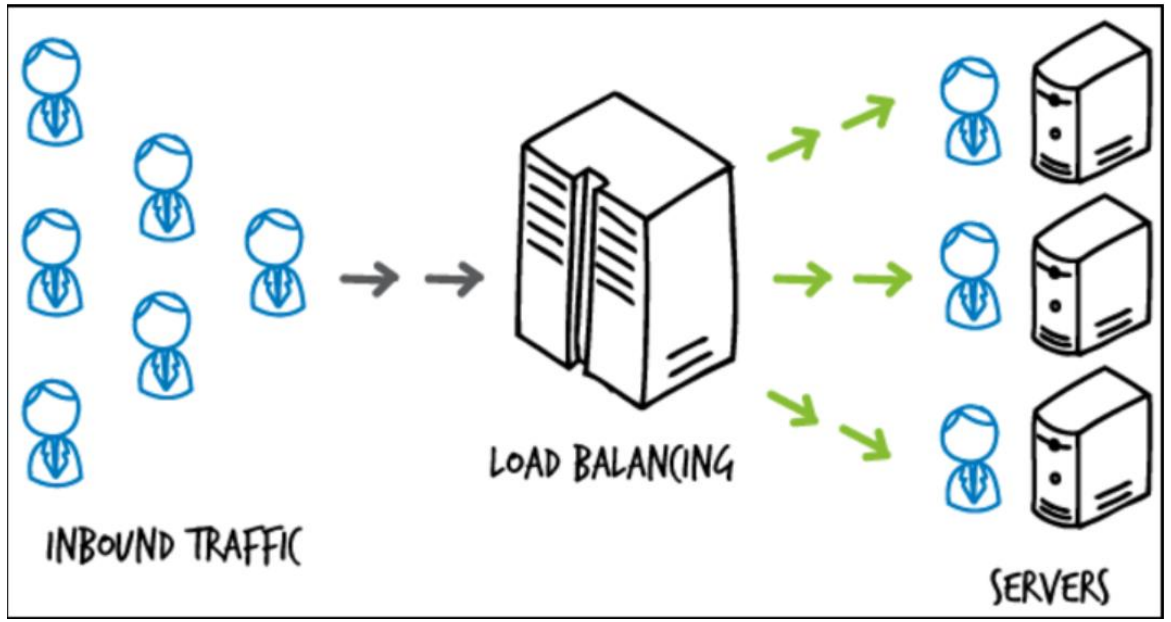
AWS Elastic Load Balancing

Jae Hyeon Kim

Load Balancing

- 부하 분산(Load Balancing)이란 처리해야 할 업무 혹은 요청 등을 나누어 처리하는 것
- VLAN을 이용한 Layer-2 Load Balancing
- Routing Protocol을 이용한 Layer-3 Load Balancing
- Server Load Balancing

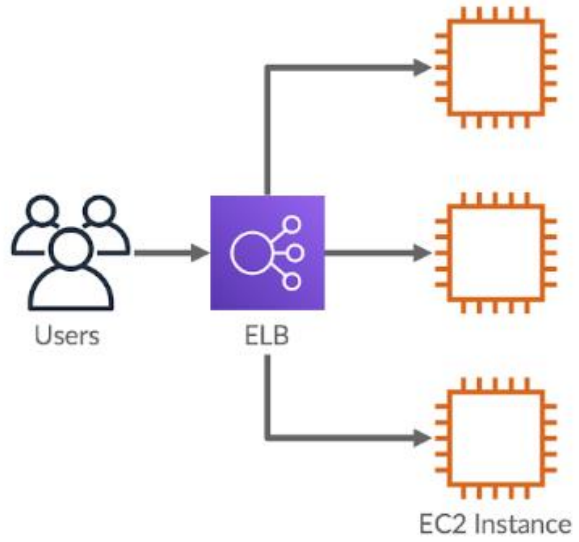


Load Balancing

- 서버 부하 분산을 담당하는 Network Switch를 L4/L7 Switch
- Cloud 환경에서는 Load Balancer

Elastic Load Balancing(ELB)

- 애플리케이션 트래픽을 Amazon EC2 인스턴스, 컨테이너, IP 주소, Lambda 함수와 같은 여러 대상에 자동으로 분산
- On-premise의 L4 switch처럼 부하 분산 뿐만 아니라 분산 대상에 대한 Health Check, 고정 세션(Sticky), SSL Offload(SSL 암복호화)등을 수행



ELB 이점

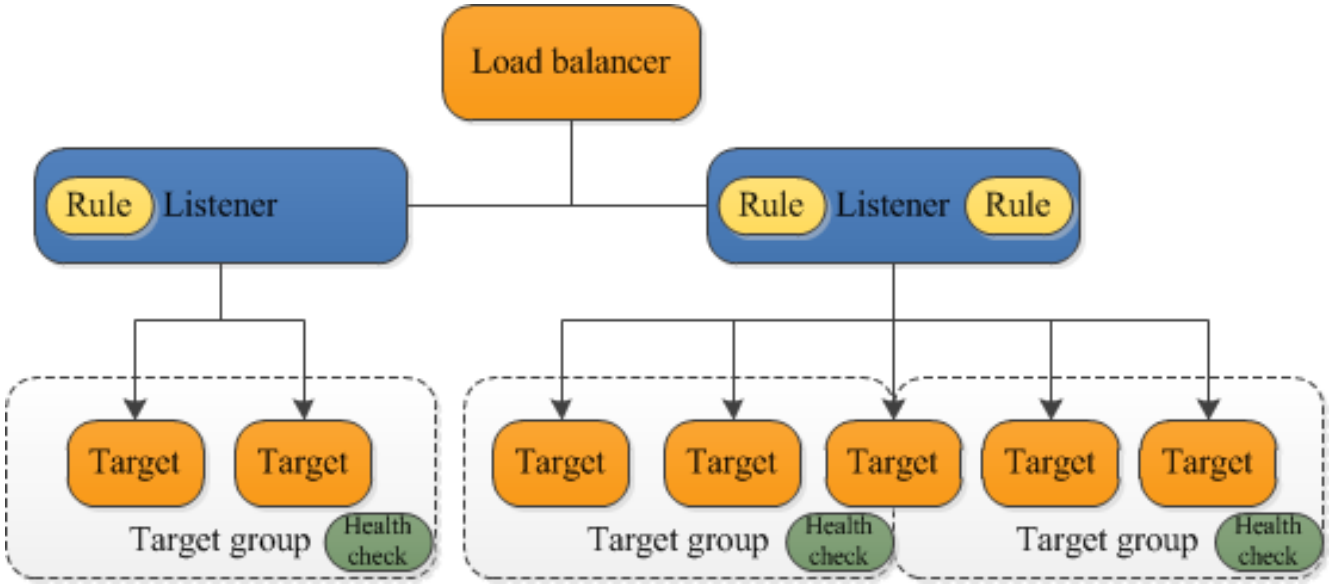
- 로드 밸런서를 사용하면 애플리케이션의 가용성과 내결함성이 높아진다
- 애플리케이션에 대한 요청의 전체적인 흐름을 방해받지 않고 필요에 따라 로드 밸런서에서 컴퓨팅 리소스를 추가 및 제거할 수 있다
- 로드 밸런서가 컴퓨팅 리소스의 상태를 모니터링하는 상태 확인을 구성할 수 있다
- 컴퓨팅 리소스가 주요 작업에 집중할 수 있도록 암호화 및 복호화 작업을 로드 밸런서로 오프로드할 수 있다

