AWS - CloudFront - SignedURLs or Signed Cookies

CloudFront signed URLs and signed cookies provide the same basic functionality: they allow you to control who can access your content.

If you want to serve private content through CloudFront and you're trying to decide whether to use signed URLs or signed cookies, consider the following.

Use **signed URLs** for the following cases:

- You want to use an RTMP distribution. Signed cookies aren't supported for RTMP distributions.
- You want to restrict access to individual files, for example, an installation download for your application.
- Your users are using a client (for example, a custom HTTP client) that doesn't support cookies.

Use **signed cookies** for the following cases:

- You want to provide access to multiple restricted files, for example, all of the files for a video in HLS format or all of the files in the subscribers' area of a website.
- You don't want to change your current URLs.

Reference:

https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/private-content-choosing-signed-urls-cookies.html

S3 Pre-signed URLs vs CloudFront Signed URLs vs Origin Access Identity (OAI)

https://tutorialsdojo.com/aws-cheat-sheet-s3-pre-signed-urls-vs-cloudfront-signed-urls-vs-origin-access-identity-oai/

AWS - CloudFront - Geo Restriction policies

You can use geo restriction - also known as **geoblocking** - to prevent users in specific geographic locations from accessing content that you're distributing through a CloudFront web distribution. To use geo restriction, you have two options:

1. Use the **CloudFront geo restriction feature**. Use this option to restrict access to all of the files that are associated with a distribution and to restrict access at the country level.

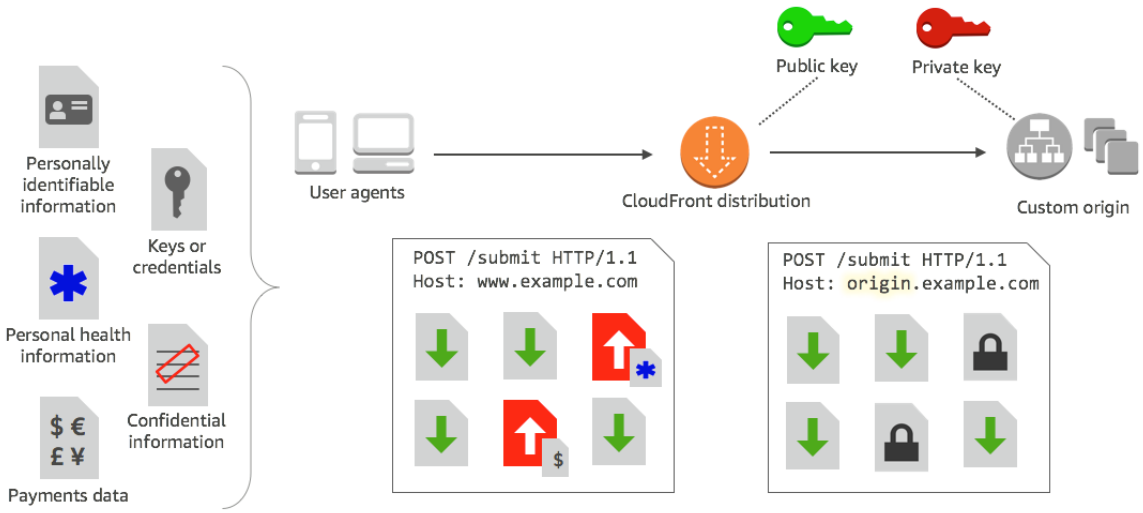
2. Use a **third-party geolocation service**. Use this option to restrict access to a subset of the files that are associated with a distribution or to restrict access at a finer granularity than the country level.

Reference:

https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/georestrictions.html

AWS - CloudFront - Field level encryption

Field-level encryption adds an additional layer of security along with HTTPS that lets you protect specific data throughout system processing so that only certain applications can see it. Field-level encryption allows you to securely upload user-submitted sensitive information to your web servers. The sensitive information provided by your clients is encrypted at the edge closer to the user and remains encrypted throughout your entire application stack, ensuring that only applications that need the data—and have the credentials to decrypt it—are able to do so.



