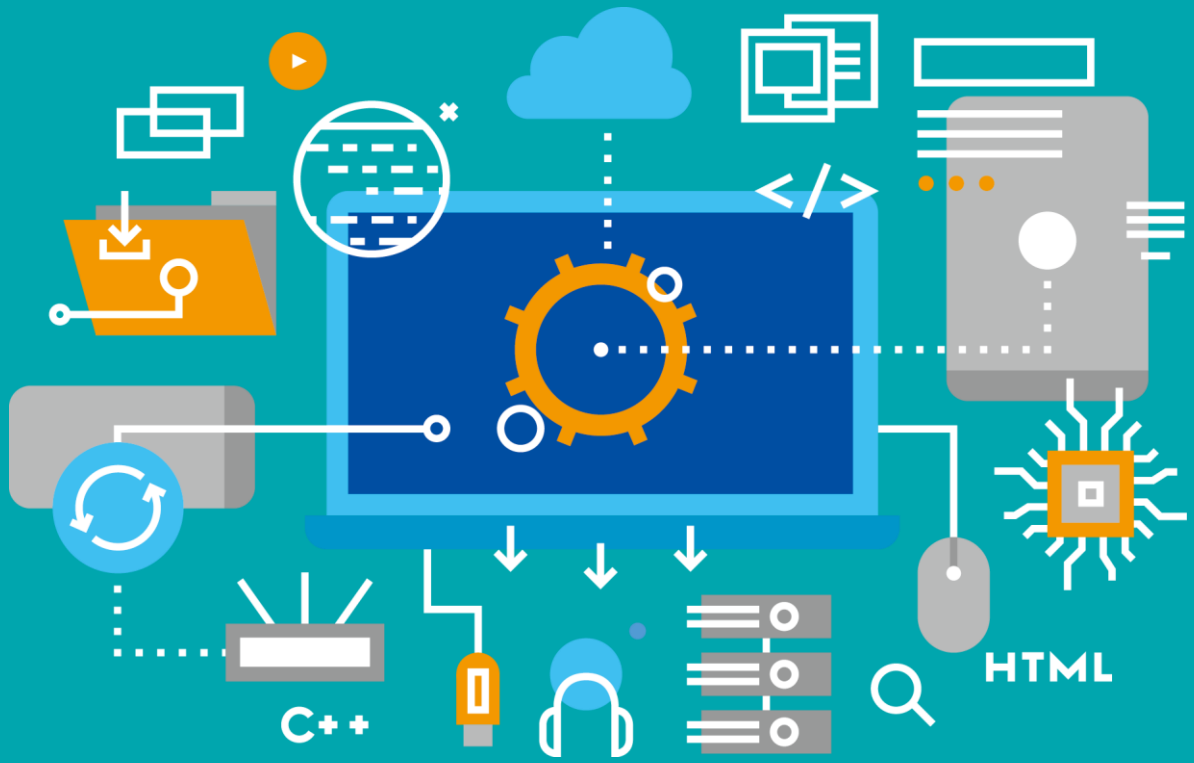
동양미래대학교

DMMU

동양미래대학교 전문기술 석사과정

클라우드와 네트워크 보안

Dongyang Mirae University



5장

로드 밸런서 :

4 계층 장비(세션 장비)

로드 밸런서/방화벽: 4 계층 장비(세션 장비)

- 2, 3계층과 달리 **통신의 방향성, 순서와 같은 통신 전반에 관한 관리** 필요
- **세션 테이블(Session Table)** : 통신의 방향성, 순서와 같은 통신 전반에 관한 정보 보관

4 계층 장비의 특징

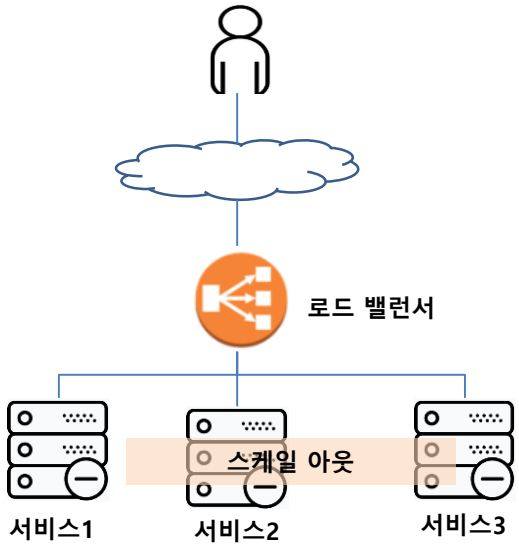
- TCP와 같은 4계층 헤더 정보 이해
- 4계층 장비에서는 세션 테이블과 세션 정보의 이해가 필수
- 4계층 이상에서 동작하는 로드 밸런서, 방화벽 같은 장비를 '**세션 장비**'라고도 부름

세션 장비 이해를 위해 필요한 요소

세션 테이블	• 세션 장비는 세션 테이블을 기반으로 운영 • 세션 정보는 세션 테이블에 남아있는 life time이 존재
Symmetric 경로 요구	• In-bound와 out-bound 경로가 일치 해야 함
정보 변경 (로드 밸런서의 경우)	• IP 주소가 변경되며 확장된 L7 로드 밸런스(ADC)는 애플리케이션 프로토콜 정보도 변경됨

로드 밸런서

- IP 주소나 4계층 정보, 애플리케이션 정보를 확인 수정하여 트래픽을 분배
- 대표 IP 주소를 서비스 IP로 갖고, 로드 밸런서가 실제 시스템의 IP로 변경하여 요청을 보냄
- 웹, 앱 애플리케이션, FWLB(Fire Wall Load Balancing), VPNLB(VPN Load Balancing)



❖ **L4 Load Balancing** : TCP/UDP (**port #**)를 기반으로 부하 분산 수행

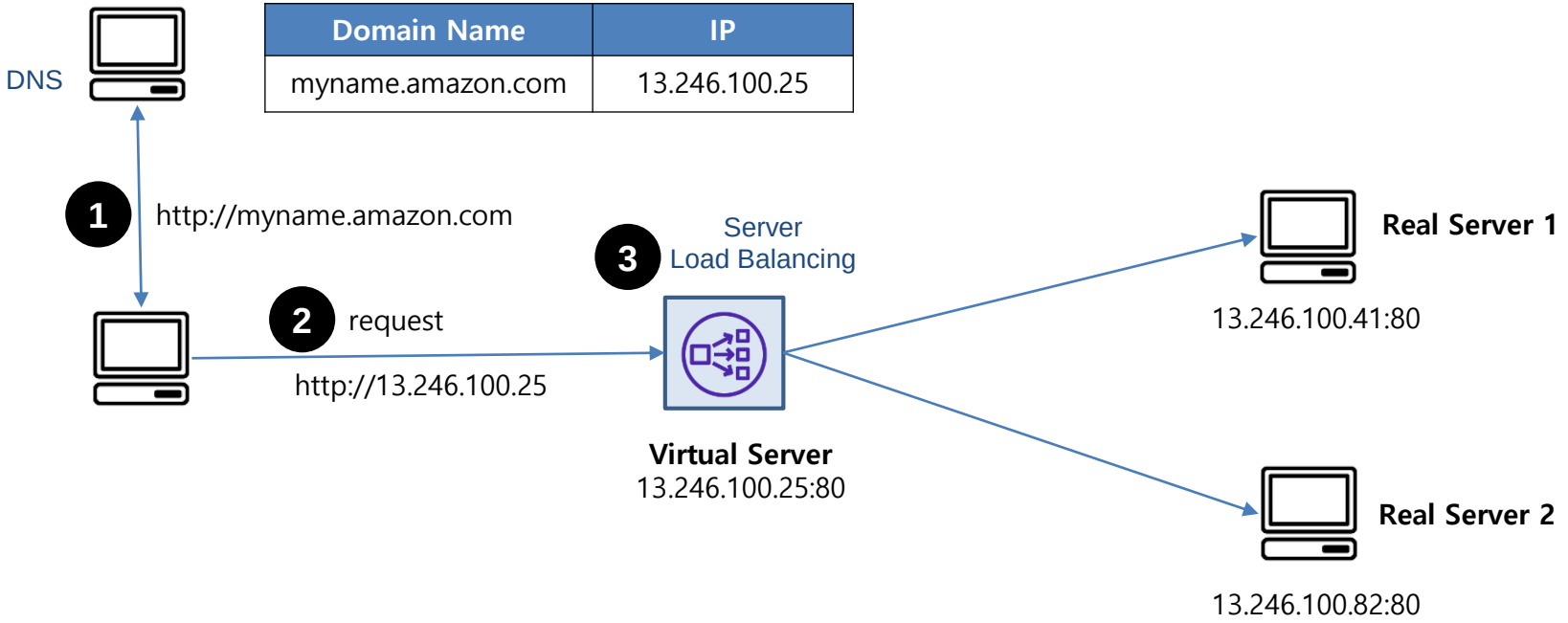
❖ **L7 Load Balancing** : HTTP, FTP, SMTP 등 **응용 프로토콜을 기반**으로 부하분산 수행

✓ **ADC(Application Delivery Controller)** : HTTP, URI 정보 기반으로 프로토콜 분석 후, 부하를 분산함. Squid, Nginx에서 수행하는 Reverse Proxy와 유사한 기능

❖ 통상, Load Balancing 장비는 L4, L7을 모두 지원하며, 설정에 따라 L4, L7로 구분됨

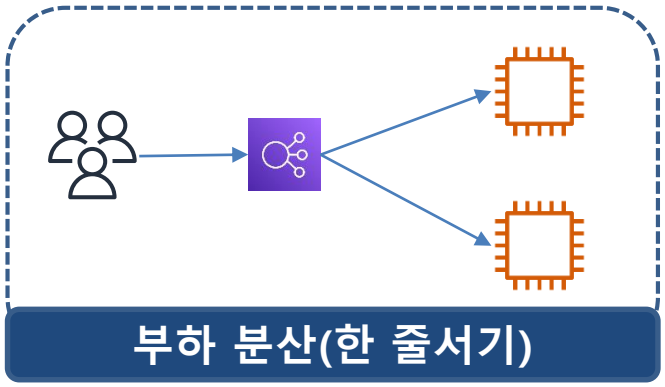
❖ **AWS의 경우** L4용 전용 컴포넌트로 NLB(Network Load Balancer)
L7용 전용 컴포넌트로 ALB(Application Load Balancer)를 구분하여 사용

로드 밸런서 On-Premise SLB 동작

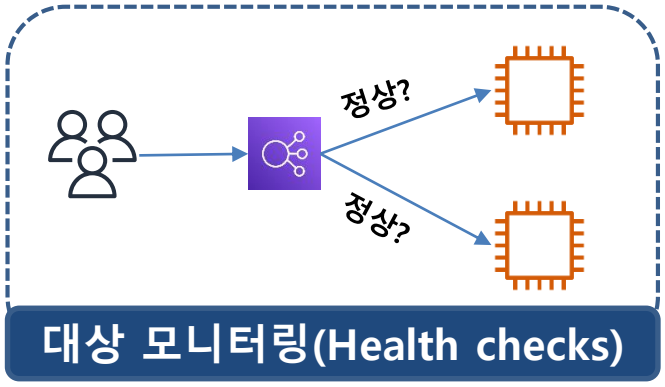


- IP 주소와 port 쌍을 분석하여 부하 분산하는 경우, L4 스위치 활용
- HTTP 프로토콜 헤더 분석을 통해 부하를 분산하는 경우, L7 스위치 활용
- L4, L7 스위치 모두 4계층 레벨을 기반으로 트래픽을 처리하므로, 통상 L4 스위치라고도 부름

Load balancer 기능



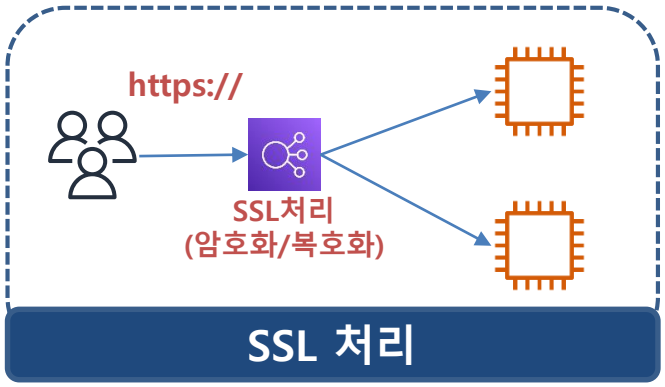
- 사용자의 접속을 자동으로 분산
- 대량의 접속이 발생하는 경우, 부하를 각각의 Target으로 분산
- AWS의 경우, Target을 서로 다른 가용영역(AZ)으로 배치할 경우, 가용성 향상



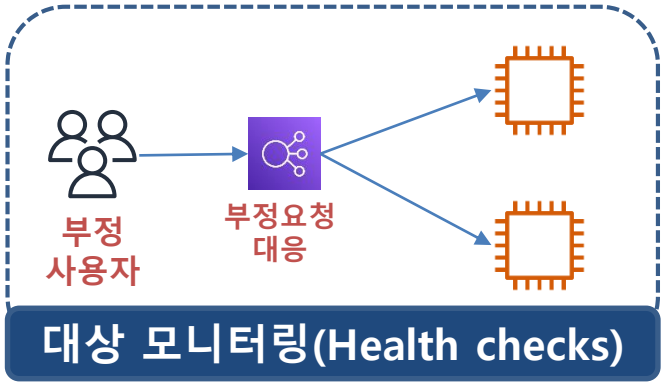
- LB는 Target에 대해 연결과 상태를 감시하고 확인
- 감시를 통해 비정상적인 동작을 감지하면 대상을 자동으로 분리

(예) 웹서버의 정해진 경로로 보낸 요청이 지정된 횟수만큼 실패하면, 해당 웹 서버로 요청을 보내지 않음

ELB (Elastic Load balancing) : AWS에서 제공하는 로드 밸런서 서비스



- SSL(Secure Sockets Layer) : 송수신 데이터를 암호화 처리
- https 프로토콜 통신시 SSL 이용(브라우저와 서비스 사이 패킷 암호화)
- 웹서버에서 암호화/복호화 처리시 성능 저하
- 로드 밸런서에 암호처리 관련 전용 시스템을 제공하여 빠른 대응



