는 PuTTY나 Xshell 또는 Tabby 같은 SSH Client Tool을 이용해서 SSH 연결한다.



- Apache Web Server를 다음과 같은 커맨드 명령 순서로 설치한다.

A. \$ sudo apt update

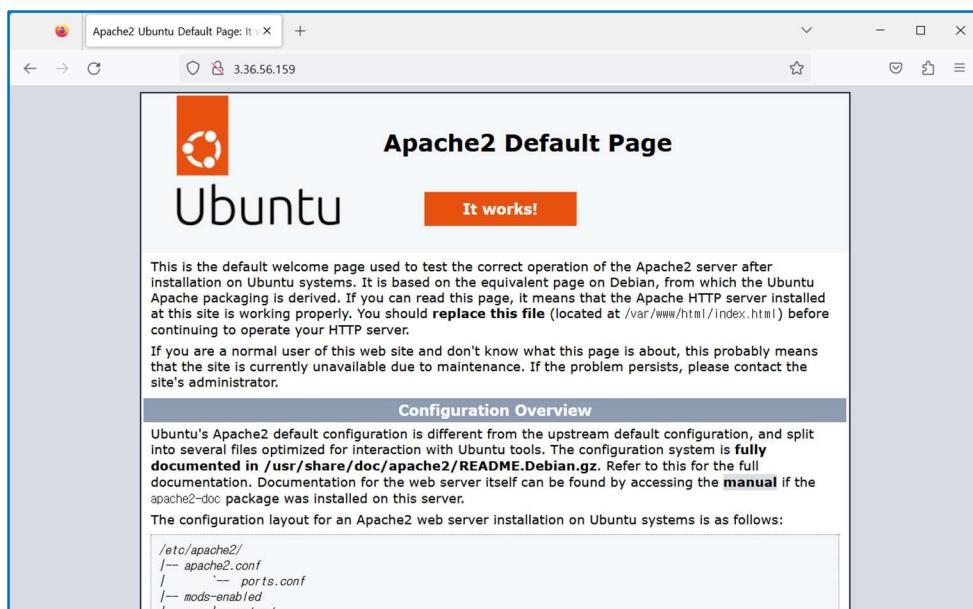
B. \$ sudo apt install -y apache2

C. \$ sudo systemctl status apache2

```
ubuntu@ip-10-0-10-47:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
  Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
  Active: active (running) since Mon 2023-10-16 06:07:19 UTC; 20s ago
    Docs: https://httpd.apache.org/docs/2.4/
          >Main PID: 2392 (apache2)
          Tasks: 55 (limit: 1141)
         Memory: 4.9M
            CPU: 29ms
       CGroup: /system.slice/apache2.service
                 └─2392 /usr/sbin/apache2 -k start
                   ├─2394 /usr/sbin/apache2 -k start
                   └─2395 /usr/sbin/apache2 -k start

Oct 16 06:07:19 ip-10-0-10-47 systemd[1]: Starting The Apache HTTP Server...
Oct 16 06:07:19 ip-10-0-10-47 systemd[1]: Started The Apache HTTP Server.
ubuntu@ip-10-0-10-47:~$
```

- 웹 브라우저를 열고 해당 EC2의 [퍼블릭 IPv4 주소]를 입력해서 Apache Web Server가 잘 실행되고 있음을 확인한다.



- 이제 기본 Apache Web 페이지를 수정해 보자. 다음의 커맨드 명령과 코드로 기본 페이지를 수정한다.

A. \$ sudo su

B. # echo "<h1>Hello, Ubuntu Apache Webserver1 from Seoul Ability Zone A</h1>" > /var/www/html/index.html

```

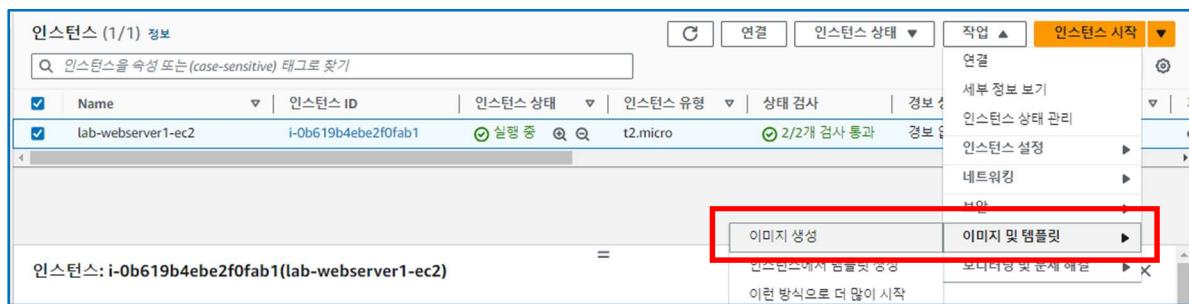
ubuntu@ip-10-0-10-47:~$ sudo su
root@ip-10-0-10-47:/home/ubuntu# echo "<h1>Hello, Ubuntu Apache Webserver1 from Seoul Ability Zone A</h1>" > /var/www/html/index.html
root@ip-10-0-10-47:/home/ubuntu# cat /var/www/html/index.html
<h1>Hello, Ubuntu Apache Webserver1 from Seoul Ability Zone A</h1>
root@ip-10-0-10-47:/home/ubuntu# exit
exit
ubuntu@ip-10-0-10-47:~$ 

```

6. 다시 페이지를 새로 고침해서 방금 수정한 Apache 기본 페이지를 확인한다.



7. 방금 생성한 Webserver 인스턴스를 한 개 더 생성하기 위해 **AMI**를 생성하기로 한다. **인스턴스 요약** 페이지에서 [작업] > [이미지 및 템플릿] > [이미지 생성]을 클릭한다.



8. [이미지 생성] 페이지에서 [이미지 이름]을 **webserver-ami**라고 입력하고 페이지를 아래로 계속 내려서 [이미지 생성] 버튼을 클릭한다.

