



# AWS CloudFront

## Table of Contents

- ▶ What is CloudFront?
- ▶ CloudFront Infrastructure
- ▶ How Does CloudFront Work?
- ▶ CloudFront Pricing
- ▶ Hands on Scenario

1

## What is CloudFront?

Cloudfront CDN dedigimiz content delivery network (icerik dagitim agi olarak cevirebiliriz).

### ▶ CloudFront What is CloudFront?



Static veya dinamik bir websitemiz oldugunu varsayalım. Bunun icerisinde videolar, imajelar var diyelim. Biz bunlari tum dunyaya yaymak istiyoruz. Ya da regional/2-3 regional bir yayin da yapabiliriz. Bunun icin request gondermek isteyen kisi bulunduгу ulkeden sizin yayini yapmak istediginiz ulkeye istek gonderiyor API araciligiyla ve sonrasında gorebiliyor. Ama mesela Amerikadan baglanmak istediginde birisi avrupaya bir zaman kaybi oluyor. Bunun icin diyoruz ki kaynaklar bir originde olsun ama ben bunu dunyanin her yerinde cacheleyim. Kullanicilar hizlica iceriklere ulasmis olsun boylece. Iste bunu cloudfrontla yapiyoruz.

- CloudFront is the **Content Delivery Network (CDN)** of AWS.
- Basically, it is a globally distributed network located on different geographical places.

## ► CloudFront

### What is CloudFront?



- The aim of this service is to **provide faster distribution** to the dynamic or the static web contents.
- World-wide distributed data centers which are called **edge locations** are the backbone of CloudFront.

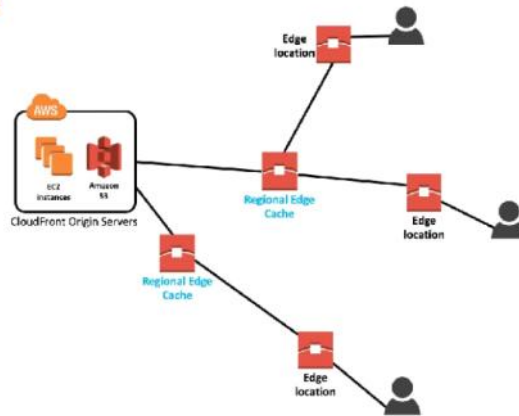
Bunu saglayan seyde Amazonun alametifarikasi tum dunyaya okyanus altindan internet aglarini ormus, dunyanin dort bir kosesine kendi agni yapmis. Altyapiyi yamis bir de onun uzerine servisler inisa etmis. Cloudfrontun kuruldugu sey tamamiyle edge locations uzerine. Edge locationlarin asil amazi cachelemek. Yani belli bir TTL boyunca hafiada tutmak. Bilgi yenilendiginde kendinde var olan bilgiyi guncelliyor. Yeni bilgiyi de belirledigimiz sure dolduktan sonra olabliyoruz.

## 2 CloudFront Infrastructure

### ► CloudFront

#### CloudFront Infrastructure

- AWS CloudFront has a world-wide network of **600+ Points of Presence (PoP)\***:
  - 600+ Edge Locations (POP)
  - 13 Regional Edge Caches
  - in 100+ cities
  - across 50+ countries



Edge locationlara ayni zamanda Points of Presence (PoP) de deniliyor. Haricinde 500 den fazla edge location var. Bunlarin haricinde 13 tane de cachelemelerin yapildiği edge locationlari var. bunlar edge locationlardan daha buyuk. Burada daha fazla malzeme depolanmis oluyor ve bu da bize hiz sagliyor. Normalde originden ulasmak istese daha uzun zaman alabilceken edge locationlardan aws in kendi agini kullanarak en yakin edge locationa gidiyor ve hizlica oradan istedigii bilgiyi temin ediyor.

\*as of April 2024.

