Amazon CloudFront

General CloudFront Concepts

CloudFront is a web service that gives businesses and web application developers an easy and cost-effective way to distribute content with low latency and high data transfer speeds.

CloudFront is a good choice for distribution of frequently accessed static content that benefits from edge delivery—like popular website images, videos, media files or software downloads.

Used for dynamic, static, streaming, and interactive content.

CloudFront is a global service:

* Ingress to upload objects.
* Egress to distribute content.

Amazon CloudFront provides a simple API that lets you:

* Distribute content with low latency and high data transfer rates by serving requests using a network of edge locations around the world.
* Get started without negotiating contracts and minimum commitments.

You can use a zone apex name on CloudFront.

CloudFront supports wildcard CNAME.

Supports wildcard SSL certificates, Dedicated IP, Custom SSL and SNI Custom SSL (cheaper).

Supports Perfect Forward Secrecy which creates a new private key for each SSL session.

Edge Locations and Regional Edge Caches

An edge location is the location where content is cached (separate to AWS regions/AZs).

Requests are automatically routed to the nearest edge location.

Edge locations are not tied to Availability Zones or regions.

Regional Edge Caches are located between origin web servers and global edge locations and have a larger cache.

Regional Edge Caches have larger cache-width than any individual edge location, so your objects remain in cache longer at these locations.

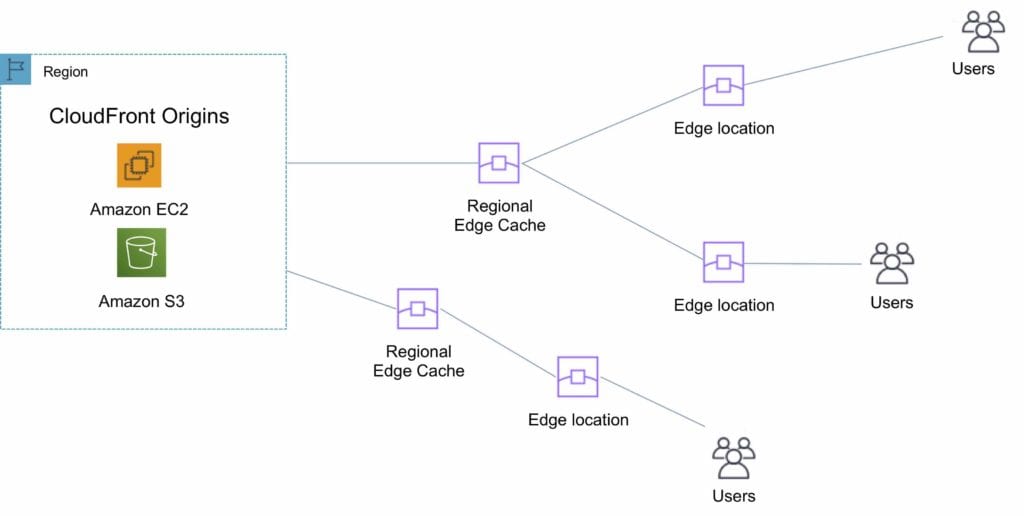
Regional Edge caches aim to get content closer to users.

Proxy methods PUT/POST/PATCH/OPTIONS/DELETE go directly to the origin from the edge locations and do not proxy through Regional Edge caches.

Dynamic content goes straight to the origin and does not flow through Regional Edge caches.

Edge locations are not just read only, you can write to them too.

The diagram below shows where Regional Edge Caches and Edge Locations are placed in relation to end users:



Origins

An origin is the origin of the files that the CDN will distribute.

Origins can be either an S3 bucket, an EC2 instance, an Elastic Load Balancer, or Route 53 – can also be external (non-AWS).

When using Amazon S3 as an origin you place all of your objects within the bucket.

You can use an existing bucket and the bucket is not modified in any way.

By default all newly created buckets are private.

You can setup access control to your buckets using:

* Bucket policies.
* Access Control Lists.

You can make objects publicly available or use CloudFront signed URLs.

A custom origin server is a HTTP server which can be an EC2 instance or an on-premise/non-AWS based web server.

When using an on-premise or non-AWS based web server you must specify the DNS name, ports and protocols that you want CloudFront to use when fetching objects from your origin.

Most CloudFront features are supported for custom origins except RTMP distributions (must be an S3 bucket).

When using EC2 for custom origins Amazon recommend:

* Use an AMI that automatically installs the software for a web server.
* Use ELB to handle traffic across multiple EC2 instances.
* Specify the URL of your load balancer as the domain name of the origin server.

