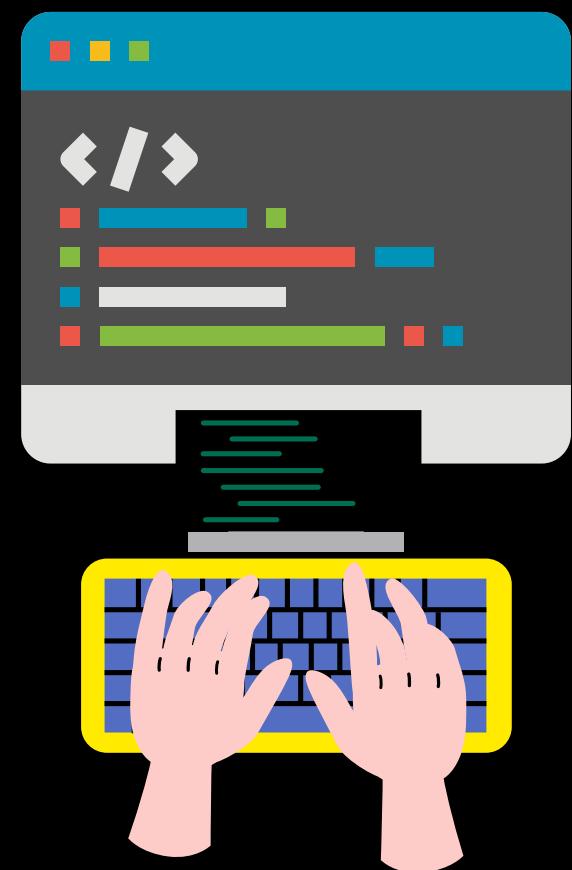




Amazon CloudFront

example.com



**I want to serve my
website Globally with
low latency**



Website can be simple like this...

Welcome to My Simple Website

[About](#) [Services](#) [Contact](#)

About Us

We are a lightweight, fast-loading website designed to demonstrate the basics of HTML and CSS. Our goal is simplicity and speed, making sure users can access information quickly.

Our Services

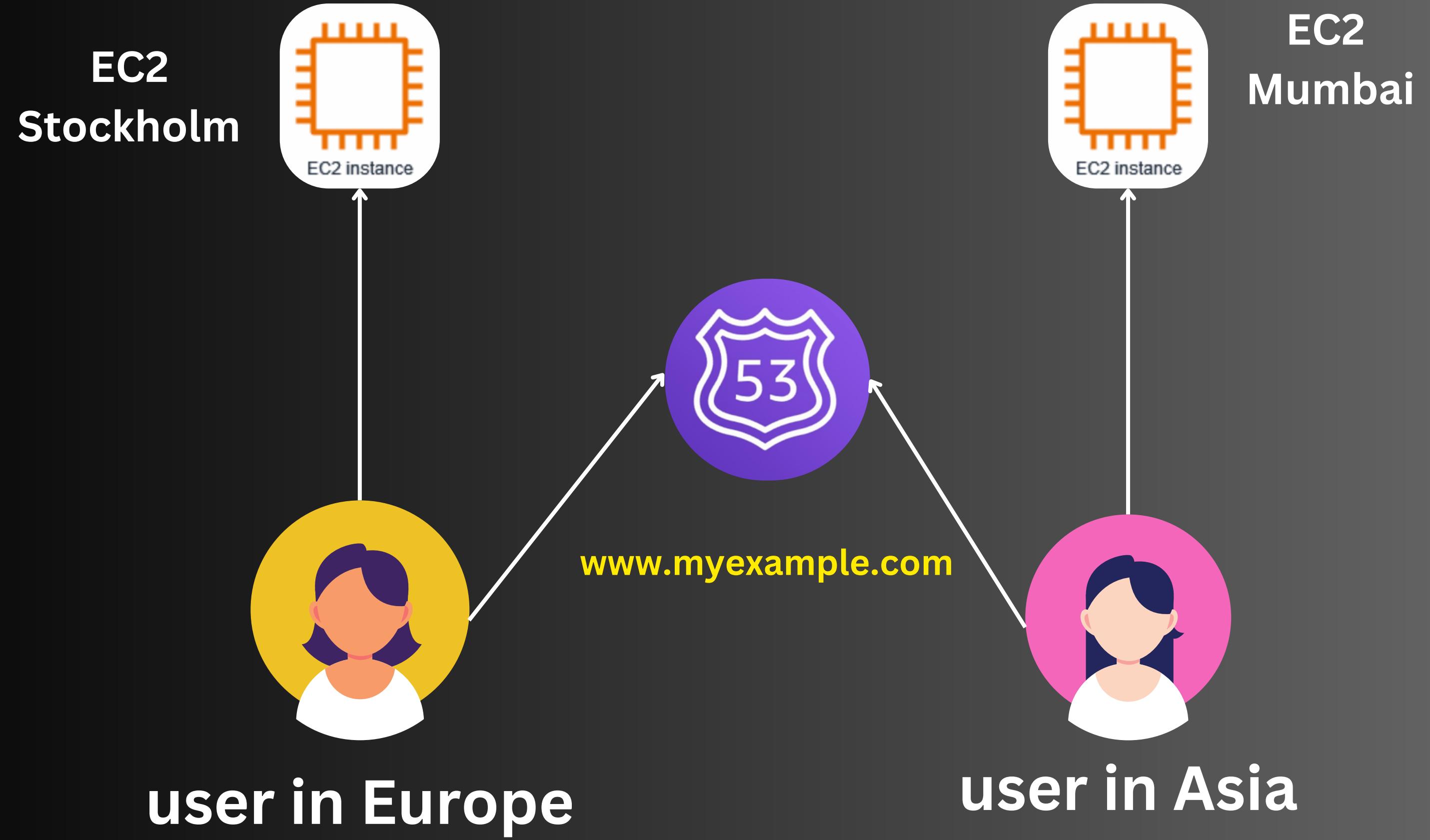
- Fast Loading Web Pages
- Minimalist Design
- Optimized Content Delivery

