

# S3 bucket

---

## S3 bucket

Amazon s3 allows people to store objects (files) in buckets directories

Bucket must be have a globally unique name

bucket are defined at the region level

naming convention (No uppercase , No underscore , 3-63 characters long , not an ip , must start with lowercase are number )

# AWS S3 Overview - Objects



- Objects (files) have a Key. The key is the FULL path:
  - <my\_bucket>/my\_file.txt
  - <my\_bucket>/my\_folder1/another\_folder/my\_file.txt
- There's no concept of "directories" within buckets (although the UI will trick you to think otherwise)
- Just keys with very long names that contain slashes ("")
- Object Values are the content of the body:
  - Max Size is 5TB
  - If uploading more than 5GB, must use "multi-part upload"
- Metadata (list of text key / value pairs – system or user metadata)
- Tags (Unicode key / value pair – up to 10) – useful for security / lifecycle

## Amazon S3 - Consistency Model

- Strong consistency as of December 2020:
- After a:
  - successful write of a new object (new PUT)
  - or an overwrite or delete of an existing object (overwrite PUT or DELETE)
- ...any:
  - subsequent read request immediately receives the latest version of the object (read after write consistency)
  - subsequent list request immediately reflects changes (list consistency)
- Available at no additional cost, without any performance impact

**Create bucket**

- Name and region**
- Configure options**
- Set permissions**
- Review**

**Name and region**

**Bucket name**

**Region** US East (Ohio)

**Copy settings from an existing bucket**

You have no buckets 0 Buckets

**Create** **Cancel** **Next**

**Amazon S3 now supports S3 Batch Operations to manage your objects with a few clicks (in preview). [Learn more](#)**

**S3 buckets**

<b>+ Create bucket</b>	<b>Edit public access settings</b>	<b>Empty</b>	<b>Delete</b>	<b>1 Buckets</b>	<b>1 Regions</b>
<input type="checkbox"/> Bucket name <input type="text"/>	Access <input type="button"/>	Region <input type="button"/>	Date created <input type="button"/>	thebucketofstephane-bigdata	Bucket and objects not public US East (Ohio) Mar 25, 2019 12:36:44 PM GMT+0100

**Upload**

- Select files**
- Set permissions**
- Set properties**
- Review**

To upload a file larger than 80 GB, use the AWS CLI, AWS SDK, or Amazon S3 REST API. [Learn more](#)

Drag and drop files and folders here

OR

**Add files**

**Upload** **Next**

AWS Services Resource Groups

Amazon S3 > thebucketofstephane-bigdata > stores > store-1

Overview

Upload + Create folder Download Actions US East (Ohio) ▾

Type a prefix and press Enter to search. Press ESC to clear.

Name	Last modified	Size	Storage class
10-22-2018	--	--	--

Viewing 1 to 1

Operations 0 In progress 3 Success 0 Error

The screenshot shows the AWS S3 console interface. At the top, the navigation bar includes the AWS logo, 'Services' dropdown, 'Resource Groups' dropdown, and links for 'Big Data Course', 'Global', and 'Support'. Below the navigation is the breadcrumb path: 'Amazon S3 > thebucketofstephane-bigdata > stores > store-1'. A left sidebar with a 'Overview' tab is partially visible. The main content area is titled 'Overview' and contains a search bar with placeholder text 'Type a prefix and press Enter to search. Press ESC to clear.' Below the search bar are four buttons: 'Upload' (blue), '+ Create folder' (white), 'Download' (white), and 'Actions' (dropdown). To the right of these buttons is the region 'US East (Ohio)' with a location pin icon. A table lists a single file named '10-22-2018'. The table has columns for 'Name', 'Last modified', 'Size', and 'Storage class', all showing '--'. Above the table, a message says 'Viewing 1 to 1'. At the bottom of the page is a dark footer bar with the text 'Operations 0 In progress 3 Success 0 Error'.

# Amazon S3 - Versioning



- You can version your files in Amazon S3
- It is enabled at the **bucket level**
- Same key overwrite will increment the “version”: 1, 2, 3....
- It is best practice to version your buckets
  - Protect against unintended deletes (ability to restore a version)
  - Easy roll back to previous version
- Notes:
  - Any file that is not versioned prior to enabling versioning will have version “null”
  - Suspending versioning does not delete the previous versions

Screenshot of the AWS S3 Bucket Properties page for the bucket "demo-stephane-s3-bucket-2020".

**Bucket Details:**

Region	Amazon resource name (ARN)	Creation date	Access
EU (Ireland) eu-west-1	arn:aws:s3:::demo-stephane-s3-bucket-2020	December 8, 2020, 17:22 (UTC+00:00)	Bucket and objects not public

**Properties Tab:**

**Bucket Versioning:**  
Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

**Multi-factor authentication (MFA) delete:**  
An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API. [Learn more](#)

**Tags (0)** [Edit](#)

Sales Services ▾

stephane-aws-course ▾ Global ▾ Support ▾

## Edit Bucket Versioning

**Bucket Versioning**

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

**Suspend**  
This suspends the creation of object versions for all operations but preserves any existing object versions.

