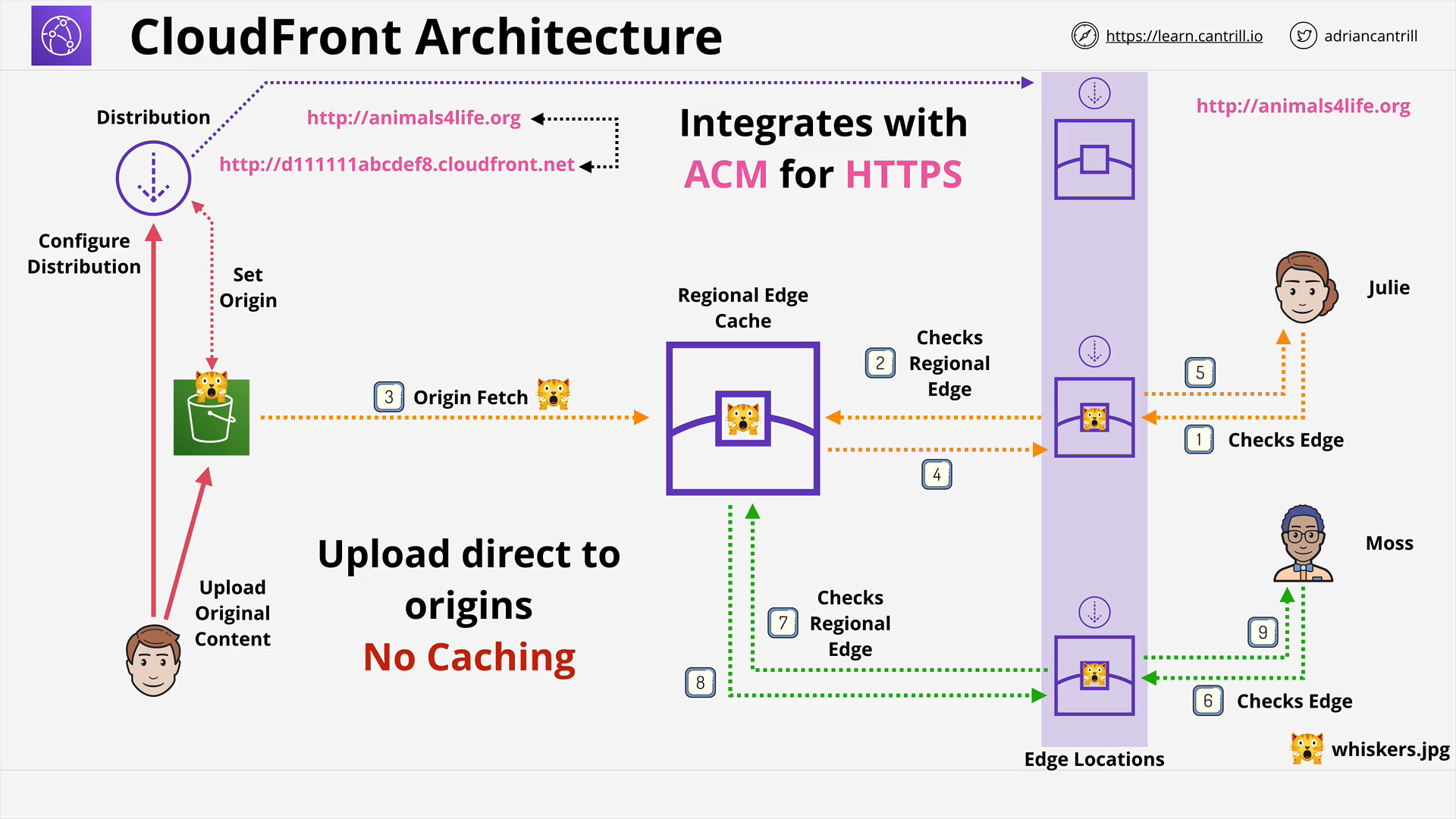
**CDN-and-Optimization**

**Architecture Basics**

* CloudFront is a global object cache (CDN)
* Download caching only
* Content is cached in locations close to customers.
* If the content is not available on the local cache when requested, CloudFront will fetch the item and cache it and deliver it locally.
* This provides lower latency and higher throughput for customers.
* Can handle **static and dynamic content.**
* **Origin** the original location of your content, can be an S3 bucket or LB.
* **Distribution** the **configuration unit of CloudFront.**
* **Edge locations** global infrastructure which hosts a cache of your data.
  + There are over 200 edge locations.