Website can be heavy like this...

Home About Services Contact

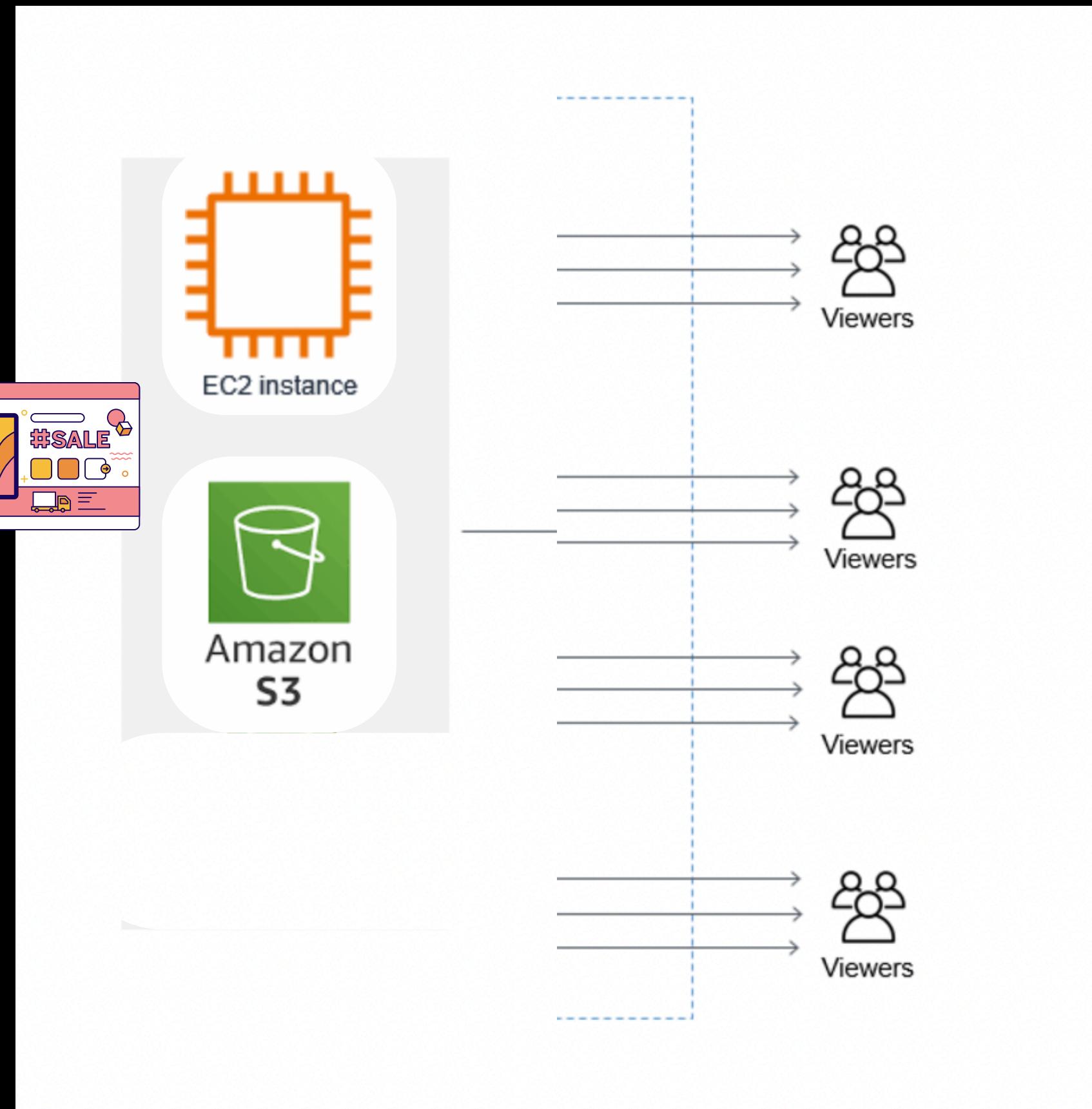