**Enable**

**After enabling Bucket Versioning, you might need to update your lifecycle rules to manage previous versions of objects.**

**Multi-factor authentication (MFA) delete**  
An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API. [Learn more](#)

Disabled

**Cancel** **Save changes**

EU (Ireland) eu-west-1	arn:aws:s3:::demo-stephane-s3-bucket-2020	December 8, 2020, 17:22 (UTC+00:00)	Bucket and objects not public
------------------------	---	-------------------------------------	-------------------------------

**Objects** **Properties** **Permissions** **Metrics** **Management** **Access points**

Drag and drop files and folders you want to upload here, or choose **Upload**.

### Objects (2)

Objects are the fundamental entities stored in Amazon S3. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

**List versions** **Delete** **Actions ▾** **Create folder** **Upload**

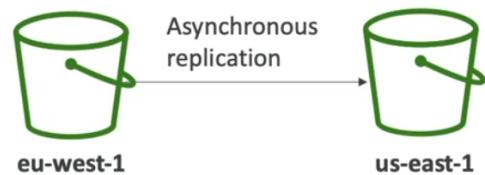
**Find objects by prefix**

<input type="checkbox"/>	Name	Type	Version ID	Last modified	Size	Storage class
<input type="checkbox"/>	<a href="#">beach.jpg</a>	jpg	Js4BL_tKpn1SnYKuNu1BmYaAXUHHLo77k	December 8, 2020, 17:31 (UTC+00:00)	85.8 KB	Standard
<input type="checkbox"/>	<a href="#">coffee.jpg</a>	jpg	null	December 8, 2020, 17:24 (UTC+00:00)	108.4 KB	Standard

in bucket versioning you can delete any file it will create delete marker you easily restore the file if you delete the market it will delete permanently

# S3 Replication (CRR & SRR)

- Must enable versioning in source and destination
- Cross Region Replication (CRR)
- Same Region Replication (SRR)
- Buckets can be in different accounts
- Copying is asynchronous
- Must give proper IAM permissions to S3



- CRR - Use cases: compliance, lower latency access, replication across accounts
- SRR – Use cases: log aggregation, live replication between production and test accounts

## S3 Replication – Notes

- After activating, only new objects are replicated
- Optionally, you can replicate existing objects using S3 Batch Replication
  - Replicates existing objects and objects that failed replication
- For DELETE operations:
  - Can replicate delete markers from source to target (optional setting)
  - Deletions with a version ID are not replicated (to avoid malicious deletes)
- There is no “chaining” of replication
  - If bucket 1 has replication into bucket 2, which has replication into bucket 3
  - Then objects created in bucket 1 are not replicated to bucket 3

under management

Lifecycle rule name	Status	Scope	Current version actions	Noncurrent versions actions	Expired object delete markers	Incomplete multipart uploads
No lifecycle rules						
There are no lifecycle rules for this bucket.						
<a href="#">Create lifecycle rule</a>						

Replication rules (0)										
Use replication rules to define options you want Amazon S3 to apply during replication such as server-side encryption, replica ownership, transitioning replicas to another storage class, and more. <a href="#">Learn more</a>										
Replication rule name	Status	Destination bucket	Destination Region	Priority	Scope	Storage class	Replica owner	Replication Time Control	KMS-encrypted objects	Replica modification sync
No replication rules										
You don't have any rules in the replication configuration.										
<a href="#">Create replication rule</a>										

Amazon S3 > Buckets > s3-stephane-bucket-origin-v2 > Replication rules > Create replication rule

## Create replication rule

### Replication rule configuration

Replication rule name

Up to 255 characters. In order to be able to use CloudWatch metrics to monitor the progress of your replication rule, the replication rule name must only contain English characters.

Status

Choose whether the rule will be enabled or disabled when created.

Enabled

Disabled

Priority

The priority value resolves conflicts that occur when an object is eligible for replication under multiple rules to the same destination. The rule is added to the configuration at the highest priority and the priority can be changed on the replication rules table.

0

**Source bucket**

#### Choose a rule scope

- Limit the scope of this rule using one or more filters
- Apply to all objects in the bucket

## Destination

#### Destination

You can replicate objects across buckets in different AWS Regions (Cross-Region Replication) or you can replicate objects across buckets in the same AWS Region (Same-Region Replication). You can also specify a different bucket for each rule in the configuration. [Learn more](#) 

- Choose a bucket in this account
- Specify a bucket in another account

#### Bucket name

Choose the bucket that will receive replicated objects.

Browse S3

#### Destination Region

-

## IAM role

Choose the bucket that will receive replicated objects.

Browse S3

#### Destination Region

US East (N. Virginia) us-east-1

## IAM role

- Choose from existing IAM roles
- Enter IAM role ARN

#### IAM role

▼



View 

## Encryption

- Replicate objects encrypted with AWS KMS

You can replicate objects that are encrypted with AWS Key Management Service (AWS KMS) keys.