볼륨 유형	디바이스	스냅샷	크기	볼륨 유형	IOPS	처리량	종료 시 삭제	암호화됨
EBS	/dev/s...	볼륨에서 새 스냅샷 생성	30	EBS 범용 SSD - gp2	100		<input checked="" type="checkbox"/> 활성화	<input type="checkbox"/> 활성화

9. 좌측 메뉴의 [이미지] > [AMI]를 클릭해서 [Amazon Machine Images(AMI)] 페이지로 이동하여 방금 생성한 webserver-ami를 확인한다.

The screenshot shows the 'Amazon Machine Images (AMI)' page with one item listed:

Name	AMI ID	AMI 이름	원본	소유자	표시 여부	상태	생성 날짜
ami-06aee060a...	webserver-...	789534828...	789534828...	프라이빗	사용 가능	2023/10/16 15:16 GMT+9	

A red box highlights the '상태' (Status) column, which shows '사용 가능' (Available).

10. 해당 AMI를 선택하고 페이지 우측 상단의 [Launch Instances from AMI] 오렌지 버튼을 클릭한다.

The screenshot shows the same 'Amazon Machine Images (AMI)' page, but now the 'webserver-ami' row has a checked checkbox in the first column. The 'Launch Instances from AMI' button in the top right corner is highlighted with a red box.

11. 방금 생성한 AMI 이미지를 가지고 다음과 같이 lab-webserver1-ec2를 생성했다. 지금 생성하는 EC2 Instance는 ap-northeast-2c Zone에 생성한다.

- A. [이름] : lab-webserver2-ec2
- B. [인스턴스 유형] : t2.micro
- C. [키 페어(로그인)] : lab-webserver-ec2-key
- D. [네트워크 설정] > [편집] : lab-vpc, lab-subnet-public2-ap-northeast-2c, [퍼블릭 IP 자동 할당] 활성화
- E. [보안 그룹] > [기존 보안 그룹 선택] : lab-sg
- F. [스토리지 구성] : 8 GiB, gp2
- G. [인스턴스 시작] 버튼 클릭

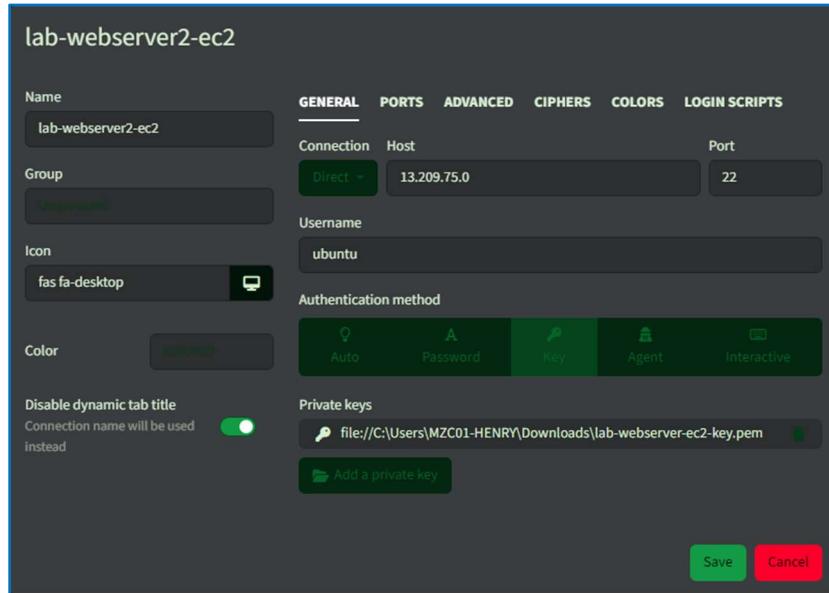
12. AMI를 이용해서 다음과 같이 lab-webserver2-ec2 새 인스턴스를 ap-northeast-2c에 잘 생성했다.

The screenshot shows the 'Instances' page with two instances listed:

Name	인스턴스 ID	인스턴스 상태	인스턴스 유형	상태 검사	정보 상태	가용 영역	퍼블릭 IPv4 DNS	퍼블릭 IPv4 ...
lab-webserver1-ec2	i-0b619b4ebef2f0fab1	실행 중	t2.micro	2/2개 검사 통과	경보 없음	+ ap-northeast-2a	ec2-3-36-56-159.ap.no...	3.36.56.159
lab-webserver2-ec2	i-077325c053274d838	실행 중	t2.micro	2/2개 검사 통과	경보 없음	+ ap-northeast-2c	ec2-13-209-75-0.ap.no...	13.209.75.0

A red box highlights the last two columns of the table.

13. 새로 생성한 lab-webserver2-ec2 인스턴스를 SSH Client Tool로 연결한다.



14. Webserver2의 터미널에서 다음과 같은 커맨드 명령을 통해 index.html을 수정한다.

A. \$ sudo su

B. # echo "<h1>Hello, Ubuntu Apache Webserver2 from Seoul Ability Zone C</h1>" > /var/www/html/index.html

```
ubuntu@ip-10-0-10-47:~$ sudo su
root@ip-10-0-10-47:/home/ubuntu# echo "<h1>Hello, Ubuntu Apache Webserver2 from Seoul Ability Zone C</h1>" > /var/www/html/index.html
root@ip-10-0-10-47:/home/ubuntu# cat /var/www/html/index.html
<h1>Hello, Ubuntu Apache Webserver2 from Seoul Ability Zone C</h1>
root@ip-10-0-10-47:/home/ubuntu# exit
exit
ubuntu@ip-10-0-10-47:~$
```

15. Web Browser를 열고 Webserver2의 **퍼블릭 IPv4** 주소로 연결해서 Seoul의 AZ C에서 접속한 결과를 확인 한다.



16. 계속해서 Webserver1의 **퍼블릭 IPv4** 주소로 연결해서 Seoul의 AZ A에서 접속한 결과를 확인한다.



ALB 생성하기

- 여기서는 Load Balancer중에서 ALB 즉, Application Load Balancer를 실습하기로 한다. EC2 서비스 페이지의 좌측 메뉴에서 [로드 밸런싱] > [로드밸런서]를 선택한다. 그리고 [로드 밸런서 생성] 버튼을 클릭한다.

