[AWS] 8. DNS와 네트워크 라우팅 Route 53, CloudFront

- **References**
  - AWS 공인 솔루션스 아키텍트 스터디 가이드 - 어소시에이트 3/e - 8장
  - https://inpa.tistory.com/entry/AWS-📚-CloudFront-개념-원리-사용-세팅-💯-정리?category=947447 (CloudFront 개념 정리)
  - https://inpa.tistory.com/entry/AWS-📚-Route-53-개념-원리-사용-세팅-💯-정리?category=947449 (Route 53 개념 정리)

※ 도메인 네임 시스템 (DNS, Domain Name System)

- **DNS (Domain Name System, 도메인 네임 시스템) 이란**
  - example.com 과 같이 사람이 읽기 편한 도메인 네임과, 92.184.216.34 같은 머신이 읽을 수 있는 IP 주소를 매핑하기 위한 시스템

- **네임스페이스 (Namespace) 란**
  - 중복된 도메인이 존재하지 않게 하기 위해 인터넷 이름 체계를 관리하는 관리 기구가 필요하다.
  - 인터넷 네임 시스템은 네임스페이스라는 도메인 네임 체계로 관리
  - 인터넷은 퍼블릭 또는 프라이빗 IP를 통하거나 톱레벨 도메인 (TLD,Top-Level Domain) 을 통해 접근할 수 있는 소규모의 네임스페이스로 구획이 나눠진 가상의 공간이라 할 수 있다.
  - Internet Protocol (IP) 과 도메인 네임의 계층 구조는 ICANN (Internet Corporation for Assigned Names and Numbers) 라는 기구에서 관장한다.

- **네임 서버 (Name Server)란**
  - 실제로 도메인 네임과 IP 주소를 연결하는 일은 네임서버가 담당
  - 모든 컴퓨터에는 로컬에서 접근할 수 있는 간단한 네임서버 데이터베이스를 지니고 있으며, 여기엔 localhost와 같은 호스트네임 엔트리와 IP주소정보가 포함된다. (Linux의 /etc/hosts 파일, 로컬 네임서버 라고 부름)
  - 로컬 네임서버에서 처리되지 못하면 네트워크 인터페이스 환경설정에 명시된 외부 DNS 네임서버로 전달된다. (/etc/resolv.conf DNS 서버 주소 명시파일, Google : 8.8.8.8, OpenDNS 208.67.222.222 등)

- **도메인과 도메인 네임**
  - 도메인(Domain)은 단일 도메인 네임으로 식별 가능한 하나 이상의 서버, 데이터 저장소 또는 디지털 리소스를 의미
  - 도메인 네임은 해당 도메인을 위해 공식 기구에 등록된 이름으로, 등록된 이름은 도메인이 가리키는 리소스를 네트워크로 직접 연결하는 데 사용 할 수 있다.

- **도메인 등록**

    - 톱레벨 네임 서버는 관련 쿼리에 응답하기 전 새 도메인 네임의 존재를 파악하고 있어야 하며, 네임 서버에 등록 된 새로운 도메인 네임을 전파하는 일은 도메인 네임 등록자 또는 레지스트라가 수행한다.
    - 도메인 네임 등록자 : 도메인 네임의 예약을 관리하는 사업자
        - Amazon Route 53은 도메인 네임 등록자와 같은 역할을 수행하는 서비스

- **도메인 레이어**
    - 도메인 네임은 여러개의 요소로 구성되며, 도메인 네임의 최우측 (.com, .org) 텍스트는 톱레벨 도메인 (TLD), 두번째는 SLD(세컨드 레벨 도메인)이라 함
    - 서브 도메인은 도메인 리소스의 하위 요소 식별에 도움을 주며, 이메일 주소, 웹 루트 디렉토리로 활용 가능하다.