To use field-level encryption, you configure your CloudFront distribution to specify the set of fields in POST requests that you want to be encrypted, and the public key to use to encrypt them. You can encrypt up to 10 data fields in a request. Hence, the correct answer for this scenario is the option that says: ***Configure the CloudFront distribution to enforce secure end-to-end connections to origin servers by using HTTPS and field-level encryption. Configure your origin to add a Cache-Control max-age directive to your objects, and specify the longest practical value for max-age to increase your cache hit ratio.***

References:

https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/field-level-encryption.html#field-level-encryption-setting-up

https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/cache-hit-ratio.html

Check out this Amazon CloudFront Cheat Sheet:

https://tutorialsdojo.com/aws-cheat-sheet-amazon-cloudfront/

Tutorials Dojo's AWS Certified Solutions Architect Professional Exam Study Guide:

https://tutorialsdojo.com/aws-cheat-sheet-aws-certified-solutions-architect-professional/

CloudFront - SSL/TLS cert management (https)

The certificate issuer you must use depends on whether you want to require HTTPS between viewers and CloudFront or between CloudFront and your origin

HTTPS between viewers and CloudFront

- You can use a certificate that was issued by a trusted certificate authority (CA) such as Comodo, DigiCert, Symantec or other third-party providers.

- You can use a certificate provided by AWS Certificate Manager (ACM)

HTTPS between CloudFront and a custom origin

- If the origin is not an ELB load balancer, such as Amazon EC2, the certificate must be issued by a trusted CA such as Comodo, DigiCert, Symantec or other third-party providers.

- If your origin is an ELB load balancer, you can also use a certificate provided by ACM.

If you're using your own domain name, such as tutorialsdojo.com, you need to change several CloudFront settings. You also need to use an SSL/TLS certificate provided by AWS Certificate Manager (ACM), or import a certificate from a third-party certificate authority into ACM or the IAM certificate store.

CloudFront - Origin Control Policy

In CloudFront, there are 3 options that you can choose as the value for your Origin Protocol Policy:

- HTTP Only,
- HTTPS Only and
- Match Viewer.

- For HTTP Only,
  - CloudFront uses only HTTP to access the origin.
- For HTTPS Only,
  - CloudFront uses only HTTPS to access the origin and
- for Match Viewer,

- CloudFront communicates with your origin using HTTP or HTTPS, depending on the protocol of the viewer request. CloudFront caches the object only once even if viewers make requests using both HTTP and HTTPS protocols.

Take note that the term "Viewer" in CloudFront basically means the requestor or the the client that sends the request. Hence, the name: "MatchViewer" basically means that CloudFront is simply matching the protocol being used by the viewer.

Match Viewer is correct because if the Origin Protocol Policy is set to Match Viewer, the CloudFront communicates with the origin using HTTP or HTTPS depending on the protocol of the viewer request.

Reference:

http://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/distribution-web-values-specify.html

CloudFront End-2-End Https with Origin

Remember that there are rules on which type of SSL Certificate to use if you are using an EC2 or an ELB as your origin. This question is about setting up an end-to-end HTTPS connection between the Viewers, CloudFront, and your custom origin, which is an ALB instance.

The certificate issuer you must use depends on whether you want to require HTTPS between viewers and CloudFront or between CloudFront and your origin:

HTTPS between viewers and CloudFront

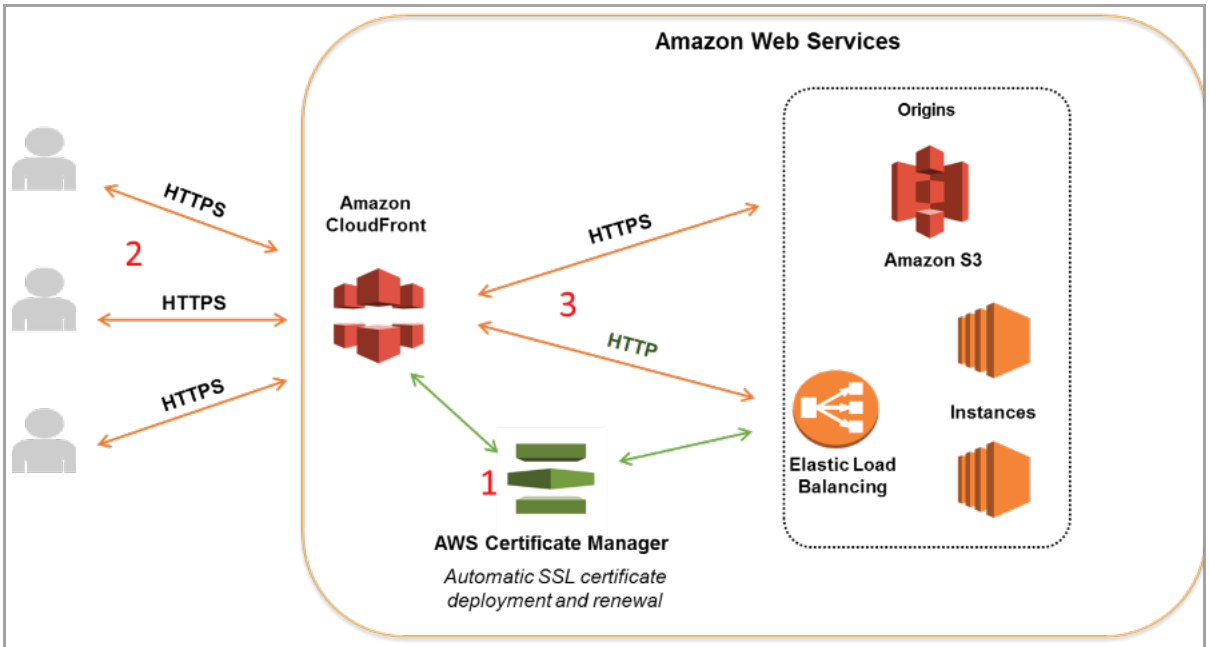
- You can use a certificate that was issued by a trusted certificate authority (CA) such as Comodo, DigiCert, Symantec or other third-party providers.