ELB 지원

- Application Load Balancers
- Network Load Balancers
- Gateway Load Balancers

Application Load Balancers

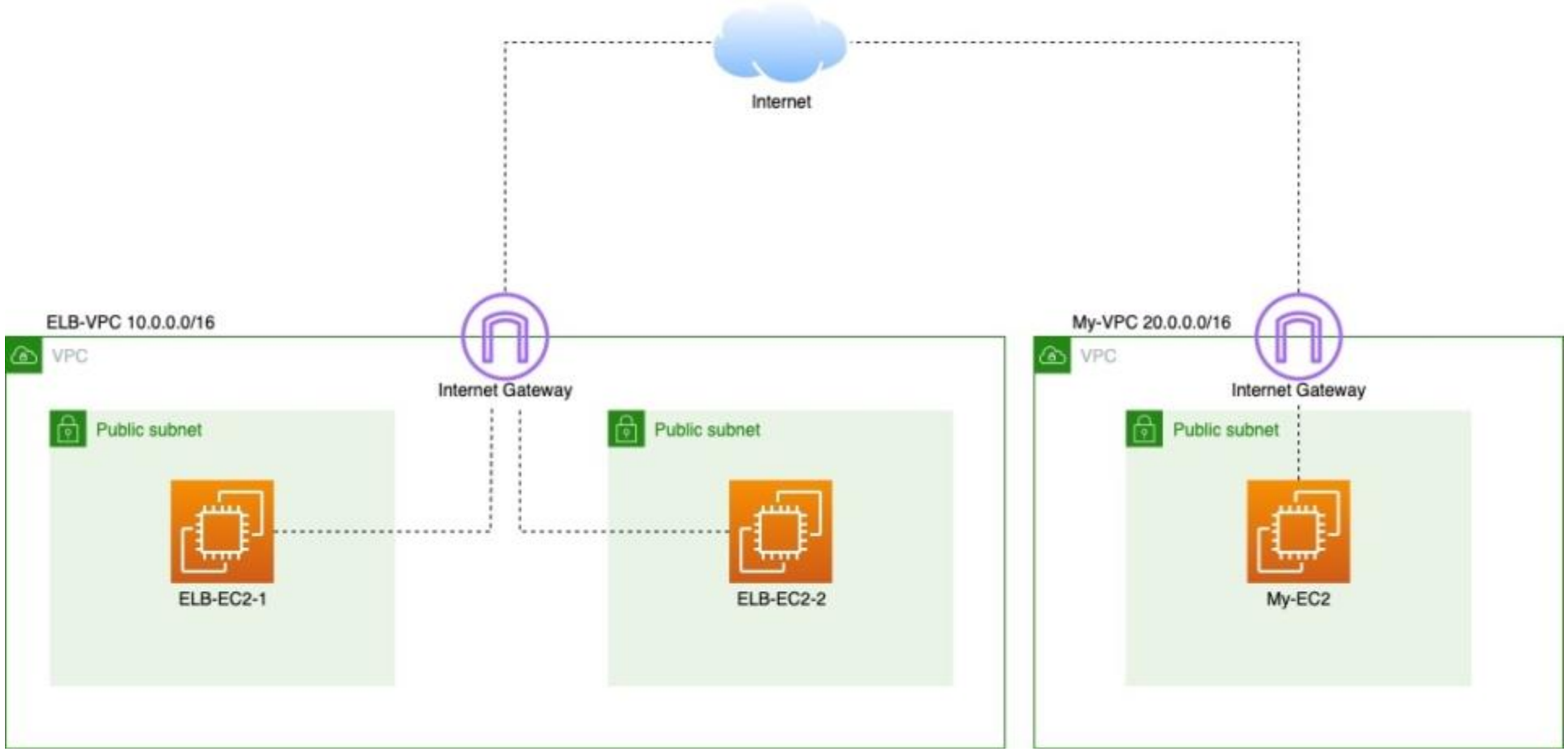
- 로드 밸런서는 수신 애플리케이션 트래픽을 여러 가용 영역의 EC2 인스턴스와 같은 여러 대상에 분산
- Http의 헤더 정보를 이용해 부하분산 실시



