Amazon S3

Contents

- S3 Introduction
- S3 Storage Tier
- S3 security and Encryption
- S3 version control
- S3 Life cycle management
- S3 Performance
- S3 static website, CORS

S3 Introduction

What is S3?

- S3 stand for **S**imple **S**torage **S**ervice.
- One of the most oldest, important service of AWS
- Providing secure, durable, highly-scalable **object storage**
- It is advertised unlimited storage
- S3 uses to store files (docs, video, text…)

S3 overview - Bucket

- Bucket is same with a directory
- Bucket name must be **Global unique**
- S3 bucket is regional scope

S3 overview - Objects

- Objects are same with files
- Objects consist of the following
  - Key (Name of Object)
    - s3://bucket_name/my_folder/another_folder/my_file
      Key
    - s3://bucket_name/my_folder/another_folder/my_file
      Prefix
  - Value (Content of Object)
    - Object size can be from 0 Bytes to 5 TB.
    - Using multi-part upload if object size > 5GB

S3 overview – Objects (cont.)

- Objects consist of the following
  - Metadata
    - Set of name-value pairs for Objects (Set at upload time, cannot modify later). Ex: Date = 20210101, x-amz-storage-class = Standard…
    - User Metatdata and System Metadata (Object creation time, Storage type, Enable Encrytion or not,…)
  - Version ID
    - For Versioning

S3 Storage Tier

S3 storage tier

	S3 Standard	S3 Intelligent-Tiering*	S3 Standard-IA	S3 One Zone-IA†	S3 Glacier	S3 Glacier Deep Archive
Designed for durability	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)
Designed for availability	99.99%	99.9%	99.9%	99.5%	99.99%	99.99%
Availability SLA	99.9%	99%	99%	99%	99.9%	99.9%
Availability Zones	≥3	≥3	≥3	1	≥3	≥3
Minimum capacity charge per object	N/A	N/A	128KB	128KB	40KB	40KB
Minimum storage duration charge	N/A	30 days	30 days	30 days	90 days	180 days
Retrieval fee	N/A	N/A	per GB retrieved	per GB retrieved	per GB retrieved	per GB retrieved
First byte latency	milliseconds	milliseconds	milliseconds	milliseconds	select minutes or hours	select hours
Storage type	Object	Object	Object	Object	Object	Object
Lifecycle transitions	Yes	Yes	Yes	Yes	Yes	Yes

S3 Standard

- Default storage tier
- 99.99% Availibility
- 99.999999999% i.e. 11 9's Durability
- For commonly purposes.
  - Ideal for performance-sensitive use cases
  - Frequently accessed data

S3 Standard

S3 Standard IA (Infrequently Access)

- For store **Infrequently Access** data (About once a month)
- Cheaper than Standard tier
- Needs extra cost for object retrieving
- Objects are available for real-time access.
- Suitable for larger objects greater than 128 KB
- Charged for minium 30 days

S3 Standard-IA

S3 One-Zone IA (Infrequently Access)

- For store infrequently access data
- Store Object data in **one AZ**
- Suitable for larger objects greater than 128 KB
- Suitable for objects can recover in case of AZ failure
- Charged for minium 30 days

S3 One-Zone IA

S3 Intelligent Tier

- Auto moving data to the most cost-effective storage tire
- Suitable for unpredictable data access pattern

S3 Intelligent-Tiering

S3 Glacier

- For low-cost data archiving
- Minimum duration period ~ 90 days
- Objects are available after minutes to hours

S3 Glacier

S3 Glacier Deep Archive

- For very low-cost data archiving
- Minimum duration period ~ 180 days
- Objects are available after 12 hours (default)

S3 Glacier Deep Archive

Exam tips

- Common purpose, need performance-sensitive, frequently access => **Standard class**
- Infrequently access, high redundancy => **IA**
- Infrequently access, low redundancy, object can recover => **One-Zone IA**
- Data archive (1 ~ 10 years), needs available minutes to hours => **Glacier**
- Data archive (> 10 years), needs available > 12 hours => **Deep Archived**

Security and Encryption

S3 security

- **Identity-based policy**
  - IAM policies – Using IAM policies to define permissions for IAM entity (IAM users, Group, Role)
- **Resource-based policy**
  - Bucket Policies – Bucket scope rules
  - Access Control Lists – Normally use to grant fine-grain permission for objects
- **NOTE**:
  - Using **Policy Evaluation Logic** to determine the permission of IAM principal

S3 security (cont.)

