

Static Website Hosting

with AWS

박찬민.

박찬민

Front-end Developer

@버드뷰(화해)

Front-end 개발자, **완전한 하나의 서비스를 만들고 싶은 욕심**을 가진 자.

하지만, Back-end 의 깊고 넓은 세계보다는 **서비스에 집중**하고 싶은 자.

그래서 **AWS Serverless**의 힘을 빌려쓰는 자.

웹사이트를 만드는 대략적인 순서

- ❑ 웹사이트를 만든다.
- ❑ 서버를 만든다.
- ❑ 서버에 웹서버를 설치한다.
- ❑ 도메인을 연결한다.

웹사이트를 만드는 대략적인 순서

❑ 웹사이트를 만든다.

Front-end 개발자가 잘 하는 것

❑ 서버를 만든다.

❑ 서버에 웹서버를 설치한다.

Front-end 개발자가 잘 모르는 것

❑ 도메인을 연결한다.

웹서버를 만든다는 것,,,

- ❑ 서버를 만든다.
- ❑ 서버에 웹서버를 설치한다.

웹서버를 만든다는 것,,,

❑ 서버를 만든다.

❑ 어디에 만들지?

- ❑ 우리집 내 컴퓨터? 데이터센터? 카페24? 클라우드?

❑ 어느 정도 성능이 필요하지?

❑ 서버 비용은 얼마나 들어가는걸까?

❑ 서버에 웹서버를 설치한다.

❑ 톰캣? Express? Nginx?

❑ 안정적인 웹서버를 만들려면 이중화도 해야해나?

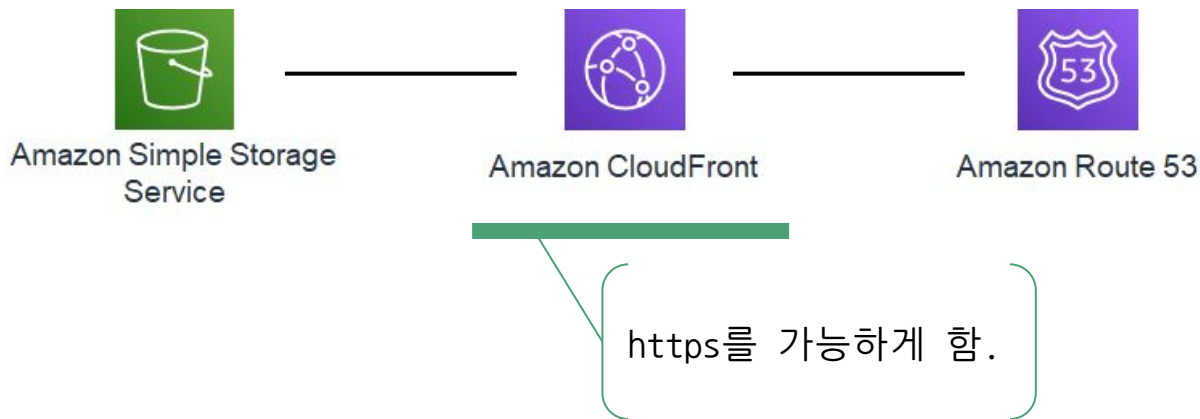
웹서버를 만든다는 것 with AWS

- ❑ S3에 파일들을 올리고 정적 웹 호스팅 기능을 켜준다!



웹서버를 만든다는 것 with AWS

- ❑ S3에 파일들을 올리고 정적 웹 호스팅 기능을 켜준다!