- You can use a certificate provided by AWS Certificate Manager (ACM)

HTTPS between CloudFront and a custom origin

- If the origin is not an ELB load balancer, such as Amazon EC2, the certificate must be issued by a trusted CA such as Comodo, DigiCert, Symantec or other third-party providers.

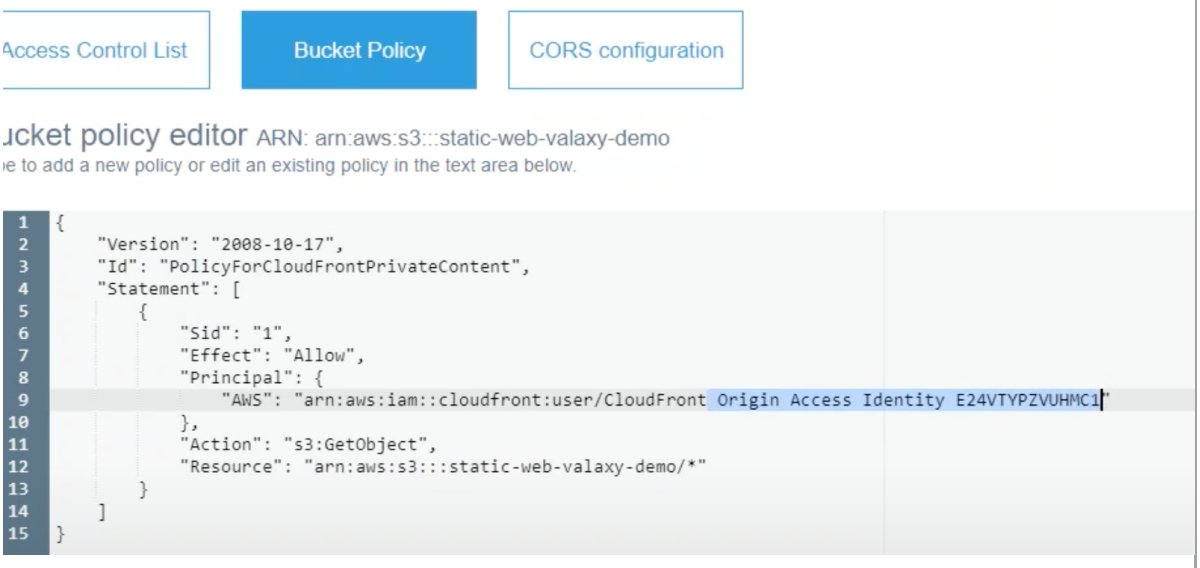
- If your origin is an ELB load balancer, you can also use a certificate provided by ACM.

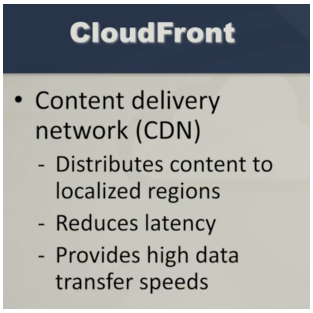
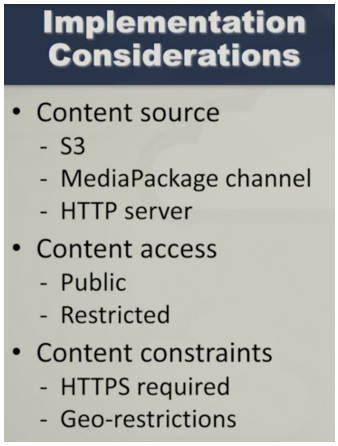