인스턴스 유형
시작 템플릿
스팟 요청
Savings Plans
예약 인스턴스
전용 호스트
용량 예약
▼ 이미지
AMI
AMI 키탈로그
▼ Elastic Block Store
블룸
스냅샷
수명 주기 관리자
▼ 네트워크 및 보안
보안 그룹
단락적 IP
배치 그룹
키 페어
네트워크 인터페이스
▼ 로드밸런싱
로드밸런서
네임 그룹

EC2 > 로드밸런서

로드밸런서

Elastic Load Balancing은 수신 트래픽의 변화에 따라 자동으로 로드 밸런서 용량을 확장합니다.

Filter 로드밸런서

이름 DNS 이름 상태 VPC ID 가용 영역 유형 생성된 날짜

로드밸런서 없음
ap-northeast-2에 로드밸런서가 없음

로드밸런서 생성

0 로드밸런서 선택됨

위에서 로드밸런서를 선택합니다.

- [로드 밸런서 유형 선택]페이지에서 로드 밸런서 유형 선택을 [Application Load Balancer]의 [생성]을 클릭 한다.

EC2 > 로드밸런서 > 로드밸런서 유형 선택

로드밸런서 유형 선택

자세한 하이라이트와 함께 전체 기능별 비교도 제공됩니다. 자세히 알아보기

로드밸런서 유형

Application Load Balancer 정보	Network Load Balancer 정보	Gateway Load Balancer 정보
HTTP 및 HTTPS 트래픽을 사용하는 애플리케이션을 위한 유연한 기능이 필요한 경우 Application Load Balancer를 선택합니다. 요청 수준에 따라 작동하는 Application Load Balancer는 마이크로서비스 및 컨테이너를 비롯한 애플리케이션 아키텍처를 대상으로 하는 고급 로우팅 및 포시 기능을 제공합니다.	애플리케이션에 초고 성능, 대규모 TLS 오프로딩, 중앙 집중화된 인증서 배포, UDP에 대한 지원 및 고정 IP 주소가 필요한 경우 Network Load Balancer를 선택합니다. 연결 수준에서 작동하는 Network Load Balancer는 안전하게 초당 수백만 개의 요청을 처리하면서도 극히 낮은 지연 시간을 유지할 수 있습니다.	GENEVE를 지원하는 서드 파티 가상 애플라이언스 플랫폼 배포 및 관리해야 할 경우 Gateway Load Balancer를 선택합니다. 이러한 애플라이언스를 사용하면 보안, 규정 준수 및 정책 제어를 개선할 수 있습니다.
생성	생성	생성

▶ Classic Load Balancer - 이전 세대

3. [Application Load Balancer 생성] 페이지에서 다음과 같이 [로드 밸런서 이름]에 lab-webserver-alb라고 값을 입력한다.

Application Load Balancer 생성 정보

Application Load Balancer는 수신 HTTP 및 HTTPS 트래픽을 요청 속성을 기반으로 Amazon EC2 인스턴스, 마이크로서비스 및 컨테이너와 같은 여러 대상에 배포합니다. 로드 밸런서는 연결 요청을 수신하면 우선 순위에 따라 리스너 규칙을 평가하여 적절한 규칙을 결정한 다음 해당되는 경우, 대상 그룹에서 규칙 작업의 대상을 선택합니다.

▶ Elastic Load Balancing의 작동 방식

기본 구성

로드 밸런서 이름
이름은 AWS 계정 내에서 고유해야 하며 로드 밸런서 생성 후에는 변경할 수 없습니다.

체계 | 정보
로드 밸런서 생성 후에는 스마트를 변경할 수 없습니다.

인터넷 경계
인터넷 경계 로드 밸런서는 인터넷을 통해 클라이언트의 요청을 대상으로 라우팅합니다. 퍼블릭 서브넷이 필요합니다. 자세히 알아보기 [\[링크\]](#)

내부
내부 로드 밸런서는 프라이빗 IP 주소를 사용하여 클라이언트의 요청을 대상으로 라우팅합니다.

IP 주소 유형 | 정보
서브넷이 사용하는 IP 주소 유형을 선택합니다.

IPv4
내부 로드 밸런서에 권장합니다.

듀얼 스택
IPv4 및 IPv6 주소를 포함합니다.

4. 페이지를 아래로 스크롤을 다운하여 [네트워크 매팅] 섹션으로 이동한다. [VPC]에서 이미 생성한 lab-vpc를 선택한다. 주의할 점은 이미 인터넷 게이트웨이가 VPC와 연결되어 있어야 한다는 것이다.

네트워크 매팅 정보

로드 밸런서는 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 밸런서 생성 후에는 선택한 VPC를 변경할 수 없으려면 대상 그룹 [\[링크\]](#)를 참조하세요.

vpc-0c697c443eaeb7444
IPv4: 10.0.0.0/16

5. [매팅] 섹션에서는 [ap-northeast-2a]와 [ap-northeast-2c]를 체크하고 이미 생성했던 서브넷을 확인한다.

매핑 | 정보
가용 영역을 2개 이상 선택하고 영역당 하나의 서브넷을 선택합니다. 로드 밸런서는 이러한 가용 영역의 대상으로만 트래픽을 라우팅합니다. 로드 밸런서 또는 VPC에서 지원하지 않는 가용 영역은 선택할 수 없습니다.

ap-northeast-2a (apne2-az1)

서브넷

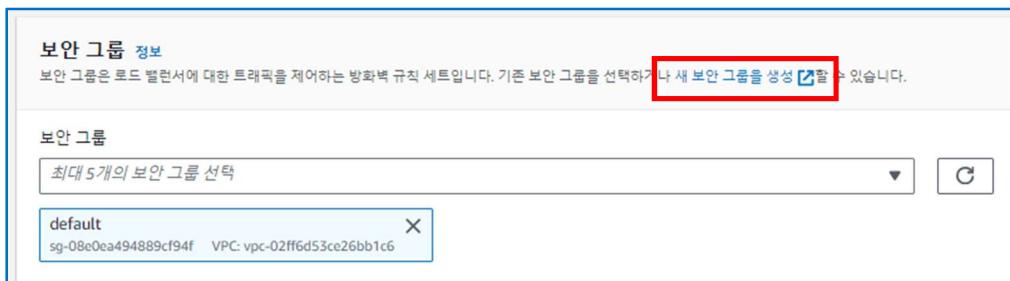
IPv4 주소
AWS에서 할당

ap-northeast-2c (apne2-az3)