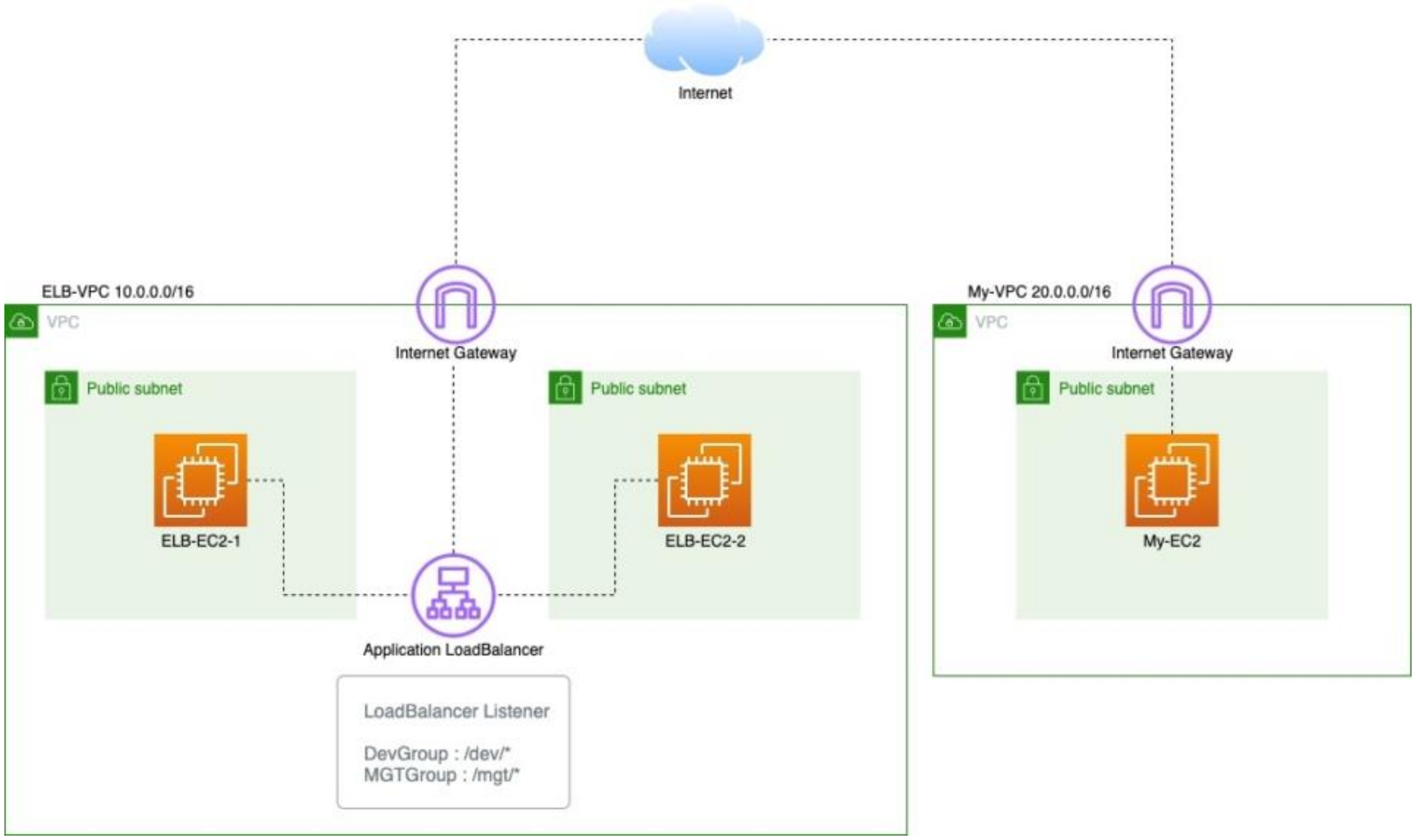
Application Load Balancers

- Application Layer의 프로토콜을 다루는 로드밸런서

Application Load Balancers



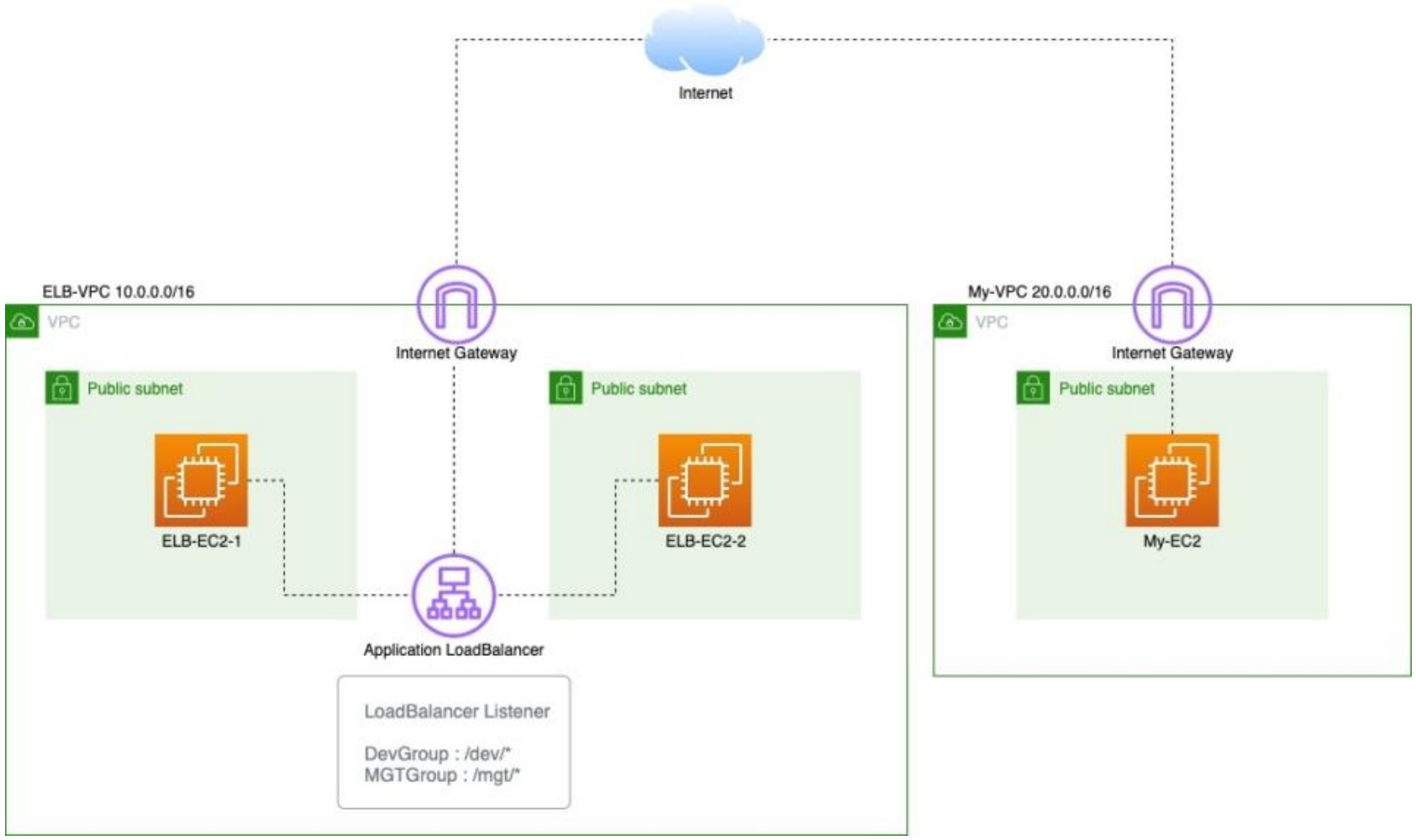
Application Load Balancers