S3 static website:

* Enter the S3 static website hosting endpoint for your bucket in the configuration.
* Example: http://<bucketname>.s3-website-<region>.amazonaws.com.

Objects are cached for 24 hours by default.

The expiration time is controlled through the TTL.

The minimum expiration time is 0.

Static websites on Amazon S3 are considered custom origins.

AWS origins are Amazon S3 buckets (not a static website).

CloudFront keeps persistent connections open with origin servers.

Files can also be uploaded to CloudFront.

**High availability with Origin Failover:**

* Can set up CloudFront with origin failover for scenarios that require high availability.
* Uses an origin group in which you designate a primary origin for CloudFront plus a second origin that CloudFront automatically switches to when the primary origin returns specific HTTP status code failure responses.
* For more info, check this [**article**](https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/high_availability_origin_failover.html).
* Also works with Lambda@Edge functions.

Distributions

To distribute content with CloudFront you need to create a distribution.

The distribution includes the configuration of the CDN including:

* Content origins.
* Access (public or restricted).
* Security (HTTP or HTTPS).
* Cookie or query-string forwarding.
* Geo-restrictions.
* Access logs (record viewer activity).

There are two types of distribution.

Web Distribution:

* Static and dynamic content including .html, .css, .php, and graphics files.
* Distributes files over HTTP and HTTPS.
* Add, update, or delete objects, and submit data from web forms.
* Use live streaming to stream an event in real time.

RTMP:

* Distribute streaming media files using Adobe Flash Media Server’s RTMP protocol.
* Allows an end user to begin playing a media file before the file has finished downloading from a CloudFront edge location.
* Files must be stored in an S3 bucket.

To use CloudFront live streaming, create a web distribution.

For serving both the media player and media files you need two types of distributions:

* A web distribution for the media player.
* An RTMP distribution for the media files.

S3 buckets can be configured to create access logs and cookie logs which log all requests made to the S3 bucket.

Amazon Athena can be used to analyze access logs.

CloudFront is integrated with CloudTrail.

CloudTrail saves logs to the S3 bucket you specify.

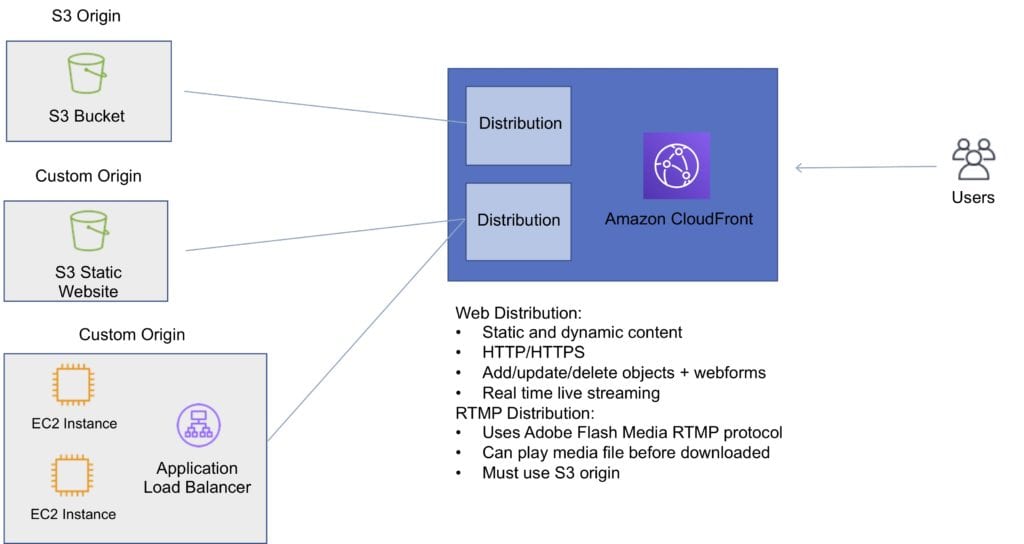
CloudTrail captures information about all requests whether they were made using the CloudFront console, the CloudFront API, the AWS SDKs, the CloudFront CLI, or another service.

CloudTrail can be used to determine which requests were made, the source IP address, who made the request etc.

To view CloudFront requests in CloudTrail logs you must update an existing trail to include global services.

To delete a distribution it must first be disabled (can take up to 15 minutes).

The diagram below depicts Amazon CloudFront Distributions and Origins:



Cache Behavior

Allows you to configure a variety of CloudFront functionality for a given URL path pattern.