# S3 Storage Classes

- Amazon S3 Standard - General Purpose
  - Amazon S3 Standard-Infrequent Access (IA)
  - Amazon S3 One Zone-Infrequent Access
  - Amazon S3 Glacier Instant Retrieval
  - Amazon S3 Glacier Flexible Retrieval
  - Amazon S3 Glacier Deep Archive
  - Amazon S3 Intelligent Tiering
- 
- Can move between classes manually or using S3 Lifecycle configurations

## S3 Storage Classes Comparison

	Standard	Intelligent-Tiering	Standard-IA	One Zone-IA	Glacier Instant Retrieval	Glacier Flexible Retrieval	Glacier Deep Archive
Durability	99.999999999% == (11 9's)						
Availability	99.99%	99.9%	99.9%	99.5%	99.9%	99.99%	99.99%
Availability SLA	99.9%	99%	99%	99%	99%	99.9%	99.9%
Availability Zones	>= 3	>= 3	>= 3	1	>= 3	>= 3	>= 3
Min. Storage Duration Charge	None	None	30 Days	30 Days	90 Days	90 Days	180 Days
Min. Billable Object Size	None	None	128 KB	128 KB	128 KB	40 KB	40 KB
Retrieval Fee	None	None	Per GB retrieved	Per GB retrieved	Per GB retrieved	Per GB retrieved	Per GB retrieved

# S3 Storage Classes – Price Comparison

## Example: us-east-1

	Standard	Intelligent-Tiering	Standard-IA	One Zone-IA	Glacier Instant Retrieval	Glacier Flexible Retrieval	Glacier Deep Archive
Storage Cost (per GB per month)	\$0.023	\$0.0025 - \$0.023	%0.0125	\$0.01	\$0.004	\$0.0036	\$0.00099
Retrieval Cost (per 1000 request)	GET: \$0.0004 POST: \$0.005	GET: \$0.0004 POST: \$0.005	GET: \$0.001 POST: \$0.01	GET: \$0.001 POST: \$0.01	GET: \$0.01 POST: \$0.02	GET: \$0.0004 POST: \$0.03  Expedited: \$10 Standard: \$0.05 Bulk: free	GET: \$0.0004 POST: \$0.05  Standard: \$0.10 Bulk: \$0.025
Retrieval Time	Instantaneous					Expedited (1 – 5 mins) Standard (3 – 5 hours) Bulk (5 – 12 hours)	Standard (12 hours) Bulk (48 hours)
Monitoring Cost (per 1000 objects)		\$0.0025					

Amazon S3 > Buckets > s3-storage-classes-demos-2022

### s3-storage-classes-demos-2022

Objects Properties Permissions Metrics Management Access Points

**Lifecycle rules (0)**  
 Use lifecycle rules to define actions you want Amazon S3 to take during an object's lifetime such as transitioning objects to another storage class, archiving them, or deleting them after a specified period of time. [Learn more](#)

C	View details	Edit	Delete	Actions	Create lifecycle rule

No lifecycle rules  
 There are no lifecycle rules for this bucket.

Create lifecycle rule

**Replication rules (0)**

create a name

click apply all objects in the bucket

and on the move current version of object to storage classes

Choose the actions you want this rule to perform. Per-request fees apply. [Learn more](#) or see [Amazon S3 pricing](#)

- Move current versions of objects between storage classes
  - Move noncurrent versions of objects between storage classes
  - Expire current versions of objects
  - Permanently delete noncurrent versions of objects
  - Delete expired object delete markers or incomplete multipart uploads
- These actions are not supported when filtering by object tags or object size.

### Transition current versions of objects between storage classes

Choose transitions to move current versions of objects between storage classes based on your use case scenario and performance access requirements. These transitions start from when the objects are created and are consecutively applied. [Learn more](#)

Choose storage class transitions

Days after object creation

Standard-IA

30

Remove

Intelligent-Tiering

60

Remove

Add transition

### Review transition and expiration actions

#### Current version actions

Day 0

- Objects uploaded



Day 30

- Objects move to Standard-IA



Day 60

- Objects move to Intelligent-Tiering



Day 180

- Objects move to Glacier Flexible Retrieval (formerly Glacier)

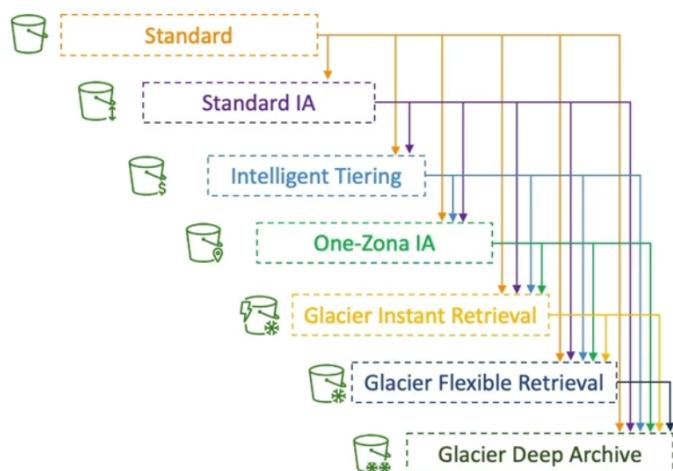
#### Noncurrent versions actions

Day 0

No actions defined.