S3로 웹호스팅 하기

1. S3 Bucket 생성

(1) http > Bucket 이름을 사용할 도메인 주소로 설정

버킷 이름 ⓘ

static-web.walkinpcm.com

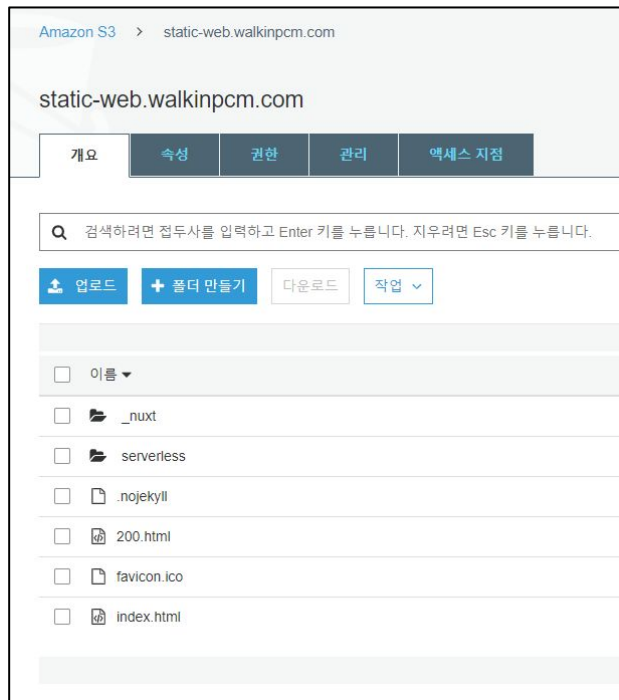
(2) https > Bucket 이름을 도메인 주소로 하지 않게 설정

버킷 이름 ⓘ

static-web-httpss

S3로 웹호스팅 하기

2. S3 Bucket에 웹사이트 파일 업로드설정



S3로 웹호스팅 하기

3. Public 권한 설정

(1) ‘ 퍼블릭 액세스 차단 ’ 설정 변경

모든 퍼블릭 액세스 차단
비활성화

편집

- ☒ 새 ACL(엑세스 제어 목록)을 통해 부여된 버킷 및 객체에 대한 퍼블릭 액세스 차단
활성화
- ☒ 임의의 ACL(엑세스 제어 목록)을 통해 부여된 버킷 및 객체에 대한 퍼블릭 액세스 차단
활성화
- ☐ 새 퍼블릭 버킷 또는 액세스 지점 정책을 통해 부여된 버킷 및 객체에 대한 퍼블릭 액세스 차단
비활성화
- ☐ 임의의 퍼블릭 버킷 또는 액세스 지점 정책을 통해 부여된 버킷 및 객체에 대한 퍼블릭 및 교차 계정 액세스 차단
비활성화

S3로 웹호스팅 하기

3. Public 권한 설정

(2) ‘ 버킷 정책 ’ 설정

```
1  {  
2    "Version": "2012-10-17",  
3    "Statement": [  
4      {  
5        "Sid": "AddPerm",  
6        "Effect": "Allow",  
7        "Principal": "*",  
8        "Action": "s3:GetObject",  
9        "Resource": "arn:aws:s3::static-web.walkinpcm.com/"  
10     }  
11   ]  
12 }
```

S3로 웹호스팅 하기

4. 정적 웹 호스팅 기능 enable

정적 웹 사이트 호스팅

✕

엔드포인트 : <http://static-web.walkingpcm.com.s3-website-ap-northeast-2.amazonaws.com>

☒ 이 버킷을 사용하여 웹 사이트를 호스팅합니다. [세부 정보](#)

인덱스 문서 [?](#)

오류 문서 [?](#)

리디렉션 규칙(선택 사항) [?](#)

☐ 요청 리디렉션 [세부 정보](#)

☐ 웹 사이트 호스팅 사용 안 함

☒ 버킷 호스팅

취소

저장

S3로 웹호스팅 하기



도메인 연결

1. Route53에서 새로운 레코드 생성

(*) S3 Bucket 이름과 동일한 도메인을 입력하면,
별칭 대상 목록에 해당하는 S3 Bucket이 나타남.

레코드 세트 생성

이름: static-web.walkinpc.com.

유형: A - IPv4 주소

별칭: ☒ 예 ☐ 아니요

별칭 대상:

- 리소스에 대 — S3 웹 사이트 엔드포인트 —
- CloudFront static-web.walkinpc.com (s3-website)
- Elastic Beanstalk ELB Application Load Balancer
- ELB Load Balancing ELB Classic Load Balancer
- S3 웹 사이트 사용 가능한 대상 없음
- 이 호스팅 ELB Network Load Balancer
- VPC 종단점 사용 가능한 대상 없음
- API Gateway 사용 가능한 대상 없음
- 2.amazonaws.com
- Global Accelerator 사용 가능한 대상 없음
- 자세히 알아 — CloudFront 배포 —

도메인 연결

`static-web.walkinpcm.com`

https 를 지원하려면,,,

- ❑ S3 버킷 이름이 도메인 형식이면 안됨!
- ❑ ACM(AWS Certificate Manager)에서 인증서 생성! 무료 !!!
- ❑ S3와 CloudFront를 연결
- ❑ Route53에서 별칭을 S3로 하지 않고 CloudFront로 설정

SSL 인증서 생성

1. ACM에서 인증서 생성

(*) 반드시 **버지니아 리전**에서 생성!!! (그래야 CloudFront에서 사용가능)

(*) 필요에 따라 도메인 설정

도메인 이름 추가

SSL/TLS 인증서(예: www.example.com)로 보호하고 싶은 사이트의 전체 도메인 이름과 서브도메인(예: www.example.com, site.example.com, images.example.com)을 보호합니다.