- LB는 부정한 접근을 감지하여 방지
- 로드밸런서에 부정한 접근에 대응하는 전용 시스템 제공으로 효율적 대응

로드 밸런서

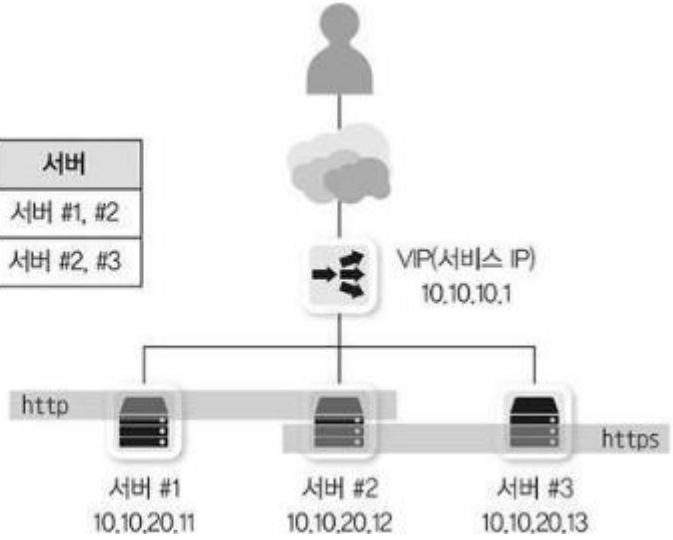
	L4 로드밸런서	L7 로드밸런서
네트워크 계층	Layer 4 전송계층(Transport layer)	Layer 7 응용계층(Application layer)
특징	> TCP/UDP 포트 정보를 바탕으로 함	> TCP/UDP 정보는 물론 HTTP의 URI, FTP의 파일명, 쿠키 정보 등을 바탕으로 함
장점	> 데이터 안을 들여다보지 않고 패킷 레벨에서만 로드를 분산하기 때문에 속도가 빠르고 효율이 높음 > 데이터의 내용을 복호화할 필요가 없기에 안전함 > L7 로드밸런서보다 가격이 저렴함	> 상위 계층에서 로드를 분산하기 때문에 훨씬 더 섬세한 라우팅이 가능함 > 캐싱 기능을 제공함 > 비정상적인 트래픽을 사전에 필터링할 수 있어 서비스 안정성이 높음
단점	> 패킷의 내용을 살펴볼 수 없기 때문에 섬세한 라우팅이 불가능함 > 사용자의 IP가 수시로 바뀌는 경우라면 연속적인 서비스를 제공하기 어려움	> 패킷의 내용을 복호화해야 하기에 더 높은 비용을 지불해야 함 > 클라이언트가 로드밸런서와 인증서를 공유해야하기 때문에 공격자가 로드밸런서를 통해서 클라이언트에 데이터에 접근할 보안 상의 위험성이 존재함

로드 밸런서

- L4 스위치

- 4계층에서 동작하면서 로드 밸런싱 기능을 가진 스위치
- 내부 동작은 4계층 로드 밸런서이지만 외형은 스위치 처럼 여러 포트를 가짐
- L4 스위치는 부하 분산, 성능 최적화, 리다이렉션 기능 제공

VIP(서비스 IP)	서비스 포트	서버
10.10.10.1	http(80)	서버 #1, #2
10.10.10.1	https(443)	서버 #2, #3



❖ **가상 서버, 가상 IP** :
- 사용자가 바라보는 실제 서비스와 서버 IP 주소

❖ **리얼 서버, 리얼 IP** :
- 실제 서비스를 수행하는 서버와 서버 IP 주소

❖ 사용자가 가상 서버의 가상 IP로 서비스 요청을 하면 로드 밸런서는 가상 IP를 리얼 IP로 변환하고 정책에 따라 부하 분산

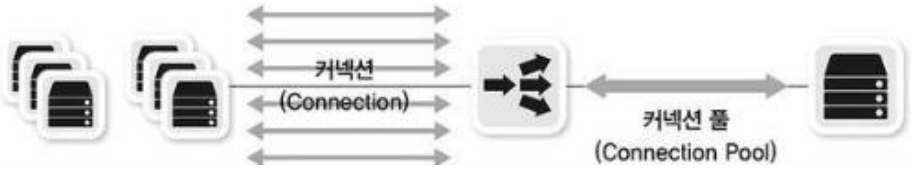
로드 밸런서

- ADC(Application Delivery Controller)

- Application 계층(7 계층)에서 동작하는 로드 밸런서
- **애플리케이션 프로토콜 헤더와 내용**을 바탕으로 다양한 부하 분산, 정보 수정, 정보 필터링이 가능하며 Proxy로 동작
- L4에서 L7까지 로드 밸런싱 기능을 제공하며, Failover(장애 극복), 리다이렉션(Redirection), 캐싱(Caching), 압축(Compression), 콘텐츠 변환 및 재 작성, 인코딩 변환 등 기능 수행
- WAF(Web Application Firewall) 기능이나 HTML, XML 검증과 변화 기능도 플러그 인 형태로 수행 가능

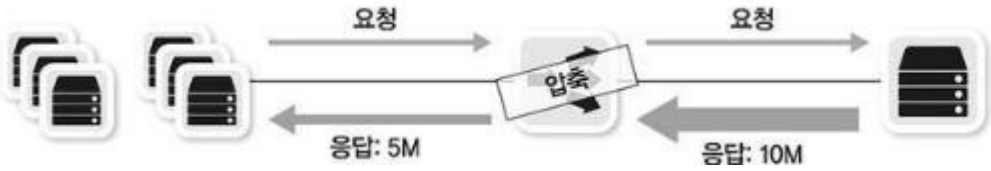
로드 밸런서

- L4 스위치 vs ADC



【L4 스위치 성능 향상 : TCP reuse, Connection Pooling】

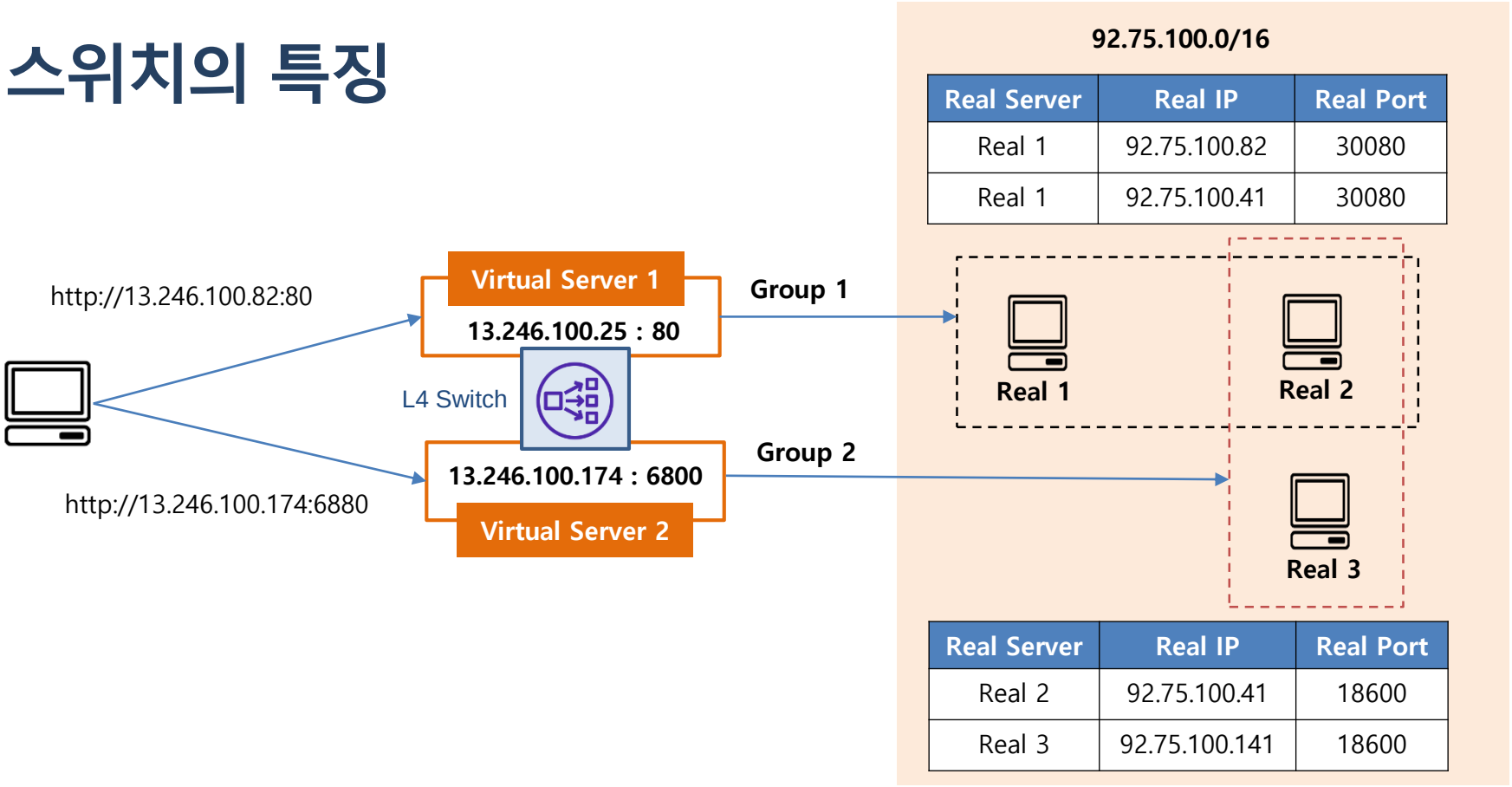
【ADC 성능 최적화 : 캐싱 기능】



【ADC 성능 최적화 : 압축 기능】

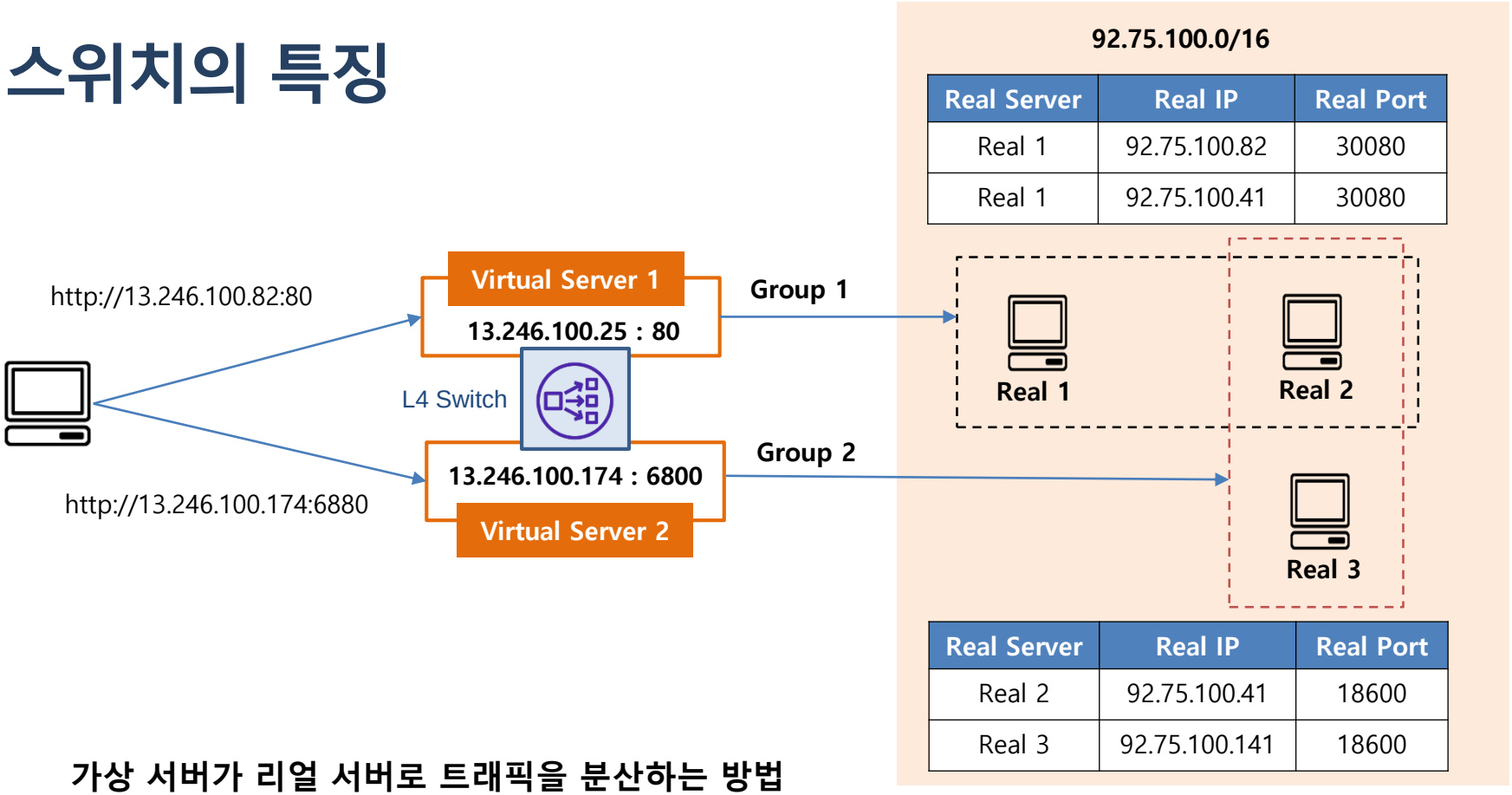
【ADC 성능 최적화 : SSL off loading 기능】

L4 스위치의 특징



- L4 switch 내부에 다수의 가상 서버 생성 가능
- 가상 서버가 트래픽을 분산하는 리얼 서버는 92.75.100.0/16 내부에 위치함
- 리얼 서버는 여러 그룹에 중복 사용 가능 (real 2는 group 1과 group 2에 중복)
- 가상 서버의 vport(80, 6800)는 리얼 서버의 rport(30090, 18600)으로 상이하게 설정 가능