# Amazon S3 – Moving between Storage Classes

- You can transition objects between storage classes
- For infrequently accessed object, move them to Standard IA
- For archive objects that you don't need fast access to, move them to Glacier or Glacier Deep Archive
- Moving objects can be automated using a Lifecycle Rules



## Amazon S3 – Lifecycle Rules



- Transition Actions – configure objects to transition to another storage class
  - Move objects to Standard IA class 60 days after creation
  - Move to Glacier for archiving after 6 months
- Expiration actions – configure objects to expire (delete) after some time
  - Access log files can be set to delete after a 365 days
  - Can be used to delete old versions of files (if versioning is enabled)
  - Can be used to delete incomplete Multi-Part uploads
- Rules can be created for a certain prefix (example: s3://mybucket/mp3/\*)
- Rules can be created for certain objects Tags (example: Department: Finance)

## Amazon S3 – Lifecycle Rules (Scenario 1)

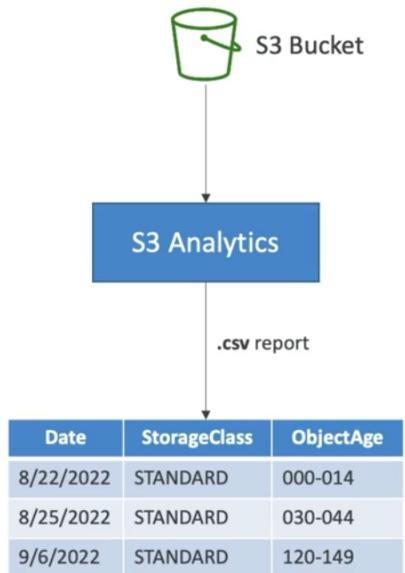
- Your application on EC2 creates images thumbnails after profile photos are uploaded to Amazon S3. These thumbnails can be easily recreated, and only need to be kept for 60 days. The source images should be able to be immediately retrieved for these 60 days, and afterwards, the user can wait up to 6 hours. How would you design this?
- S3 source images can be on Standard, with a lifecycle configuration to transition them to Glacier after 60 days
- S3 thumbnails can be on One-Zone IA, with a lifecycle configuration to expire them (delete them) after 60 days

## Amazon S3 – Lifecycle Rules (Scenario 2)

- A rule in your company states that you should be able to recover your deleted S3 objects immediately for 30 days, although this may happen rarely. After this time, and for up to 365 days, deleted objects should be recoverable within 48 hours.
- Enable S3 Versioning in order to have object versions, so that “deleted objects” are in fact hidden by a “delete marker” and can be recovered
- Transition the “noncurrent versions” of the object to Standard IA
- Transition afterwards the “noncurrent versions” to Glacier Deep Archive

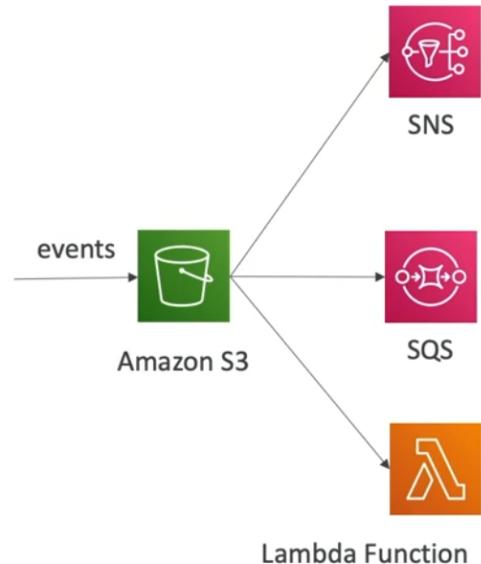
# Amazon S3 Analytics – Storage Class Analysis

- Help you decide when to transition objects to the right storage class
- Recommendations for Standard and Standard IA
  - Does NOT work for One-Zone IA or Glacier
- Report is updated daily
- 24 to 48 hours to start seeing data analysis
- Good first step to put together Lifecycle Rules (or improve them)!

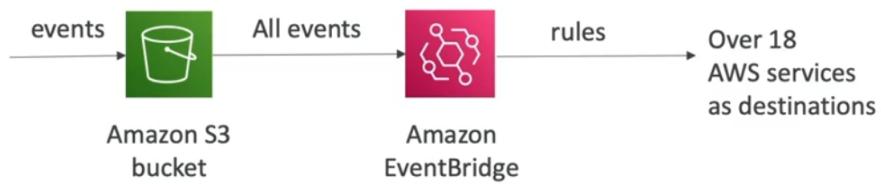


## S3 Event Notifications

- S3:ObjectCreated, S3:ObjectRemoved, S3:ObjectRestore, S3:Replication...
- Object name filtering possible (\*.jpg)
- Use case: generate thumbnails of images uploaded to S3
- Can create as many “S3 events” as desired
- S3 event notifications typically deliver events in seconds but can sometimes take a minute or longer



# S3 Event Notifications with Amazon EventBridge



- Advanced filtering options with JSON rules (metadata, object size, name...)
- Multiple Destinations – ex Step Functions, Kinesis Streams / Firehose...
- EventBridge Capabilities – Archive, Replay Events, Reliable delivery

Configure in CloudTrail [Edit](#) [Delete](#) [Create event notification](#)

Name	Event types	Filters	Destination type	Destination
No event notifications				
Choose <a href="#">Create event notification</a> to be notified when a specific event occurs.				
<a href="#">Create event notification</a>				

**Amazon EventBridge**  
For additional capabilities, use Amazon EventBridge to build event-driven applications at scale using S3 event notifications. [Learn more](#) or see [EventBridge pricing](#)

[Edit](#)

Send notifications to Amazon EventBridge for all events in this bucket  
Off

## Create event notification Info

To enable notifications, you must first add a notification configuration that identifies the events you want Amazon S3 to publish and the destinations where you want Amazon S3 to send the notifications.