도메인 이름*

제거

✕

이 인증서에 다른 이름 추가

이 인증서에 추가 이름을 추가할 수 있습니다. 예를 들어, "www.example.com"

CloudFront 배포 생성

1. Web distribution 타입으로 생성

2. 설정해줄 속성

- Origin Domain Name: S3의 웹 호스팅 기능에서 할당된 도메인 **직접 입력**
- Viewer Protocol Policy: Redirect HTTP to HTTPS
- Alternate Domain Names(CNAMEs): 사용할 도메인 주소
- SSL Certificate: Custom SSL Certificate를 선택하고,
목록에서 ACM에서 생성한 인증서 선택
- Default Root Object: 보통 'index.html'

Route53 레코드 생성

1. CloudFront 배포를 생성할 때 사용했던 도메인 이름으로 생성
2. 별칭 목록에서 방금 생성한 CloudFront 배포 선택

`static-web-https.walkinpcm.com`

없는 URL로 접근한다면?

- ❑ 없는 URL로 접근하면 에러가 발생함.
(Origin에서 test 리소스를 찾을 수 없기 때문에)



없는 URL로 접근한다면?

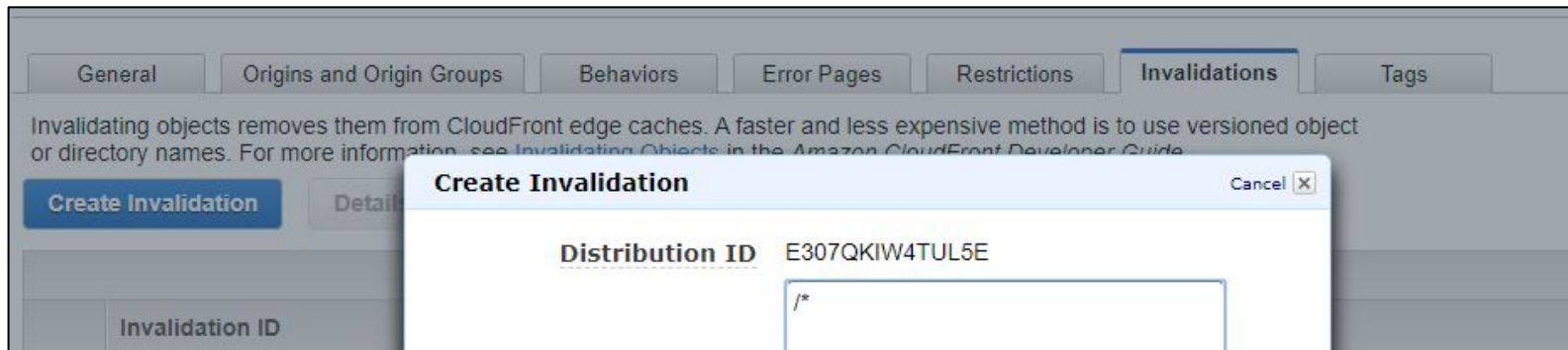
- ❑ CloudFront에 Custom Error Response를 설정 할 수 있음.
- ❑ 아래와 같이 403, 404 에러에 대해 설정.

Custom Error Response Settings

HTTP Error Code	404: Not Found ▼
Error Caching Minimum TTL (seconds)	300
Customize Error Response	<input checked="" type="radio"/> Yes <input type="radio"/> No
Response Page Path	/index.html
HTTP Response Code	200: OK ▼

새로운 빌드를 배포하려면?

- ❑ S3에 새로운 빌드파일 업로드
- ❑ 하지만 업로드 직후에 CloudFront에는 아직 예전 빌드파일이 남아있음
- ❑ CloudFront에서 새로운 빌드파일을 가져가게 하려면?
- ❑ Invalidations 기능을 이용! 기존에 저장된 파일을 삭제함.



S3 기반의 웹호스팅 장점

→ 서버 관리 불필요

서버를 프로비저닝하거나 유지 관리할 필요가 없습니다. 설치, 유지 또는 관리할 소프트웨어나 런타임이 없습니다.

→ 사용량만큼 지불

서버 단위가 사용량에 대해 요금을 지불합니다.

→ 자동화된고가용성

S3는 99.99%의 가용성과 99.999999999%의 내구성을 지원합니다. 그래서 별도로고가용성을 위한 설계를 할 필요가 없습니다.

“ 감사합니다 :) ”