- **Networking**
  - Support VPC Endpoints for private connection
- **Logging and Audit**
  - S3 API can be logged by CloudTrail
  - S3 Access Logs can be stored in another Bucket
- **User Security**
  - MFA Delete: Require MFA code for termination object (preventing accidentally deletion)
  - Pre-Signed URLs: URLs are valid for a limited time

S3 encryption

- **Encryption in Transit**
  - SSL/TLS
- **Encryption at Rest (Server Side Encryption - SSE)**
  - S3 Managed Keys – SSE-S3
  - KMS Managed Keys – SSE-KMS
  - Customer Managed Keys – SSE-C
- **Client Side Encryption**
  - Objects are encrypted before uploading to S3 by users

S3 versioning

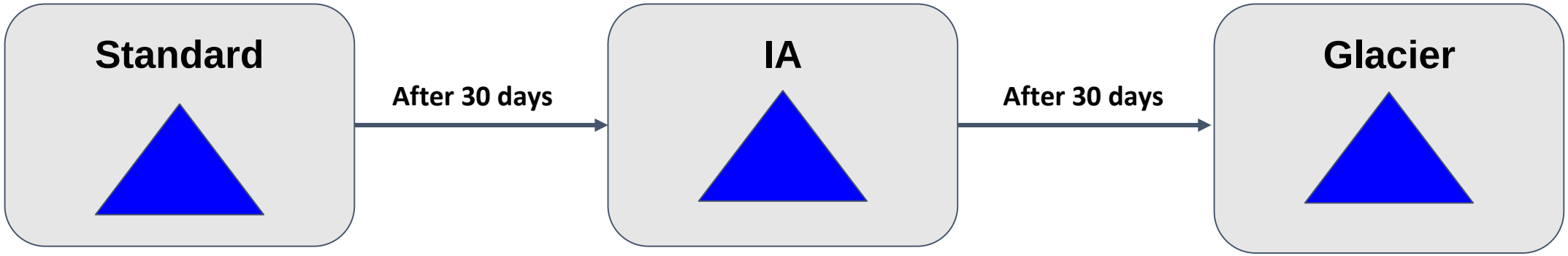
S3 versioning

- Store all versions of an object (including all write/update action and delete)
- Once enabled, **Versioning cannot be disabled**, only suspended
- Versioning can intergrate with **Life Cycle Management** rules
- Provide **MFA delete capability** for object termination action (Required MFA code to delete objects)

S3 Lifecycle Management

S3 Lifecycle Management

- Auto moving objects between the difference storage class
- For cost-effective object store
- Can be used with **Versioning** feature