### General configuration

#### Event name

Event name can contain up to 255 characters.

#### Prefix - optional

Limit the notifications to objects with key starting with specified characters.

#### Suffix - optional

Limit the notifications to objects with key ending with specified characters.

### Event types

Specify at least one event for which you want to receive notifications. For each group, you can choose an event type for all events, or you can choose one or more individual events.

#### Event types

Specify at least one event for which you want to receive notifications. For each group, you can choose an event type for all events, or you can choose one or more individual events.

#### Object creation

 All object create events  
s3:ObjectCreated:\*

- Put  
s3:ObjectCreated:Put
- Post  
s3:ObjectCreated:Post
- Copy  
s3:ObjectCreated:Copy
- Multipart upload completed  
s3:ObjectCreated:CompleteMultipartUpload

#### Object removal

 All object removal events  
s3:ObjectRemoved:\*

- Permanently deleted  
s3:ObjectRemoved:Delete
- Delete marker created  
s3:ObjectRemoved:DeleteMarkerCreated

#### Object restore

**Destination**

**ⓘ** Before Amazon S3 can publish messages to a destination, you must grant the Amazon S3 principal the necessary permissions to call the relevant API to publish messages to an SNS topic, an SQS queue, or a Lambda function. [Learn more](#) 

**Destination**  
Choose a destination to publish the event. [Learn more](#) 

**Lambda function**  
Run a Lambda function script based on S3 events.

**SNS topic**  
Send notifications to email, SMS, or an HTTP endpoint.

**SQS queue**  
Send notifications to an SQS queue to be read by a server.

**Specify SQS queue**

**Choose from your SQS queues**

**Enter SQS queue ARN**

**SQS queue**

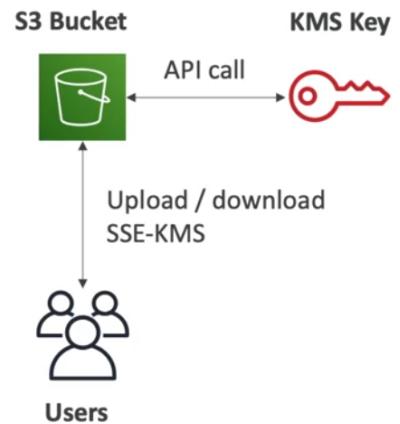


## S3 – Baseline Performance

- Amazon S3 automatically scales to high request rates, latency 100-200 ms
- Your application can achieve at least 3,500 PUT/COPY/POST/DELETE and 5,500 GET/HEAD requests per second per prefix in a bucket.
- There are no limits to the number of prefixes in a bucket.
- Example (object path => prefix):
  - bucket/folder1/sub1/file => /folder1/sub1/
  - bucket/folder1/sub2/file => /folder1/sub2/
  - bucket/1/file => /1/
  - bucket/2/file => /2/
- If you spread reads across all four prefixes evenly, you can achieve 22,000 requests per second for GET and HEAD

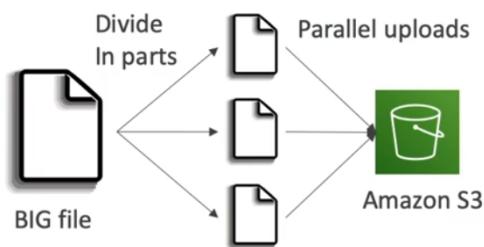
# S3 – KMS Limitation

- If you use SSE-KMS, you may be impacted by the KMS limits
- When you upload, it calls the `GenerateDataKey` KMS API
- When you download, it calls the `Decrypt` KMS API
- Count towards the KMS quota per second (5500, 10000, 30000 req/s based on region)
- You can request a quota increase using the Service Quotas Console



## S3 Performance

- Multi-Part upload:
  - recommended for files > 100MB, must use for files > 5GB
  - Can help parallelize uploads (speed up transfers)
- S3 Transfer Acceleration
  - Increase transfer speed by transferring file to an AWS edge location which will forward the data to the S3 bucket in the target region
  - Compatible with multi-part upload