L4 스위치의 특징

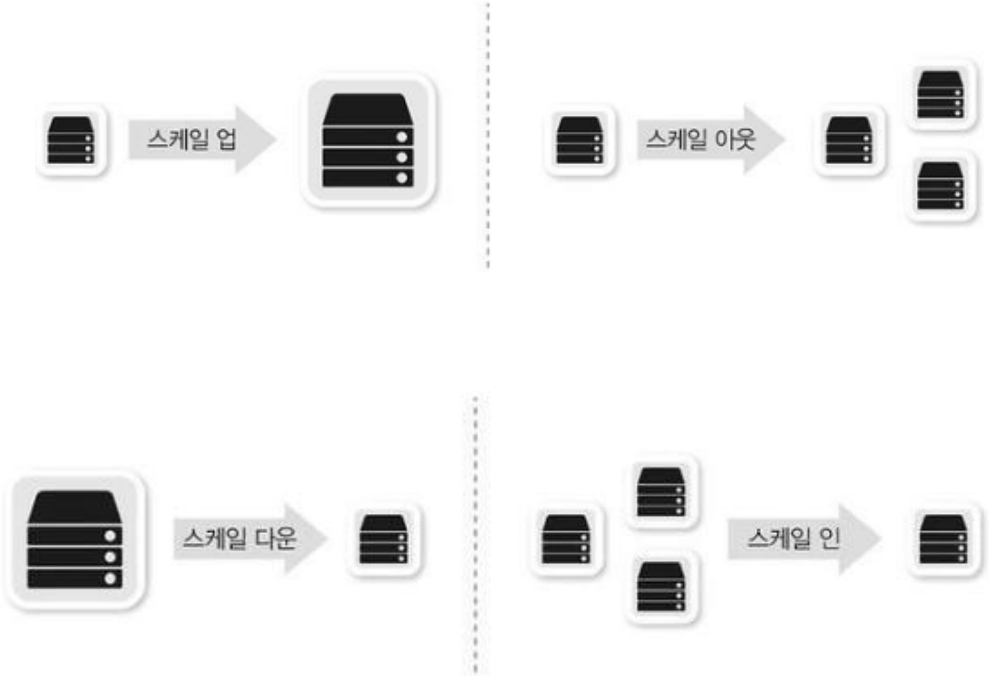


가상 서버가 리얼 서버로 트래픽을 분산하는 방법

- **라운드 로빈(Round Robin)** : 순차적으로 리얼 서버에 분산
- **IP 해시(Hash)** : 클라이언트의 IP 주소를 해시 함수의 변수로 활용하여, 고정된 리얼 서버로 분산
- **Least Connection(최소 연결)** : 클라이언트가 신규 연결 요청하면, 리얼 서버의 기존 연결 수를 분석해서 그 수가 가장 작은 서버로 분산

로드 밸런서

【참고】 Scale Up / Scale Out / Scale Down / Scale In



▼ 표 6-1 시스템 확장 방법인 스케일 업과 스케일 아웃의 장.단점 비교

	스케일 업(Scale-Up)	스케일 아웃(Scale-Out)
설명	하드웨어 성능 자체를 업그레이드하거나 더 높은 성능의 시스템으로 마이그레이션 하는 방법	여러 대의 서버로 로드를 분산하는 방법. 서비스 자체를 구분해 나누거나 같은 서비스를 분산해 처리하는 방법이 있다.
장점	부품을 쉽게 추가할 수 있으면 시스템 설계 변경 없이 서비스 사용량을 쉽게 늘릴 수 있다(주로 기존 대형 유닉스 시스템에서 사용함)	스케일 업 방식보다 더 적은 비용으로 시스템 확장이 가능하다. 여러 대의 시스템에 로드를 적절히 분산해 하나의 시스템에 장애가 발생하더라도 서비스에 미치는 영향이 없도록 결함허용(Fault Tolerance)을 구현할 수 있다.
단점	부품 추가가 어렵다(최근 x86), 시스템이 커질수록 비용이 기하급수적으로 증가한다.	스케일 아웃을 위해 별도의 복잡한 아키텍처를 이해하고 운영해야만 할 수 있다. 프로세스나 네트워크 장비가 추가로 필요할 수 있다.

방화벽

- 네트워크 중간에 위치해서 장비를 통과하는 트래픽을 조건에 따라 허용(Permit)하거나 차단(Deny)하는 장비
- 3, 4계층에서 동작하고 세션을 인지, 관리하는 SPI(Stateful Packet Inspection) 엔진을 기반으로 동작하는 장비

❖ 방화벽의 기본 정책은
- 인터넷으로 나가는 모든 패킷은 허용
- 내부로 들어오는 모든 패킷을 차단

❖ Session 정보를 관리하여,
- **통과하는 패킷이 내부에서 시작한 것인지**
- 외부에서 시작한 것인지를 판단하여 관리

※ 패킷 상태 정보를 인지하여 stateful로 동작하는 장비의 경우,

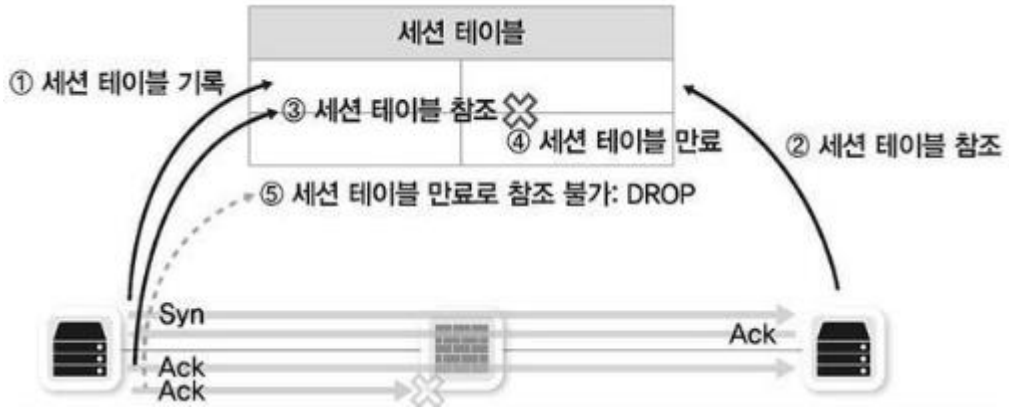
상태 정보를 갖고 있어 상태 테이블(State Table) 또는 해당 상태에 대한 세션 값을 유지하여 세션 테이블(Session table)이라고 부름

4계층 장비를 통과할 때 유의점 (세션 관리)

- 세션 테이블 유지, 세션 정보 동기화

- 세션 테이블의 세션 정보는 일정 시간만(**Session Timeout**) 유지
- 세션 테이블에 세션 정보가 없으면 방화벽은 패킷을 차단

【세션 테이블의 세션 만료 시간이 애플리케이션 만료 시간 보다 짧은 경우의 예】



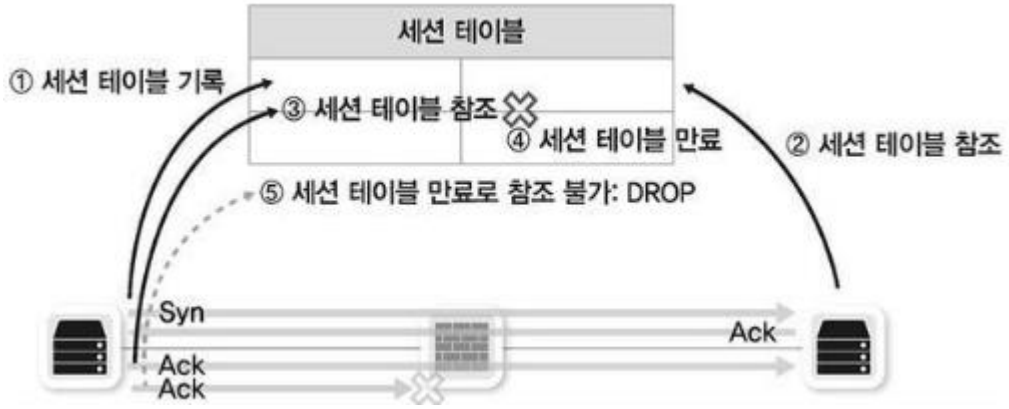
1. 3방향 핸드셰이크를 통해 정상적으로 세션 설정
   ① 방화벽에서 세션 설정 과정을 확인하고 세션 테이블 기록
2. ②, ③ 세션 테이블을 참조해 방화벽에서 패킷 통과
3. 일정 시간 동안 통신 없음
4. ④ 세션 타임으로 세션 테이블 만료
5. 세션 만료 후 애플리케이션 통신 시작
6. ⑤ 세션이 만료되어 방화벽에서 패킷 드롭

문제 해결 방안 ?

4계층 장비를 통과할 때 유의점 (세션 관리)

- 세션 테이블 유지, 세션 정보 동기화

【세션 테이블의 세션 만료 시간이 애플리케이션 만료 시간 보다 짧은 경우의 예】



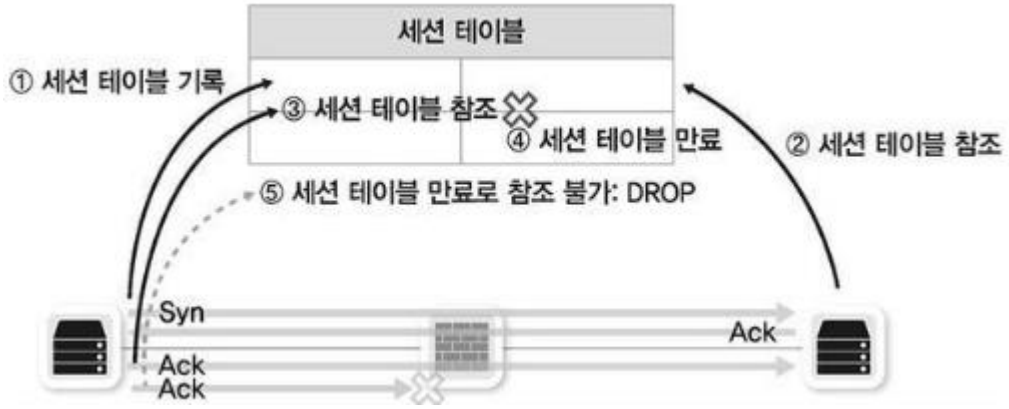
[문제 해결 방안] 세션 장비 운영자

- ❖ **세션 장비의 세션 만료 시간을** 애플리케이션의 세션 만료 시간에 맞추어 **늘려 준다**. – 애플리케이션 개발자와 협의
- ❖ 세션 테이블 **세션 정보가 없는 경우에도 패킷을 차단하지 않고 통과** 시킴(정책 변화) – 보안에 취약해짐
- ❖ 세션 만료시, 세션 장비는 양 종단에 세션 종료를 통보함. 양 종단 장비는 해당 세션을 끊고 필요시 재 설정함

4계층 장비를 통과할 때 유의점 (세션 관리)

- 세션 테이블 유지, 세션 정보 동기화

【세션 테이블의 세션 만료 시간이 애플리케이션 만료 시간 보다 짧은 경우의 예】



[문제 해결 방안] 개발자

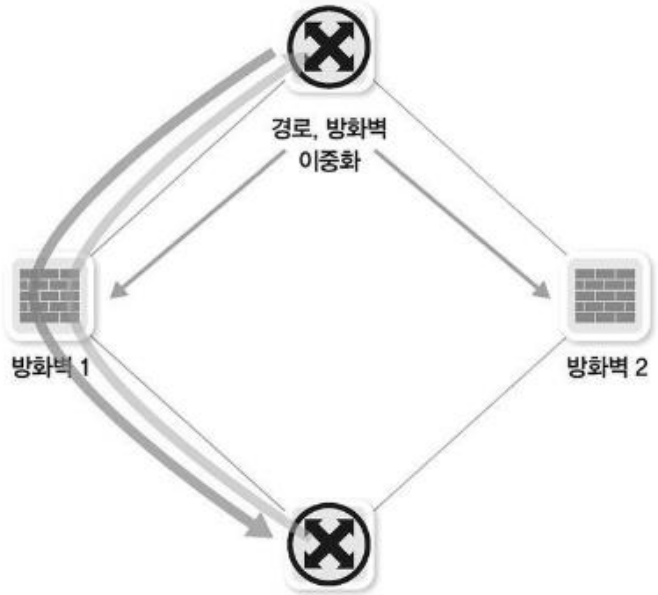
❖ 패킷을 주기적으로 보내 세션을 유지(Health Check)

4계층 장비를 통과할 때 유의점 (세션 관리)

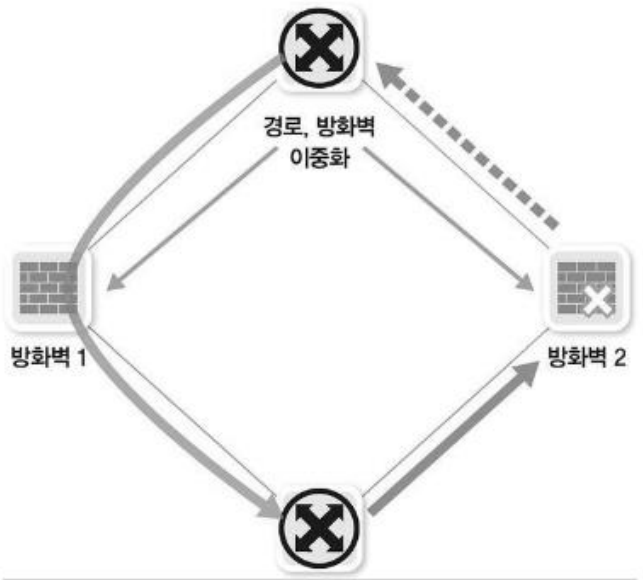
- 비대칭 경로 문제

- **대칭 경로(Symmetric Path)** : 인 바운드 패킷과 아웃 바운드 패킷이 같은 장비 통과
- **비대칭 경로(Asymmetric Path)** : 인 바운드 패킷과 아웃 바운드 패킷이 다른 장비 통과

❖ 네트워크 안정성을 위해 회선과 장비를 이중화 운영함으로 다른 장비를 통과할 수 있음



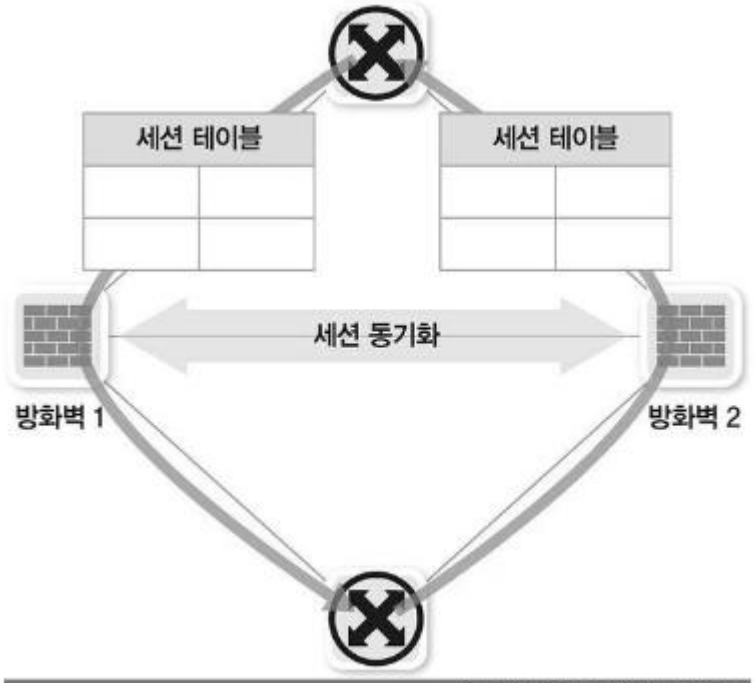
▲ 그림 6-15 대칭 경로. 인바운드, 아웃바운드 패킷이 한 장비를 통과해 통신에 문제가 없다.