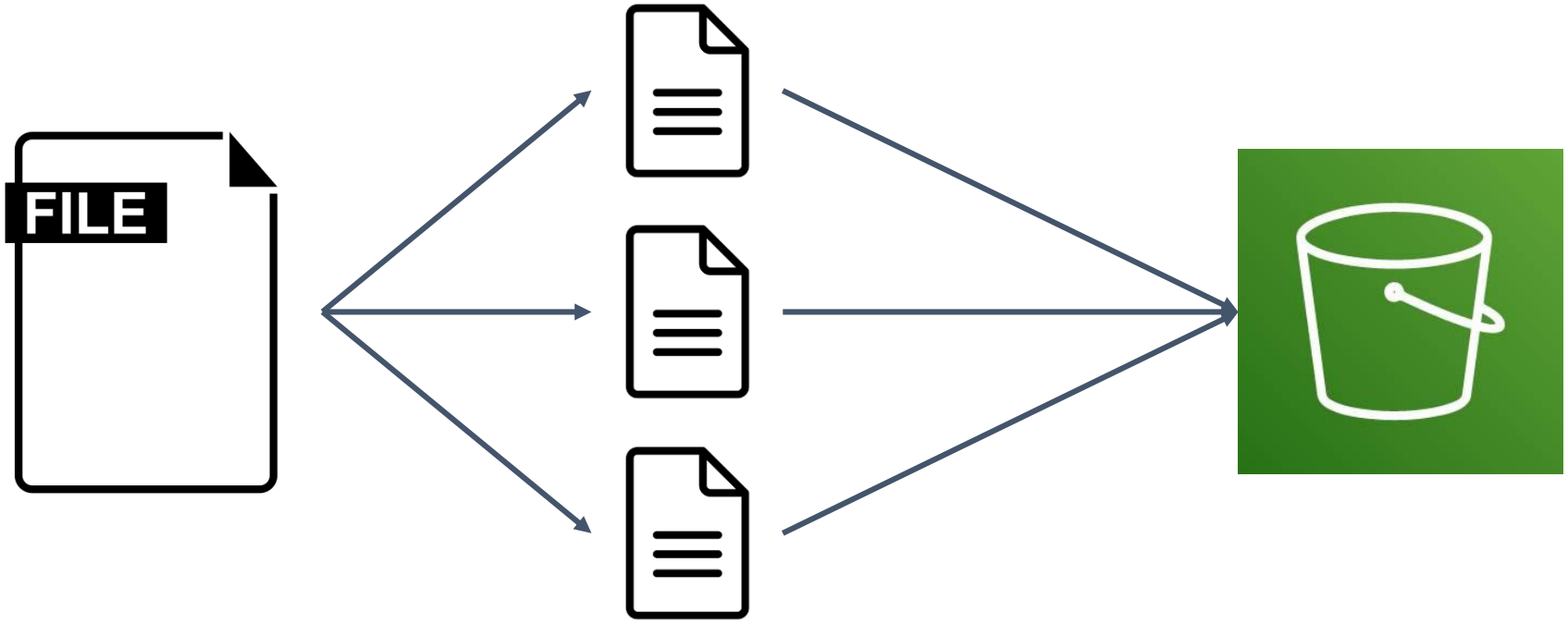
S3 performance

S3 baseline performance

- Each **prefix** allows 3,500 PUT/COPY/POST/DELETE and 5,500 GET/HEAD requests in a bucket
- No limit for number of prefixes in a bucket
- Prefix example.
  - s3://bucket/**folder1/sub1/**file1.txt
  - s3://bucket/**folder2/sub2/**file2.txt
- More prefixes, you get more better performance