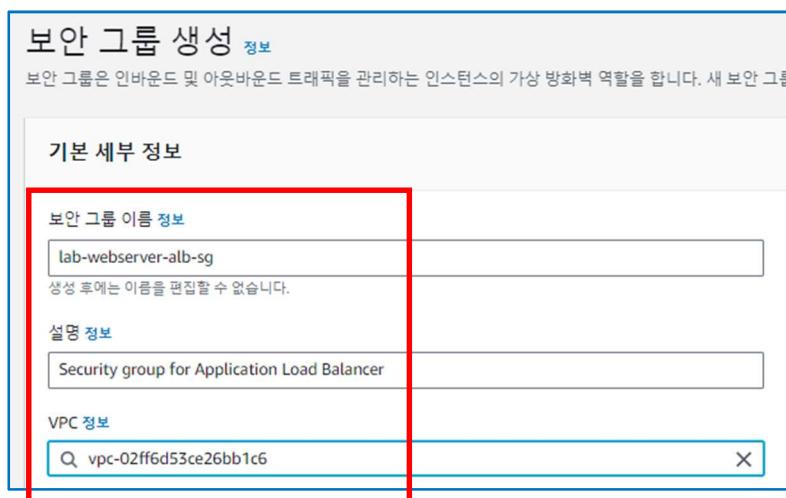
서브넷

IPv4 주소
AWS에서 할당

6. [보안 그룹]에서는 [새 보안 그룹 생성]링크를 클릭한다.



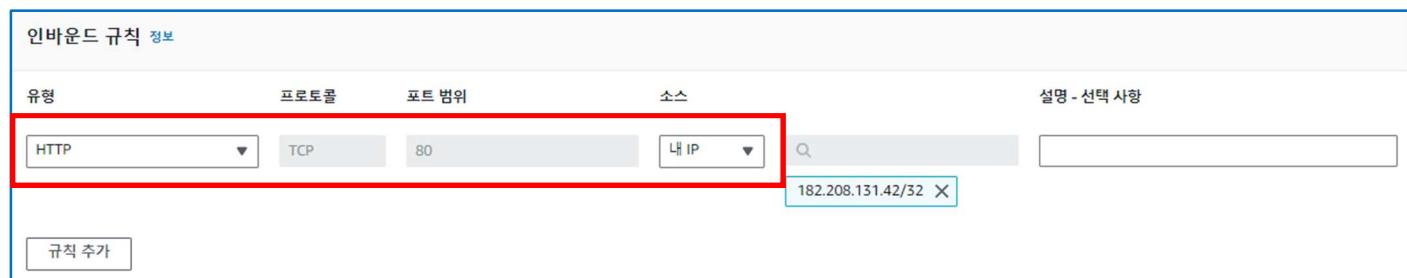
7. [보안 그룹 생성]페이지에서 [보안 그룹 이름]을 **lab-webserver-alb-sg**라고 입력하고 간단하게 설명에 "Security group for Application Load Balancer"이라고 입력한 후, [VPC]는 이미 생성한 **lab-vpc**를 선택한다. [설명]은 필수이며 한글은 입력처리가 되지 않기 때문에 영문으로 간단하게 남긴다.



8. [인바운드 규칙] 섹션에서 [규칙 추가]를 클릭하여 다음과 같은 값으로 설정한다.

A. [유형] : HTTP

B. [소스] : 내 IP



9. 페이지를 계속 아래로 스크롤 다운하여 [보안 그룹 생성]을 클릭한다.

10. 다시 [Application Load Balancer 생성] 페이지로 돌아와서 [새로고침] 버튼을 클릭하여 방금 생성한 보안 그룹을 선택하고 기존의 **default** 보안 그룹은 삭제한다.

보안 그룹 정보
보안 그룹
최대 5개의 보안 그룹 선택
lab-webserver-alb-sg X
sg-0c7c26264a4eccd74 VPC: vpc-02ff6d53ce26bb1c6

11. [리스너 및 라우팅] 섹션에서 현재 리스너가 **HTTP:80**을 확인하고 **[대상 그룹 생성]** 링크를 클릭한다.

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

▼ 리스너 HTTP:80

프로토콜 포트 기본 작업 정보
HTTP : 80 다음으로 전달: 대상 그룹 선택
1-65535 대상 그룹 생성 C

리스너 추가

12. [그룹 세부 정보 지정] 페이지에서 **[대상 유형 선택]**을 인스턴스로 선택한다.

EC2 > 대상 그룹 > 대상 그룹 생성

그룹 세부 정보 지정
로드 밸런서는 요청을 대상 그룹의 대상으로 라우팅하고 대상에 대한 상태 확인을 수행합니다.

1단계
그룹 세부 정보 지정
2단계
대상 등록

기본 구성
대상 그룹이 생성된 후에는 이 섹션의 설정을 변경할 수 없습니다.

대상 유형 선택
 인스턴스

- 특정 VPC 내의 인스턴스에 대한 로드 밸런싱을 지원합니다.
- Facilitates the use of Amazon EC2 Auto Scaling [\[Link\]](#) to manage and scale your EC2 capacity.

IP 주소

- VPC 및 온프레미스 리소스에 대한 로드 밸런싱을 지원합니다.
- 동일한 인스턴스에 있는 여러 IP 주소 및 네트워크 인터페이스로의 라우팅을 지원합니다.
- マイクロ서비스 기반 아키텍처를 통한 유연성을 제공하여 애플리케이션 간 통신을 간소화합니다.
- IPv6 대상을 지원하여 중단간 IPv6 통신 및 IPv4에서 IPv6로의 NAT를 활성화합니다.

Lambda 함수

- 단일 Lambda 함수로 라우팅을 지원합니다.
- Application Load Balancer에만 액세스할 수 있습니다.

Application Load Balancer

- Network Load Balancer가 특정 VPC 내에서 TCP 요청을 수락하고 라우팅할 수 있는 유연성을 제공합니다.
- Application Load Balancer로 고정 IP 주소 및 PrivateLink를 손쉽게 사용할 수 있습니다.