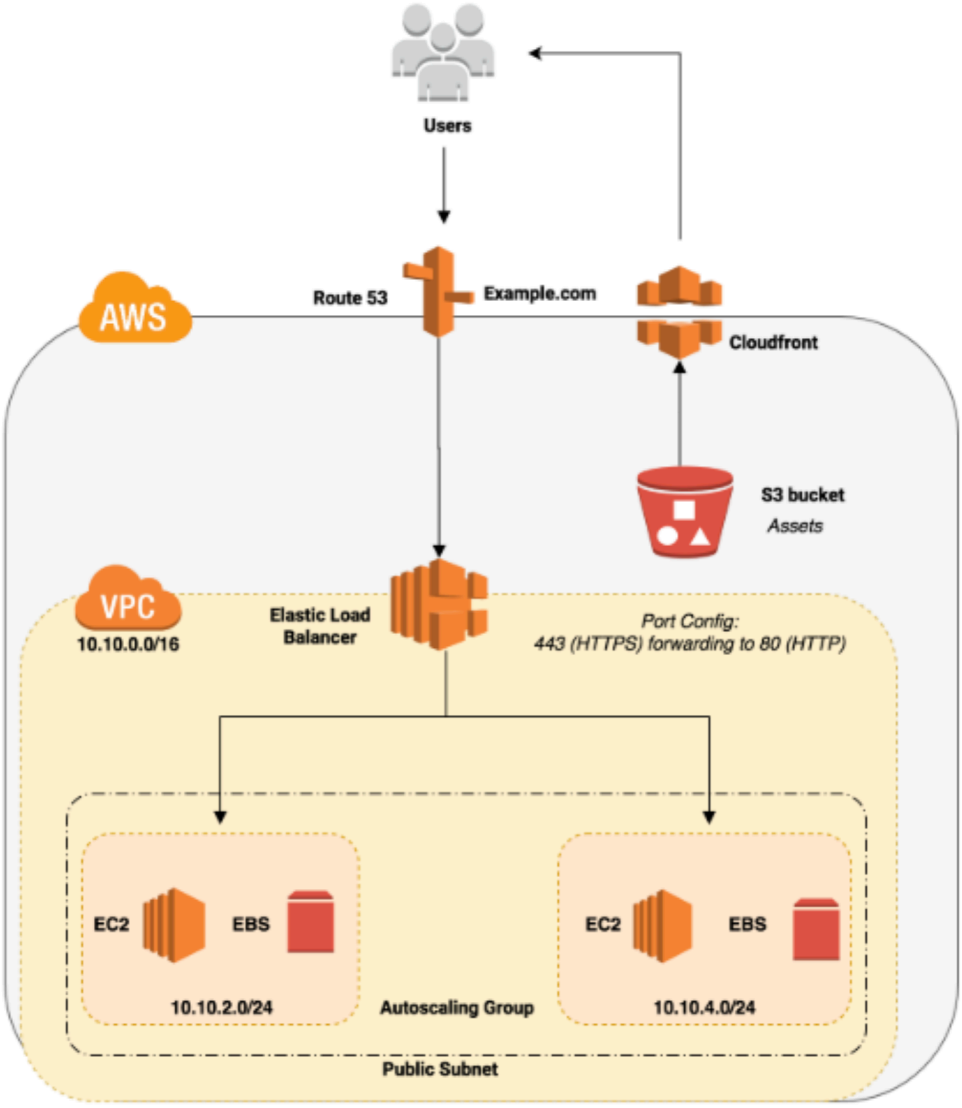
If you're using your own domain name, such as tutorialsdojo.com, you need to change several CloudFront settings. You also need to use an SSL/TLS certificate provided by AWS Certificate Manager (ACM), or import a certificate from a third-party certificate authority into ACM or the IAM certificate store. Lastly, you should set the Viewer Protocol Policy to HTTPS Only in CloudFront.