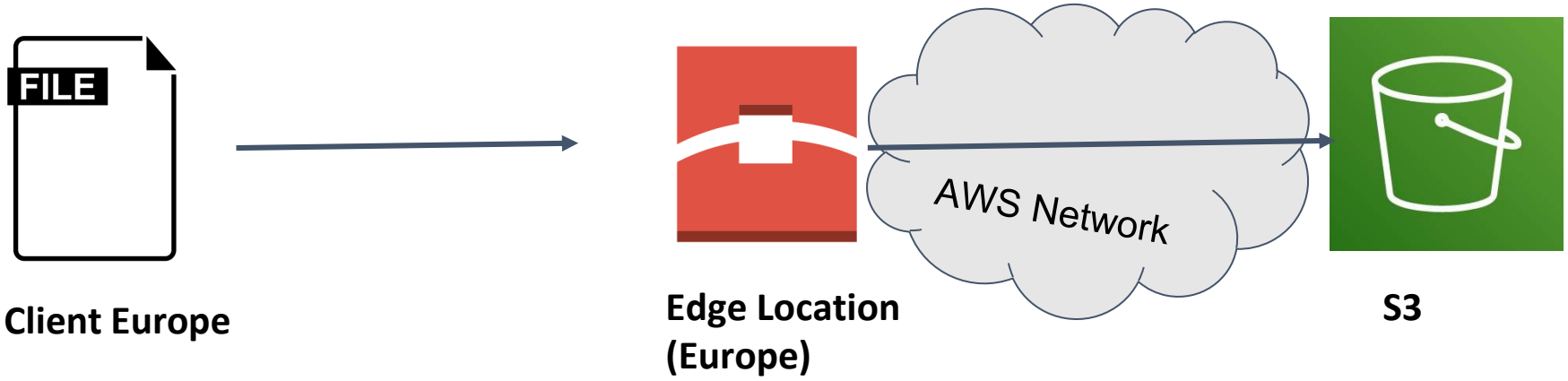
S3 - Multipart upload

- Recommend for files > 100MB, must use for files > 5GB
- Multipart upload can help to optimize throughput by parallel uploading



S3 - Transfer Acceleration

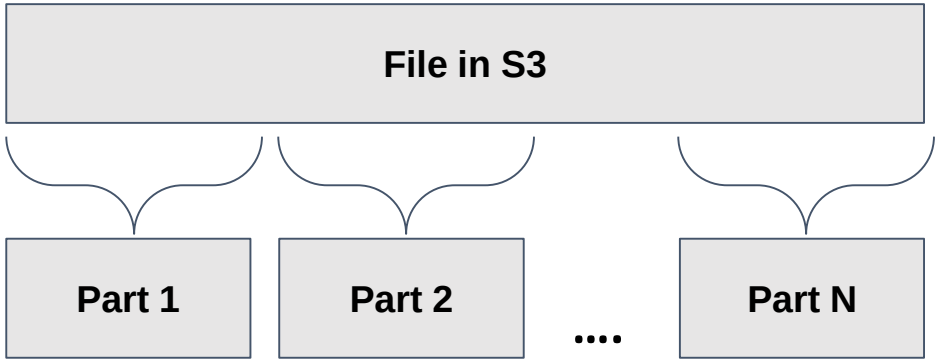
- Using for uploading objects
- Using Edge location as the proxy for S3 and clients.



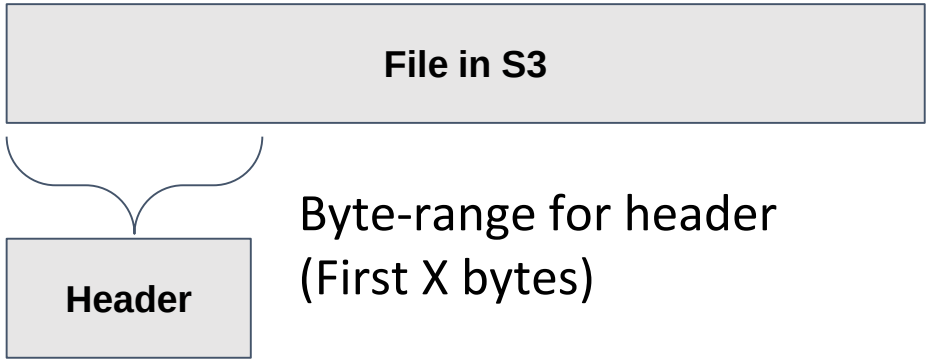
S3 - Byte-Range Fetches

- Parallelized requests to fetch difference byte ranges from one object
- Achieving higher aggregate throughput
- Better resilience in case of failures