CLARUSWAY  
WAY TO REINVENT YOURSELF

## ► CloudFront

### CloudFront Infrastructure

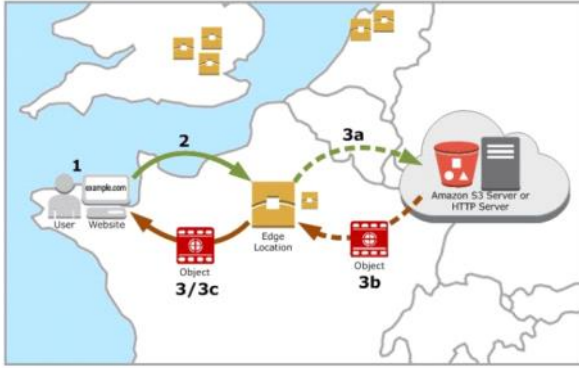


- These facilities are spread over 100+ cities across 50 countries\*.  
\* Sep 2023

3

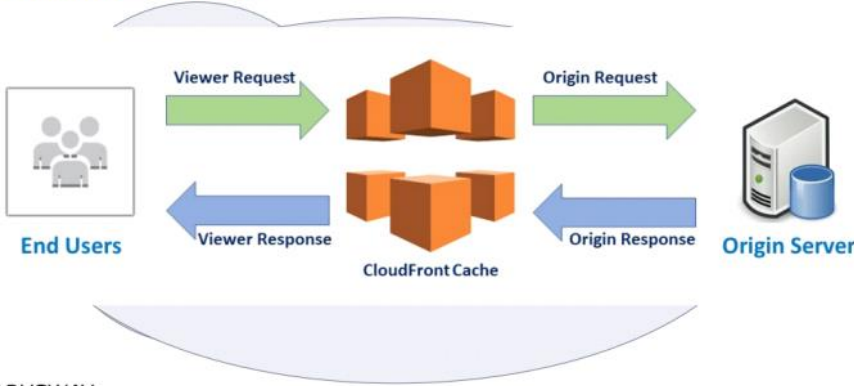
# How does CloudFront work?

## CloudFront How does it work?



Bir noktadan bir user example.com sitesinde girmek istedigini varsayalım. Bu içerikler de Almanyada S3 Kolinde tutuluyor. Normalde userin kolindeki içeriklere ulasmasi zaman alacak bir durum. Biz edge locationlari ayarliyoruz. Buralara bilgileri içerikleri cacheliyoruz. Bu siteye girmek isteyen ilk kisi istek yolladiginda artik istegi edge location karsiliyor. Bu bilgi yoksa hemen originden bilgiyi cekiyor ve daha sonra ayni isteklere ededen cevap veriyor. Yani artik bu yolculuk her defasinda yapilmak yerine daha kısa bir sekilde bilgiye ulasilmis olunuyor.

## CloudFront How does it work?



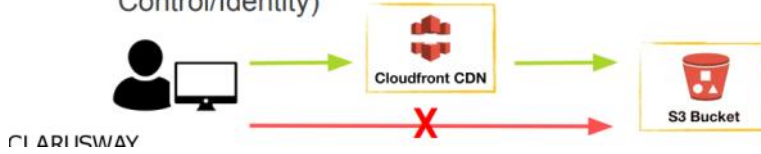
Enduser bizim contentimize ulasan kisi. Origin server AWS icinde de olabilir http bir adres olabilir yani bizim içeriklerimizin olduğu yer. Bu olmazsa cloudfront hicbir işe yaramaz. Amacimiz zaten buradan bilgileri dunyaya yaymak. Originin elinde dagitabilecegi birsey varsa cloudfront çalışır. Cloudfrontun tek amaci dagitmak. Tum dunyaya ya da regional bir planlama dahilinde dagitmak. Viewer request end userin yani ilk kullanici attigi ilk request. Origin request olmasaydi bu viewer request direkt olarak origin servere gidecekti. Viewer requesti karsilayan cloudfront edge locationimiz oluyor. Bu cacheleme eger yapildiyse origin servera gitmeden viewer responsini veriyor ama yoksa eger requesti alip originden response alip end usera iletıyor.

CLASHWAY

## CloudFront

### Origins

- AWS Origin
  - Elastic Load Balancer
  - MediaStore Container
  - MediaPackage Container
  - API Gateway
  - S3 Bucket (Enables OAC/OAI-Origin Access Control/Identity)
- Custom Origin (HTTP)
  - EC2 Instance
  - Any HTTP server
  - S3 Static Website