S3 ORIGIN ACCESS IDENTITY (OAI)

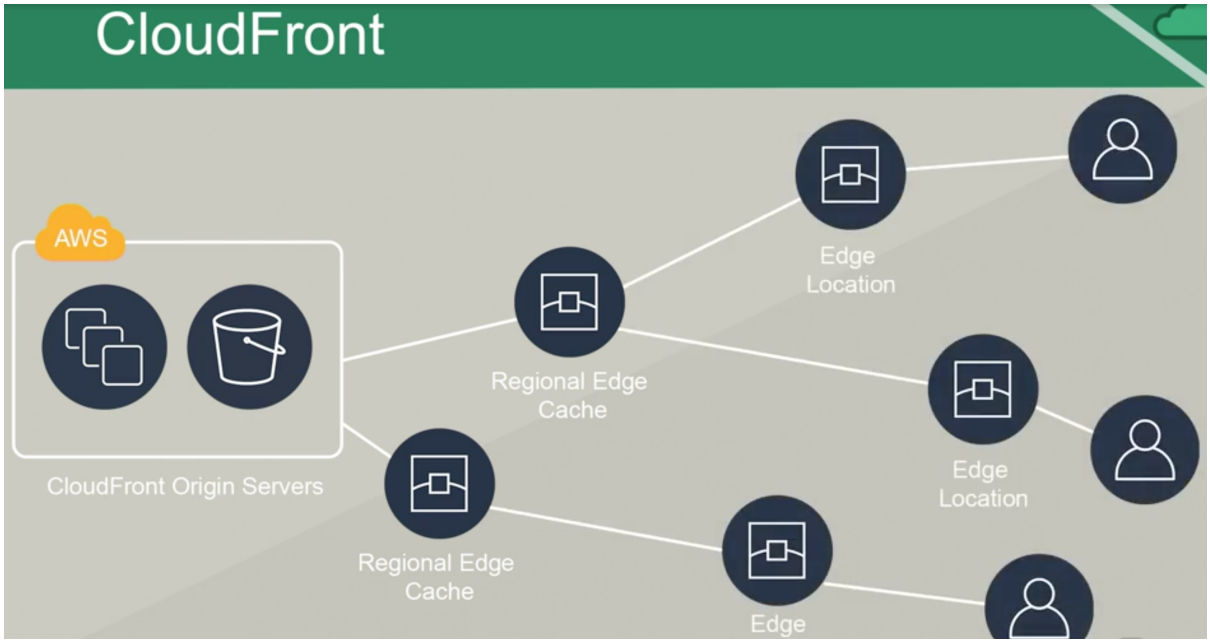
- Create Cloud-front distribution
- Create Origin Access identity (OAI)
- Attach OAI to Cloud-front distribution
- Update S3 bucket policy



Cloud Front https://aws.amazon.com/premiumsupport/knowledge-center/cloudfront-serve-static-website/	S3 with Cloud front https://aws.amazon.com/blogs/networking-and-content-delivery/amazon-s3-amazon-cloudfront-a-match-made-in-the-cloud/	AWS Global Accelerator https://aws.amazon.com/global-accelerator/faqs/
		


Users
AWS
Route 53
Example.com
Cloudfront
S3 bucket
Assets
VPC
10.10.0.0/16
Elastic Load Balancer
Port Config:
443 (HTTPS) forwarding to 80 (HTTP)
EC2
EBS
10.10.2.0/24
Autoscaling Group
EC2
EBS
10.10.4.0/24
Public Subnet

Brief



https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/IntroductionUseCases.html