  + They can be one or more racks in a third party server system.
  + Normally 90% storage with some small compute.
* **Regional Edge Cache**
  + Larger version of an edge location.
  + Support a number of local edge locations.
  + Designed to hold more data to cache things which are accessed less often.
  + Provides another layer of caching.



**Caching Optimization**

Parameters can be passed on the url such as query string parameter. An example is ?language=en and ?language=es

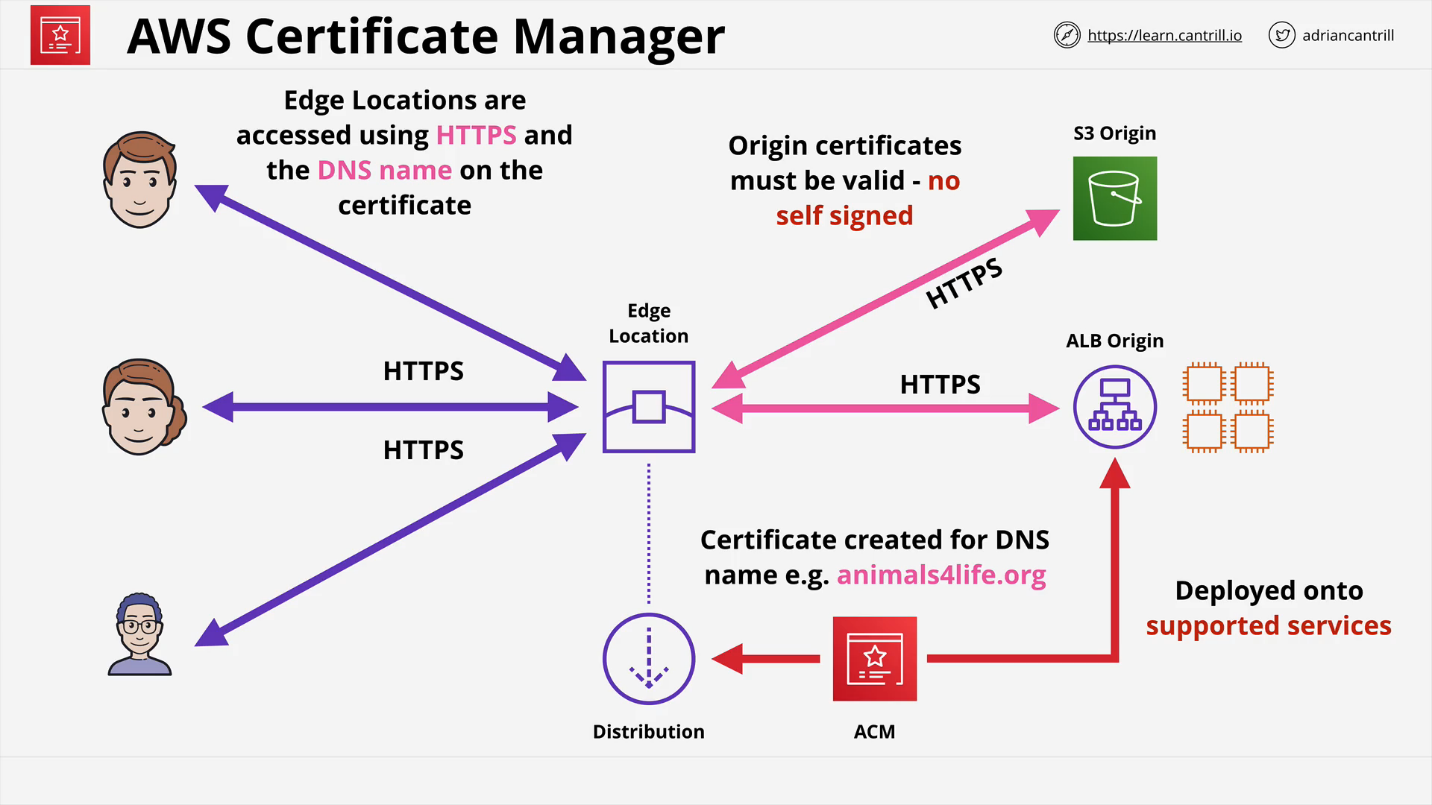
Caching will cache each string parameter storing two different objects. You must use the same string parameters again to retrieve them. If you remove them and the object is not caching it will need to be fetched first.