- **전체 주소 도메인 네임 (FQDN, Fully Qualified Domain Name)**
    - 도메인 네임 그대로 요청을 전달하려는 경우 FQDN을 사용
    - FQDN은 서브도메인과 TLD를 포함 도메인의 절대 주소 정보를 모두 지니며, TLD 뒤 후속 닷을 추가하여 FQDN임을 명시

- **존과 존 파일**
    - Zone 은 DNS 도메인을 정의한 것
        - Route 53은 존 을 호스팅 영역(Hosted zone)이라 부른다.
    - Zone File 은 도메인 내에서 DNS 주소에 매핑되는 리소스를 설명하는 텍스트 파일

지시자	용도
Name	정의된 도메인 네임 또는 서브도메인 네임
TTL	레코드 만료전 유효시간(time to live)
Record Class	레코드의 네임스페이스, 보통 IN을 사용(Internal)
Record Type	레코드에 의해 정의 된 레코드 타입(A, CNAME 등)

- **레코드 타입**
  - DNS 서버가 해당 패킷을 받았을 때 어떤식으로 처리할지를 나타내는 지침

종류	설명
A	해당 도메인 주소가 가지는 IP(1:1)
CNAME	별칭을 부여한 특정 도메인 주소
MX	메일을 주고 받기 위한 서비스 레코드
TXT	일반적인 텍스트 내용을 기록, String
SOA	도메인의 시작점(Start Of Authority)
HINFO	호스트의 CPU 정보와 운영체제 정보
MINFO	메일박스와 메일리스트 정보
PTR	IP주소에 대한 호스트명
UINFO	사용자정보
ANY	호스트에 관련된 모든 레코드들의 정보
AAAA	IPv6 버전 A 레코드
SRV	비슷한 TCP/IP 서비스를 제공하는 다수의 서버 위치 정보
NS	영역을 풀이할 수 있는 dns서버 목록

※ Amazon Route 53

- **Route 53 이란**
  - 가용성, 확장성이 뛰어난 클라우드 DNS(Domain Name System) 웹 서비스
  - 도메인 구입부터 네임서버 등록, DNS에 필요한 모든 기능이 있고, 추가 모니터링 기능을 제공
  - 다른 도메인 등록기관에 비해 비슷하거나 저렴한 편이고, 부가적인 기능 제공 및 안정성, GUI를 제공해 관리가 수월하다

- Route 53은 사용자의 요청을 Amazon EC2 인스턴스, Elastic Load Balancing 로드벨런서, S3 버킷 등 AWS 에서 실행되는 인프라에 효과적으로 연결하고, AWS 외부의 인프라로 라우팅 하는데 Route 53 사용이 가능
- 동적으로 사용자에게 노출 될 DNS 레코드 타입과 값 조정 가능
- 각종 로드벨런싱 기능 제공
- 트래픽 흐름을 사용하면 지연시간기반 라우팅 기능 제공

- **Route 53 주요 기능**
  - 도메인 등록
  - DNS 관리
  - 가용성 모니터링 (네트워크 헬스체크)
  - 트래픽 관리 (라우트 정책 등)

- **도메인 등록**
  - 도메인은 ICANN 인증 등록사업자 또는 레지스트라중 어느곳에서도 등록이 가능하지만 Route 53 에서는 간단하게 도메인 등록작업을 마칠 수 있다.
  - 기존 도메인을 AWS로 이전하는 경우기존 레지스트라의 어드민 인터페이스에 있는 도메인 이전 설정을 변경하고 이에 대한 인증 코드 요총후 Route 53에 해당 인증코드를 제공하여 이전

- **DNS 관리**
  - Hosted zone(호스트 영역) 설정을 통해 사용자가 브라우저, 이메일클라이언트 또는 프로그래밍 방식으로 도메인 네임을 호출해 사용할 수 있도록 돕는다.
  - 기존 방식이 미리 설정 된 존 파일을 임포트 해서 수정하는 방식이었다면 Route 53은 호스팅 영역을 생성한 뒤 콘솔 또는 AWS CLI에서 설정할 수 있도록 한다.
  - 퍼블릭 또는 프라이빗 호스팅 영역 방식

- 프라이빗 호스팅 영역은 AWS VPC를 통해서만 해당 리소스를 접근 할 수 있도록 함, 외부 사용자의 접근을 허용할 필요가 있다면 퍼블릭 호스팅 영역으로 설정한다.
- 자동으로 SOA 레코드를 생성하여 4개의 네임서버 주소를 제공하며, 그 다음부터 새 레코드 세트 정의를 통해 여러분이 원하는 도메인과 서브 도메인의 관계를 설정하고 리소스를 연결하는 등 작업을 수행하면 된다.