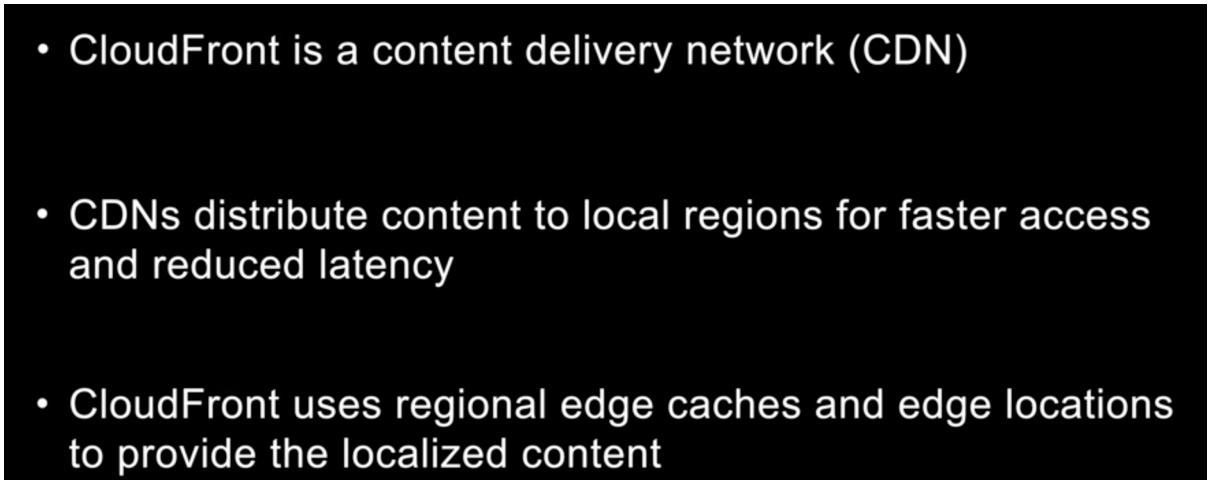
CloudFront Use Cases

PDF | Kindle | RSS

Using CloudFront can help you accomplish a variety of goals. This section lists just a few, together with links to more information, to give you an idea of the possibilities.

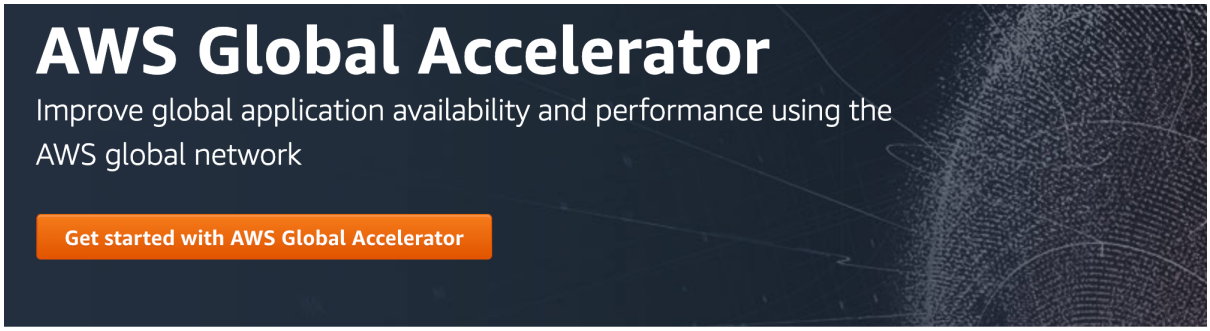
Topics

- Accelerate Static Website Content Delivery
- Serve Video On Demand or Live Streaming Video
- Encrypt Specific Fields Throughout System Processing
- Customize at the Edge
- Serve Private Content by using Lambda@Edge Customizations



AWS Global Accelerator:

- Improve the performance of TCP/UDP traffic.
- Continuously monitor the End-points of the application and identify the un-healthy end-point then redirects to healthy end-points in less than 30 seconds.
- https://aws.amazon.com/global-accelerator/features/



AWS Global Accelerator is a service that improves the availability and performance of your applications with local or global users. It provides static IP addresses that act as a fixed entry point to your application endpoints in a single or multiple AWS Regions, such as your Application Load Balancers, Network Load Balancers or Amazon EC2 instances.

AWS Global Accelerator uses the AWS global network to optimize the path from your users to your applications, improving the performance of your traffic by as much as 60%. You can test the performance benefits from your location with a speed comparison tool. AWS Global Accelerator continually monitors the health of your application endpoints and redirects traffic to healthy endpoints in less than 30 seconds.

AWS Global Accelerator features

PAGE CONTENT

- Static anycast IP addresses
- Fault tolerance using network zones
- Global performance-based routing
- TCP Termination at the Edge
- Bring your own IP (BYOIP)
- Fine-grained traffic control
- Continuous availability monitoring
- Client affinity
- DDoS resiliency at the edge

Static anycast IP addresses

AWS Global Accelerator provides you with static IP a
AWS Regions. These IP addresses are anycast from A
same time. This enables traffic to ingress onto the A
addresses to regional AWS resources or endpoints, s
Elastic IP addresses. AWS Global Accelerator's IP adc
addresses, you don't need to make any client-facing
are assigned to your accelerator for as long as it exis

Fault tolerance using networl

AWS Global Accelerator has a fault-isolating design
AWS Global Accelerator allocates two IPv4 static adc
Availability Zones, these network zones are isolated
unique IP subnet. If one IP address from a network z
certain client networks, your client applications can

Global performance-based rc

AWS Global Accelerator uses the vast, congestion-fr
endpoint in the closest AWS Region to the user. If th

AWS - Global Accelerator

- **AWS Global Accelerator** is a service that improves the availability and performance of your applications with local or global users.
- It provides static IP addresses that act as a fixed entry point to your application endpoints in a single or multiple AWS Regions, such as your Application Load Balancers, Network Load Balancers or Amazon EC2 instances.