▲ 그림 6-16 비대칭 경로. 인바운드 패킷과 아웃바운드 패킷이 한 장비를 통과하지 않아 세션 정보가 없어 패킷이 드롭된다.

4계층 장비를 통과할 때 유의점 (세션 관리)

- 비대칭 경로 문제 해결 방안(1)

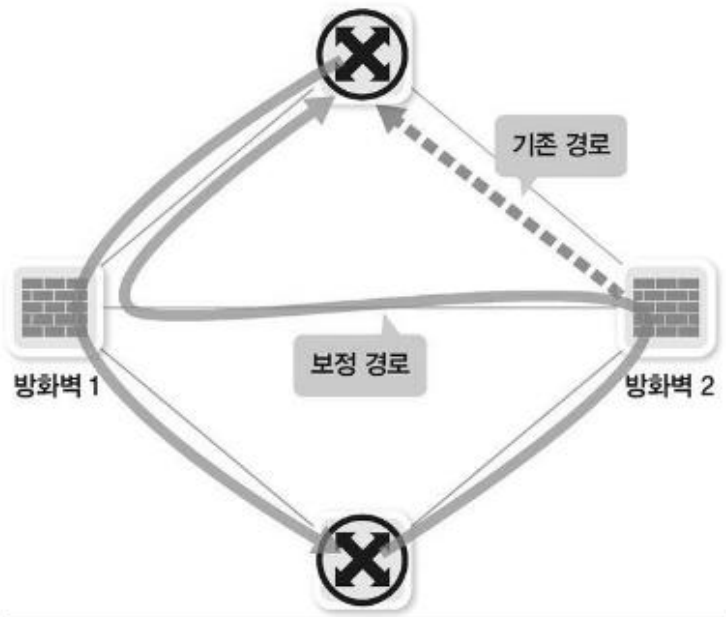


▲ 그림 6-17 세션 동기화 기능을 이용하면 비대칭 경로인 경우에도 통신이 가능하다.

- **세션 테이블을 동기화 함**
- (장점) 패킷 경로를 변경할 필요 없음
- (단점) 세션 동기화 시간보다 패킷 응답이 빠르면 오 동작
- 응답 시간이 비교적 긴 인터넷 게이트웨이 방화벽 사용시 유용

4계층 장비를 통과할 때 유의점 (세션 관리)

- 비대칭 경로 문제 해결 방안(2)

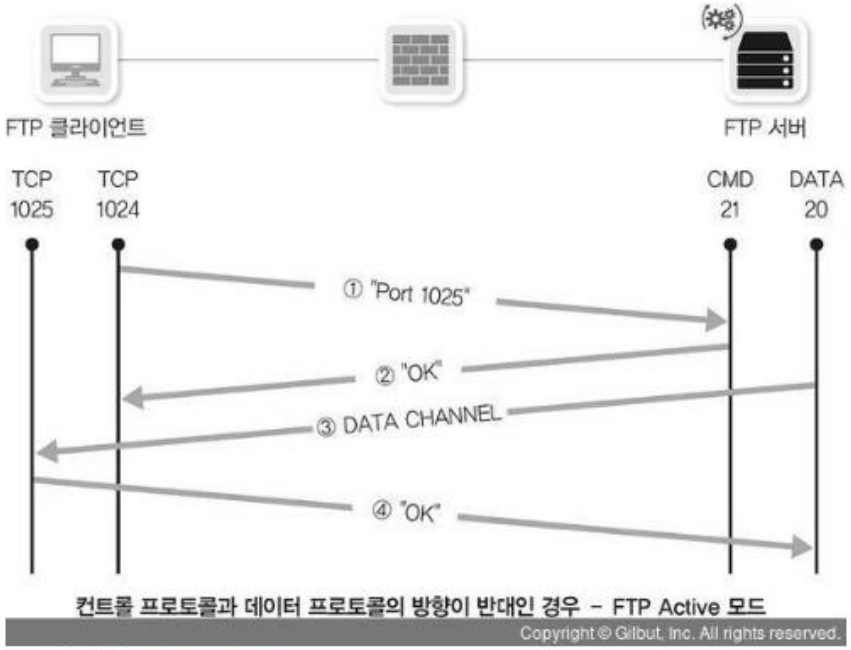


▲ 그림 6-18 경로 보정 기능(MAC 리라이팅, 터널링)으로 비대칭 경로를 예방할 수 있다.

- **인 바운드 패킷 통과 없이 아웃 바운드 패킷이 장비로 들어 온 경우**
- 인 바운드 패킷이 들어 온 다른 장비 쪽으로 패킷을 보내 경로를 보정함
- (단점) 다른 방화벽으로 패킷을 보내기 위한 방화벽간 통신용 링크 필요
- MAC 주소를 변경(Rewriting) 등의 방식으로 경로 보정

4계층 장비를 통과할 때 유의점 (세션 관리)

- 하나의 통신에 2개 이상의 세션이 사용되는 경우



▲ 그림 6-20 FTP Active 모드

1. 클라이언트가 FTP 서버에 접속. 클라언트는 1023번 이상의 TCP 포트를 사용, 서버는 TCP 21번 포트를 사용
2. ① 클라이언트가 서버에 데이터를 1025번 포트를 사용해 수신하겠다고 알림
3. ② 서버는 클라이언트에 1025번 포트를 사용해 송신하겠다고 응답
4. ③ 서버에서 데이터를 보냄, 클라이언트에서 응답하고 데이터를 수신

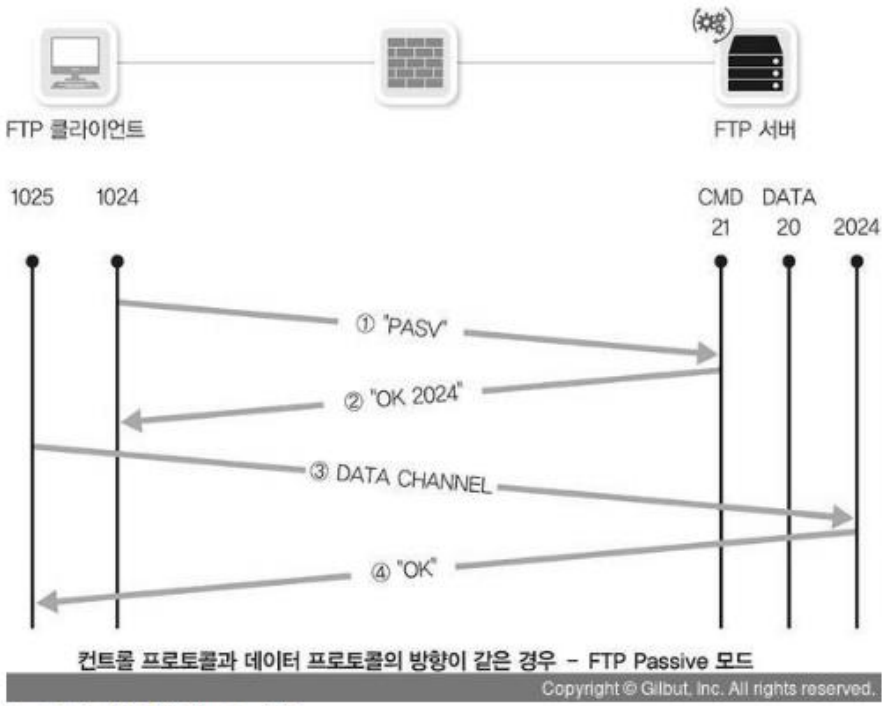
통상, 서버에서 방화벽 설정을 해주면
클라이언트에서는 서비스이용이 가능해야 함

❖ 클라이언트에 방화벽이 있는 경우 – 클라이언트에서도 방화벽 추가 설정이 필요하다는 문제
- 1025번 포트로 유입되는 패킷을 허용하도록 설정해 주어야 함

❖ 서버에 방화벽이 있는 경우
- 21번(FTP 서버) 포트로 유입되는 패킷을 허용하도록 설정해 주어야 함

4계층 장비를 통과할 때 유의점 (세션 관리)

- 하나의 통신에 2개 이상의 세션이 사용되는 경우



▲ 그림 6-21 FTP Passive 모드

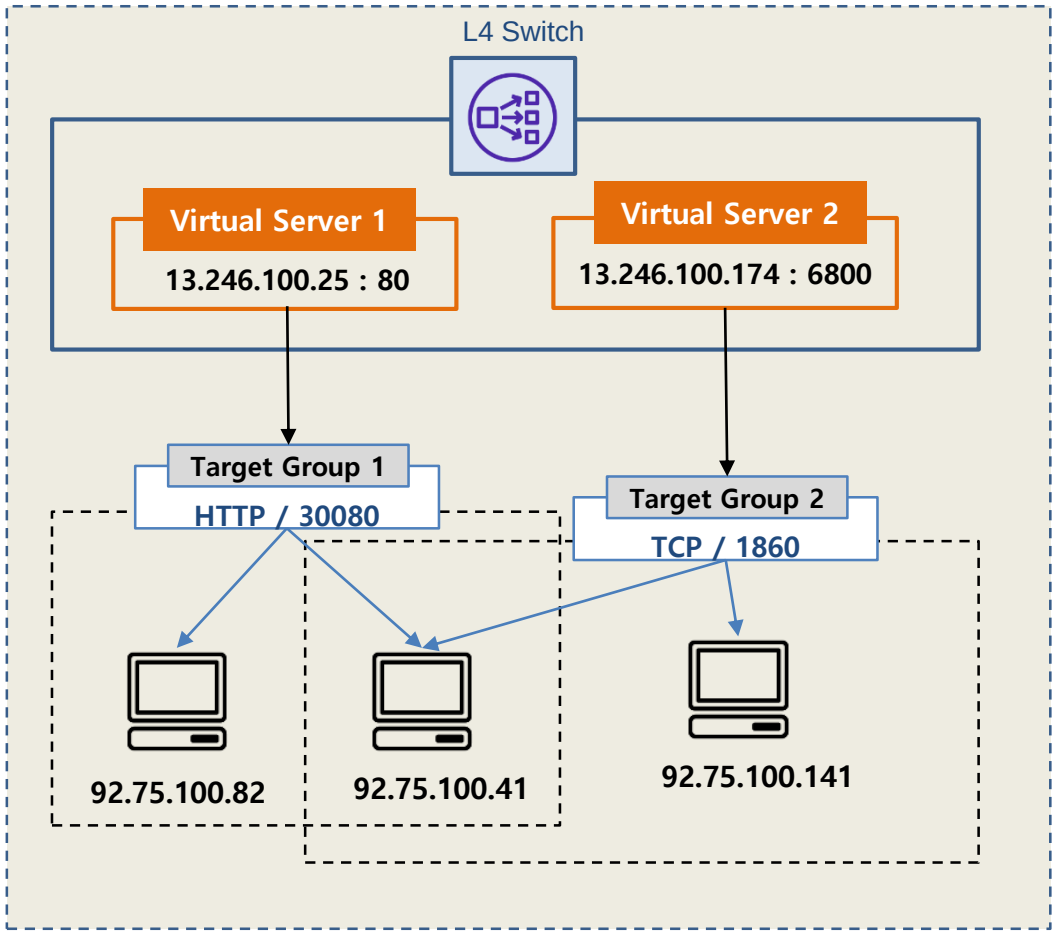
1. 클라이언트가 서버에 접속. 클라언트는 1023번 이상의 TCP 포트를 사용, 서버는 TCP 21번 포트를 사용(Active 모드와 동일)
2. ① 클라이언트가 Passive 모드를 사용하겠다고 알림
3. ② 서버는 클라이언트에 데이터 수신에 사용할 포트를 알림. 2024번 포트를 사용해 수신하겠다고 응답
4. ③ 클라이언트에서 서버에 데이터를 요청. ② 과정에서 서버에서 알려준 2024번 포트에 요청.
5. 데이터 전송

통상, 서버에서 방화벽 설정을 해주면
클라이언트에서는 서비스이용이 가능해야 함

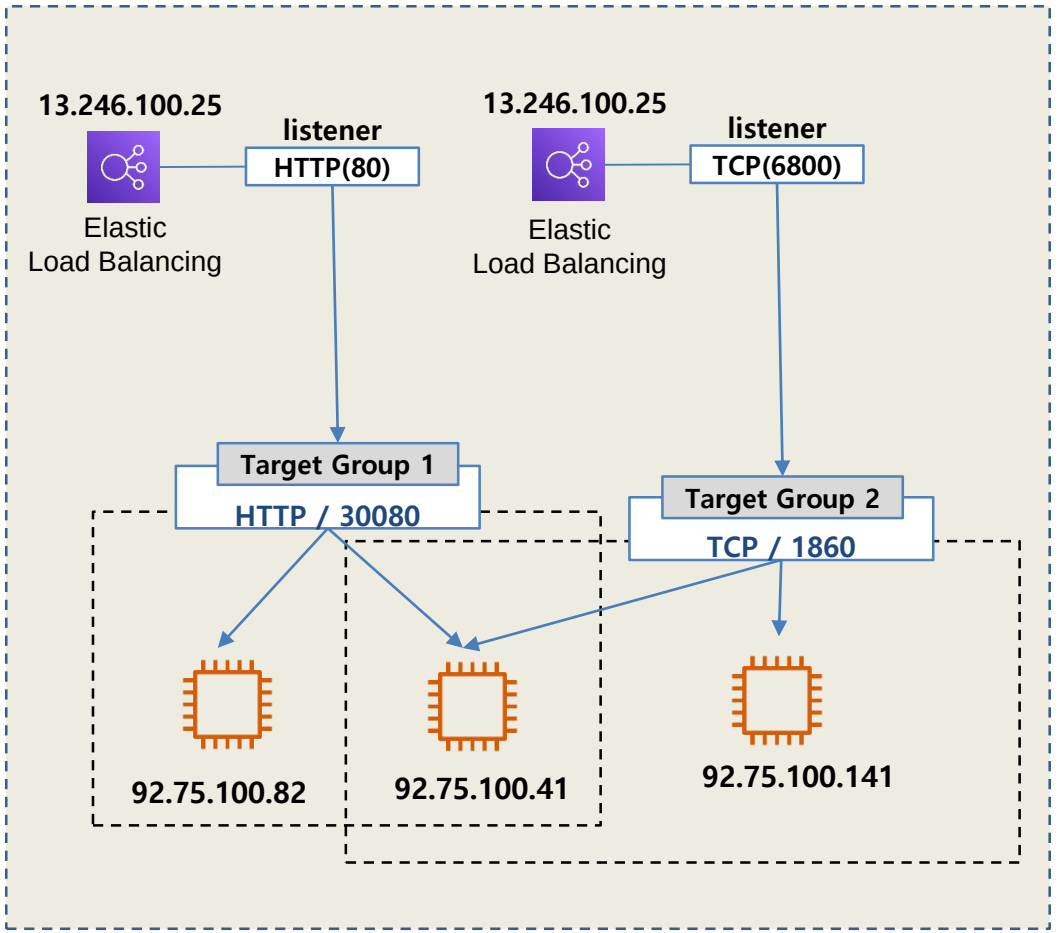
- ❖ 클라이언트에 방화벽이 있는 경우 : 클라이언트에서는 별도 설정 필요 없음
- ❖ 서버에 방화벽이 있는 경우
  - 21번(FTP 서버), 2024번 포트로 유입되는 패킷을 허용하도록 설정해 주어야 함
  - 통상, 서버에서는 데이터 전송을 위한 포트 범위를 설정