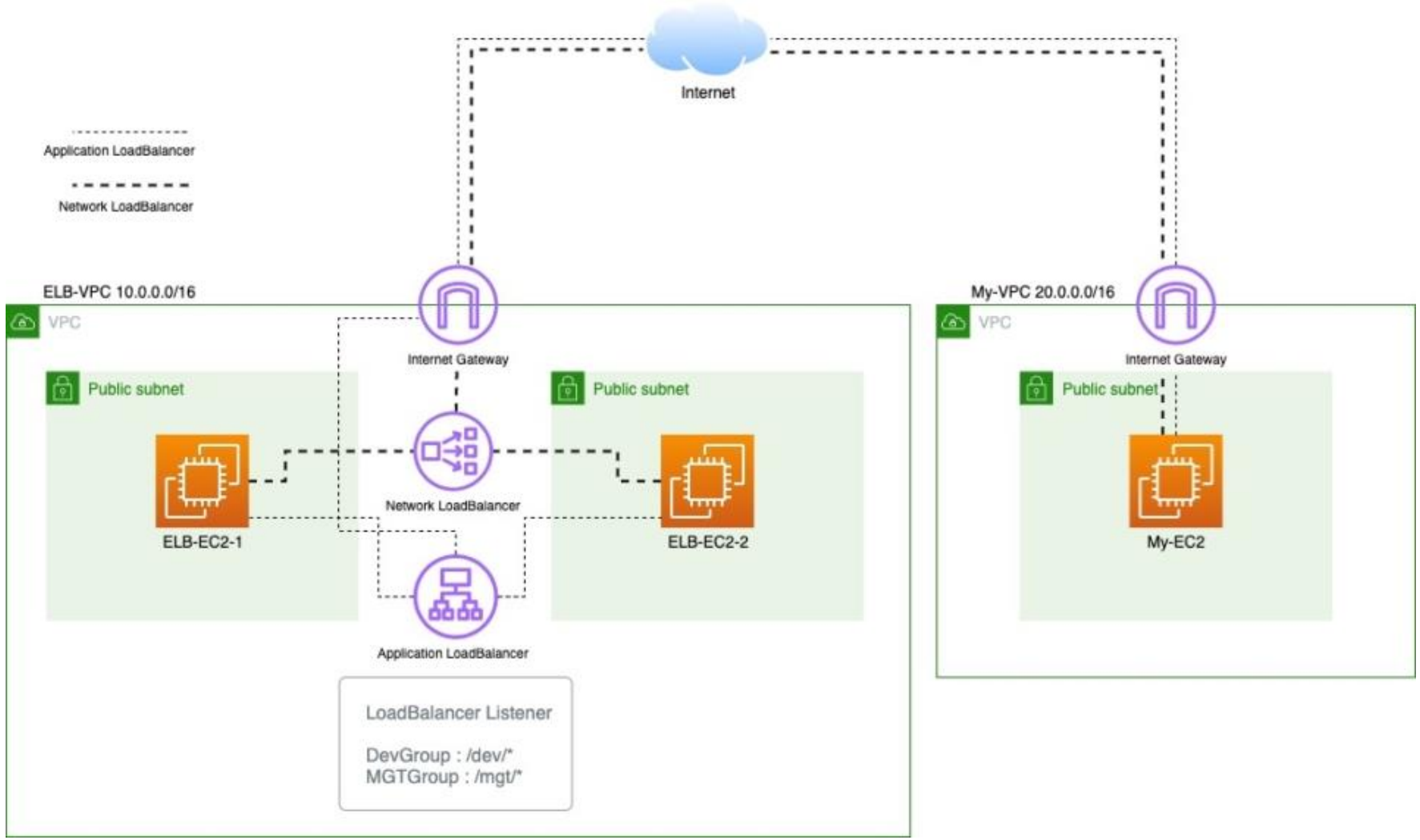
Network Load Balancers

- NLB는 OSI 모델의 네 번째 계층에서 작동
- 초당 수백만개의 요청 처리 가능
- 리스너 프로토콜: TCP, TLS, UDP, TCP_UDP
- 대상 그룹 프로토콜: TCP, TLS, UDP, TCP_UDP
- NLB는 고정 IP를 가짐