Welcome to the World of Nature

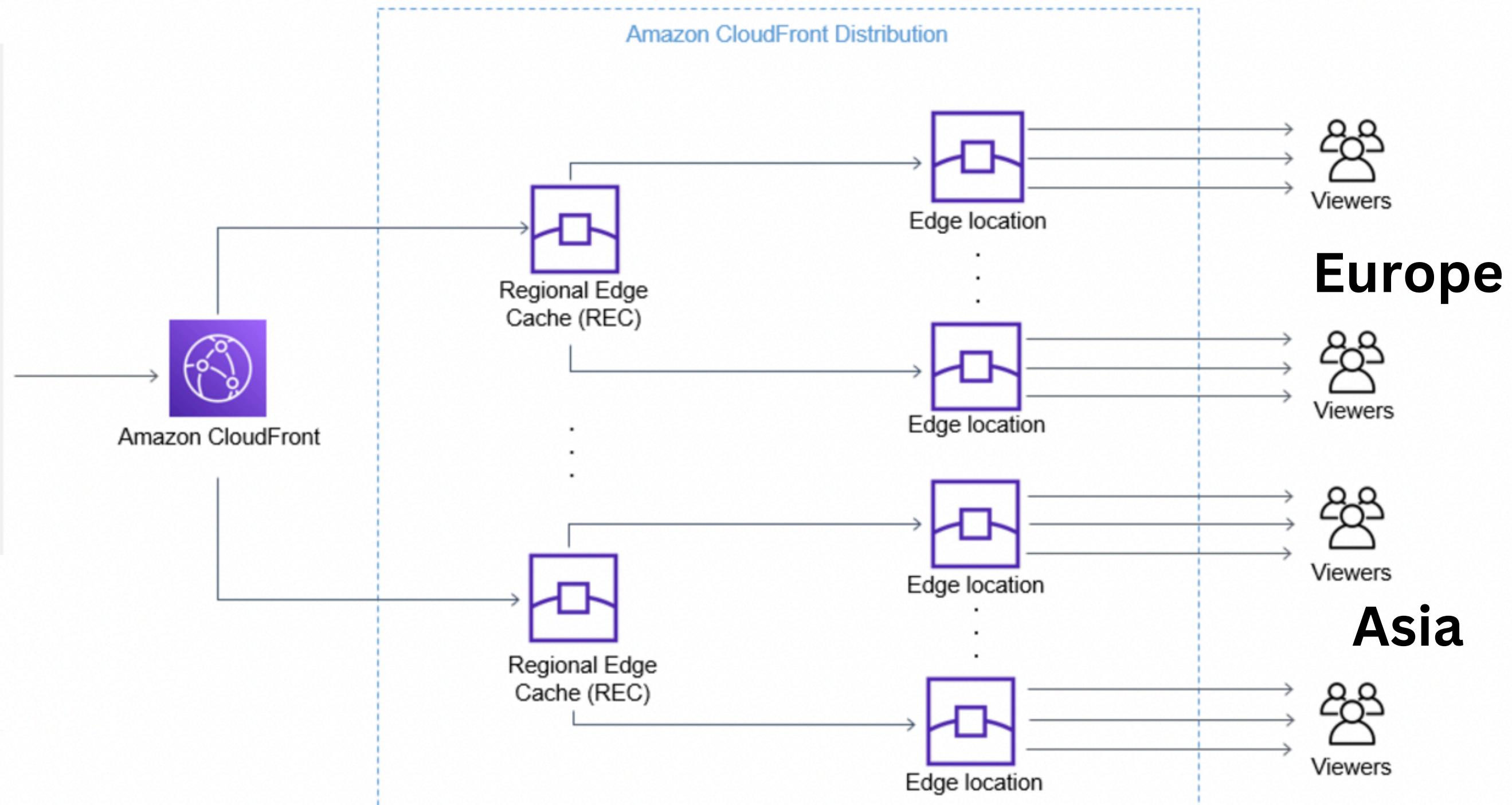
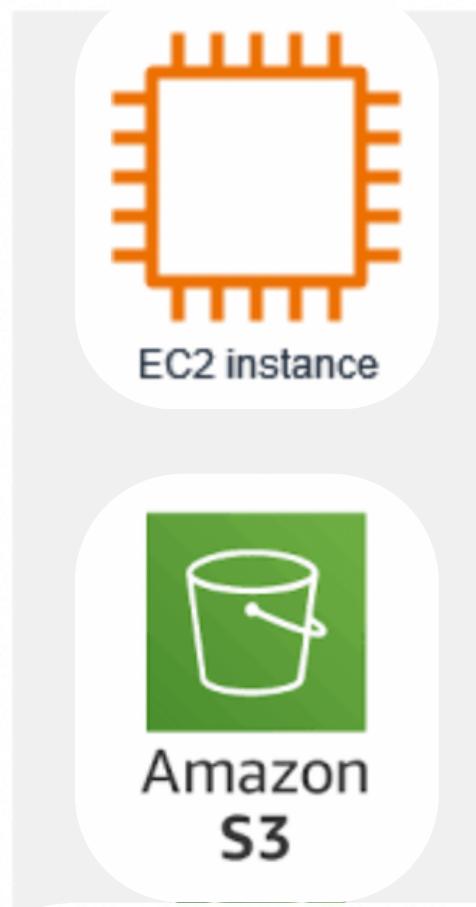