- **가용성 모니터링**
  - 연결 된 리소스의 헬스체크 모니터링 기능을 제공한다.
  - 새 레코드 세트 생성 시, 라우팅 정책 선택 옵션이 제공되며, Simple 정책을 선택 시 헬스 체크와 정책을 간단하게 연결 가능
  - 레코드 세트에 연결된 리소스의 성능을 주기적으로 점검하며, 검증 시 반응이 없을 경우 Route 53은 해당 리소스가 오프라인 상태가 됐다고 판단, 트래픽을 백업 리소스로 우회 시킨다.

- **라우팅 정책 (트래픽 관리)**

정책	설명
심플 라우팅 (Simple Routing)	모든 요청을 지정한 IP 주소 또는 도메인 네임으로 전송 (Default 설정)
가중치 라우팅 (Weighted Policy)	설정 한 비율에 따라 다수의 리소스에 트래픽을 분산 처리, 별도의 레코드 세트를 생성하여 레코드 세트마다 동일한 세트 ID 값을 할당한 뒤 인스턴스 마다 가중치 값을 입력
지연 라우팅 (Latency-based Routing)	다수의 AWS 리전에서 실행되는 리소스를 조합해 최고의 사용자 경험을 제공하기 위한 방법, 각 리전에 병렬적으로 리소스 배치, 세트ID를 동일하게 설정시 전송지연이 낮은 쪽으로 트래픽을 전송한다.
실패대응 라우팅 (Failover Routing)	헬스체크 결과 정상적으로 작동하는 리소스에 우선적으로 트래픽을 전송하는 방식
지리적 라우팅 (Geolocation Routing)	지연 라우팅과 달리 지리적 라우팅은 대륙, 국가, 미국의 주 등 요청이 발신된 지역을 기준 (좁은 단위) 으로 라우팅 방식을 결정, Route 53은 종종 요청 IP 주소의 발신지를 식별하지 못하는 경우가 있어 기본 레코드 설정에서 이러한 부분을 보완할 수 있도록 설정해야 한다.

정책	설명
다변량 라우팅 (Multivalue Routing)	고가용성을 제공하기 위함, 다수의 변수 값을 지닌 레코드 세트로 하나의 리소스를 가리키도록 하고 이를 헬스체크와 연계 하여 고가용성 확보

- **Traffic Flow**
  - 라우팅 정책 조합을 시각화 하는 콘솔 기반 그래픽 인터페이스
  - 라우팅 템플릿 , 템플릿 커스터마이징 기능 제공

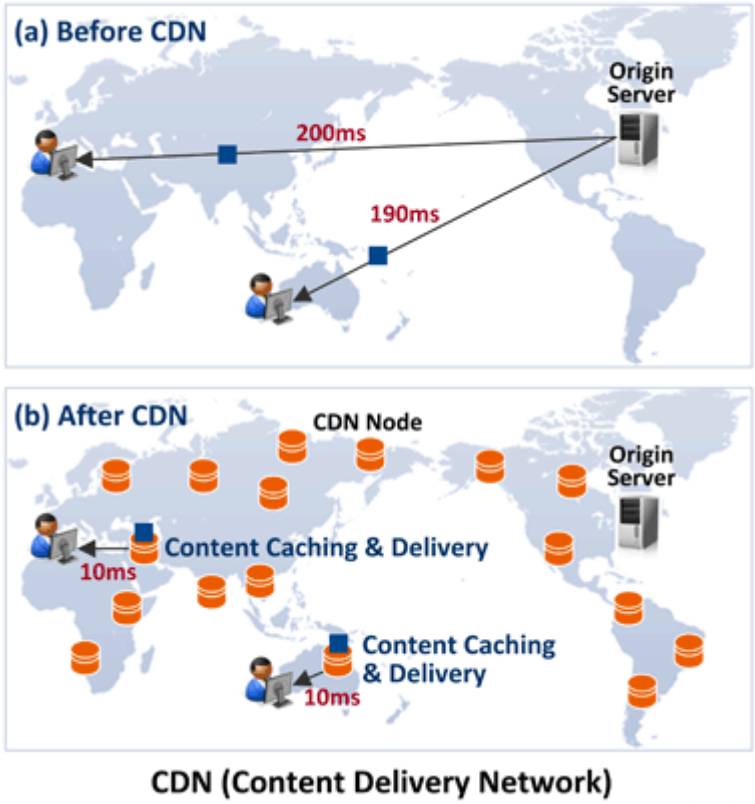
- **Route 53 Resolver**
  - AWS 리소스와 AWS 외부의 리소스를 통합해 하이브리드 인프라를 구성하는 경우 사용하여 라우팅을 관리
  - Resolver는 양방향 주소 쿼리를 이용 AWS 리소스와 AWS 외부에 존재하는 온프레미스 리소스의 라우팅을 관리하여, 프라이빗 및 퍼블릭 플랫폼 기반의 워크로드를 간편하게 처리