- AWS Global Accelerator uses the AWS global network to optimize the path from your users to your applications, improving the performance of your TCP and UDP traffic.
- AWS Global Accelerator continually monitors the health of your application endpoints and will detect an unhealthy endpoint and redirect traffic to healthy endpoints in less than 1 minute.

- ***AWS Global Accelerator*** service is more suitable for non-HTTP use cases, such as gaming (UDP), IoT (MQTT), or Voice over IP, as well as for HTTP use cases that specifically require static IP addresses or deterministic, fast regional failover.
- Moreover, there is no direct way that you can integrate AWS Global Accelerator with Amazon S3.
- It's more suitable to use Amazon CloudFront instead in this scenario.

References:

https://aws.amazon.com/premiumsupport/knowledge-center/cloudfront-serve-static-website/

https://aws.amazon.com/blogs/networking-and-content-delivery/amazon-s3-amazon-cloudfront-a-match-made-in-the-cloud/

https://aws.amazon.com/global-accelerator/faqs/

- Many applications, such as gaming, media, mobile applications, and financial applications, need very low latency for a great user experience.
- To improve the user experience, AWS Global Accelerator directs user traffic to the nearest application endpoint to the client, thus reducing internet latency and jitter.
- It routes the traffic to the closest edge location via Anycast, then by routing it to the closest regional endpoint over the AWS global network. AWS Global Accelerator quickly reacts to changes in network performance to improve your users' application performance.

- AWS Global Accelerator and Amazon CloudFront are separate services that use the AWS global network and its edge locations around the world.
- CloudFront improves performance for both cacheable content (such as images and videos) and dynamic content (such as API acceleration and dynamic site delivery).
- Global Accelerator improves performance for a wide range of applications over TCP or UDP by proxying packets at the edge to applications running in one or more AWS Regions.
- Global Accelerator is a good fit for non-HTTP use cases, such as gaming (UDP), IoT (MQTT), or Voice over IP, as well as for HTTP use cases that specifically require static IP addresses or deterministic, fast regional failover.
- Both services integrate with AWS Shield for DDoS protection.

References:

https://aws.amazon.com/global-accelerator/

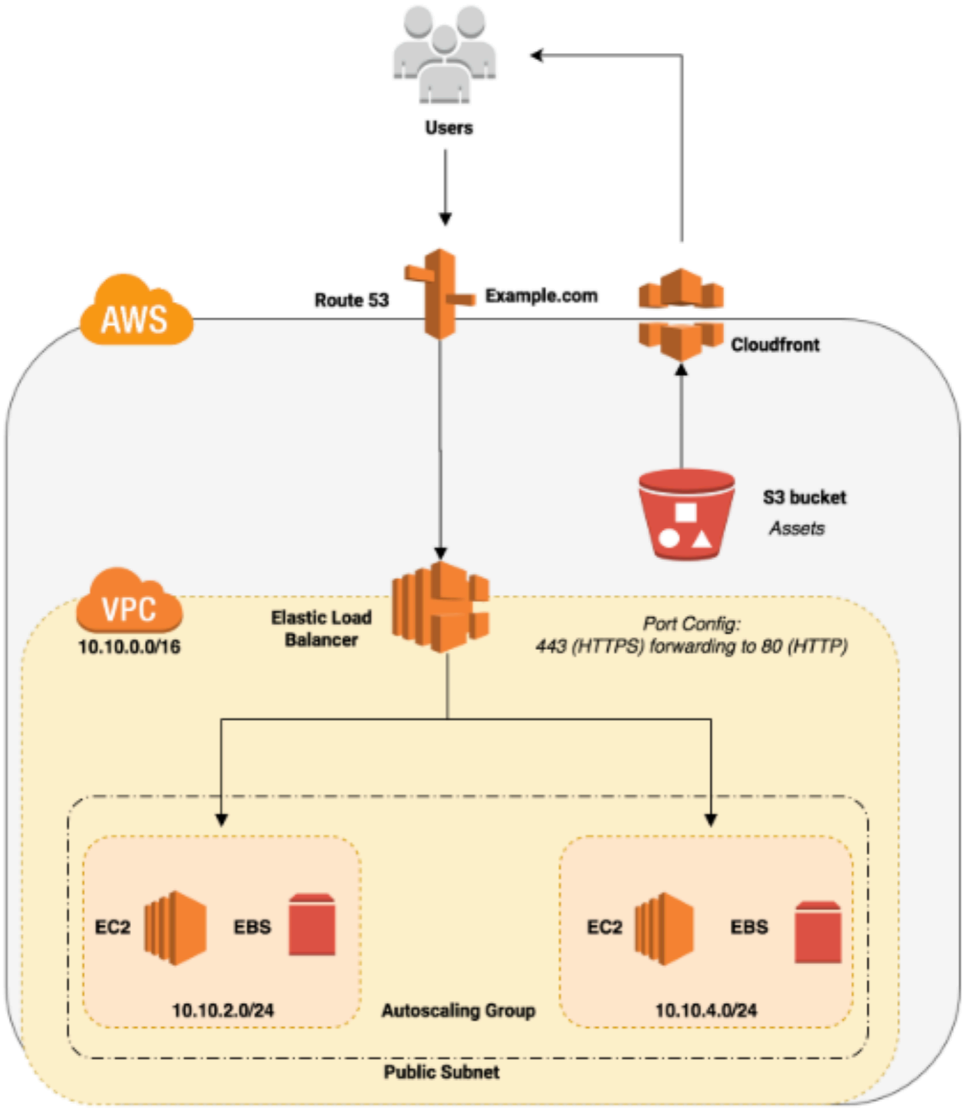
https://aws.amazon.com/global-accelerator/faqs/