Solution

AWS CloudFront is a Content Delivery Network (**CDN**) that speeds up the delivery of web content to users by caching it at servers (edge locations) close to them, improving load times and performance globally.





AWS CloudFront primarily caches static content like **images, CSS, JavaScript, and videos**. It can also cache dynamic content (e.g., HTML or API responses) if configured with caching policies and headers.

By default, sensitive or user-specific data and backend logic are not cached. Cache behavior is controlled via TTLs, cache behaviors, and origin headers.



Key Features of a Content Delivery Network – Performance, Security – Amazon CloudFront

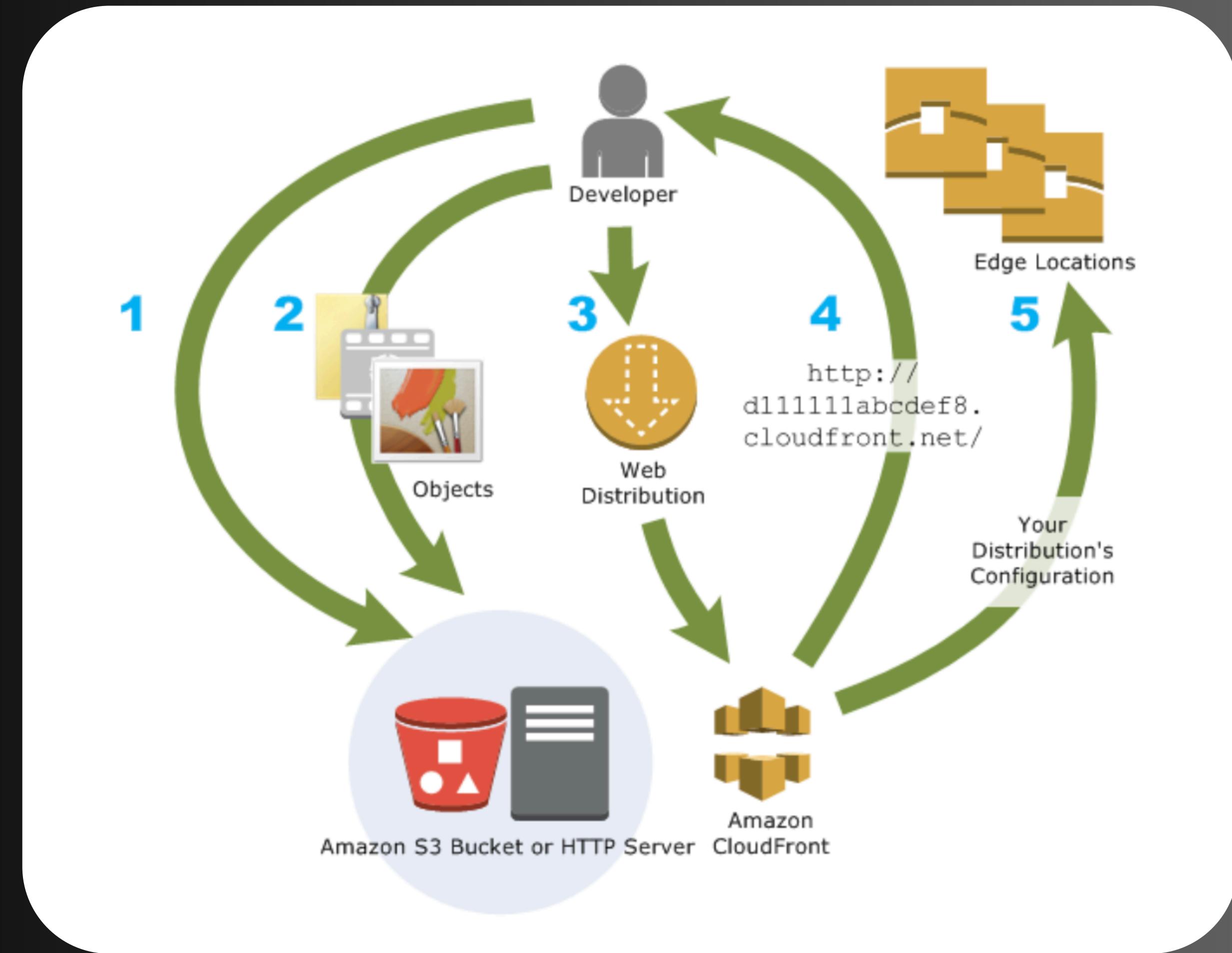
Learn about the key features for Amazon CloudFront's global content delivery network (CDN).

