# 로드 밸런싱

인터넷의 발달로 데이터 통신이 활발해지면서 한대의 서버로 모든 트래픽을 감당하기 어려워졌다. 수많은 사용자를 동시에 처리하고 정확한 데이터를 제공하기 위해 여러 대의 서버에 동일한 데이터를 저장하고 트래픽을 분산하여 처리한다. 한 서버에 트래픽이 몰리는 상황이 발생하지 않으려면 로드 밸런싱이 필요하다.

기존 서버로 트래픽을 감당할 수 없는 경우 이에 대처할 수 있는 방법은 다음과 같다.

- 기존 서버의 하드웨어 성능을 올리는 Scale-up 방식
- 기존 서버와 동일하거나 낮은 성능의 서버를 증설하는 Scale-out 방식

하드웨어 향상에 드는 비용이 높기도하고, 서버가 여러 대이면 무중단 서비스를 제공하는 환경을 구성하기에도 용이하기에 Scale-out으로 서버를 증설한다. 여러 대의 서버를 사용하면 트래픽을 균등하게 분산시키는 로드 밸런싱이 필요하다.

## 로드 밸런싱

서버가 처리해야 할 요청(Load)를 여러 대의 서버로 분산하여(Balancing) 처리하는 것을 의미한다. 로드 밸런싱을 통해 애플리케이션의 가용성, 확장성, 보안, 성능이 향상될 수 있다.

로드밸런서가 요청을 배정할 서버를 선택하는 알고리즘은 다음과 같다.

**Round Robin**

요청을 순서대로 돌아가며 배정하는 방식이다. 여러 대의 서버가 동일한 스펙을 갖고 있고, 서버와의 연결이 길지 않은 경우에 적합하다.

**Weighted Round Robin**

서버마다 가중치를 매기고 가중치가 높은 서버에 요청을 우선적으로 배분한다. 서버마다 트래픽 처리 능력이 다른 경우 처리 능력이 높은 서버의 가중치를 높게 설정하는 방식으로 사용하기에 적합하다.

**IP Hash**

클라이언트의 IP 주소를 특정 서버로 매핑하여 요청을 처리하는 방식이다. IP 주소를 해싱하기 때문에 사용자는 항상 동일한 서버로 연결된다.

**Least Connection**

요청이 들어온 시점에 가장 적은 연결상태를 가진 서버에 우선적으로 배분한다. 자주 세션이 길어지거나 서버에 분배된 트래픽이 일정하지 않은 경우에 적합하다.

**Least Response Time**

서버의 현재 연결 상태와 응답 시간을 모두 고려하여 트래픽을 배분한다.

## L4 로드 밸런싱과 L7 로드 밸런싱

L4는 전송 계층 프로토콜의 헤더를, L7은 응용 계층 프로토콜의 헤더를 부하 분산에 이용한다.

**L4 로드 밸런서**

L4 로드 밸런서는 네트워크 계층이나 전송 계층의 정보를 바탕으로 로드를 분산한다.

- 전송 계층의 정보: IP 주소, 포트번호, MAC주소, 전송 프로토콜

**L7 로드 밸런서**

L7 로드 밸런서는 애플리케이션 계층에서 HTTP 헤더, 쿠키와 같은 사용자 요청을 기준으로 특정 서버에 트래픽을 분산할 수 있다. 패킷의 내용을 확인하고 이에 따라 클라이언트의 요청을 보다 세분화해 서버에 분산시킬 수 있다. L7 로드 밸런서는 특정 패턴을 지닌 바이러스를 감지할 수 있어서 DoS/DDos 같은 비정상 트래픽을 필터링할 수도 있다.

로드 밸런서의 주요 기능

- NAT(Network Address Translation)

  사설 IP 주소를 공인 IP 주소로 바꾼다

- Tunneling

  데이터를 캡슐화하여 연결된 노드만 캡슐화된 패킷을 구별해 이를 해제할 수 있도록 한다

- DSR(Direct Server Return)

  서버에서 클라이언트로 되돌아가는 경우 목적지 주소를 스위치의 IP 주소가 아닌 클라이언트 IP 주소로 전달해서 네트워크 스위치를 거치지 않고 클라이언트를 찾아가도록 한다.