- **Route 53 요금정책**

유형	정책
호스팅 영역 관리	호스팅 영역에 대한 월별 요금 지불 (25개까지 영역당 $0.5, 추가부터 영역당 $0.1)
DNS 쿼리 제공	ELB, CloudFront, ElasticBeanstalk, API Gateway, Alia A 레코드를 제외하고 요금 부과 (비례 할당) (표준쿼리, 지연시간 기반 라우팅 쿼리, 지역 DNS 및 지역 근접성 쿼리)
도메인 이름 관리	TLD 별 요금이 다르다. 1년 단위 등록, 계정당 최대 50개 등록 가능

※ CDN(Content Delivery Network 또는 Content Distribution Network, 컨텐츠 전송 네트워크)

- **CDN 이란**

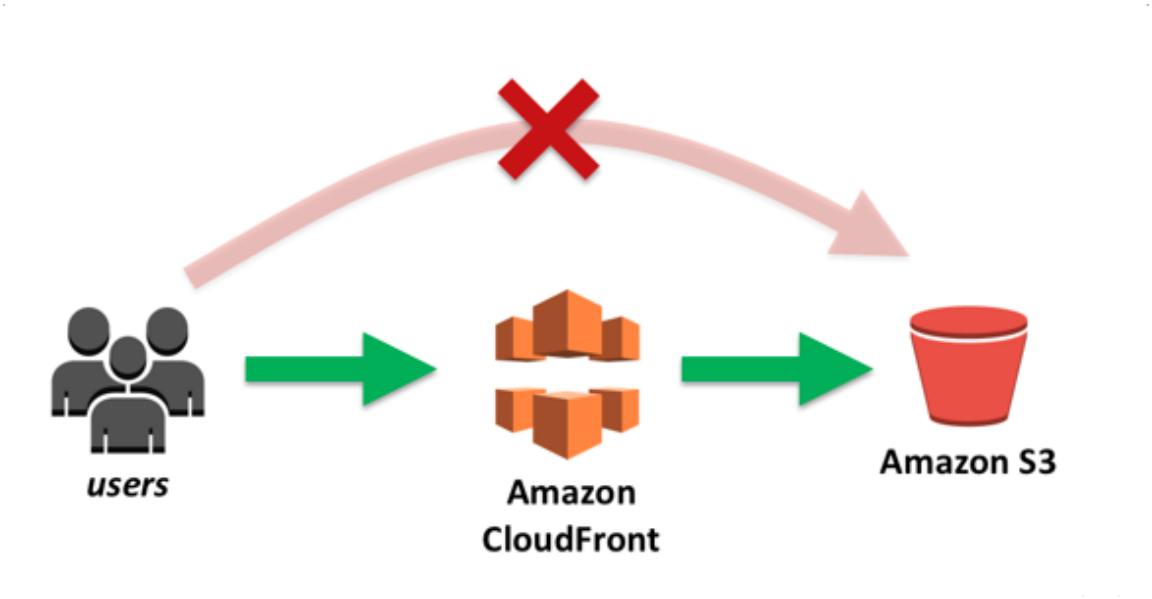


CDN (Content Delivery Network)

  - 컨텐츠를 효율적으로 전달하기 위해 여러 노드를 가진 네트워크에 데이터를 저장하여 제공하는 시스템
  - 인터넷 서비스 제공자(ISP)에 직접 연결되어 데이터를 전송하므로, 컨텐츠 병목을 피할 수 있다. (= 서버 요청이 필요 없기 때문에 서버의 부하를 낮추는 효과)
  - 웹페이지, 이미지, 동영상 등의 컨텐츠를 본래 서버에서 받아와 캐싱해 두고, 해당 컨텐츠에 대한 요청이 들어오면 캐싱해 둔 컨텐츠를 제공한다.
  - 컨텐츠 제공 서버와 실제 요청 지점간 지리적 거리가 매우 멀거나, 통신환경이 안좋은 경우 빠른 컨텐츠 제공이 가능하다.