If string parameters aren't involved in the caching, you can select no to forward them to the origin.

If the application does use query string parameters, you can use all of them for caching or just selected ones.

We can perform **manual invalidations of caches** so that all the cached content can be flush out, so that incase new contents are uploaded those are reflected in the cloudfront-urls

**AWS Certificate Manager (ACM)**

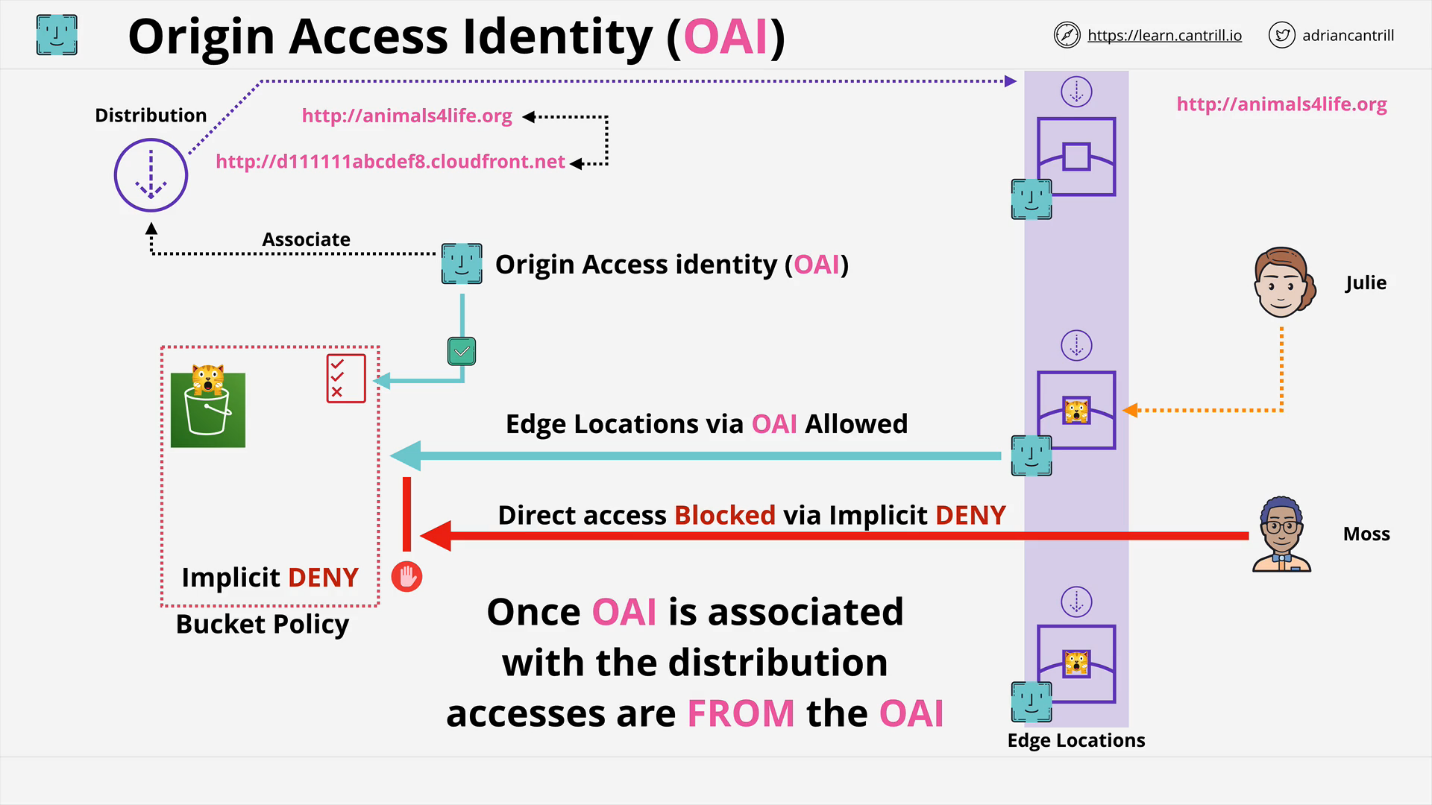
* HTTP lacks encryption and is insecure
* HTTPS uses SSL/TLS layer of encryption added to HTTP
* Data is encrypted in-transit
* Certificates allow servers to prove their identity
* Signed by a trusted authority (CA).
* To be secure, a **website generates a certificate, and has a CA sign it. The website then uses that certificate to prove its authenticity.**
* ACM allows you to **create, renew, and deploy certificates.**
* Supported AWS services ONLY (**CloudFront** and **ALB, NOT EC2)**
* If it's not a managed service, ACM doesn't support it.
* **CloudFront must have a trusted and signed certificate.** Can't be self-signed.