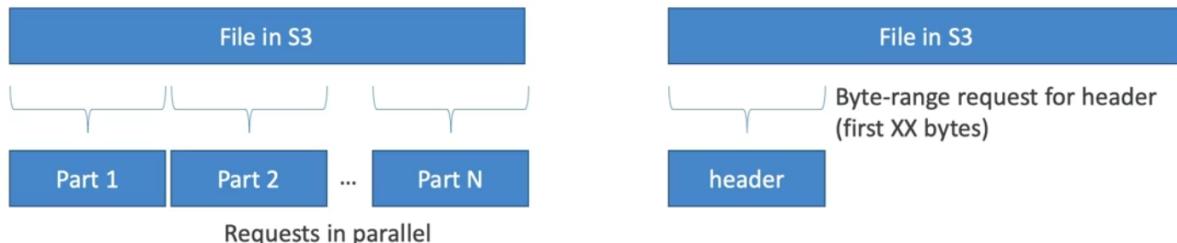


# S3 Performance – S3 Byte-Range Fetches

- Parallelize GETs by requesting specific byte ranges
- Better resilience in case of failures

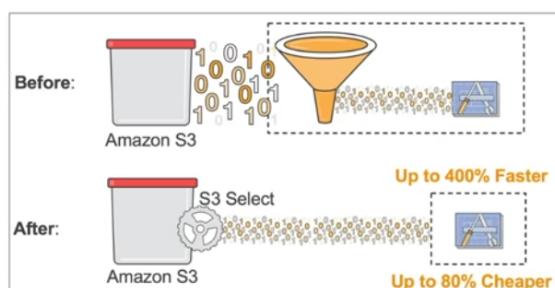
Can be used to speed up downloads

Can be used to retrieve only partial data (for example the head of a file)

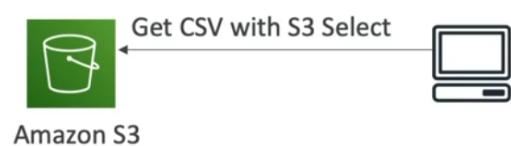


## S3 Select & Glacier Select

- Retrieve less data using SQL by performing server-side filtering
- Can filter by rows & columns (simple SQL statements)
- Less network transfer; less CPU cost client-side



<https://aws.amazon.com/blogs/aws/s3-glacier-select/>



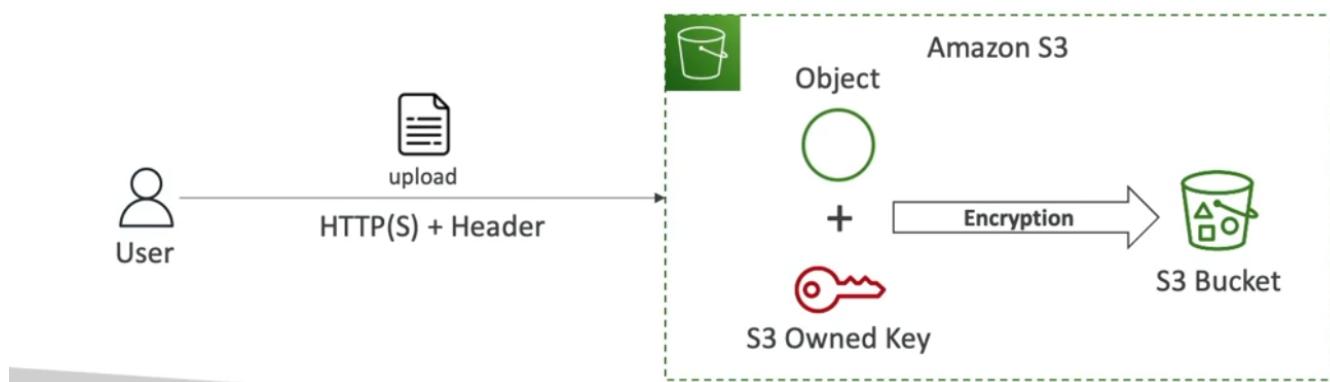
# Amazon S3 – Object Encryption



- You can encrypt objects in S3 buckets using one of 4 methods
- Server-Side Encryption (SSE)
  - Server-Side Encryption with Amazon S3-Managed Keys (SSE-S3)
    - Encrypts S3 objects using keys handled, managed, and owned by AWS
  - Server-Side Encryption with KMS Keys stored in AWS KMS (SSE-KMS)
    - Leverage AWS Key Management Service (AWS KMS) to manage encryption keys
  - Server-Side Encryption with Customer-Provided Keys (SSE-C)
    - When you want to manage your own encryption keys
- Client-Side Encryption
- It's important to understand which ones are for which situation for the exam