Network Load Balancers

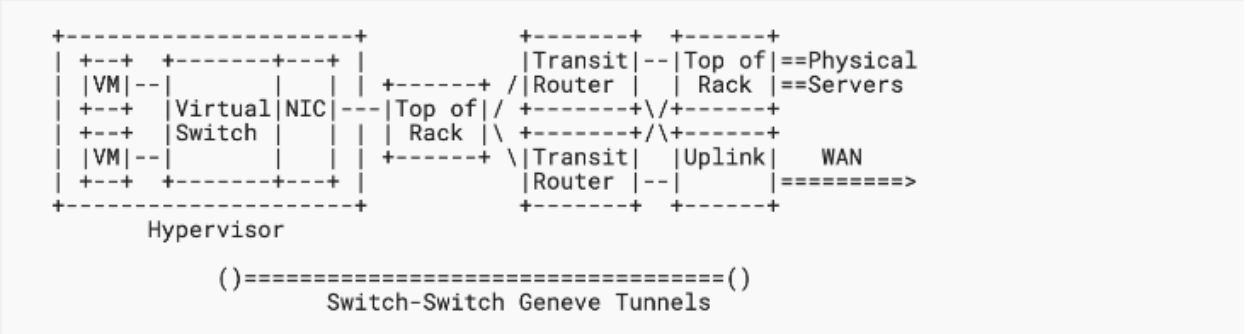


Network Load Balancers

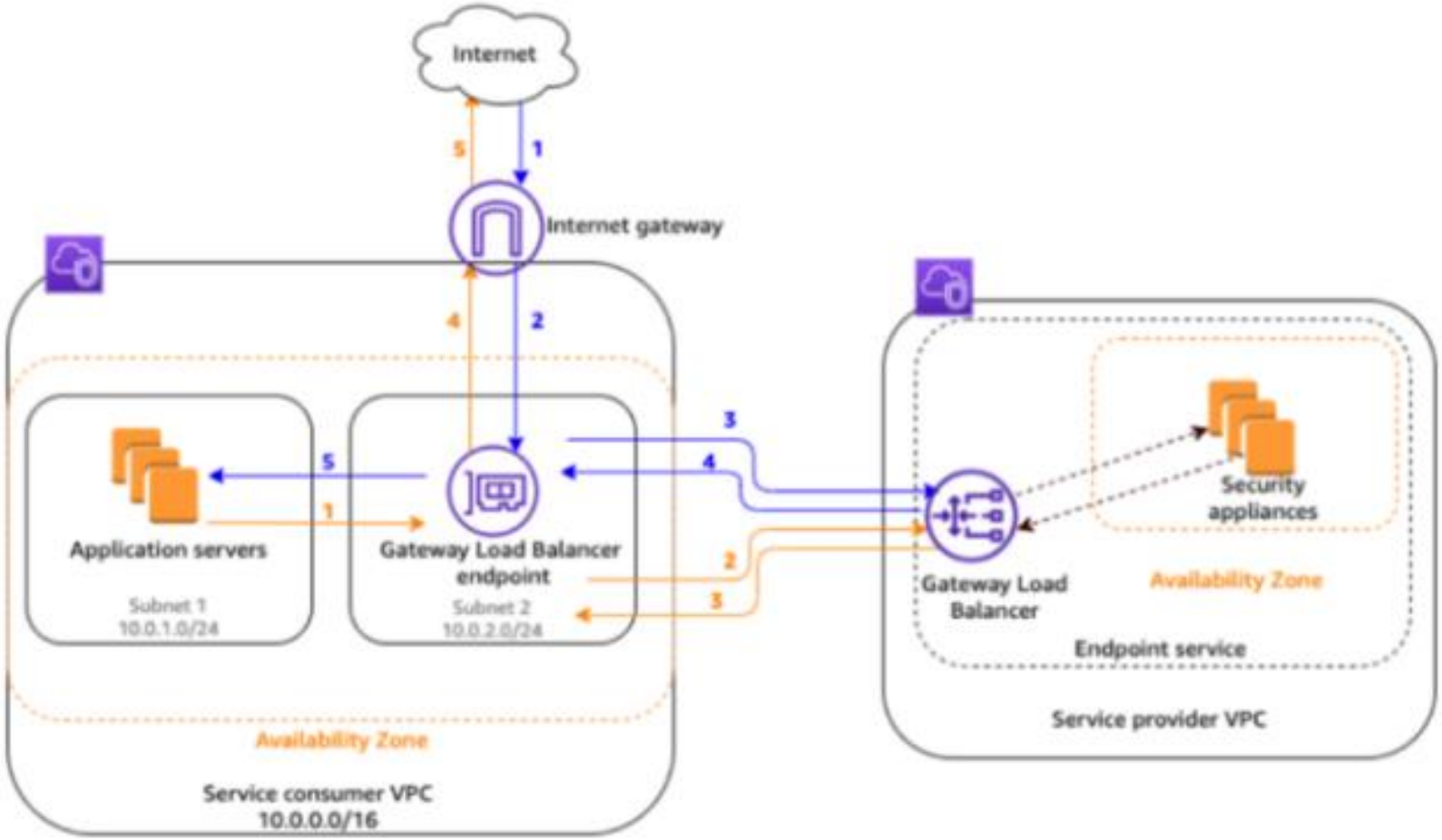


Gateway Load Balancers

- 네트워크 트래픽을 다루는 다양한 가상 어플라이언스(방화벽, 침입 탐지 및 방지 시스템)들을 쉽고 비용 효율적으로 배포, 확장 그리고 관리 해주는 완전 관리형 서비스
- OSI 모델 세 번째 계층에서 작동
- 모든 포트에서 모든 IP 패킷을 수신 대기하고 리스너 규칙에 지정된 대상 그룹으로 트래픽을 전달
- 리스너 프로토콜: 지정 x
- 대상 그룹 프로토콜: GENEVE



Gateway Load Balancers



Gateway Load Balancers

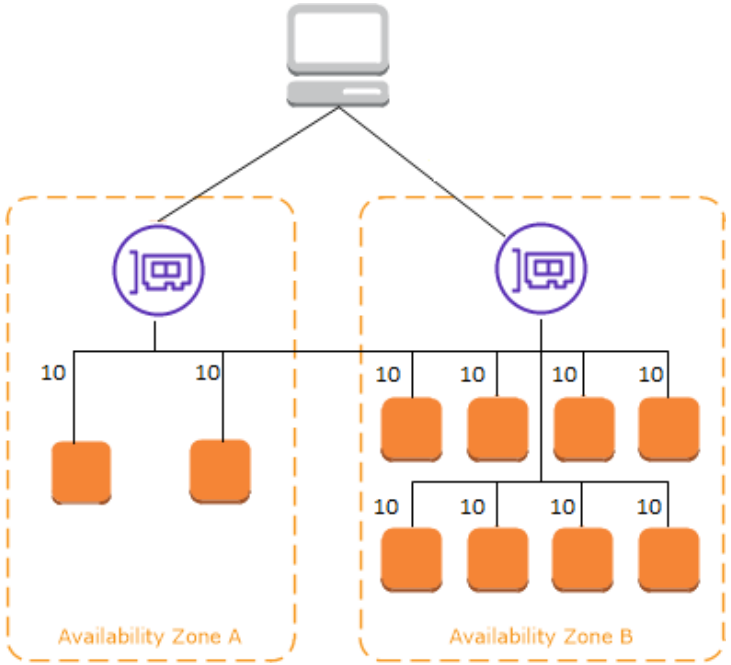
- GWLB는 Network Gateway, LB, 가상 어플라이언스들에 대한 Auto Scaling
- VPC Endpoint와 같은 역할

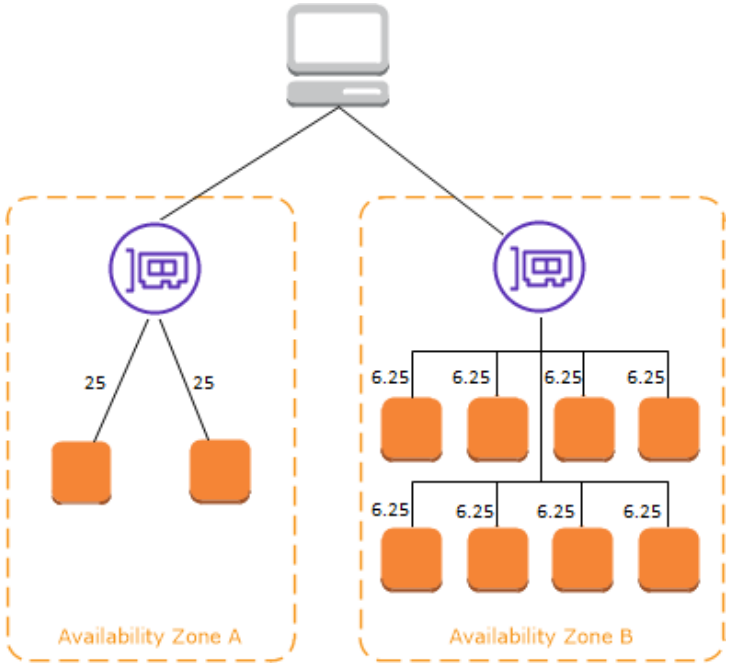
ELB 작동 방식

- LB에 대해 가용 영역을 활성화 하면 가용 영역에 LB 노드가 생성
- LB의 노드는 클라이언트의 요청을 등록된 대상으로 분산

Cross-zone Load Balancing

- Cross-zone LB가 활성화되면 각 노드는 활성화된 모든 가용 영역의 등록된 대상에 트래픽을 분산
- Cross-zone LB가 비활성화되면 각 노드는 가용 영역에 등록된 대상에만 트래픽을 분산





Cross-zone Load Balancing

- Application Load Balancer를 사용하면 Cross-zone Load Balancing이 항상 활성화
- Network Load Balancer 및 Gateway Load Balancer에서는 Cross-zone Load Balacing이 기본적으로 비활성화

Request Routing

1. 클라이언트는 LB에 접근하기 위해 DNS서버를 사용해 LB의 도메인 이름을 확인

2. 해당 DNS서버는 LB 노드의 IP주소를 클라이언트에 반환

3. 그 IP주소는 LB 노드의 IP주소

4. 클라이언트는 LB에 요청을 보내는 데 사용할 IP주소를 결정 및 접근

5. 요청을 수신한 LB 노드는 정상 등록된 대상을 선택하고 사설 IP주소를 사용하여 대상으로 요청을 보냄(리스너 작동)

ALB Routing Algorithm

- 우선순위에 따라 Listener rule을 평가
- 라우팅 알고리즘을 따라 대상그룹에서 대상을 선택
- 기본 라우팅 알고리즘은 Round-Robin
- Least Outstanding Requests 방식은 처리되지 않은 요청을 가장 적게 가지고 있는 EC2 인스턴스에게 할당하는 방식