Cloudfrontun birseyi deliver edebilmesi için bir origin bir merkez lazım. Buraada iki farklı originimiz var. 1- AWS in kendi servisleri. 2- custom origin Buarada sunu dusunebiliriz neden EC2 customin altinda? Bu bir AWS servisi degil mi? Evet ec2 AWS kendi servisi ancak EC2lerin ayaga kalkmasi için kullandigi Ipler kimsenin sahsinin degil. Dunyanin mali gibi dusunebiliriz. Bu yuzden AWS altinda degil custom origin altinda goruyoruz. AWS altinda elastic load balancer var. load balancerlarda bir endpoint var ve ona gore dagitim yaptirabiliyoruz. MediaStore container ve mediapackage container canli medya paylasimi yapabilecegimiz servisler. S3 bucket ile S3 static website arasidaki fark ne? normalde s3 e bucket üzerinden ulasmak demek API uzerinden ulasmak demek. API request edip oradan ulasim gerceklestiyor. Yani s3 bucketteki imagea ulastigimiz zaman REST API kullanmis oluyoruz. Ama S3un static websitesiyle ayaga kaldirilmis bir siteyi acip onun icindeki image a ulastigim zaman website endpointi kullanmis oluyoruz. Bir bucket icindeki image ulasmak için https baglantisi oluyordu ama static website oldugunda http baglantisi oluyor ki bu https gibi guvenli degil.

S3 bucketta allow ya da denyler koyabiliyoruz s3 static websitea koyamadigimiz bir durum. S3 bucketta sunu duyebiliyoruz mesela: sadece cloud icinde ulasim olsun ama disaridan kimse ulasmasin. Iste buna OAC/OAI yani origin access control/identiy deniliyor.

### Key differences between a website endpoint and a REST API endpoint

An Amazon S3 website endpoint is optimized for access from a web browser. The following table summarizes the key differences between a REST API endpoint and a website endpoint.

Key difference	REST API endpoint	Website endpoint
Access control	Supports both public and private content	Supports only publicly readable content
Error message handling	Returns an XML-formatted error response	Returns an HTML document
Redirection support	Not applicable	Supports both object-level and bucket-level redirects
Requests supported	Supports all bucket and object operations	Supports only GET and HEAD requests on objects
Responses to GET and HEAD requests at the root of a bucket	Returns a list of the object keys in the bucket	Returns the index document that is specified in the website configuration
Secure Sockets Layer (SSL) support	Supports SSL connections	Does not support SSL connections

## CloudFront

### Signed URLs & Signed Cookies

- Signed URLs
  - Restrict access to individual files
  - For clients that doesn't support cookies
- Signed Cookies
  - Provide access to multiple restricted files
  - No changes in your current URL



Biz S3 te kisa sureli goruntulemek istedigimizde presigned URL kullaniyorduk. Burada AWS bize token veriyordu linki inceledigimiz zaman gorduk bunu. O token ile ozel olarak giris saglamis oluyorduk. Signed URL zaten bu anlama geliyor, imzalanmis URL. Iste biz cloufrontta da bu signed URL olayini ayarlayabiliyoruz. Aslinda ekstradan para kazanma yontmi gibi. Yani mesela karsimiza bir reklam cikti. Hem sarki soyleyip hem klip ceken programi 100 dolara gormek için tiklayin vs gibi. O tikladigimizda acilan URL signed URL. Biz tikladigimizda karsimiza cikan o URL i kimse gormuyor. Signed URL de objeye gore bir URL yaratildi. Bu tek bir obje için gecerli. Ama mesela bir egitim programi duzenledim ve bu egitime uye olan parasini odayen insanlar bu egitim videolarini ucretsiz izleyebilir. 1000 tane video var diyelim. Her biri için URL olusturmak yerine bir cookies geliyor ve bu cookies kabul ettigimiz zaman bu linke girdigimizde cookies arkada kinki onayliyor ve izni veriyor.

SINAVLARDAYLA SOYLE GELEBILIR!!!! GIZLI ICERIKLERIMIZ VAR. BUNU PAYLASMAK ISTIYORUZ AMA YENI BIR URL OLUSTURMAK ISTEMİYORUZ. BUNU NASIL YAPARIZ GIBI BIR SORU CIKABILIR KI CEVAP SIGNED COOKIES.

## 4 CloudFront Pricing

## ► CloudFront Pricing

Free Tier	CloudFront Savings Bundle	Custom Pricing
<p>Always free</p> <p>1 TB of data transfer out 10,000,000 HTTP or HTTPS Requests 2,000,000 CloudFront Function Invocations</p> <p>Each month</p>	<p>Amazon CloudFront charges traffic served based on the following dimensions. The CloudFront Security Savings Bundle is a flexible self-service pricing plan that helps you save up to 30% on your CloudFront bill in exchange for a monthly spend commitment for a one-year term. This savings is not limited to data delivered by CloudFront but applies to all CloudFront usage types, including CloudFront Functions and Lambda@Edge. The CloudFront Security Savings Bundle also includes free AWS Web Application Firewall (WAF) usage up to 10% of your committed amount.</p>	<p>Custom Pricing</p> <p>Custom discounted pricing is available for customers willing to commit to a minimum of 10 TB of data transfer per month for 12 months or longer. Discounts vary based on the amount of the commitment. Interested in signing up for discounted pricing?</p> <p>Contact Us</p>