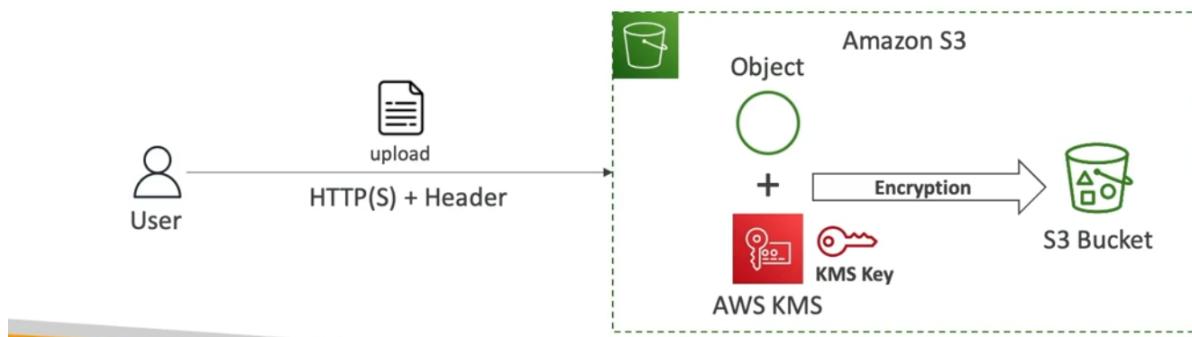
## Amazon S3 Encryption – SSE-S3

- Encryption using keys handled, managed, and owned by AWS
- Object is encrypted server-side
- Encryption type is AES-256
- Must set header "x-amz-server-side-encryption": "AES256"



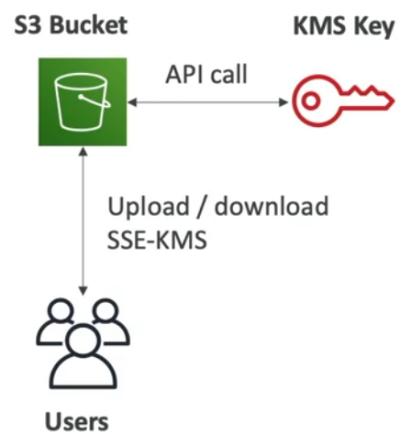
# Amazon S3 Encryption – SSE-KMS

- Encryption using keys handled and managed by AWS KMS (Key Management Service)
- KMS advantages: user control + audit key usage using CloudTrail
- Object is encrypted server side
- Must set header "x-amz-server-side-encryption": "aws:kms"



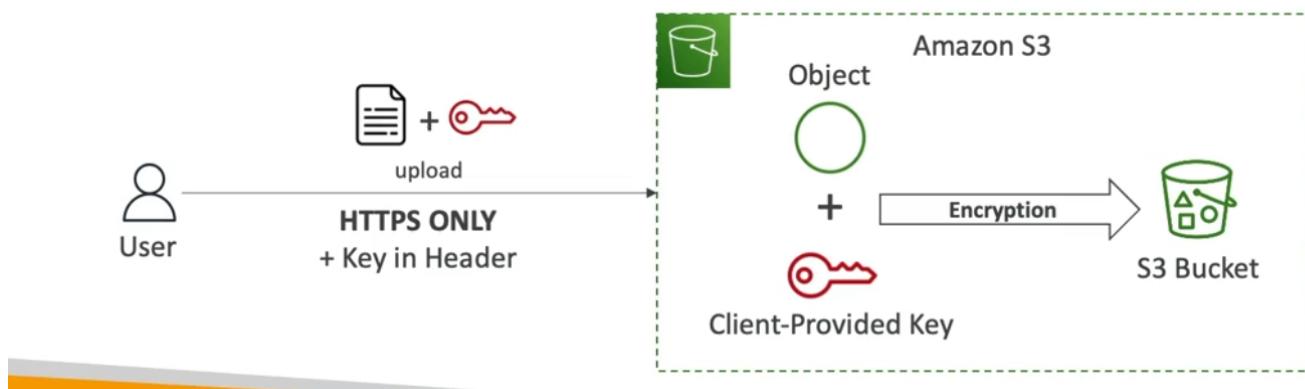
## SSE-KMS Limitation

- If you use SSE-KMS, you may be impacted by the KMS limits
- When you upload, it calls the `GenerateDataKey` KMS API
- When you download, it calls the `Decrypt` KMS API
- Count towards the KMS quota per second (5500, 10000, 30000 req/s based on region)
- You can request a quota increase using the Service Quotas Console



# Amazon S3 Encryption – SSE-C

- Server-Side Encryption using keys fully managed by the customer outside of AWS
- Amazon S3 does NOT store the encryption key you provide
- HTTPS must be used
- Encryption key must provided in HTTP headers, for every HTTP request made



# Amazon S3 Encryption – Client-Side Encryption

- Use client libraries such as Amazon S3 Client-Side Encryption Library
- Clients must encrypt data themselves before sending to Amazon S3
- Clients must decrypt data themselves when retrieving from Amazon S3
- Customer fully manages the keys and encryption cycle



# Amazon S3 – Encryption in transit (SSL/TLS)

- Encryption in flight is also called SSL/TLS
- Amazon S3 exposes two endpoints:
  - HTTP Endpoint – non encrypted
  - HTTPS Endpoint – encryption in flight
- HTTPS is recommended
- HTTPS is mandatory for SSE-C
- Most clients would use the HTTPS endpoint by default



under properties -> serverside encryption

**Server-side encryption**

Server-side encryption protects data at rest. [Learn more](#)

**Server-side encryption**

Do not specify an encryption key

Specify an encryption key

**Encryption key type**

To upload an object with a customer-provided encryption key (SSE-C), use the AWS CLI, AWS SDK, or Amazon S3 REST API.