※ Amazon CloudFront

- **CloudFront 란**



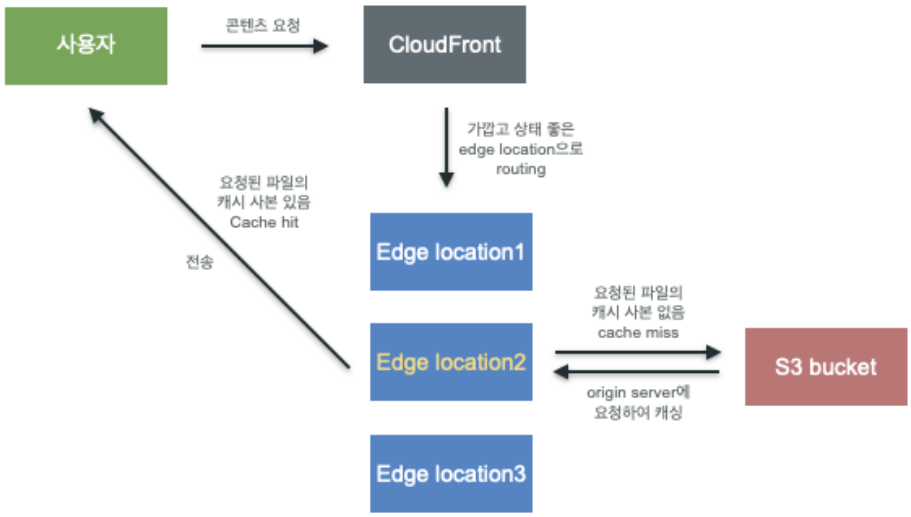
  - 개발자 친화적 환경에 짧은 지연시간관 빠른 전송속도로 데이터, 동영상, 애플리케이션 및 API 를 전세계 고객에게 안전하게 전송하는 고속 컨텐츠 전송 네트워크 서비스 (CDN)
  - CDN서비스 이외에도 기존 보안기능(Anti-DDoS)을 제공

- **CloudFront 의 컨텐츠 배포 방식**
  - 배포의 종류는 콘텐츠의 미디어 타입에 따라 달라진다
    - 웹페이지나 그래픽 컨텐츠의 경우 web distribution 을 선택
    - S3에 저장 된 Adobe RTMP (Real-Time Messagine Protocol) 기반 비디오 컨텐츠의 경우 RTMP distribution 선택
  - 배포 환경 설정 시 무료로 사용할 수 있는 ACM (AWS Certificate Manager) SSL/TLS 암호화 인증서를 추가할지 선택 가능하다 (스니핑, 중간자 공격 방어에 도움)

- **Cloud Front 가 지원하는 컨텐츠 원본 또는 오리진**
  - Amazon S3 버킷 : 접속 가능한 모든 S3 버킷
  - AWS MediaPackage 채널 엔드포인트 : 비디오 패키징 및 원본 추적 기능
  - AWS MediaStore 컨테이너 엔드포인트 : 미디어 최적화 스토리지 서비스

- **엣지 로케이션**
  - 컨텐츠가 캐싱되고 유저에게 제공되는 지점
  - AWS가 CDN을 제공하기 위해 만든 서비스인 CloudFront의 캐시 서버(데이터 센터의 전 세계 네트워크)
  - CloudFront는 이 엣지 로케이션을 통해 컨텐츠를 제공
  - CloudFront를 통해 사용자 요청이 들어오면 가장 지연시간이 낮은 엣지 로케이션으로 라우팅 되어 컨텐츠 전송 성능이 뛰어나다.

- **CloudFront 동작방식**



  - AWS 백본 네트워크를 통해 컨텐츠를 가장 효과적으로 서비스할 수 있는 엣지로 각 사용자 요청을 라우팅 하여 배포속도를 높임.