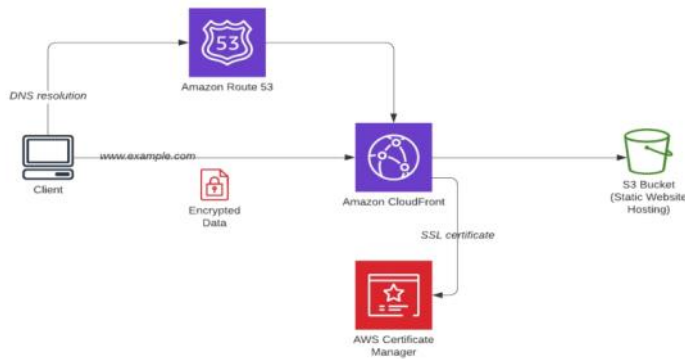
## ► CloudFront Pricing



Region bazlı bir ücretlendirme söz konusu ama biz free tierdan faydalaniyoruz suanda 1 TB a kadar.

- Pricing varies depending on **region** and **usage**. Thus, to be more cost effective setup your configuration regarding your goal, methods, end users and geographic locations.

## ► CloudFront Hands on Scenario



Burada amac bicim iceriklerimizi cloudfront araciligiyla dunyanin herhangi bir yerine hizlica iletmek. Bunun icin icerik lazim dedik. Biz s3 bucket kullanacagiz. Biz static website olusturacagiz ve onun uzerinden bir image gosterimi yapacagiz. Amazon cloudfront ismi d ile baskar amazon-cloudfront.net ile biter. Http olan sitemizi https yapmak icin bir certificate belirleyecegiz guvenlik icin. Daha sonra endpointimiz olan adresle route53 uzerinden yayin yapacagiz cunku musterinin bunu kolayca aklinda tutmasini istiyoruz diger turlu bize verdigi adres aklida kalir bir adres degil.



### Settings

**Price class** [Info](#)  
Choose the price class associated with the maximum price that you want to pay.

☒ Use all edge locations (best performance)  
☐ Use only North America and Europe  
☐ Use North America, Europe, Asia, Middle East, and Africa

**Alternate domain name (CNAME) - optional**  
Add the custom domain names that you use in URLs for the files served by this distribution.

[Remove](#)

[Add item](#)

[To add a list of alternative domain names, use the bulk editor.](#)

**Custom SSL certificate - optional**  
Associate a certificate from AWS Certificate Manager. The certificate must be in the US East (N. Virginia) Region (us-east-1).

[Choose certificate](#) [Refresh](#)

[Request certificate](#)

Burada naked domain namei yazmak onemli aksi takdirde route53 bunu algilamiyor ve yayın yapmiyor.

**Custom SSL certificate - optional**  
Associate a certificate from AWS Certificate Manager. The certificate must be in the US East (N. Virginia) Region (us-east-1).

[Choose certificate](#) [Refresh](#)

**None**

**ACM certificates**

blog.clarusway.us (a65c114e-1477-4012-876b-759cd2484cdb)
clarusway.us (bef3302a-50d0-4025-a62d-dd230e665719)
*.clarusway.us (2418b374-8fb9-4c1b-b711-cc0d3a1218c4)
*.clarusway.us (f872f3db-ba9d-46df-8c6a-9d165036e8ee)
clarusway.us (8d77bf6f-d498-4a7a-90dd-9371e4daafac)

**IAM certificates**

Burada da aldığımız sertifikayı seçiyoruz.  
\* li domain değil naked olanı seçeriz.

### Create record [Info](#)

**Quick create record** [Switch to wizard](#)

**Record 1** [Delete](#)

**Record name** [Info](#)   **Record type** [Info](#)

Keep blank to create a record for the root domain.

☒ Alias

**Route traffic to** [Info](#)

[Choose endpoint](#) [Choose Region](#)

**Routing policy** [Info](#)

[Add another record](#)

Route53 te yeni bir record olustururken amazonun bir servisinden fadalanacagimiz icin alias kullaniyoruz.  
Endpointi cloudfront seciyoruz ve region otomatik olarak geliyor.