Speed up downloads



Download partial data

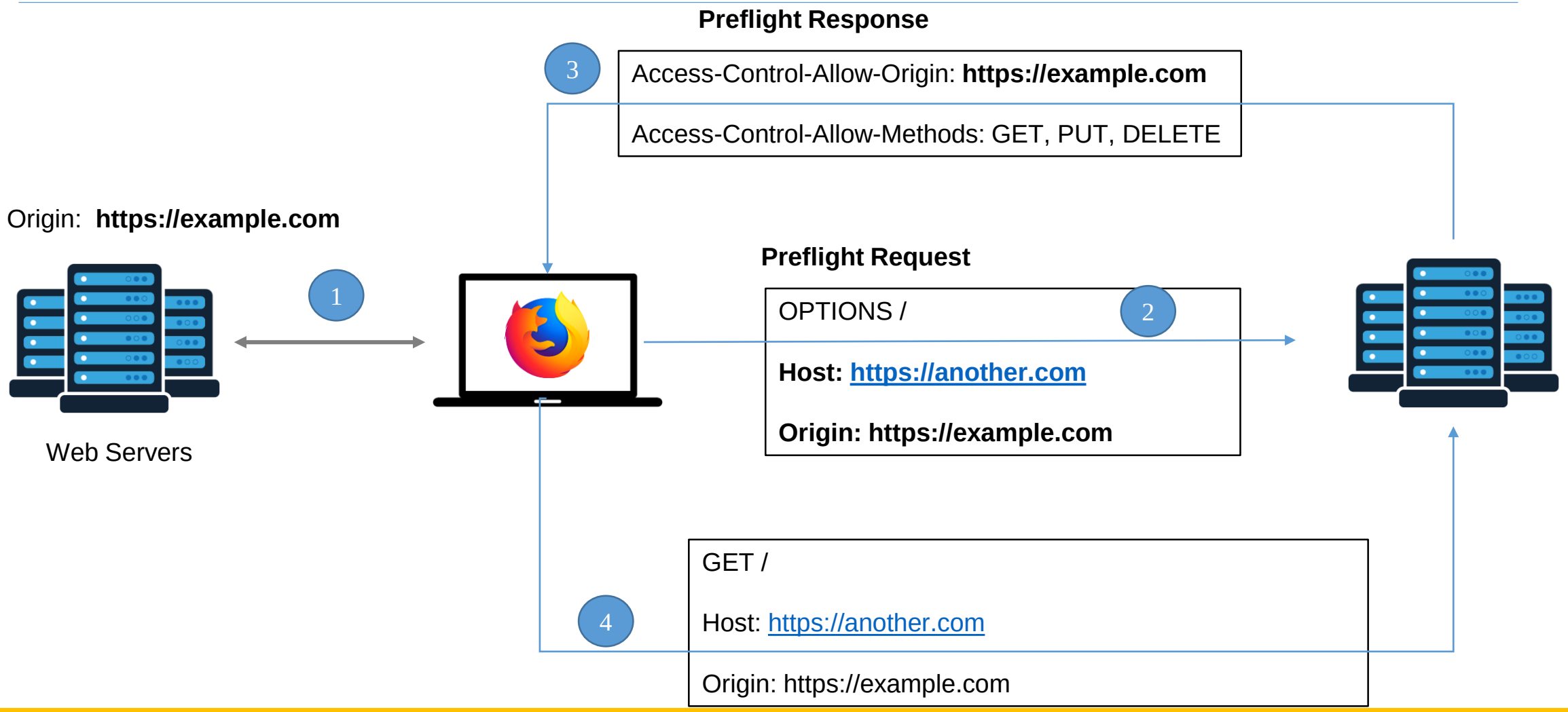


S3 CORS

CORS

- An Origin: <scheme> "://" <hostname> ":" <port> (Ex: https://example.com)
- CORS stands for **C**ross-**O**rigin **R**esource **S**haring
- Same Origin: https://example.com/site1, https://example.com/site2
- Difference Origin: https://example.com/site1, https://another.com/app1
- The requests need to be allowed by target origin using CORS header (Ex: **Access-Controll-Allow-Origin**)

CORS (cont.)


Preflight Response
3
Access-Control-Allow-Origin: https://example.com
Access-Control-Allow-Methods: GET, PUT, DELETE
Origin: https://example.com
Preflight Request
1
OPTIONS /
2
Host: https://another.com
Origin: https://example.com
Web Servers
4
GET /
Host: https://another.com
Origin: https://example.com

S3 CORS

- If the clients request CORS, S3 bucket (enabled static website) need to be enabled CORS
- You can only specific an Origin or all (*)