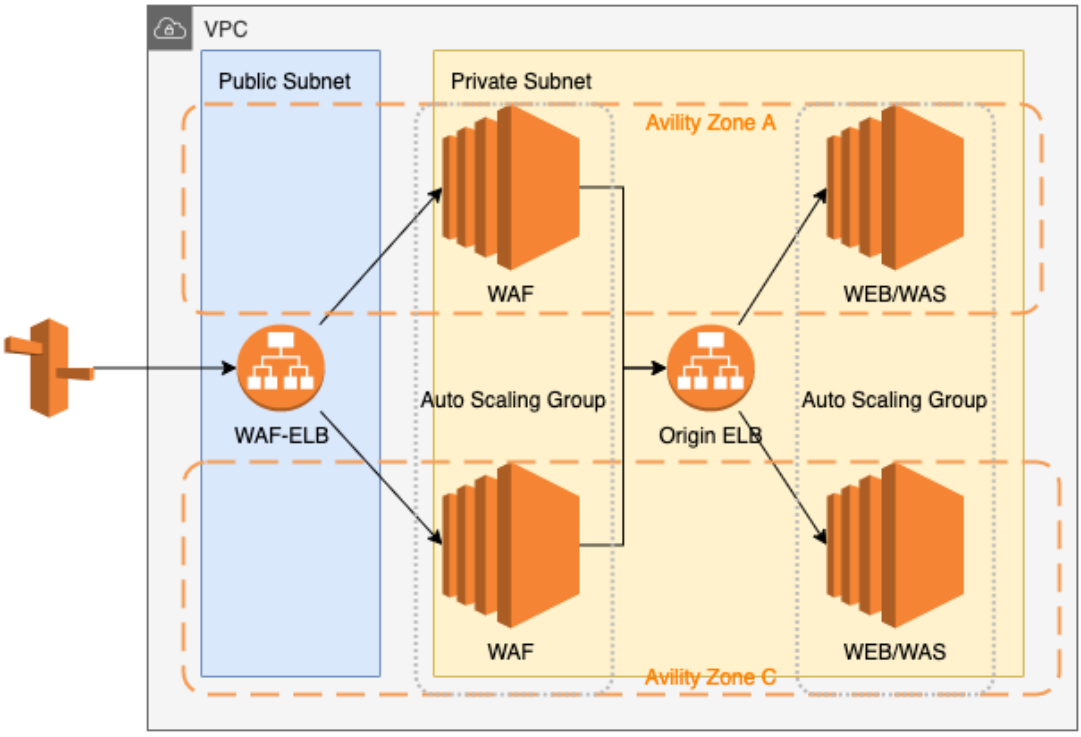
        - 컨텐츠가 엣지 로케이션에 없는 경우 : 컨텐츠의 최종 버전에 대한 소스로 지정된 오리진에서 컨텐츠 검색하여 제공받아 전달
        - 컨텐츠가 엣지 로케이션에 있는 경우 : 바로 전달
    - 엣지 로케이션 내에 서비스를 등록하는 AWS 서비스가 CloudFront

- **CloudFront 장점**
    - AWS 네트워크를 사용하면 사용자의 요청이 반드시 통과해야 하는 네트워크 수가 줄어 성능이 향상
    - 파일의 첫 바이트를 로드하는데 걸리는 지연시간이 줄어들고, 데이터 전송속도가 빨라짐
    - 파일의 사본이 전세꼐 여러 엣지 로케이션에 유지 되 므로 안정성과 가용성이 향상된다.
    - 보안성 향상
        - 오리진 서버에 대한 종단 간 연길의 보안이 보장 (HTTPS)
        - 서명된 URL 및 쿠키 사용 옵션으로 자체 사용자 지정 오리진에서 프라이빗 컨텐츠를 제공하도록 할 수 있음.

