CDN 구조

1. 사용자가 웹서버에 콘텐츠 요정
2. 요청은 중간 서버를 거쳐

Origin서버에 요청

1. Origin은 요청을 받아 해당 컨텐츠를

Edge에 저장

1. Edge는 해당 컨텐츠를 저장하고

사용자에게 전달

1. 사용자가 같은 요청을 할 시Edge는

Orgin에 요청하지 않고 Edge서버에서 바로 처리

1. Edge가 2, 3차면 릴레이라고 한다.

CDN을 이루는 것

Edge(컨테이너 + 오케스트레이션, 프록시 웹 서버, 프로비져닝 툴, Purge epi)

Origin(Origin 웹서버)

전체(모니터링 시스템)

컨테이너

* 커널 공유 기능을 이용해 애플리케이션, 라이브러리, 서비스를 넣어 패키지 형식으로 배포할 수 있게 끔 만든 가상화 기술

CDN 기능

* Cache는 요청에 빠르게 응답하기 위해 서버와 클라이언트 사이에서 응 답 콘텐츠의 사본을 저장하는 공간을 의미한다
* 프록시는 Client와 Sever 사이 통신을 중계
* 받은 요청을 어떤 서버에 다시 요청을 할 것인지 결정
* CDN에 배포된 서버들은 CDN 룰에 맞게 보안 설정 O 🡺 Ddos 공격 해킹 위험에 대한 보안 강화

Nginx 프록시 설정

세가지 블록 구조를 가지고 있음

1. Http 블록
   1. 가장 범위 큼, http에 대한 요청을 해당 설정 적용
   2. 설정한 블록에서 선언한 설정은 서버 지역의 기본 값이 된다
2. 서버 블록
   1. 하나의 호스트를 선언하는데 사용, 한 개 이상의 지역을 설정
3. 지역 블록
   1. 특정 URL을 처리하는 방법 정의
   2. 설정이 중복 될 때 🡺 지역 > 서버 > http 블록으로 우선 순위가 있음

Nginx 에서의 캐시 구조

* 파일 형태로 콘텐츠 캐시
* 대부분 2단계 이상의 계층 구조 사용
* 캐시 사용 여부와 같은 데이터를 저장하는 공유 메모리 공간, 디스크로 이동할 필요 없이 캐시 되어 있는지 판별(Key\_zone)
* 콘텐츠의 저장 기간을 설정해 주는 Inactive가 있음

Cache 상태

* Hit 컨텐츠가 캐쉬 면 상단에 요청 X
* MISS 컨텐츠가 캐쉬 면 상단에 요청 O
* Expired 캐쉬된 컨텐츠 만료 🡺 상단에 요청 O(MISS에 포함)
* Revalidated 캐쉬된 콘텐츠 만료, 자신의 콘텐츠 다시 반환 🡺 상단에 요청 X
* Bypass 캐쉬 X 바로 상단에 요청
* Stale 컨텐츠 만료, 상단에 요청 🡺 상간 서버 응답 없음 🡺 기존 파일 전달

Nginx 프록시(Proxy)

1. 프록시 패스
   1. 프록시 하는 서버와 url정함
   2. 도메인, 서버명을 사용할 수 있음
2. Upstream
   1. 상단에 보낼 서버를 결정