NLB Routing Algorithm

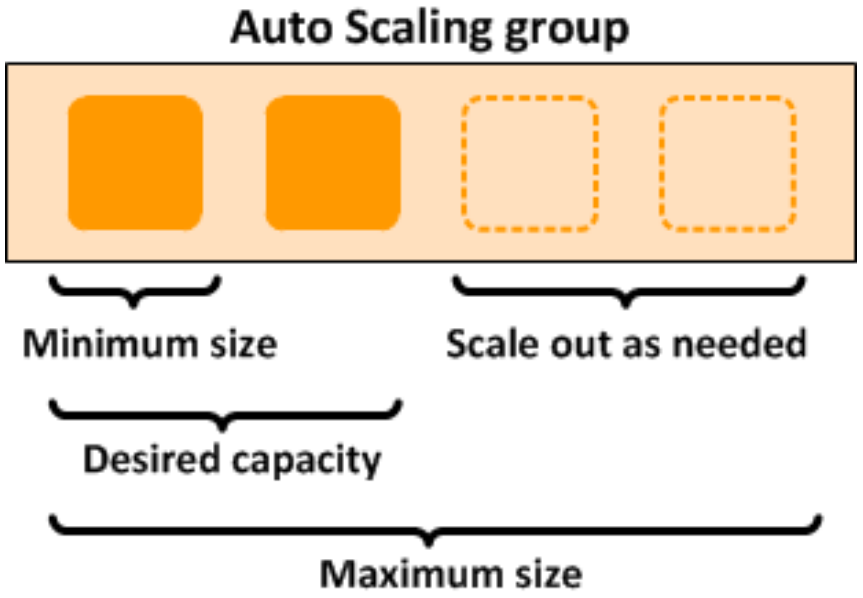
- Flow Hash Algorithm을 사용해 대상 그룹에서 대상 선택
- 연결 수명 동안 각 개별 TCP 연결을 단일 대상으로 라우팅

Flow Hash Algorithm

- 5-Tuple(Source IP Address, Source Port, Destination IP Address, Destination Port, Protocol)을 기반으로 한 알고리즘
- TCP는 5-Tuple에 더해 TCP Sequence Number까지 사용

Amazon EC2 Auto Scaling

- 변화하는 수요에 동적으로 대응하고 비용을 최적화
- Auto Scaling Group이라는 EC2인스턴스 모음을 생성



Auto Scaling 구성 요소

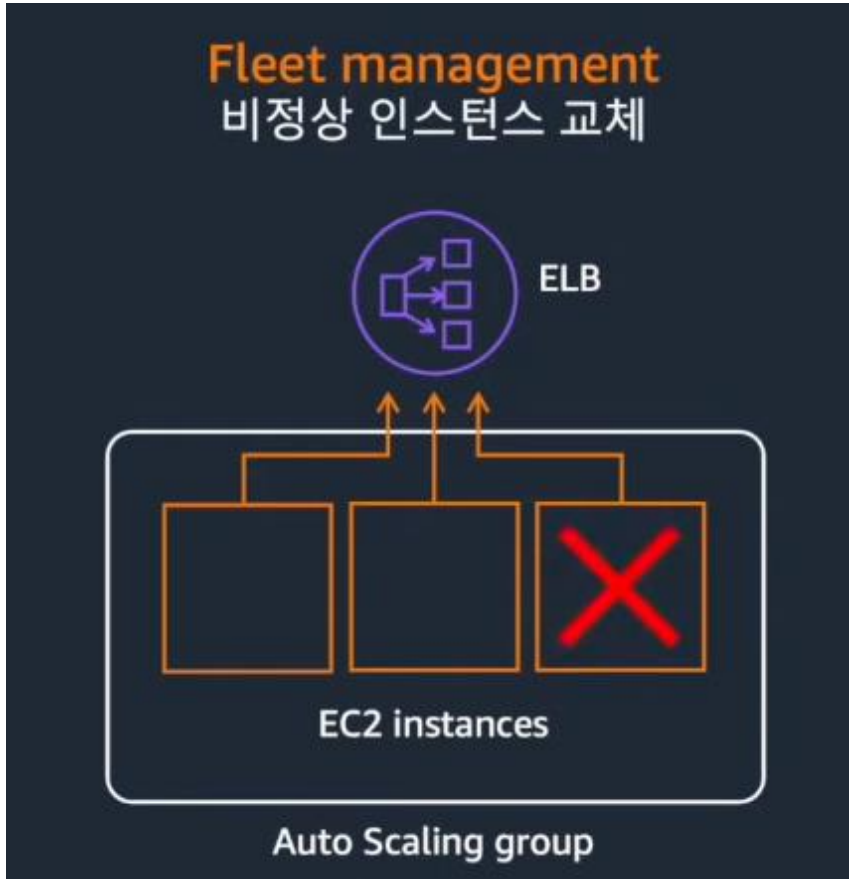
- Groups
- Configuration templates
- Scaling options