- **CloudFront 의 기능**
    - 정적 & 동적 컨텐츠 분별 제공 → 경로 패턴으로 URL에 따라 정적/동적 컨텐츠 분기처리
        - 정적(Static) 컨텐츠 → 캐싱으로 접근속도 최적화
        - 동적(Dynamic) 컨텐츠
            - 네트워크 최적화, 연결 유지, Gzip 압축 등을 사용
            - 서버와 통신시 전처리 작업 (DNS Lookup, TCP Connection, Time to First Byte 등)을 CloudFront 에서 네트워크 최적화
            - 내용 최적화가 아닌 통신을 최적화하여 속도를 최적화 시킴
    - HTTPS 지원

        - Origin에서 HTTPS 를 지원하지 않더라도 CloudFront 내 HTTPS 통신을 지원할 수 있도록 구성 가능 (Proxy 서버 역할)
    - 지리적 제한 설정
        - 특정 지역의 컨텐츠 접근을 제한 가능
    - 다른 서비스와 연계
        - AWS WAF, Lambda@Edge 등과 연동 가능하다

- **AWS WAF (Web Application Firewall) 란**



    - 고객의 정의한 조건에 따라 웹 요청을 허용, 차단 또는 모니터링하는 규칙을 구성하여 공격으로부터 웹 애플리케이션을 보호하는 웹 애플리케이션 방화벽

- **Lambda@Edge 란**
    - 엣지 로케이션에서 돌아가는 람다

        - 요청 및 응답 시간에 헤더를 변경.
        - 헤더에 쿠키 세부 정보를 추가. 요청 및 응답에 따라 AB 테스트를 수행.
        - 헤더 세부 정보를 기반으로 URL을 다른 사이트로 리디렉션
        - 헤더에서 사용자 에이전트를 가져오고 브라우저, OS 등의 세부 정보를 찾을 수 있다.

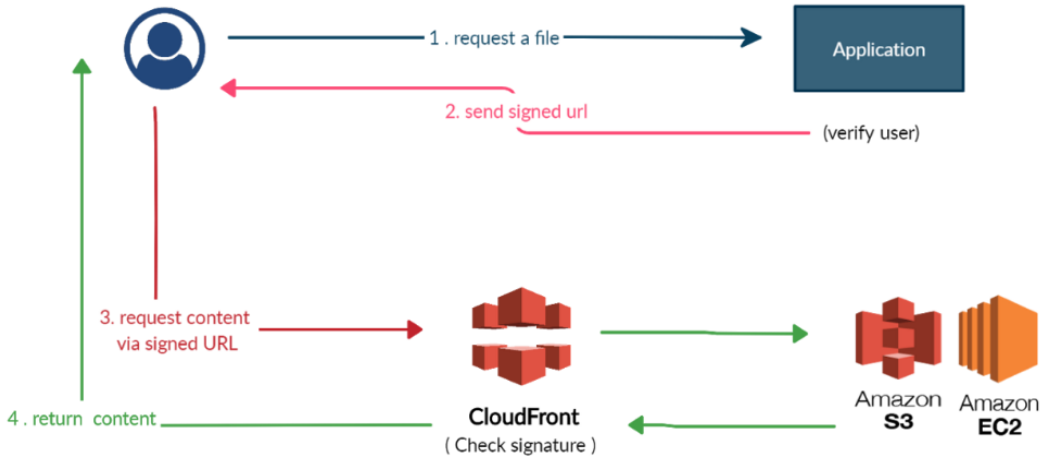
- **CloudFront Function**
    - Lambda@Edge의 6분의 1 비용으로 경량 javascript 실행
    - 아주 간단한 액션에 사용
    - 캐싱, 헤더조작

- **CloudFront 리포팅**
    - 주요 CloudFront 이용 지표 확인
        - 캐시 상태, 컨텐츠 요청 수, Top Referrer
    - CloudFront의 뷰어 정보는 헤더에 더해 Origin에 전송
        - 디바이스 타입
        - IP Address
        - Contry/도시/위도/경도/타임존 등