AWS 분산제어 : Load Balancing

On-Premise



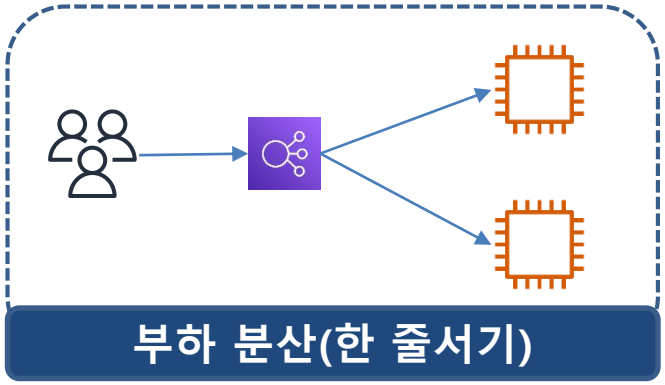
AWS



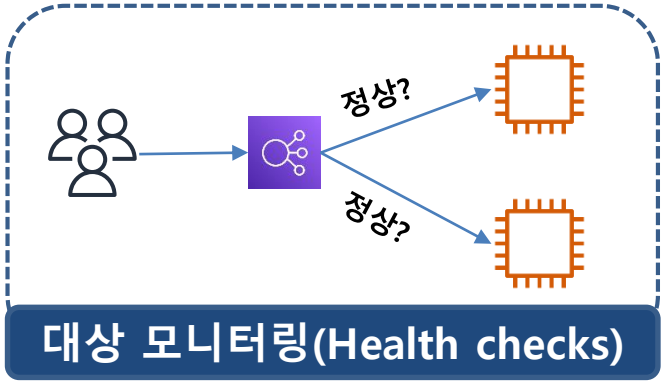
※ L4 switch 내부에 가상 서버를 여러 개 생성 가능

ELB (Elastic Load balancing) 기능

AWS에서 제공하는 로드 밸런서 서비스



- 사용자의 접속을 자동으로 분산
- 대량의 접속이 발생하는 경우, 부하를 각각의 Target으로 분산
- 각 Target을 서로 다른 가용영역(AZ)으로 배치할 경우, 가용성 향상

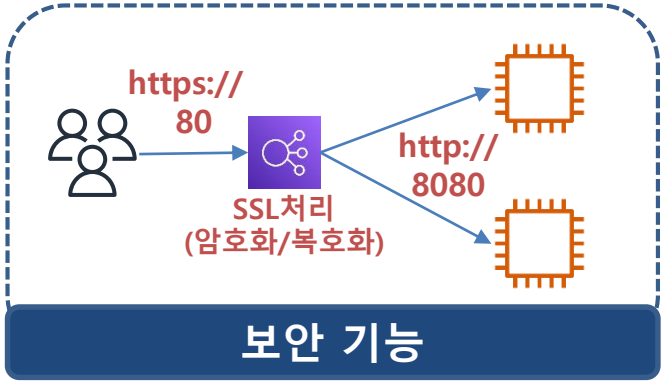


- ELB는 Target에 대해 연결과 상태를 감시하고 확인
- 감시를 통해 비정상적인 동작을 감지하면 대상을 자동으로 분리

(예) 웹서버의 정해진 경로로 보낸 요청이 지정된 횟수만큼 실패하면, 해당 웹 서버로 요청을 보내지 않음

ELB (Elastic Load balancing) 기능

AWS에서 제공하는 로드 밸런서 서비스



- AWS의 기본적인 보안 서비스 적용 가능
- HTTPS를 HTTP로 변환
  - SSL 암호화 및 복호화 처리를 ELB에서 수행
  - 웹서버의 부하를 줄이거나 인증서 관리 비용 줄임
- Port 번호 변환 (80 -> 8080)
  - Well known port(0~1023)를 통한 통신은 강력한 사용자 권한 필요
  - ELB 내부의 웹서버는 1024번 이상 포트 번호를 이용하여 일반 권한 사용자로 작동 시킴
- NLB(Network Load Balancer)는 보안 그룹 설정 불가

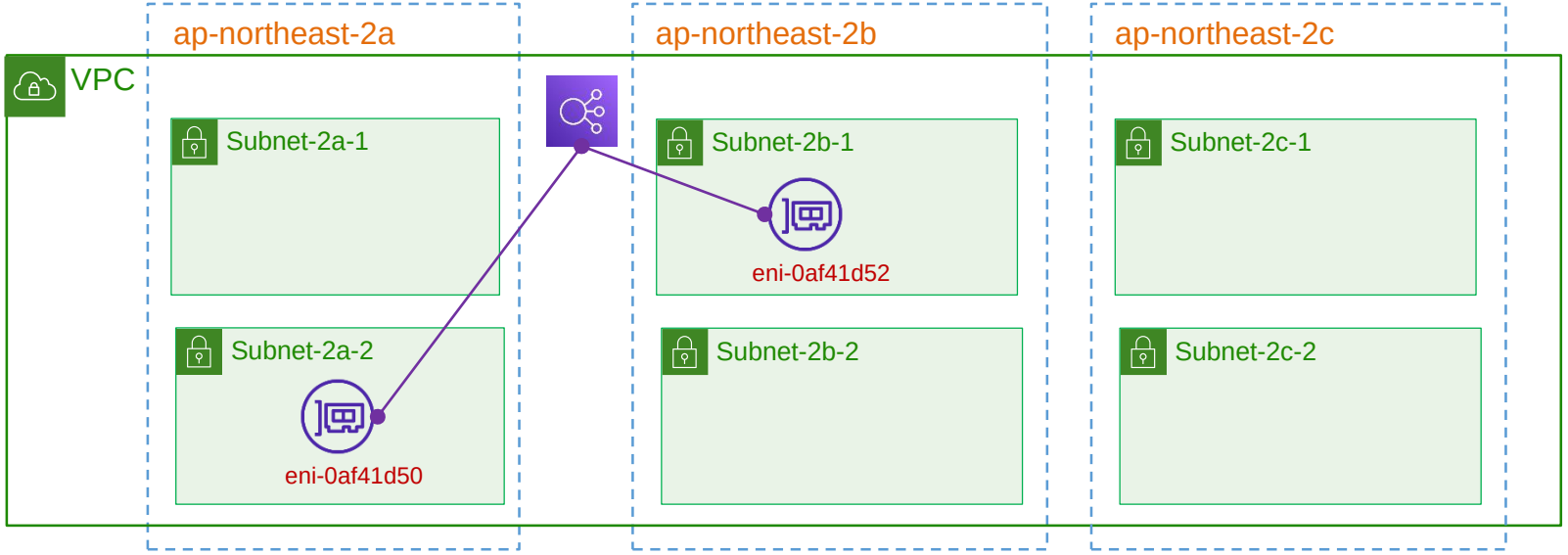
ELB (Elastic Load balancing) 유형

AWS에서 제공하는 로드 밸런서 서비스

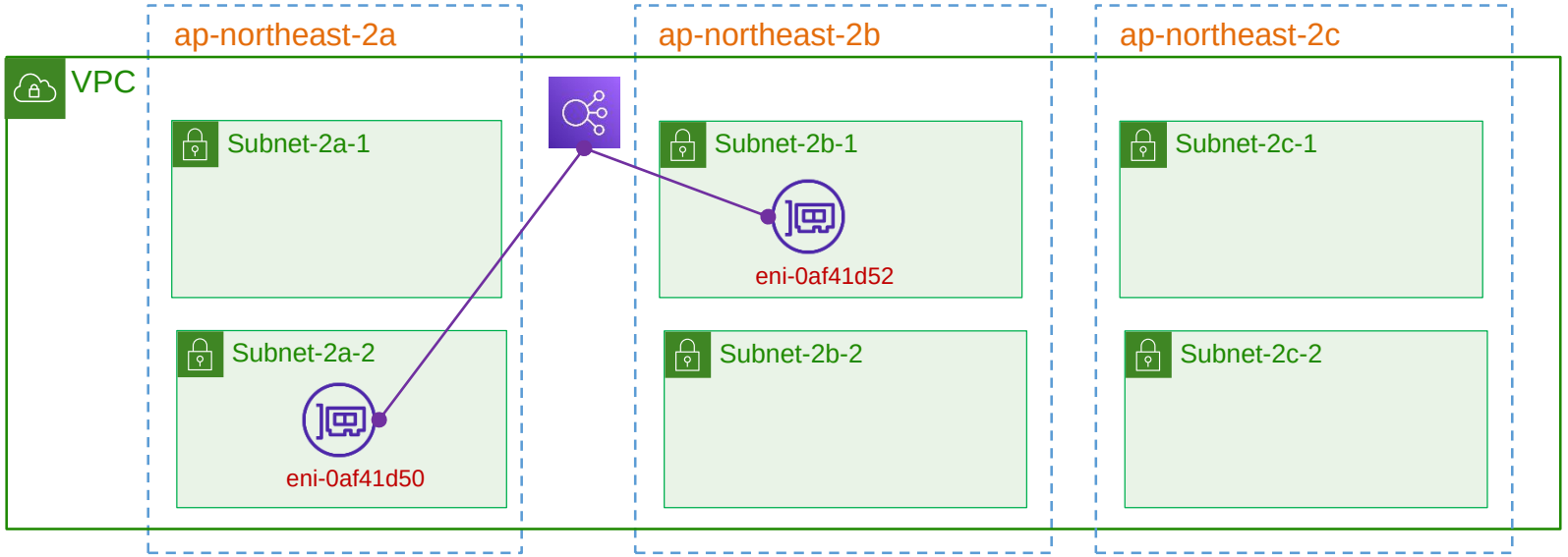
이름	설명
ALB (Application Load balancer)	• HTTP트래픽과 HTTPS 트래픽 부하분산 가능 • L7으로 동작하여, 마이크로 서비스나 컨테이너 등 다양한 응용프로그램에서도 대응 가능 • 사용 예 : 웹사이트, REST API를 제공하는 사이트
NLB (Network Load balancer)	• L4에서 동작하여, HTTP/HTTPS, TCP, UDP, TLS 트래픽 부하분산 가능 • 수백만 건 이상 요청이 발생하는 대규모 트래픽에서도 속도가 빠름 • 사용 예 : 게임, 채팅 등
CLB (Classic Load balancer)	• ALB, NLB 서비스 이전부터 제공되던 구형 로드 밸런서 • L4 및 L7에서 동작 • 예전 아키텍쳐 사용 등 특별한 경우를 제외하고는 ALB, NLB사용 권장
GWLB (Gateway Load balancer)	• AWS에서 제공하는 타사 보안 제품의 배포 및 관리 가능 • L3에서 동작

ELB (Elastic Load balancing) 특징

- **ELB 접속은 IP가 아닌 도메인으로 접속** – 하나의 장비가 아니라 AWS 리소스가 조합된 서비스
- **로드 밸런서 노드(Load Balancer Node) :** ELB 생성과 동시에 ELB용 ENI가 생성됨
- 노드 생성 위치는 ELB 생성시 선택한 가용영역의 서브넷임
- 가용영역별 1개 서브넷만 생성 가능 (**가용영역별 1개 노드**가 생성됨)
- ELB의 실제 역할은 노드가 수행함 (실제 각 가용 영역의 노드에서 로드밸런싱 수행)
- 클라이언트가 실제 접속하는 ELB IP는 노드의 IP
- ALB는 반드시 2개 이상의 가용 영역을 선택해야 함