Auto Scaling 의 이점

- Fleet management
- Dynamic scaling

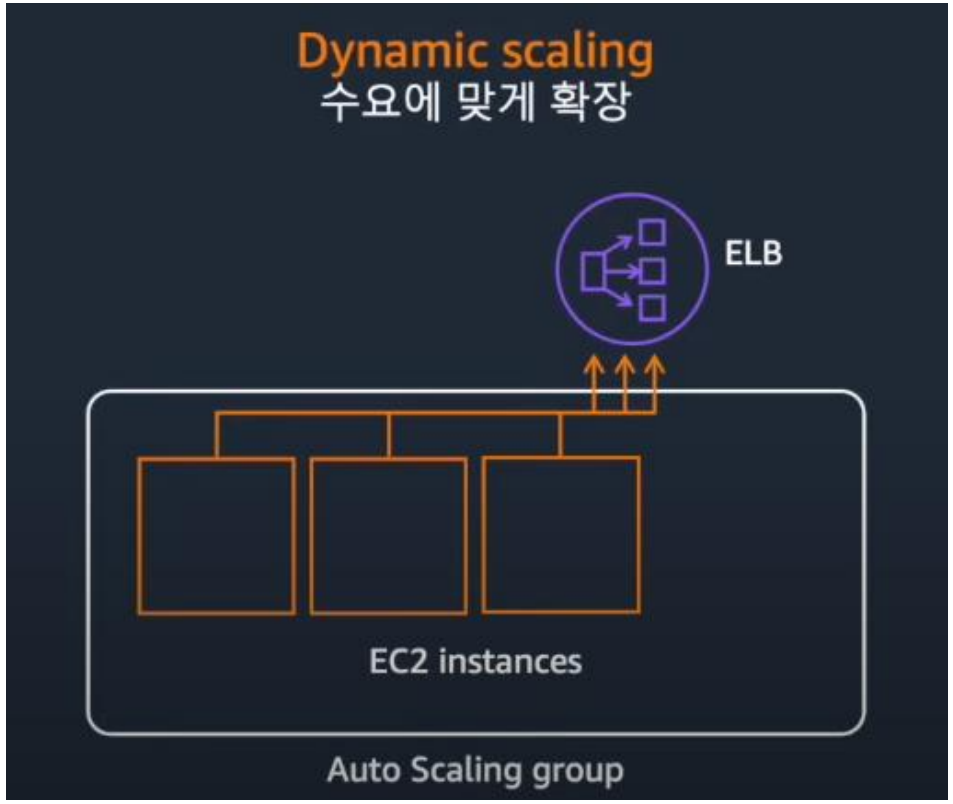
Fleet management

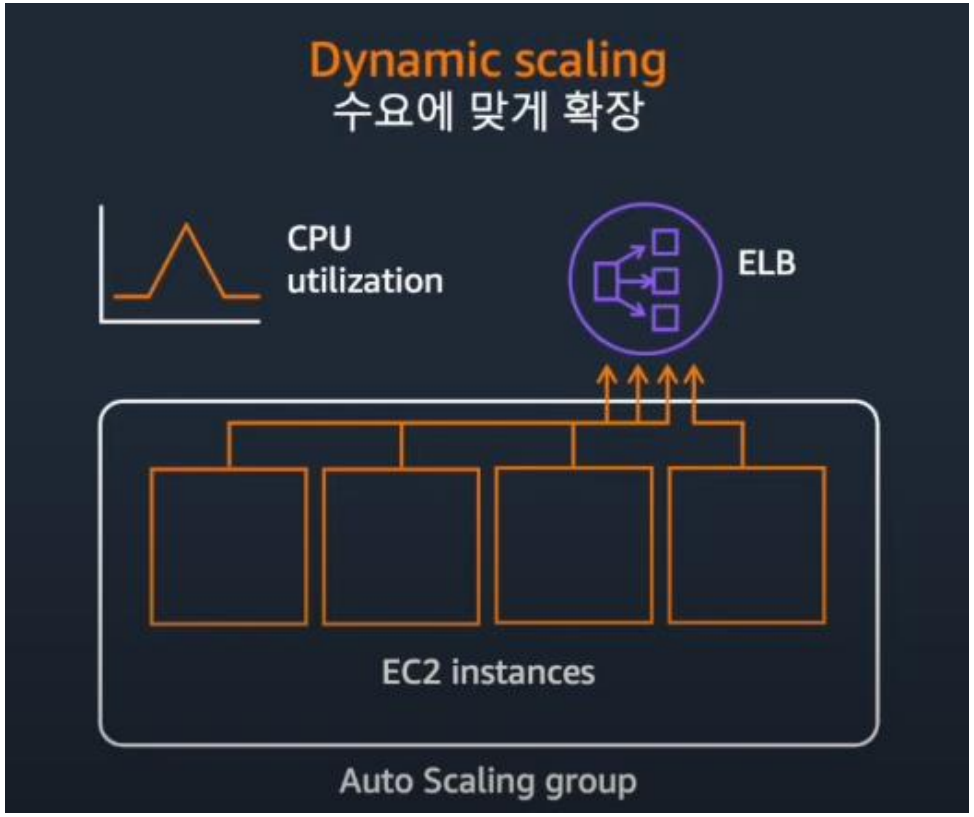
- 비정상 인스턴스 교체



Dynamic scaling

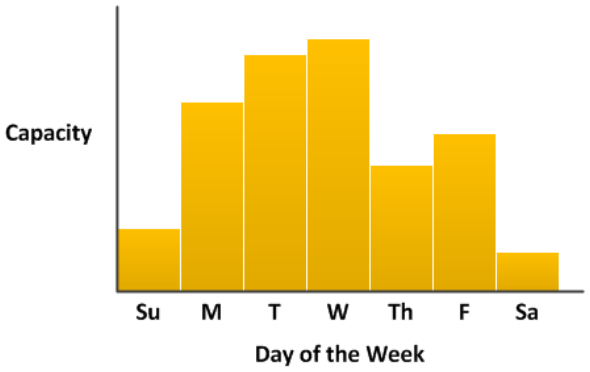
- 수요에 맞게 확장

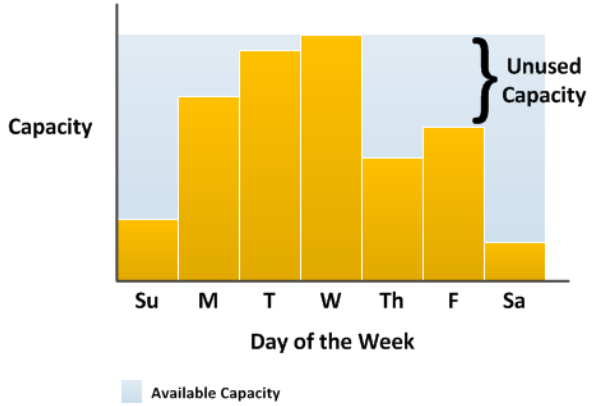


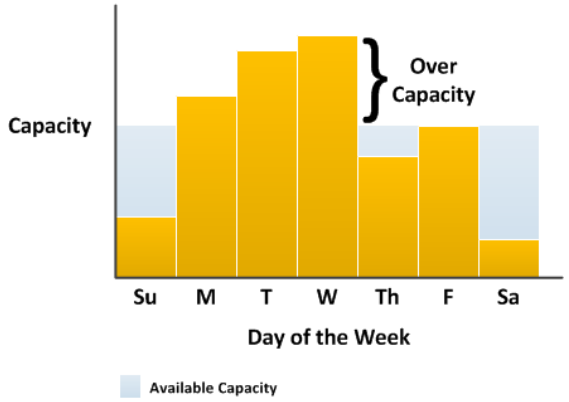


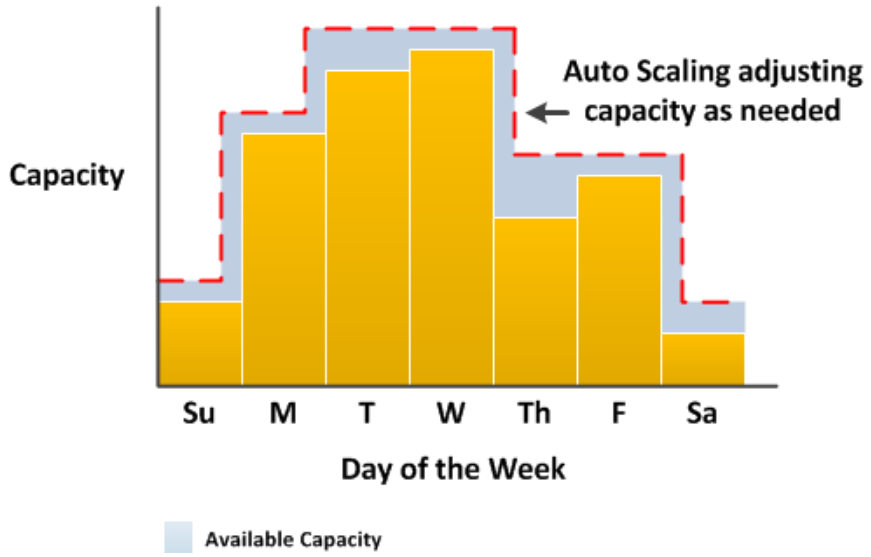
Auto Scaling 예시

- 가변적인 수요에 대응