**Amazon S3-managed keys (SSE-S3)**

An encryption key that Amazon S3 creates, manages, and uses for you. [Learn more](#)

**AWS Key Management Service key (SSE-KMS)**

An encryption key protected by AWS Key Management Service (AWS KMS). [Learn more](#)

**Additional checksums**

Checksum functions are used for additional data integrity verification of new objects. [Learn more](#)

**Additional checksums**

**Off**

Amazon S3 will use a combination of MD5 checksums and Etags to verify data integrity.

**On**

Specify a checksum function for additional data integrity validation

**Tags (0)**

Track storage cost or other criteria by tagging your bucket. [Learn more](#)

Key	Value
No tags associated with this resource.	

**Default encryption**

Automatically encrypt new objects stored in this bucket. [Learn more](#)

Default encryption
Disabled

**Intelligent-Tiering Archive configurations (0)**

Enable objects stored in the Intelligent-Tiering storage class to tier-down to the Archive Access tier or the Deep Archive Access tier which are optimized for objects that will be rarely accessed for long periods of time. [Learn more](#)

[View details](#) [Edit](#) [Delete](#) [Create configuration](#)

Find Intelligent-Tiering Archive configurations

## Amazon S3 – Default Encryption vs. Bucket Policies

- One way to “force encryption” is to use a bucket policy and refuse any API call to PUT an S3 object without encryption headers

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyIncorrectDecryptionHeader",
      "Effect": "Deny",
      "Principal": "*",
      "Action": [ "s3:PutObject" ],
      "Resource": [ "arn:aws:s3:::examplebucket/*" ],
      "Condition": {
        "StringNotEquals": {
          "s3:x-amz-server-side-encryption": "AES256"
        }
      }
    }
  ]
}

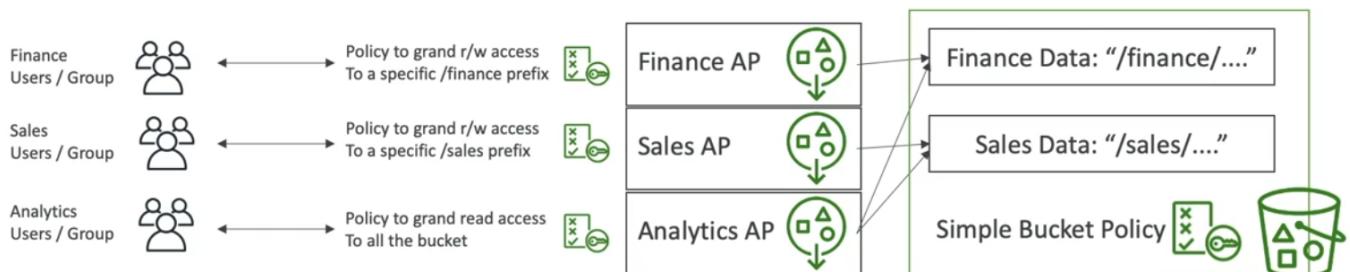
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyUnencryptedObjectUploads",
      "Effect": "Deny",
      "Principal": "*",
      "Action": [ "s3:PutObject" ],
      "Resource": [ "arn:aws:s3:::examplebucket/*" ],
      "Condition": {
        "Null": {
          "s3:x-amz-server-side-encryption": true
        }
      }
    }
  ]
}
```

- Another way is to use the “default encryption” option in S3

## S3 – Access Points

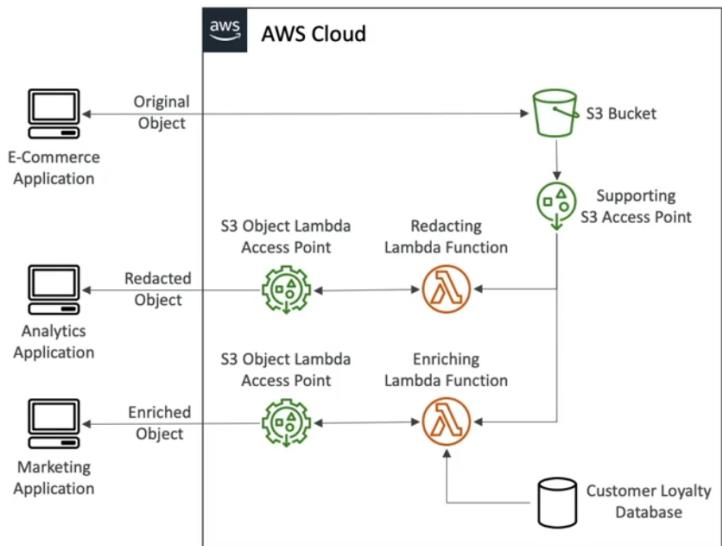


- Each Access Point gets its own DNS and policy to limit who can access it
  - A specific IAM user / group
  - One policy per Access Point => Easier to manage than complex bucket policies



# S3 Object Lambda

- Use AWS Lambda Functions to change the object before it is retrieved by the caller application
- Only one S3 bucket is needed, on top of which we create S3 Access Point and S3 Object Lambda Access Points.



- Use Cases:

- Redacting personally identifiable information for analytics or non-production environments.
- Converting across data formats, such as converting XML to JSON.
- Resizing and watermarking images on the fly using caller-specific details, such as the user who requested the object.