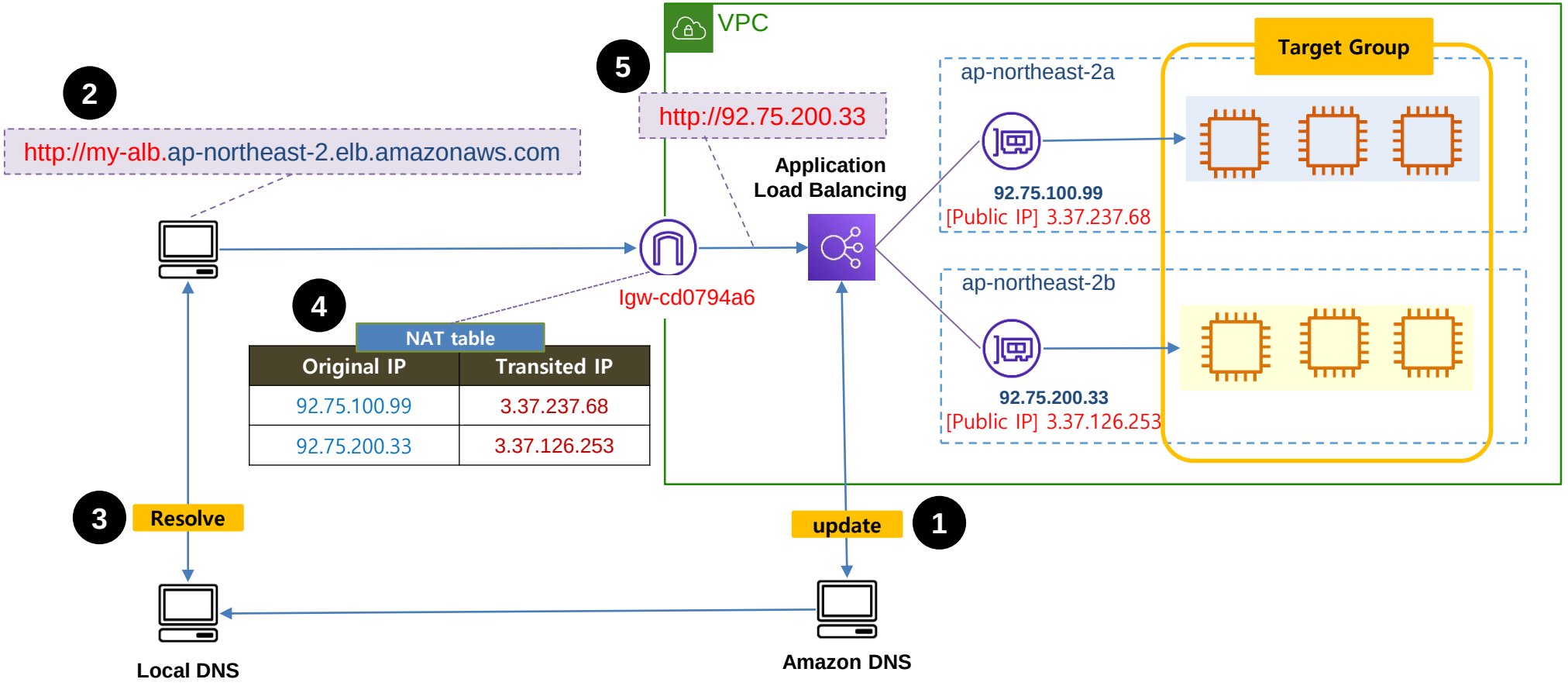


ELB (Elastic Load balancing) 특징

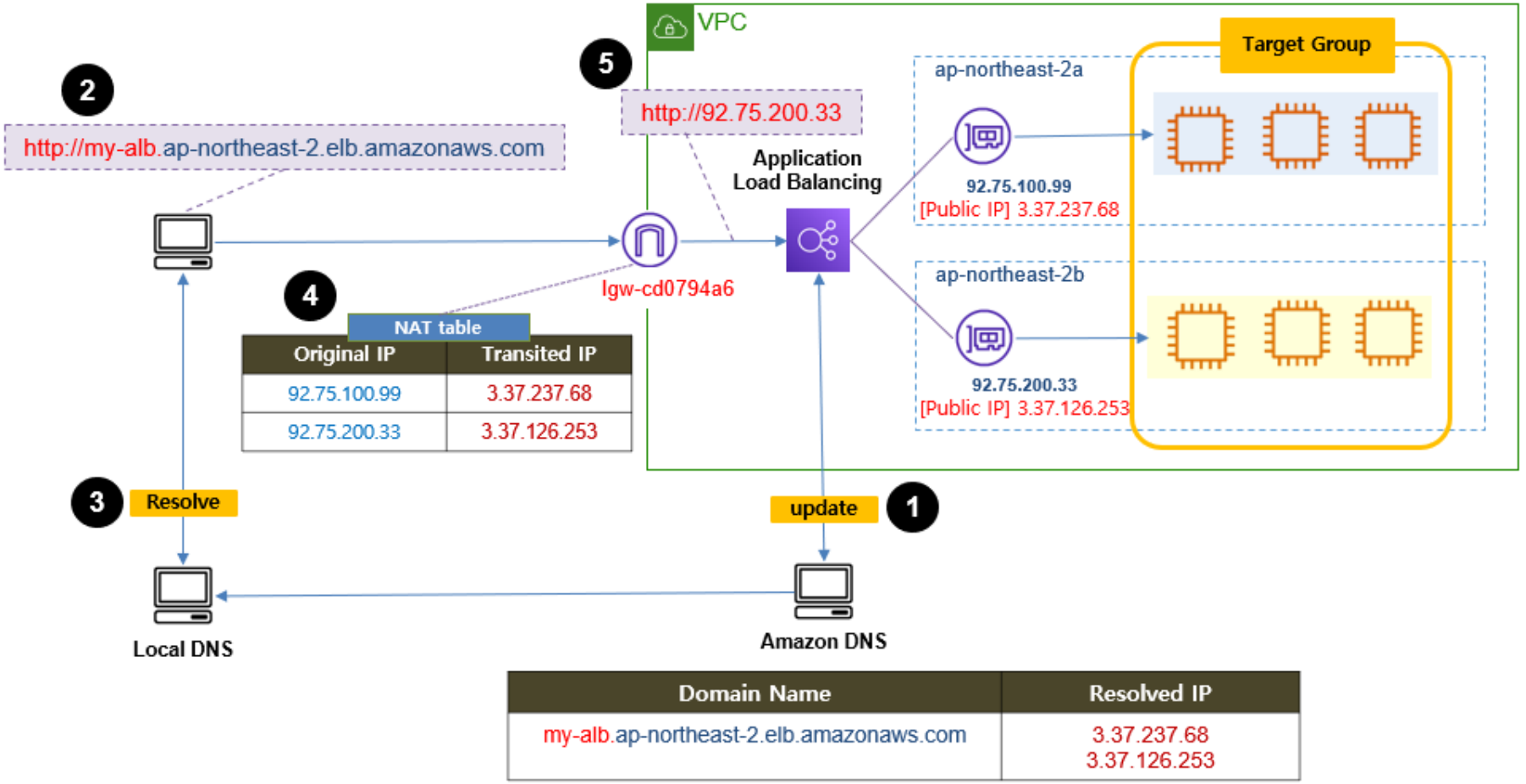


- ALB는 2개 이상의 가용영역을 유지해야 함 (ap-northeast-2a와 ap-northeast-2b 의 2개 가용 영역을 선택)
- 각 가용 영역에서 노드 **(Load Balancer Node)**가 설치될 subnet을 하나씩 선택 (Subnet-2a-2, Subnet-2b-1 선택)
  - ① 각 가용영역에 1개씩 설치된 노드가 로드밸런싱을 수행함
  - ② 선택하지 않은 가용영역 ap-northeast-2c로의 노드밸런싱은 불가능
  - ③ Subnet-2a-1에 놓은 인스턴스로 로드밸런싱은 가능함
- NLB는 최소 1개 이상의 가용 영역만 유지해도 됨

ELB (Elastic Load balancing) 동작 시나리오

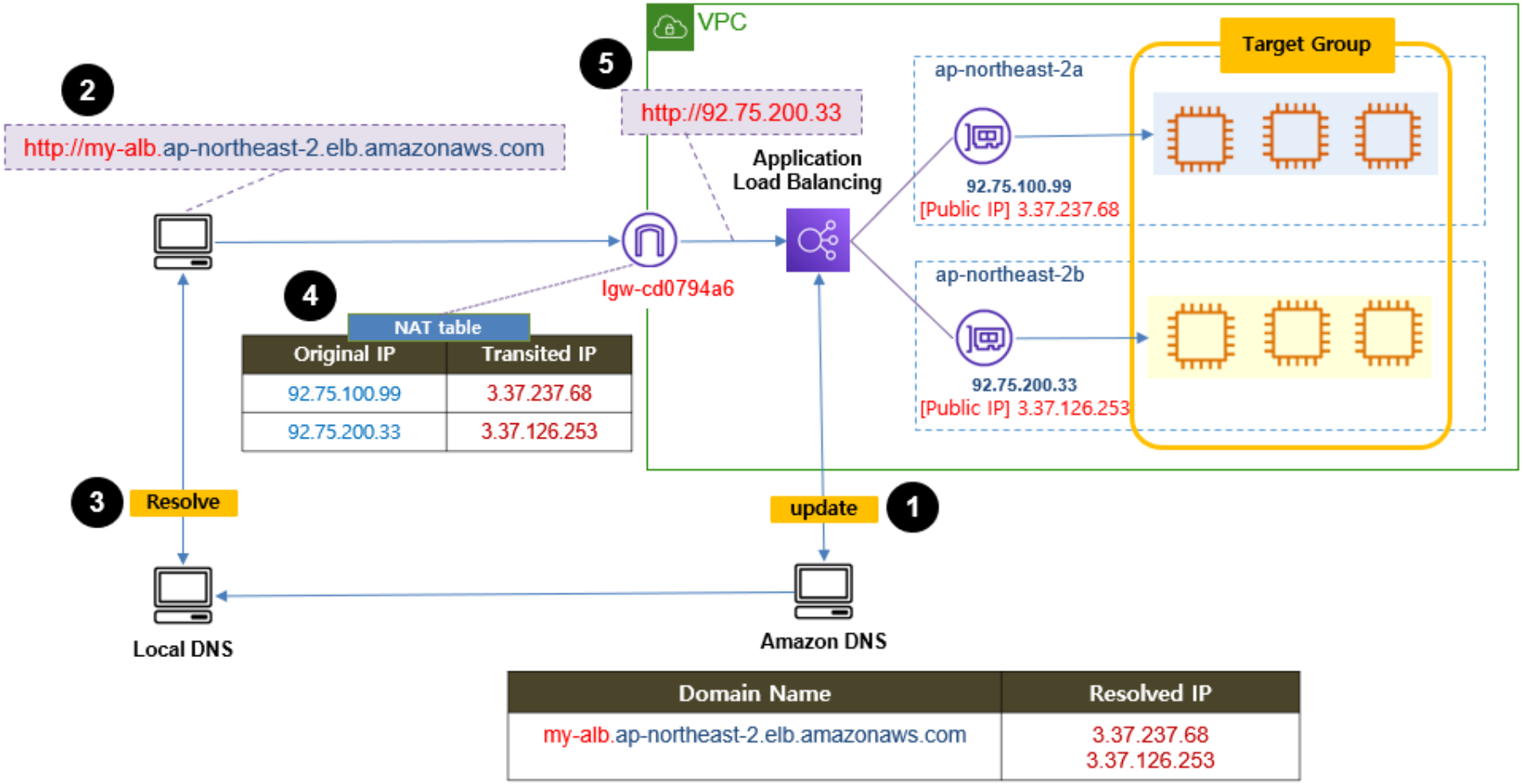


Domain Name	Resolved IP
my-alb.ap-northeast-2.elb.amazonaws.com	3.37.237.68 3.37.126.253



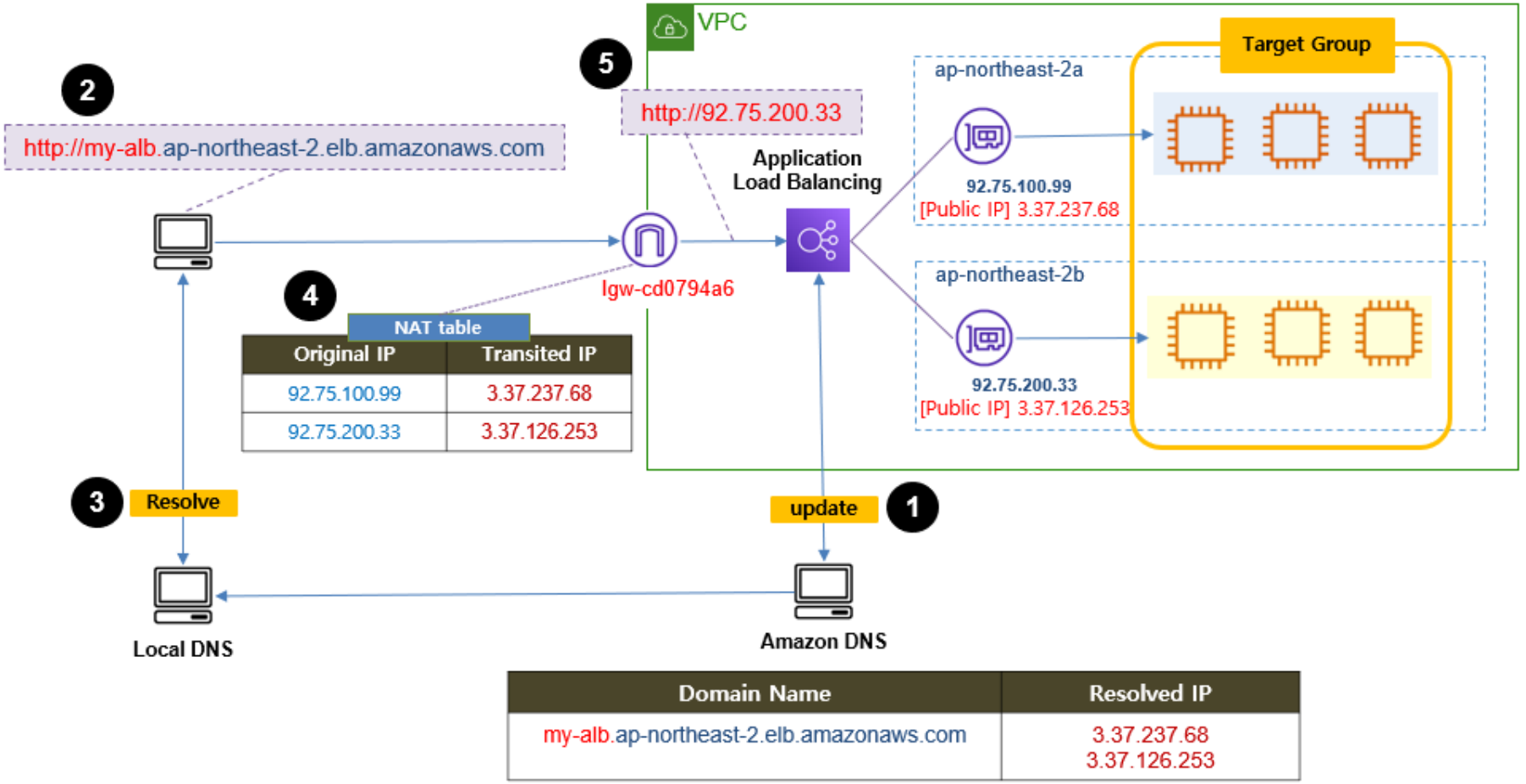
1

- ALB를 생성하고, 노드가 설치될 가용영역과 subnet을 선택하면 가용영역에 하나의 노드가 생성됨
- 노드가 생성되거나 변동되면 Amazon DNS에 등록(레코드가 갱신)
- ALB 도메인 이름에 해당하는 노드의 IP 주소 레코드가 등록됨
  - ❖ 도메인 이름 : my-alb.ap-northeast-2.elb.amazonaws.com