13. [대상 그룹 이름]을 **lab-webserver-alb-tg**라고 입력하고 [프로토콜]은 **HTTP**로 [포트]는 **80**임을 확인하며, [VPC]는 이미 생성했던 **lab-vpc**, 마지막으로 [프로토콜 버전]은 **HTTP1**을 선택한다.

대상 그룹 이름
lab-webserver-alb-tg
하이픈을 포함하여 최대 32자리의 영숫자 문자를 사용할 수 있지만 이름이 하이픈으로 시작하거나 끝나지 않아야 합니다.

프로토콜 포트
HTTP : 80
1-65535

IP 주소 유형
표시된 IP 주소 유형의 대상만 이 대상 그룹에 포함될 수 있습니다.

IPv4
Each instance has a default network interface (eth0) that is assigned the primary private IPv4 address. The instance's primary private IPv4 address is the one that will be applied to the target.

IPv6
Each instance you register must have an assigned primary IPv6 address. This is configured on the instance's default network interface (eth0). Learn more [\[\]](#)

VPC
대상 그룹에 포함할 인스턴스가 있는 VPC를 선택합니다. 위에서 선택한 IP 주소 유형을 지원하는 VPC만 이 목록에서 사용할 수 있습니다.

lab-vpc
vpc-02ff6d53ce26bb1c6
IPv4: 10.0.0.16

프로토콜 버전

HTTP1
HTTP/1.1을 사용하여 대상으로 요청을 전송합니다. 요청 프로토콜이 HTTP/1.1 또는 HTTP/2일 때 지원됩니다.

HTTP2
HTTP/2를 사용하여 대상으로 요청을 전송합니다. 요청 프로토콜이 HTTP/2 또는 gRPC일 때 지원되지만 gRPC 전용 기능은 사용할 수 없습니다.

gRPC
gRPC를 사용하여 대상으로 요청을 전송합니다. 요청 프로토콜이 gRPC일 때 지원됩니다.

14. [상태 검사]가 **HTTP** 프로토콜로 지정돼 있음을 확인하고 [다음]을 클릭한다.

상태 검사
연결된 Load Balancer가 상태 테스트를 위해 등록된 대상에 아래 설정에 따라 요청을 주기적으로 전송합니다.

상태 검사 프로토콜
HTTP

상태 검사 경로
기본 경로 "/"를 사용하여 루트를 ping하거나 원하는 경우 사용자 지정 경로를 지정합니다.
/

최대 1024자까지 허용됩니다.

▶ 고급 상태 검사 설정

▶ 태그 - 선택 사항
대상 그룹에 태그를 추가하는 것을 고려하십시오. 태그를 사용하면 AWS 리소스를 분류하여 좀 더 쉽게 관리할 수 있습니다.

취소 다음

15. 대상 등록에 보이는 두 개의 인스턴스 **lab-webserver1-ec2**와 **lab-webserver2-ec2**를 [아래에 보류 중인 것으로 포함] 버튼을 통해 대상으로 등록한다.

0개 선택됨

선택한 인스턴스를 위한 포트
선택한 인스턴스로 트래픽을 라우팅하기 위한 포트입니다.
80
1-65535(쉼표로 여러 포트 구분)

아래에 보류 중인 것으로 포함

2개의 선택 항목이 현재 아래에 보류 중입니다. 준비가 되면 대상을 더 포함하거나 등록하십시오.

대상 보기

제거	상태 확인	인스턴스 ID	이름	포트	상태	보안 그룹	영역	프라이빗 IPv4 주소	서비스
X	대기 중	i-0b619b4ebe2f0fab1	lab-webserver1-ec2	80	실행 중	lab-sg	ap-northeast-2a	10.0.10.47	sub
X	대기 중	i-077325c053274d838	lab-webserver2-ec2	80	실행 중	lab-sg	ap-northeast-2c	10.0.20.198	sub

16. 페이지 하단의 [대상 그룹 생성] 오렌지 버튼을 클릭한다.

2개 대기 중

취소 이전 대상 그룹 생성

17. 대상 그룹이 생성되었다.

The screenshot shows the AWS EC2 Target Groups page. At the top, a green bar indicates the target group has been created successfully: '대상 그룹 생성 완료: lab-webserver-alb-tg'. Below this, the main table lists one target group:

이름	ARN	포트	프로토콜	대상 유형	로드밸런서	VPC ID
lab-webserver-alb-tg	arn:aws:elasticloadbalancing:us-east-1:123456789012:targetgroup/lab-webserver-alb-tg/sg-0e7c26264a4eccd74	80	HTTP	인스턴스	연결된 항목 없음	vpc-02ff6d53c

18. 대상 그룹이 생성되면 로드 밸런서 화면으로 다시 돌아와서 [리스너 및 라우팅]에서 새로 고침 버튼을 클릭하고 방금 생성한 **webserver-tg**를 선택한다.

The screenshot shows the 'Listeners & Routing' configuration for a load balancer. It is currently set to port 80. A red box highlights the 'Forward traffic to' section where the target group 'lab-webserver-alb-tg' is selected. Below this, there is a note about adding tags to the listener.

19. 페이지 아래로 계속 스크롤다운하여 [요약]에서 한 번 지금까지의 작업을 확인 후, 페이지 오른쪽 아래의 [로드밸런서 생성]을 클릭하여 로드밸런서를 생성한다.

The screenshot shows the 'Summary' page for creating a load balancer. It displays the configuration details: VPC (vpc-02ff6d53ce26bb1c6), subnets (ap-northeast-2a, ap-northeast-2c), and listeners (HTTP:80). A red box highlights the large orange 'Create Load Balancer' button at the bottom right.

20. [Application Load Balancer]가 잘 생성되었다. [상태]의 값이 활성일때까지 대기한다.

The screenshot shows the AWS EC2 Load Balancers page. A single load balancer named 'lab-webserver-alb' is listed. The 'Status' column for this entry is highlighted with a red box and contains a green circle with a checkmark, indicating it is active. Other columns include 'Name' (lab-webserver-alb), 'DNS Name' (lab-webserver-alb-703280...), 'VPC ID' (vpc-02ff6d53ce26bb1c6), 'Usage Type' (Application), and 'Created At' (2023년 10월 16일, 15:59 (UTC+09:00)).