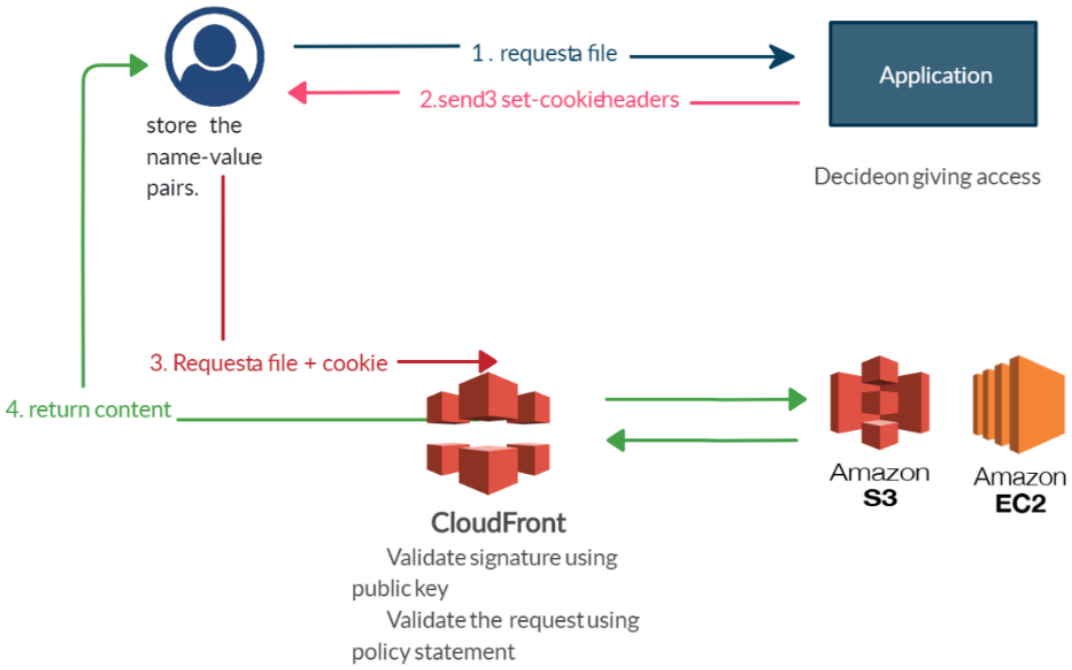
- **CloudFront 정책**
    - 캐시 정책 (Cache Control)
        - TTL 및 Cache Key 정책
        - CloudFront가 어떻게 캐싱 할지 결정
    - 원본 요청 정책 (Origin Request)
        - Origin에 쿠키 헤더 쿼리스트링중 어떤 것을 보낼지

- 응답 헤더 정책
  - CloudFront가 응답과 함께 실어보낼 HTTP Header

- **CloudFront 보안**
  - Signed URL



    - 어플리케이션에서 CloudFront의 컨텐츠에 접근할 수 있는 URL을 제공하여 컨텐츠 제공을 제어
    - URL에는 시작시간, 종료시간, IP, 파일명, URL 유효기간 등의 정보를 담을 수 있다.
    - 이 URL 이외의 접근을 막고, 허용된 유저에게만 URL을 전달하여 제어
    - 단 하나의 파일 또는 컨텐츠에 대한 허용만 가능
    - S3 Signed URL과 비슷한 방식
  - Signed Cookie



    - Signed URL이 하나의 컨텐츠만 제공 제어를 한다 하면, Signed Cookie 는 다수의 컨텐츠의 제공방식을 제어하고 싶을 때 사용
    - 다수의 파일 및 스트리밍 접근 허용 가능 (정기구독 프리미엄 유저만 볼 수 있는 강의 등)
  - Origin Access Identity (OAI)
    - S3 컨텐츠(정적 컨텐츠)를 CloudFront를 사용해서만 볼 수 있도록 제한하는 방법
    - S3 Bucket Policy로 CloudFront의 접근 허용을 통해 사용
  - Field Level Encryption
    - CloudFront로부터 Origin 사이의 통신을 암호화
    - 최대 10개 필드까지 가능
    - 공개키 방식으로 암호화 → CloudFront에 공개키를 제공 후 오리진에서 Private Key로 해독

- **CloudFront 무효화**
  - 파일 무효화 (invalidation) 란

        - TTL이 지나기 전 강제로 캐시를 삭제하는 것
        - 주로 새로운 파일을 업데이트 한 후 TTL을 기다리지 못할 상황에 사용
    - 추가 비용이 발생
    - CloudFront API, 콘솔, Third-Party 툴 등을 사용한 파일 무효화 기능 사용 가능

- **CloudFront 삭제**
    - 전세계 배포되는 서비스이기에 바로 삭제는 불가능하며 비활성화 후 삭제를 진행해야 한다. 이역시 시간이 오래걸림.