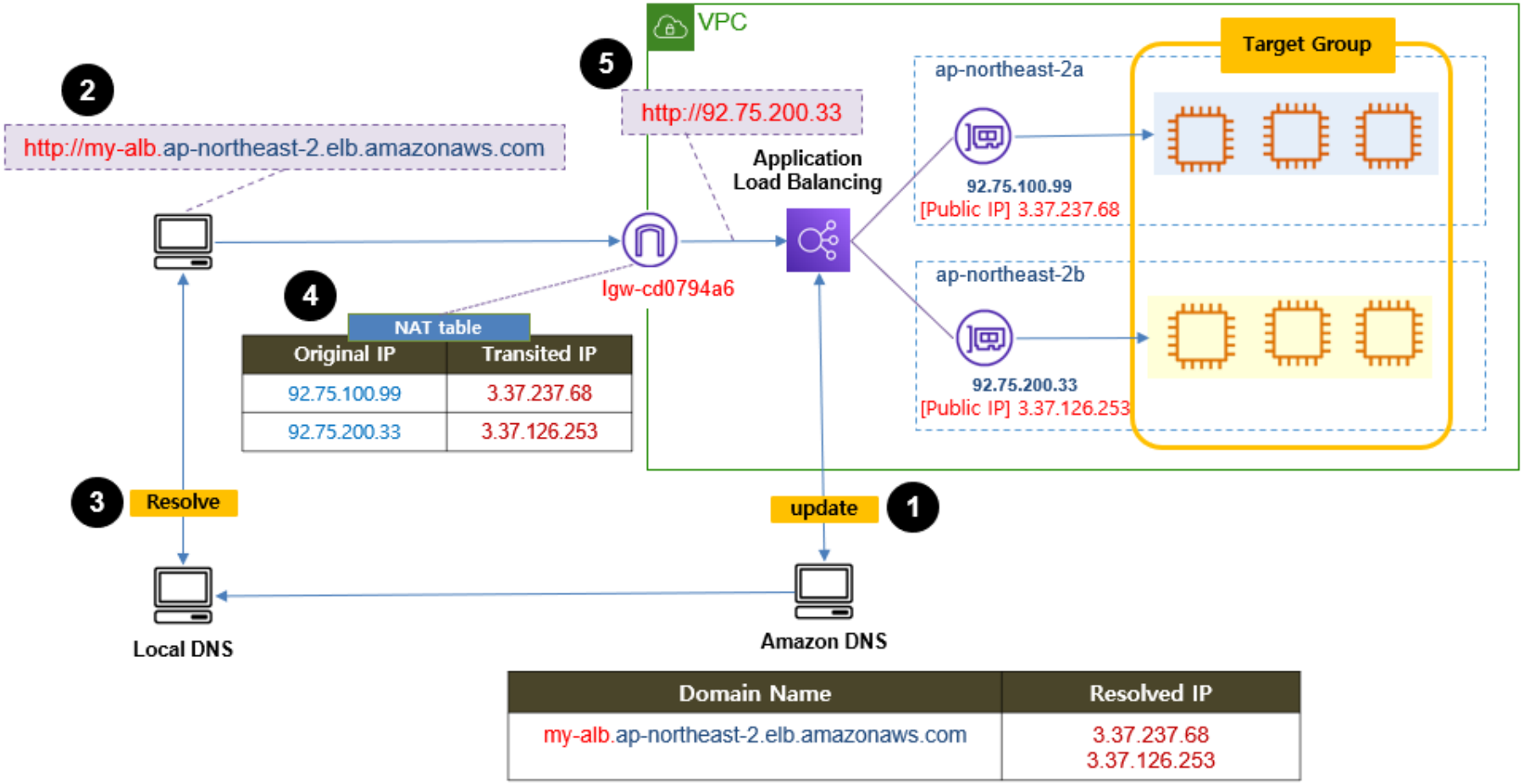
2 • 클라이언트가 ALB의 DNS 이름으로 웹 접속을 요청
- ❖ http://my-alb.ap-northeast-2.elb.amazonaws.com

3 • 클라이언트는 DNS에 도메인을 쿼리하여 IP를 불러옴
- ❖ 이때 ELB가 사용하는 2개 노드의 IP가 번갈아 가며 선택된다.



4

- 가용영역 2b의 노드가 선택되면 http://3.37.126.253으로 접속한다.
- 이 웹 접속은 IGW의 NAT 테이블을 참조하여 IP를 변환
  - ❖ http://92.75.200.33

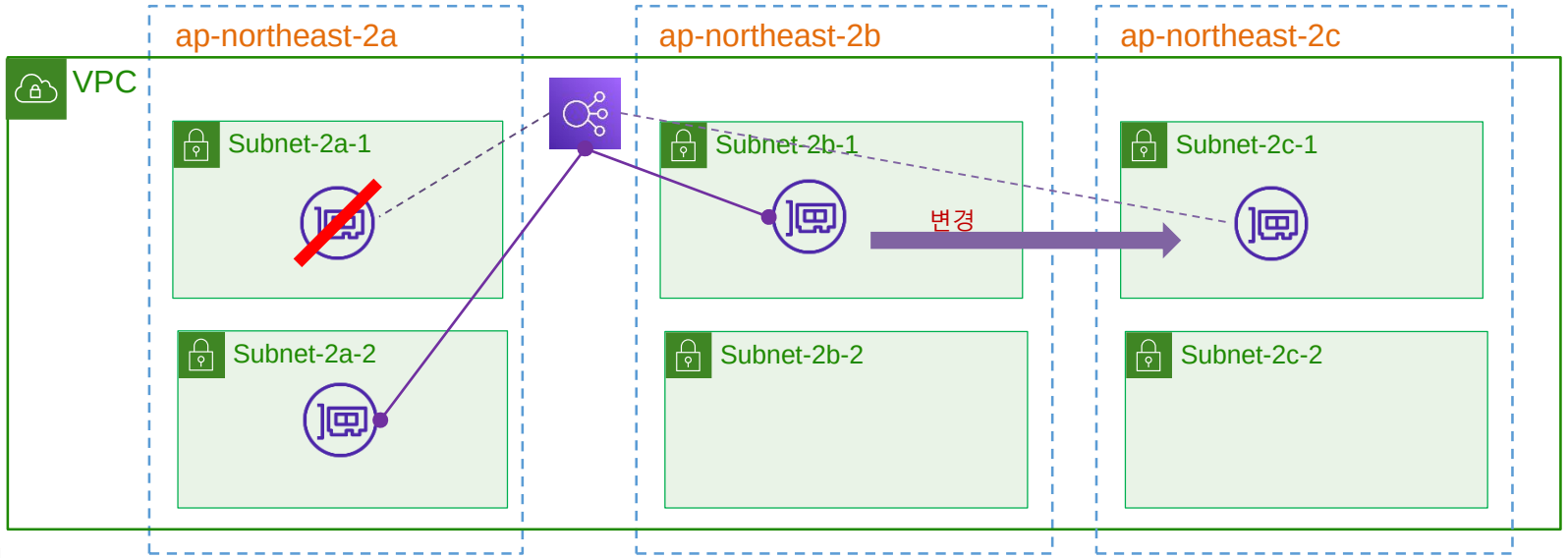


5

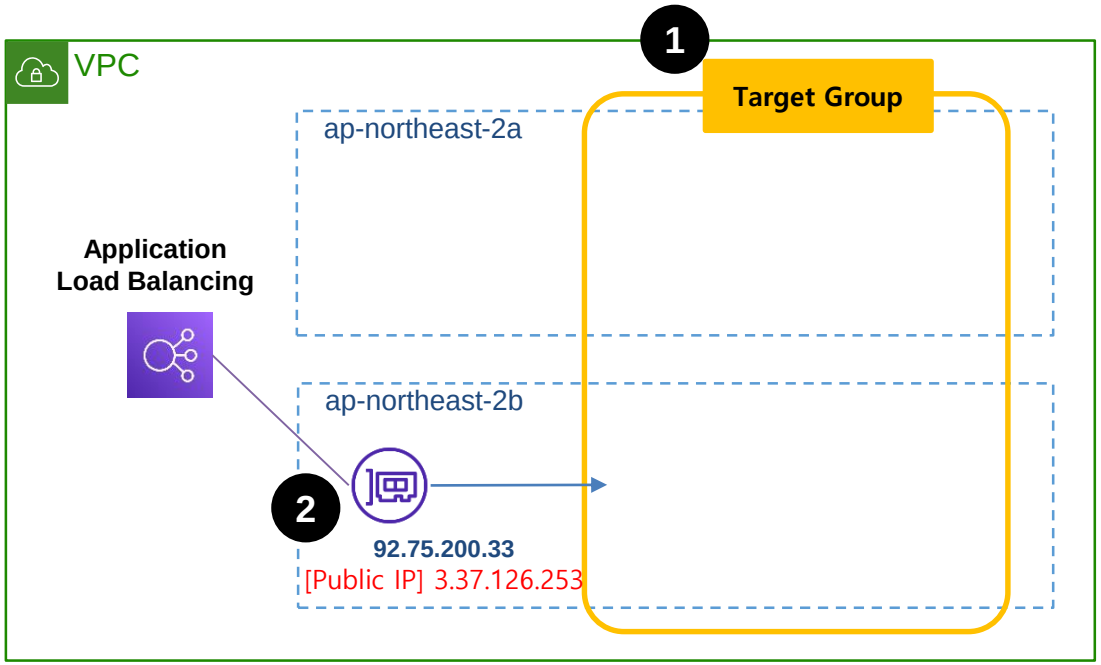
- 가용영역 2b에 있는 노드 (http://92.75.200.33)에서 웹 접속 요구를 받아
- Target Group으로 로드밸런싱 한다.

ALB와 NLB

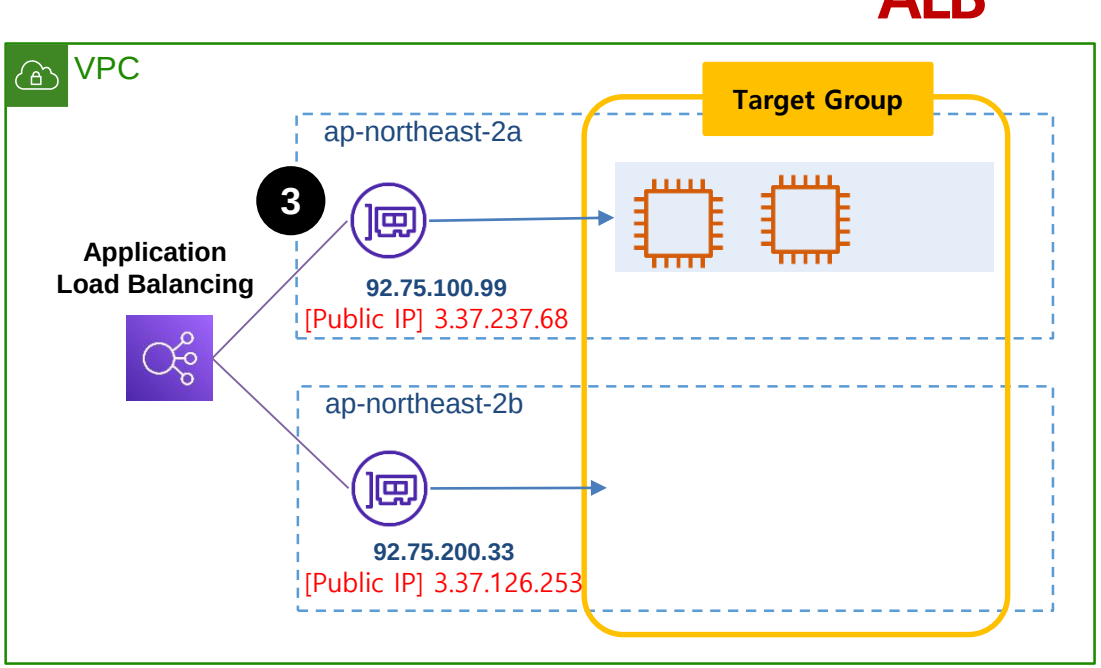
Domain Name	ALB	NLB	CLB	GWLB
가용영역 선택	최소 2개	최소 1개		
가용영역별 선택 가능한 Subnet 수	1개 (가용 영역별 1개의 노드만 생성 가능)			
가용영역 범위	추가, 변경, 삭제 가능	추가만 가능	추가, 변경, 삭제 가능	변경 불가
노드에 EIP 연결 가능	불가	가능	불가	불가



ALB와 NLB



ALB



1
- ALB 생성시 2개의 가용영역을 선택(2a, 2b)
- ALB에 대상이 없는 대상그룹(target group)을 연결

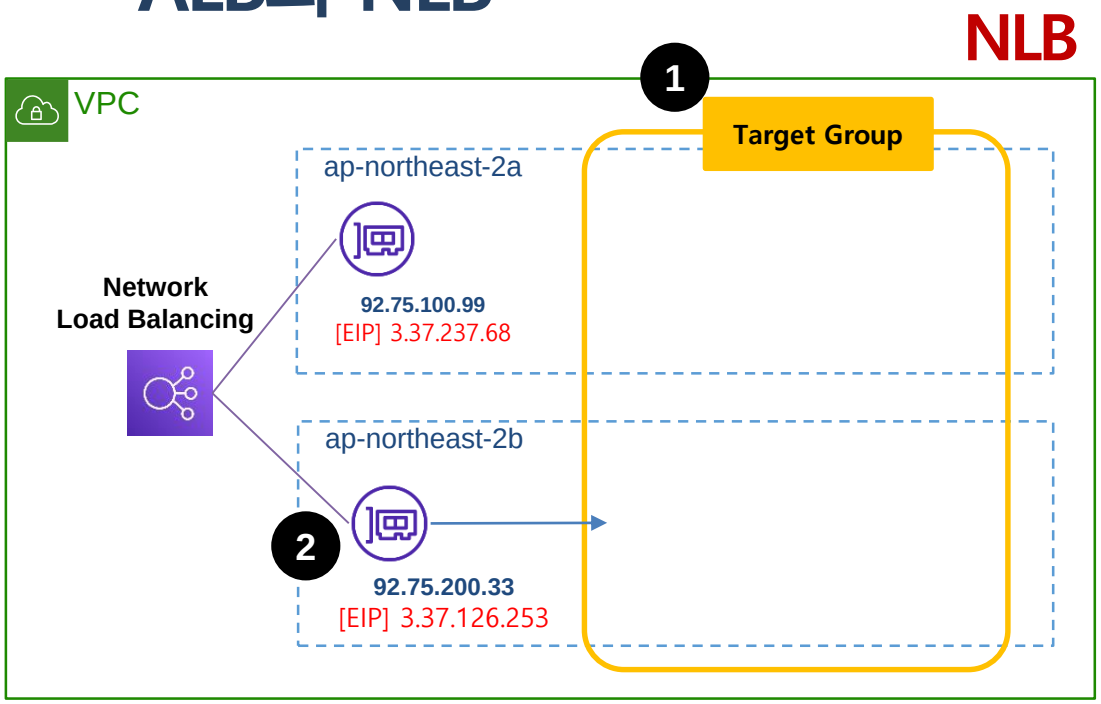
2
- ALB는 노드를 모든 가용영역에 생성하지 않고, 랜덤하게 한곳에만 생성 (예: 가용영역 2b에 생성)