**Origin Access Identity (OAI)**

1. Identity **can be associated with a CloudFront distribution.**
2. The **edge locations gain this identity.**
3. **Create or adjust the bucket policy on the S3 origin.** Add an **explicit allow for the OAI.** Can remove any other explicit allows on the OAI. This leaves the implicit deny.

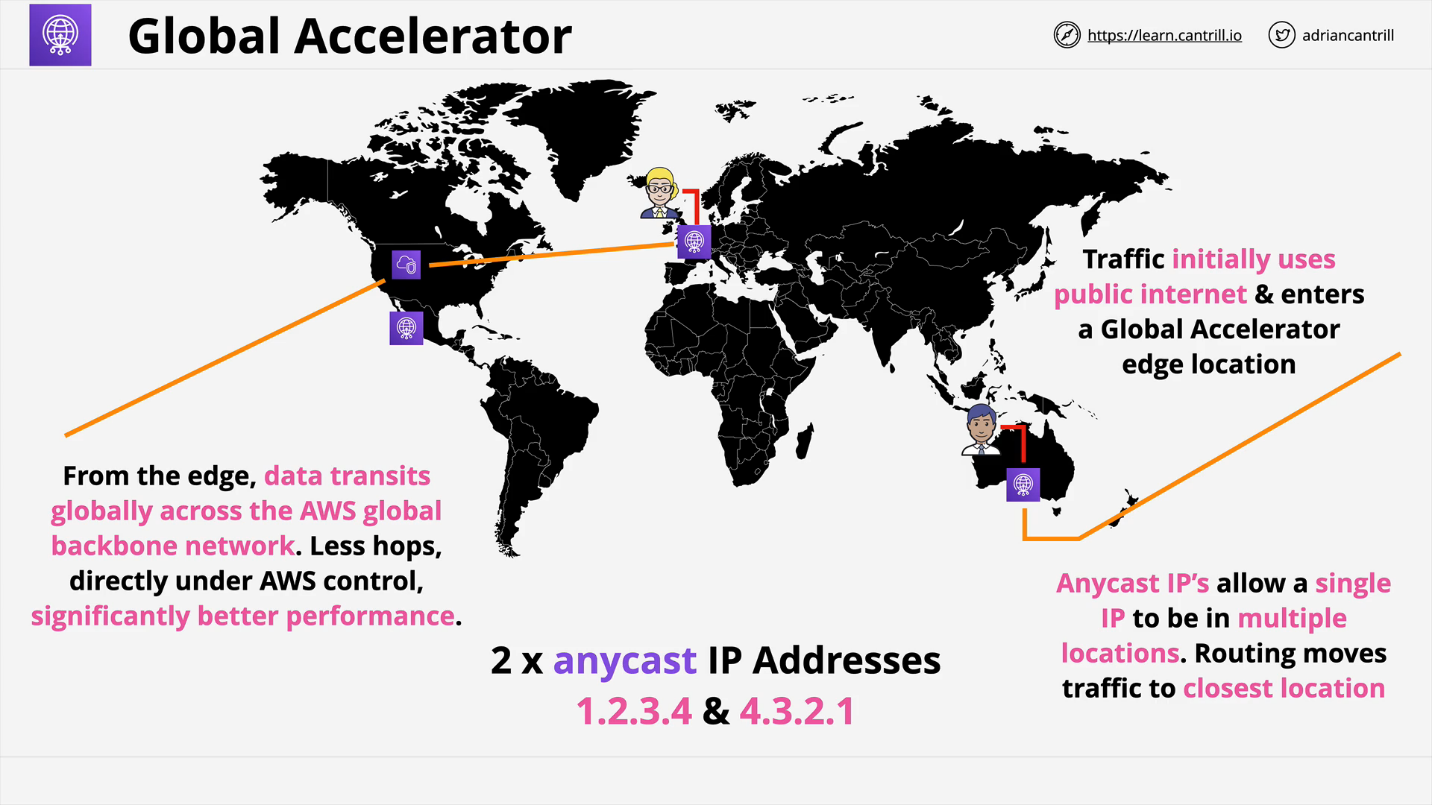
As long as accesses are coming from the edge locations, **it will know they are from the OAI** and allow them. Any direct attempts will not use the OAI and will only get the implicit deny.