For each cache behavior you can configure the following functionality:

* The path pattern (e.g. /images/\*.jpg, /images\*.php).
* The origin to forward requests to (if there are multiple origins).
* Whether to forward query strings.
* Whether to require signed URLs.
* Allowed HTTP methods.
* Minimum amount of time to retain the files in the CloudFront cache (regardless of the values of any cache-control headers).

The default cache behavior only allows a path pattern of /\*.

Additional cache behaviors need to be defined to change the path pattern following creation of the distribution.

You can restrict access to content using the following methods:

* Restrict access to content using signed cookies or signed URLs.
* Restrict access to objects in your S3 bucket.

A special type of user called an Origin Access Identity (OAI) can be used to restrict access to content in an Amazon S3 bucket.

By using an OAI you can restrict users so they cannot access the content directly using the S3 URL, they must connect via CloudFront.

You can define the viewer protocol policy:

* HTTP and HTTPS.
* Redirect HTTP to HTTPS.
* HTTPS only.

You can define the Allowed HTTP Methods:

* GET, HEAD.
* GET, HEAD, OPTIONS.
* GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE.

For web distributions you can configure CloudFront to require that viewers use HTTPS.

Field-Level Encryption:

* Field-level encryption adds an additional layer of security on top of HTTPS that lets you protect specific data so that it is only visible to specific applications.
* Field-level encryption allows you to securely upload user-submitted sensitive information to your web servers.
* The sensitive information is encrypted at the edge closer to the user and remains encrypted throughout application processing.

Origin policy:

* HTTPS only.
* Match viewer – CloudFront matches the protocol with your custom origin.
* Use match viewer only if you specify Redirect HTTP to HTTPS or HTTPS only for the viewer protocol policy.
* CloudFront caches the object once even if viewers makes requests using HTTP and HTTPS.

Object invalidation:

* You can remove an object from the cache by invalidating the object.
* You cannot cancel an invalidation after submission.
* You cannot invalidate media files in the Microsoft Smooth Streaming format when you have enabled Smooth Streaming for the corresponding cache behavior.

Objects are cached for the TTL (always recorded in seconds, default is 24 hours, default max is 1 year).

Only caches for GET requests (not PUT, POST, PATCH, DELETE).

Dynamic content is cached.

Consider how often your files change when setting the TTL.

Invalidation can be used to immediately revoke cached objects – chargeable.

Deletions propagate.

Restrictions

Blacklists and whitelists can be used for geography – you can only use one at a time.

There are two options available for geo-restriction (geo-blocking):

* Use the CloudFront geo-restriction feature (use for restricting access to all files in a distribution and at the country level).
* Use a 3rd party geo-location service (use for restricting access to a subset of the files in a distribution and for finer granularity at the country level).

AWS WAF

AWS WAF is a web application firewall that lets you monitor HTTP and HTTPS requests that are forwarded to CloudFront and lets you control access to your content.

With AWS WAF you can shield access to content based on conditions in a web access control list (web ACL) such as:

* Origin IP address.
* Values in query strings.

CloudFront responds to requests with the requested content or an HTTP 403 status code (forbidden).

CloudFront can also be configured to deliver a custom error page.

Need to associate the relevant distribution with the web ACL.

Security

PCI DSS compliant but recommended not to cache credit card information at edge locations.

HIPAA compliant as a HIPAA eligible service.

Distributed Denial of Service (DDoS) protection:

* CloudFront distributes traffic across multiple edge locations and filters requests to ensure that only valid HTTP(S) requests will be forwarded to backend hosts. CloudFront also supports geoblocking, which you can use to prevent requests from particular geographic locations from being served.

Domain Names

CloudFront typically creates a domain name such as a232323.cloudfront.net.

Alternate domain names can be added using an alias record (Route 53).

For other service providers use a CNAME (cannot use the zone apex with CNAME).

Moving domain names between distributions:

* You can move subdomains yourself.
* For the root domain you need to use AWS support.

Charges

There is an option for reserved capacity over 12 months or longer (starts at 10TB of data transfer in a single region).

You pay for:

* Data Transfer Out to Internet.
* Data Transfer Out to Origin.
* Number of HTTP/HTTPS Requests.
* Invalidation Requests.
* Dedicated IP Custom SSL.
* Field level encryption requests.

You do not pay for:

* Data transfer between AWS regions and CloudFront.
* Regional edge cache.
* AWS ACM SSL/TLS certificates.
* Shared CloudFront certificates.