21. 로드밸런서를 통하여 웹 애플리케이션에 접근하기 이전, 위에서 생성하였던 웹 서버가 로드 밸런서의 트래픽만 받게 하기 위하여 보안 그룹 인바운드 규칙 편집 작업을 수행한다. 보안 그룹 메뉴에서 lab-sg 보안 그룹을 선택 후, [인바운드 규칙 편집]을 클릭한다.

The screenshot shows the AWS Security Groups page for a security group named 'lab-sg'. The 'Inbound Rules' tab is selected. It displays two inbound rules:

Name	보안 그룹 규칙 ID	IP 범위	유형	프로토콜	포트 범위	소스
-	sgr-0f17429da2ce7b374	IPv4	HTTP	TCP	80	0.0.0.0/0
-	sgr-0bda21ab1521bb883	IPv4	SSH	TCP	22	0.0.0.0/0

22. 기존 HTTP 인바운드 규칙을 [삭제] 버튼을 클릭하여 삭제한다.

The screenshot shows the AWS Security Groups page for the 'lab-sg' group, specifically the 'Inbound Rules' section. The first rule, which defines port 80 for HTTP traffic, has its delete button highlighted with a red box. This rule is associated with the security group ID 'sgr-0f17429da2ce7b374'.

23. 그리고 새로운 규칙을 생성한다. [규칙 추가]를 클릭하여 로드밸런서의 트래픽만 받을 수 있도록 다음과 같이 설정한다.

A. [유형] : HTTP

B. [소스] : 사용자 지정

C. 목록에서 lab-webserver-alb-sg를 지정

그리고 오른쪽 하단에 있는 [규칙 저장] 버튼을 클릭한다.

The screenshot shows the AWS CloudWatch Metrics Insights search interface. At the top, there's a search bar with the query: `aws.ec2.*.cpu_utilization > 80 AND aws.ec2.*.status_code == 404`. Below the search bar, there are two tabs: "Metrics" and "Logs". Under the "Metrics" tab, there are two results listed:

- Result 1: `aws.ec2.us-east-1.us-east-1a.ec2-metrics:AWS/ElasticLoadBalancing/RequestCount` - Last 1 hour ago, Last 1 hour ago
- Result 2: `aws.ec2.us-east-1.us-east-1a.ec2-metrics:AWS/ElasticLoadBalancing/RequestCount` - Last 1 hour ago, Last 1 hour ago

At the bottom of the page, there are navigation links: "Metrics", "Logs", "CloudWatch Metrics", "CloudWatch Metrics Insights", and "CloudWatch Metrics Insights Help".

24. 다시 로드밸런서 메뉴로 들어와 방금 생성한 로드밸런서의 상세 페이지에서 [DNS 이름]을 복사하여 웹 브라우저에 붙여 넣는다.

The screenshot shows the AWS Load Balancer console. It displays the details of the load balancer named "lab-webserver-alb".

이름	DNS 이름	상태	VPC ID	가용 영역	유형	생성된 날짜
lab-webserver-alb	lab-webserver-alb-703280...	활성	vpc-02ff6d53ce26bb1c6	2 가용 영역	application	2023년 10월 16일, 15:59 (UTC+09:00)

Below the table, there are tabs for "세부 정보", "리스너 및 규칙", "네트워크 매팅", "보안", "모니터링", "통합", "속성", and "태그".

In the "세부 정보" section, there are several details:

- 로드밸런서 유형: Application
- 상태: 활성
- VPC: vpc-02ff6d53ce26bb1c6
- IP 주소 유형: IPv4
- 체계: Internet-facing
- 호스팅 영역: ZWKZPGT148KDX
- 가용 영역:
 - subnet-00741cb2ca85f0f48 (ap-northeast-2a)
 - subnet-0416a1c4941baa1a5 (ap-northeast-2c)
- 생성된 날짜: 2023년 10월 16일, 15:59 (UTC+09:00)

At the bottom, there's a section for "로드밸런서 ARN": `arn:aws:elasticloadbalancing:ap-northeast-2:789534828835:loadbalancer/app/lab-webserver-alb/da64071d7c336ffd`. To the right of this, there's a box labeled "DNS 이름 정보" containing the value: `lab-webserver-alb-703280660.ap-northeast-2.elb.amazonaws.com (A 레코드)`.

25. 아래와 같은 웹 화면을 확인할 수 있다.



26. 웹 화면 새로 고침하면 Application Load Balancer에서 기본으로 제공하는 라운드 로빈 알고리즘에 따라 화면에 보이는 Availability Zone 값이 변경되는 것을 확인할 수 있다.



Application Load Balancer Test on Failure

1. 이미 실행중인 Webserver 인스턴스에서 lab-webserver1-ec2를 선택한 후 [인스턴스 상태] > [인스턴스 중지]를 클릭한다.

The screenshot shows the CloudWatch Metrics interface for two instances. The top navigation bar includes '인스턴스 (1/2) 정보' and '인스턴스 상태 ▲' (Instance Status ▲). The '인스턴스 상태' dropdown is highlighted with a red box and has '인스턴스 중지' (Stopped) selected. Below the dropdown, there are sections for '인스턴스 재부팅' (Reboot Instance), '인스턴스 최대 절전 모드' (Maximum Power Savings Mode), and '인스턴스 종료' (Termination). The table lists two instances: 'lab-webserver1-ec2' (ID: i-0b619b4ebe2f0fab1) and 'lab-webserver2-ec2' (ID: i-077325c053274d838). Both instances are currently running (status: 실행 중).