3
- 로드밸런싱 대상 인스턴스가 가용영역 2a에 등록되면
- 가용영역 2a에 노드를 추가 생성함
- 기존 대상 인스탄스가 중지되면, 노드를 제거하기도 함

❖ ALB는 노드를 가변 적으로 운영함

ALB는 노드에 고정 public IP(EIP)를 할당하지 않고 자동할당 public IP를 할당함

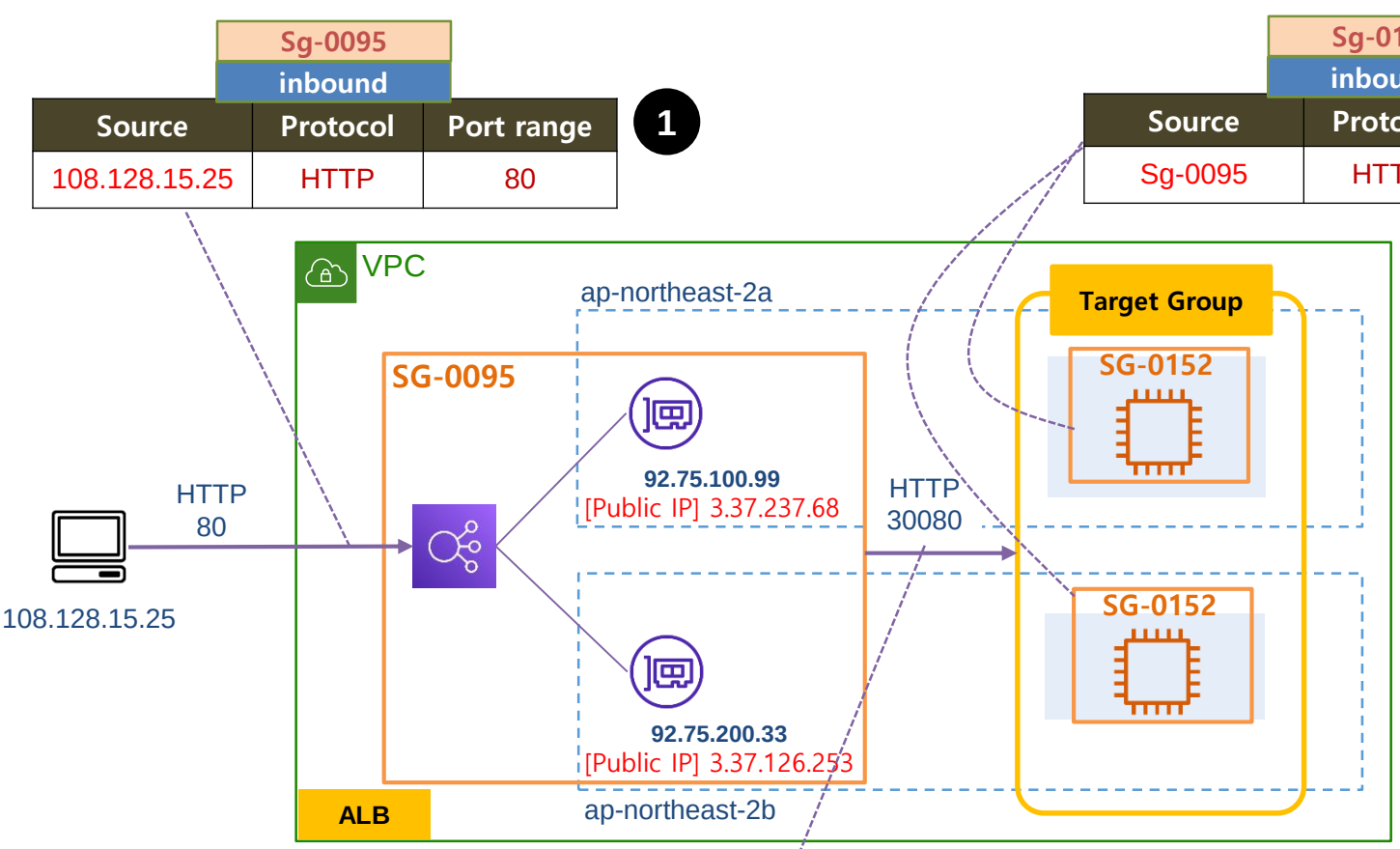
ALB와 NLB



1 • NLB 생성시 2개의 가용영역을 선택(2a, 2b)
• ALB에 대상이 없는 대상그룹(target group)을 연결
2 • NLB는 로드밸런싱 대상의 유무와 상관없이 모든 가용영역에 노드를 생성함

NLB는 노드에 고정 public IP(EIP)를 할당함

ELB 보안 그룹(SG: security group) 설정 - ALB

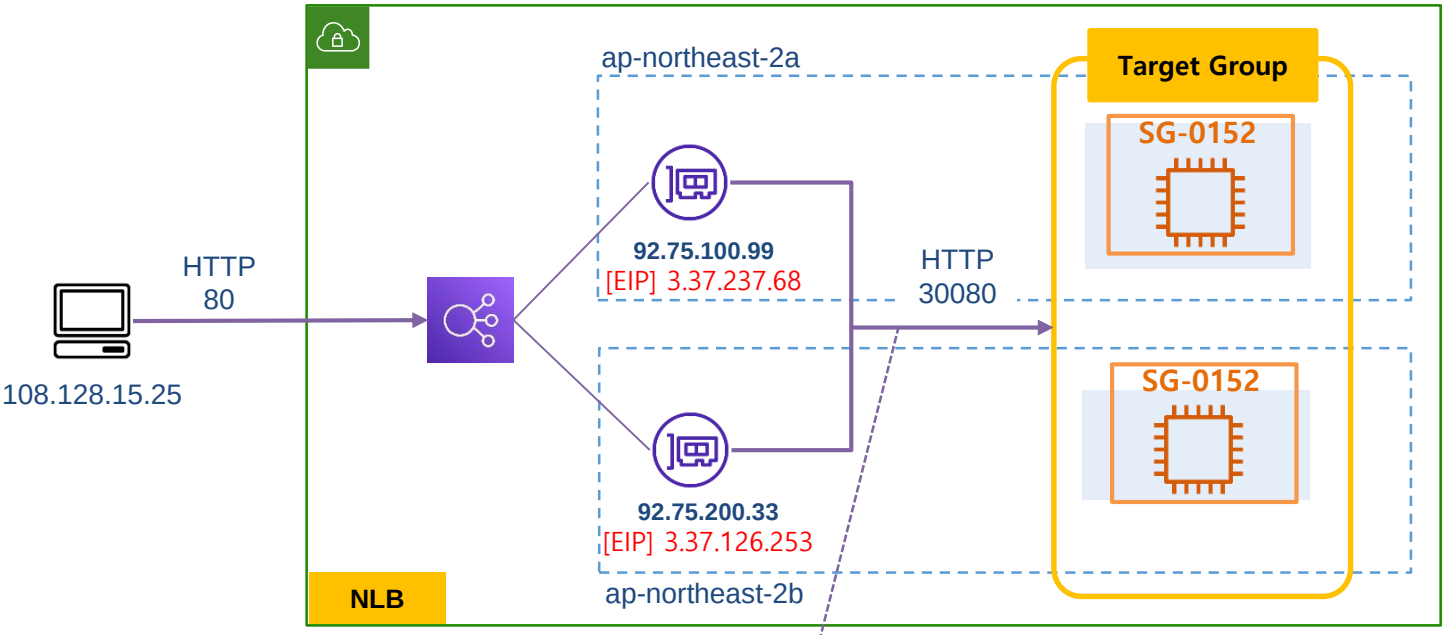


1 Inbound :
- 클라이언트의 IP와 ALB 리스너 프로토콜 접근 허용

2 outbound :
- ALB가 로드밸런싱할 대상 IP와 프로토콜
- 모든 목적지 IP 지정 보다는 동일 보안그룹 적용 대상(sg-0125) 지정(효과적)

3 inbound :
- ALB가 노드의 IP를 일일이 지정하기 보다는 ALB가 사용하는 SG(SG-0095)를 지정

ELB 보안 그룹(SG: security group) 설정 - NLB



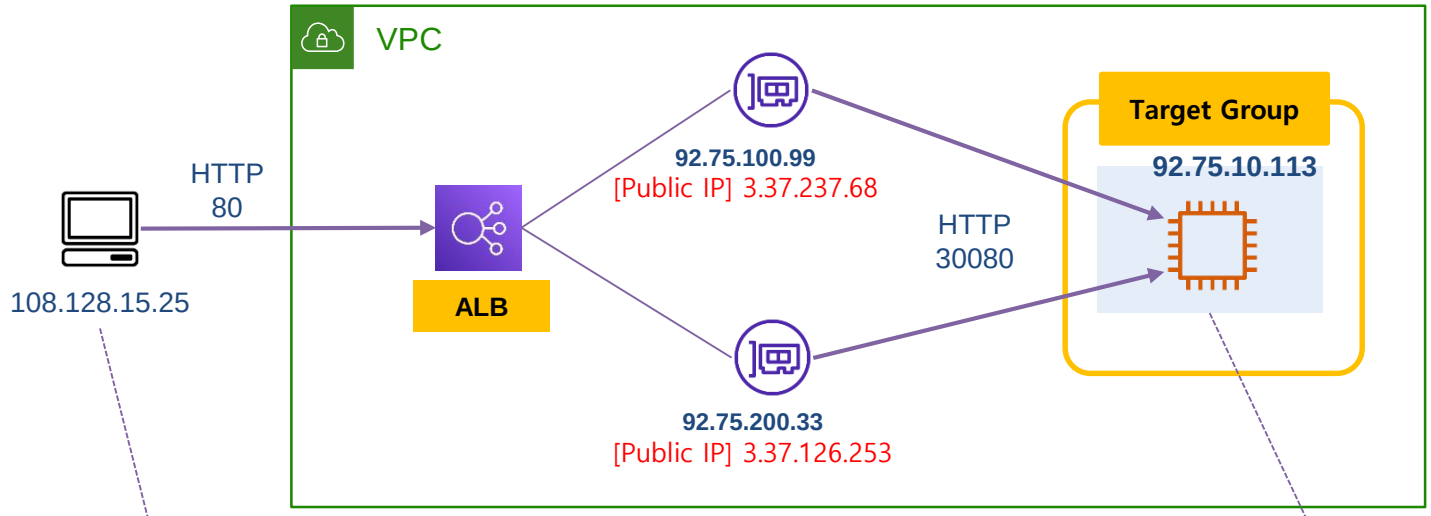
Sg-0152 inbound 1

Source	Protocol	Port range
108.128.15.25	HTTP	30080

1 **Inbound :**

- NLB는 sg를 사용하지 않음
- '클라이언트 IP 보존' 설정하면, 대상 인스턴스는 인바운드 트래픽의 발신 IP로 클라이언트 IP로 받을 수 있음
- '클라이언트 IP 보존' 설정하지 않으면, 일일이 NLB 노드 IP를 기술해야 함

ALB 라우팅 세션



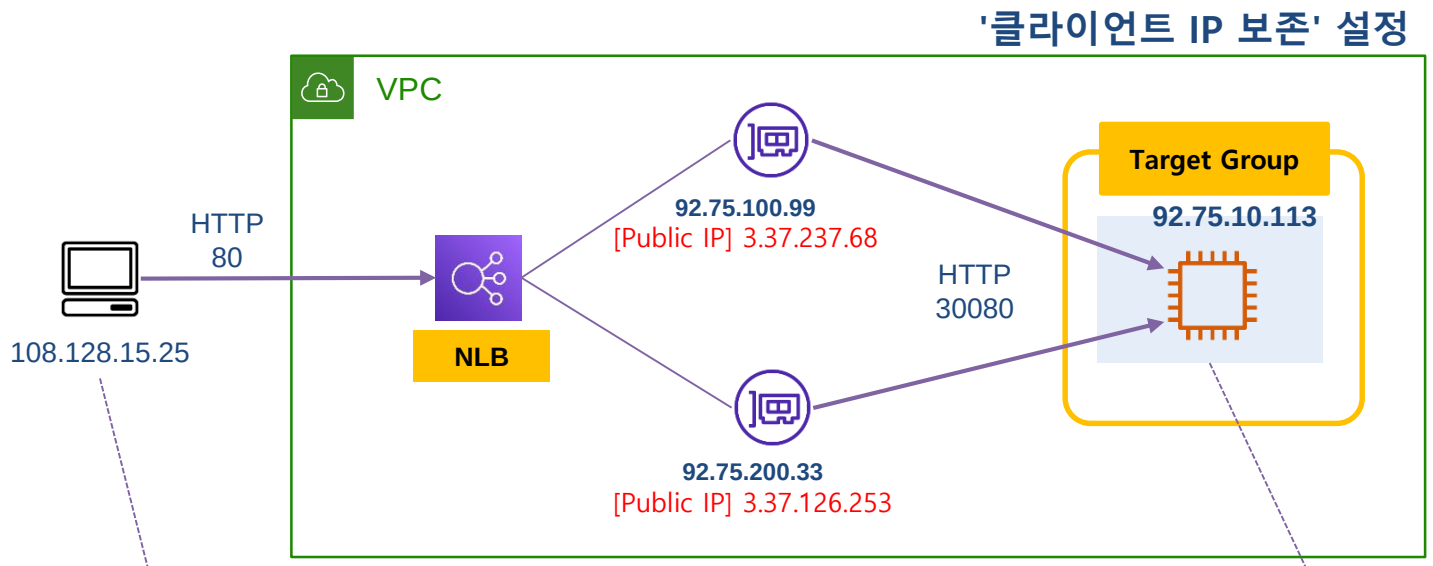
> netstat -an

protocol	Local	Foreign
TCP	108.128.15.25:50025	3.37.237.68:80
TCP	108.128.15.25:50037	3.37.237.253:80

> netstat -an

protocol	Local	Foreign
TCP	92.75.10.113:30080	92.75.100.99:43376
TCP	92.75.10.113:30080	92.75.200.33:43611

NLB 라우팅 세션



> netstat -an

protocol	Local	Foreign
TCP	108.128.15.25:50025	3.37.237.68:80
TCP	108.128.15.25:50037	3.37.237.253:80

> netstat -an

protocol	Local	Foreign
TCP	92.75.10.113:30080	108.128.15.25:50025
TCP	92.75.10.113:30080	108.128.15.25:50037