Best practice is to **create one OAI per CloudFront distribution to manage permissions**.

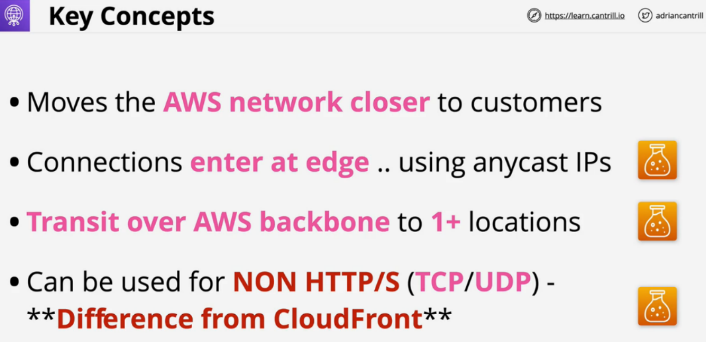


**AWS Global Accelerator**

* Move the **AWS network closer to customers.**
* **Designed to optimize the flow of data** from users to your AWS infrastructure.
* Generally customers who are **further away from your infrastructure go through more internet based hops** and this means a **lower quality connection.**
* Normal IP addresses are unicast IP addresses. These refer to one thing.
* Global Accelerator starts with 2 **anycast** IP address
  + Special IP address
  + Anycast IPs allow a single IP to be in multiple locations.
  + **Traffic initially uses public internet and enters Global Accelerator at the closest edge location.**
  + Traffic then flows globally across the AWS global backbone network.
* Global accelerator is a network product, can use TCP/UDP.

**Pro Exam Tip:**

**Global Accelerator** doesn’t cache content or data, it doesn’t provide any http/https based capability, it’s network product. SO if the question talks about transiting network data over TCP/UDP as quickly and efficiently as possible through a global network it’s likely global accelerator. **Network Optimization.**

If the question mentions content delivery, caching, pre-signed urls or any of the other services cloudfront provides then the answer would be cloudfront. Http/Https, content delivery, manipulation of the content then it’s cloud front.