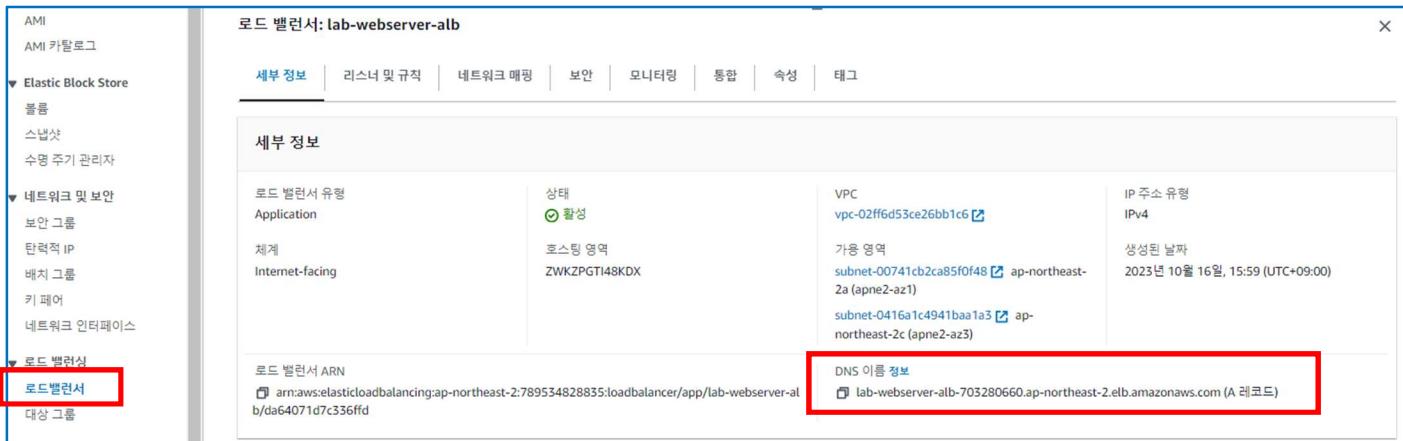
2. 해당 인스턴스가 현재 중지됨을 확인한다.

This screenshot shows the same CloudWatch Metrics interface as above, but with the '인스턴스 상태' dropdown set to '중지됨' (Stopped). The table below shows the same two instances, with 'lab-webserver1-ec2' now listed as stopped.

3. 좌측 메뉴의 [로드 밸런싱] > [대상 그룹]을 클릭하여 대상 그룹 페이지로 이동한다. 앞 lab에서 생성한 lab-webserver-alb-tg 대상그룹의 [등록된 대상] 섹션에 보면 lab-webserver1-ec2의 [상태 확인] 값이 unused로 변경됨을 확인한다.

The screenshot shows the Application Load Balancer Target Groups page for 'lab-webserver-alb-tg'. The left sidebar includes sections for '인스턴스 유형', '시작 험플릿', '스팟 요청', 'Savings Plans', '예약 인스턴스', '전용 호스트', and '총량 예약'. The main area shows the target group details and the '등록된 대상' (Registered Targets) section. The '대상' tab is selected. It lists two targets: 'lab-webserver2-ec2' (ARN: arn:aws:elasticloadbalancing:ap-northeast-2:c... 80 HTTP 인스턴스) and 'lab-webserver1-ec2' (ARN: arn:aws:elasticloadbalancing:ap-northeast-2:c... 80 HTTP 인스턴스). The '상태 확인' (State Check) column shows 'healthy' for the first target and 'unused' for the second. A red box highlights the '대상' tab and the 'unused' status of the second target.

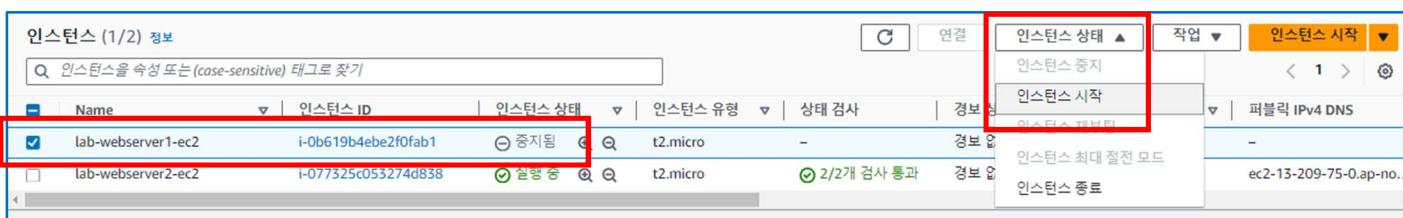
4. 좌측 메뉴 중 [로드 밸런싱] > [로드밸런서]로 이동하여 현재 lab-webserver-alb의 [DNS 이름]을 복사한다.



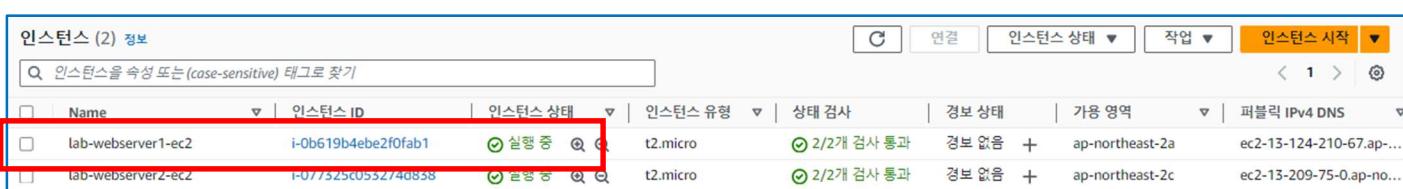
5. 복사한 [DNS 이름]을 웹 브라우저에 입력하여 웹 사이트 접속 후 여러 차례 새로고침을 수행하여 2번 서버 즉 ubuntu-webserver2-ec2로만 연결됨을 확인한다.



6. [인스턴스] > [인스턴스]로 클릭, 인스턴스 페이지로 이동하여 다시 1번 서버를 선택하고 [인스턴스 상태] > [인스턴스 시작]을 클릭하여 중지되어 있는 1번 서버를 시작한다.



7. 1번 서버 즉 lab-webserver1-ec2가 정상적으로 실행 중임을 확인한다.



8. 좌측 메뉴 중 [로드 밸런싱] > [대상 그룹]을 클릭하여 [대상 그룹] 페이지로 이동 후, [세부 정보] 탭에서 확인해보면 현재 정상적인 서버가 2대이고, 비정상과 사용되지 않음이 0임을 확인할 수 있다.