Amazon CloudFront has three types of infrastructure to securely deliver content with high performance to end users:

- CloudFront Regional Edge Caches (RECs) are situated within AWS Regions, between your applications' web server and CloudFront Points of Presence (POPs) and embedded Points of Presence. CloudFront has 13 RECs globally.
- CloudFront Points of Presence are situated within the AWS network and peer with internet service provider (ISP) networks. CloudFront has 600+ POPs in 100+ cities across 50+ countries.
- CloudFront embedded Points of Presence are situated within internet service provider (ISP) networks, closest to end viewers. In addition to CloudFront POPs, there are 600+ embedded POPs across 200+ cities in North America, Europe, and Asia.

Browsers act like a mini-CDN by caching website files (like images, CSS, and JavaScript) locally on a user's device, which speeds up loading for repeat visits.

Only helps individual users.



Included in Always Free Tier

- 1 TB of data transfer out to the internet per month
- 10,000,000 HTTP or HTTPS Requests per month
- 2,000,000 CloudFront Function invocations per month
- 2,000,000 CloudFront KeyValueStore reads per month
- Free SSL certificates
- No limitations, all features available

Pricing

CloudFront charges for data transfers out from its edge locations, along with HTTP or HTTPS requests. Pricing varies by usage type, geographical region, and feature selection.

The data transfer from your origin to CloudFront is always free when using AWS origins like Amazon Simple Storage Service (Amazon S3), Elastic Load Balancing, or Amazon API Gateway. You are only billed for the outbound data transfer from CloudFront to the viewer when using AWS origins.

For more information, see [CloudFront pricing](#) and the Billing and Savings Bundle [FAQs](#).

Key Differences

Feature	CloudFront	Multi-Location Hosting
Performance	Optimized for static content and caching.	Optimized for dynamic content near users.
Cost	Pay-per-use, often cheaper.	Higher costs for server and database setup.
Ease of Use	Easy to set up, minimal management.	Requires managing multiple server instances.
Scalability	Auto-scales globally.	Requires manual scaling per location.
Content Freshness	Cached content may require invalidation.	Dynamic content is always current.
Compliance	Less control over data residency.	Full control over where data is hosted.





CDN Cloud Service - Amazon CloudFront - AWS

Amazon CloudFront is a content delivery network (CDN) service that helps you distribute your static and dynamic content quickly and reliably with high speed performance, security, and developer ease-of-use.

```
#!/bin/bash
sudo yum update -y
```

```
# Install Apache web server (httpd)
sudo yum install -y httpd
sudo systemctl start httpd
sudo systemctl enable httpd
```

```
# Create a simple HTML file to verify the web server is running
echo "<html><h1>Welcome to Apache Web Server on Amazon Linux!</h1></html>" >
/var/www/html/index.html
```