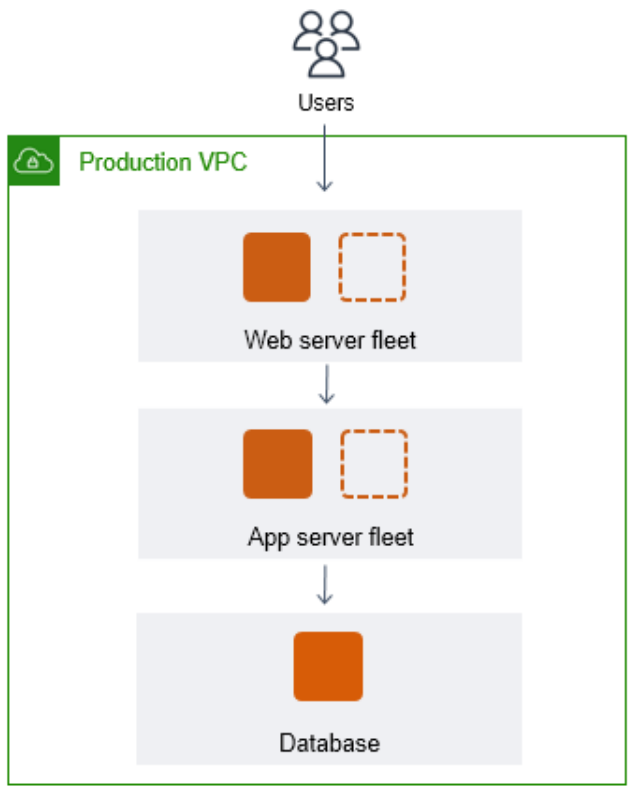


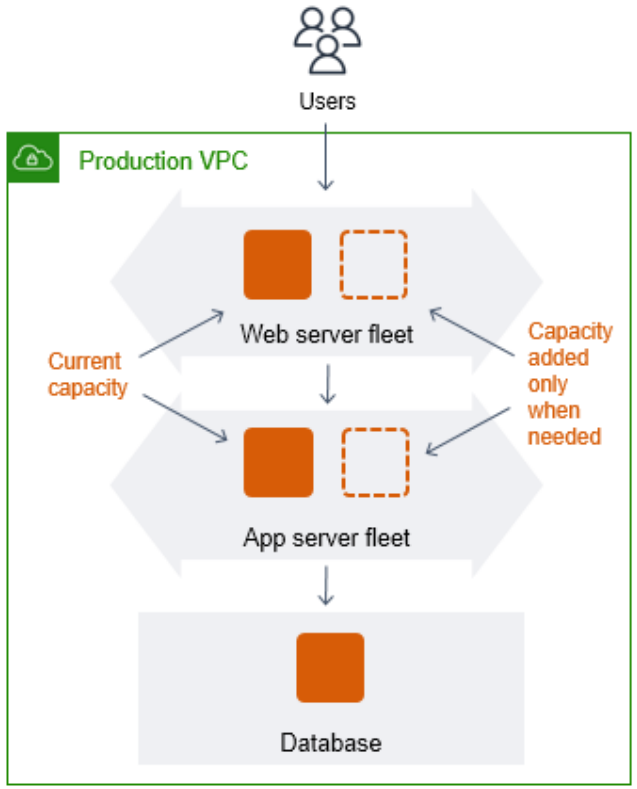




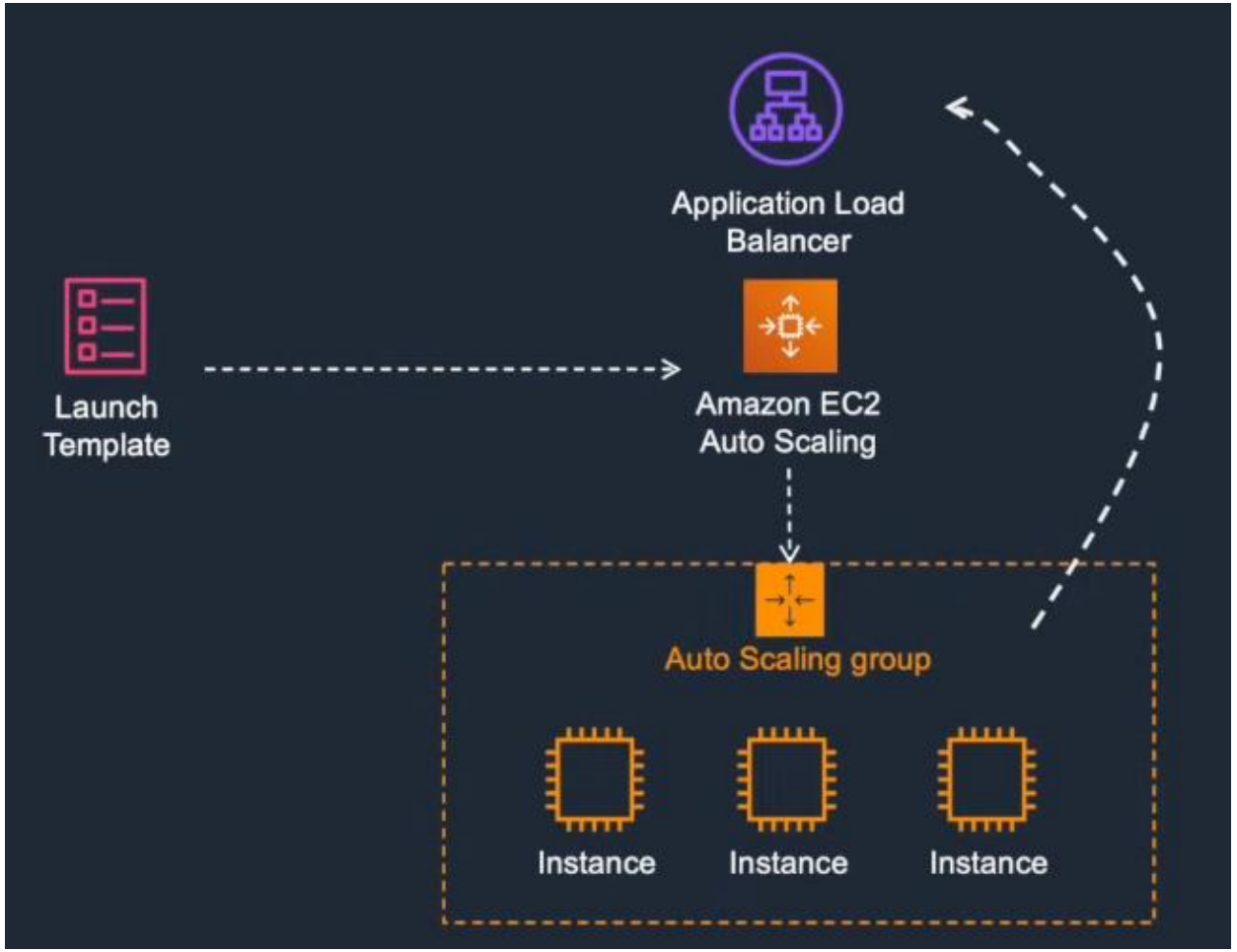
Auto Scaling 예시

- 웹 앱 아키텍처





EC2 Auto Scaling 환경 구성



Scaling Policy

- Manual Scaling
- Dynamic Scaling
  - CPU 평균 사용량이 50%가 되었을 때
  - ALB 요청 수에 따라
  - CloudWatch Metric 에 따라
- Scheduled Scaling
  - 새벽에는 트래픽이 없음
  - 내일 오후 2시에 이벤트 예정
- Predictive Scaling
  - 인스턴스의 수요를 과거 패턴으로부터 학습 -> 예측된 수요에 앞서 인스턴스 시작