대상 그룹 (1/1) 정보

이름	ARN	포트	프로토콜	대상 유형	로드 밸런서	VPC ID
lab-webserver-alb-tg	arn:aws:elasticloadbalancing:ap-northeast-2:789534828835:targetgroup/lab-webserver-alb-tg/b8abc2eba2188c4c	80	HTTP	인스턴스	lab-webserver-alb	vpc-02ff6d53ce26bb1c6

대상 그룹: lab-webserver-alb-tg

세부 정보

arn:aws:elasticloadbalancing:ap-northeast-2:789534828835:targetgroup/lab-webserver-alb-tg/b8abc2eba2188c4c

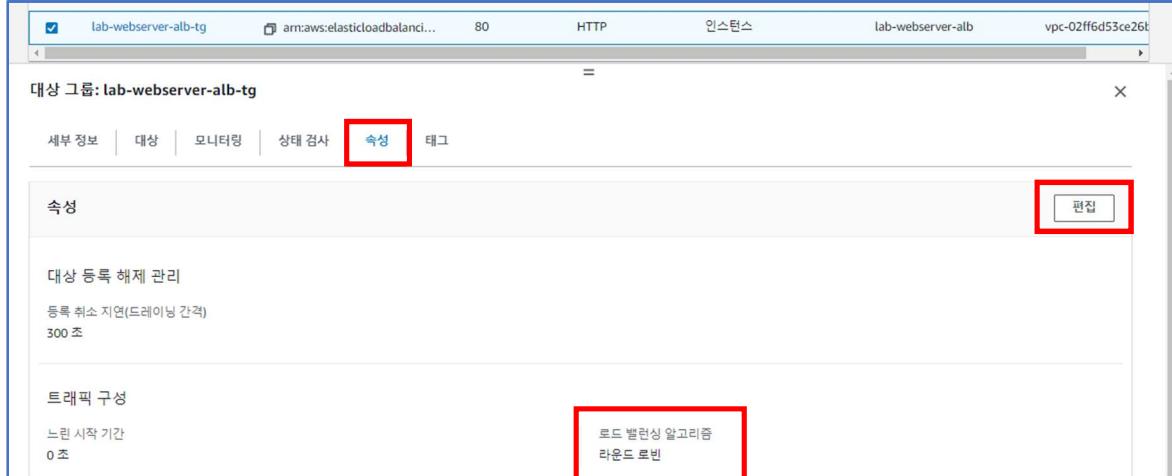
대상 유형 인스턴스	프로토콜: 포트 HTTP: 80	프로토콜 버전 HTTP1	VPC vpc-02ff6d53ce26bb1c6
IP 주소 유형 IPv4	로드 밸런서 lab-webserver-alb		
대상 합계 2	정상 2	비정상 0	사용되지 않음 0
			초기 0
			Draining 0

9. 로드 밸런서의 [DNS 이름]을 복사하여 웹 브라우저에 입력하고 웹 사이트에 접속한 페이지를 새로고침을 여러 번 수행하여 1번 서버와 2번 서버가 교차 반복됨을 확인한다.

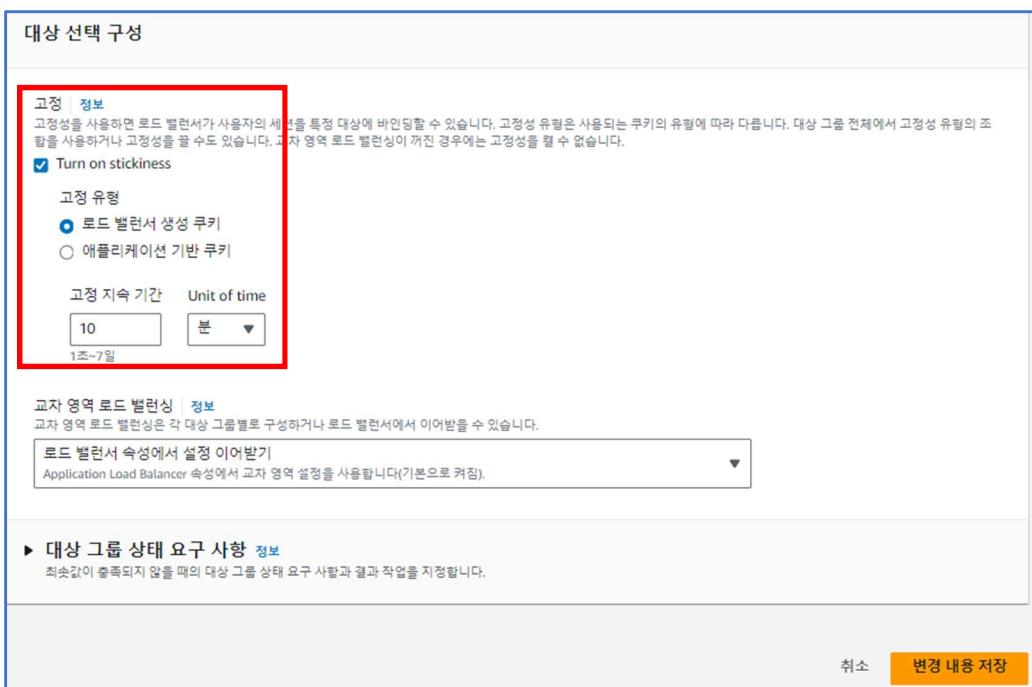


Application Load Balancer Sticky Session Setting

- 좌측 메뉴 중 [로드 밸런싱] > [대상 그룹]에서 현재 대상 그룹인 lab-webserver-alb-tg의 [속성] 탭을 선택해서 [로드 밸런싱 알고리즘]이 라운드 로빈임을 확인한다. [편집] 버튼을 클릭한다.



- [대상 그룹 속성 편집] 페이지에서 [고정]의 값을 체크해서 활성화한 후, [고정 지속 기간]을 10분으로 설정한다. 그리고 [변경 내용 저장]을 클릭하여 창을 닫는다.



- ELB의 [DNS 이름] 값을 복사하여 웹 브라우저에 입력한 후, 새로 고침을 여러 번 하여 접속한 웹 사이트가 고정되어 동일한 서버로만 접속되는지 확인한다.