Check out this AWS Global Accelerator Cheat Sheet:

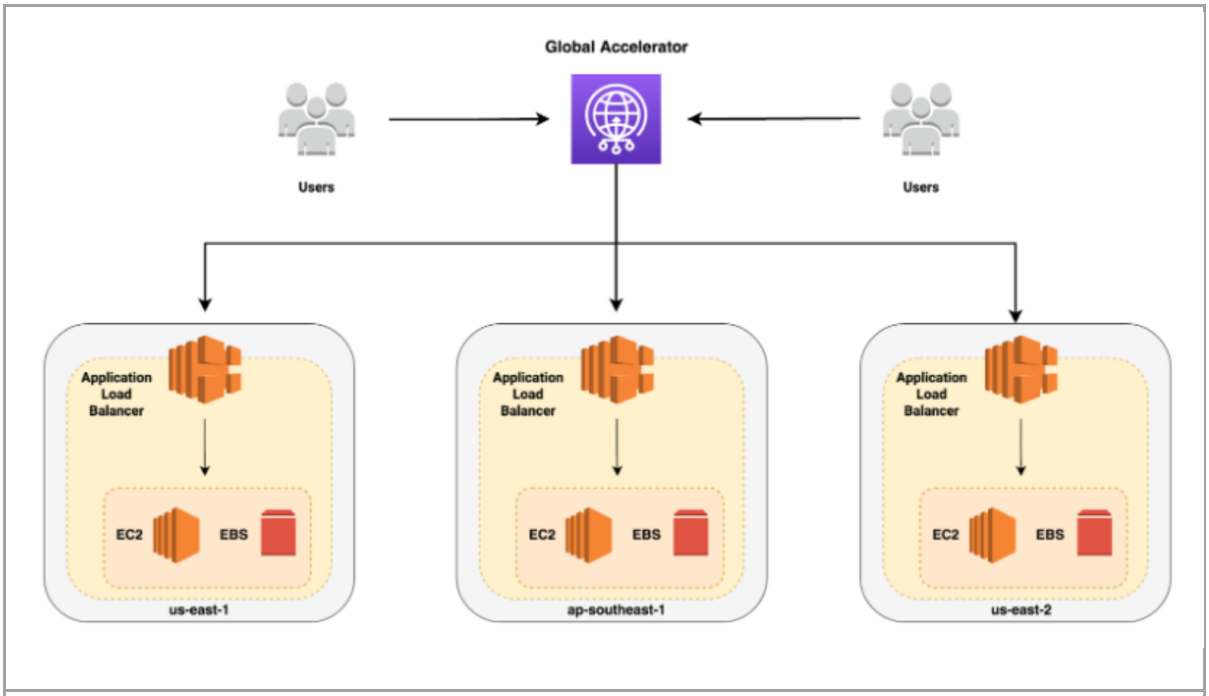
https://tutorialsdojo.com/aws-cheat-sheet-aws-global-accelerator/

GLOBAL ACCELERATOR Vs CLOUD-FRONT

CLOUD-FRONT



GLOBAL ACCELERATOR


Global Accelerator
Users
Users
Application Load Balancer
EC2
EBS
us-east-1
Application Load Balancer
EC2
EBS
ap-southeast-1
Application Load Balancer
EC2
EBS
us-east-2