S3 Storage Class

- 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 Durability Ad Availability

- Durability:
  - High durability (99.999999999%, 11 9's) of objects across multiple AZ
  - If you store 10,000,000 objects with Amazon S3, you can on average expect to incur a loss of a single object once every 10,000 years
  - Same for all storage classes

- Availability:
  - Measures how readily available a service is
  - Varies depending on storage class
  - Example: S3 standard has 99.99% availability = not available 53 minutes a year

S3 Standard – General

- 99.99% Availability
- Used for frequently accessed data
- Low latency and high throughput
- Sustain 2 concurrent facility failures

- Use Cases: Big Data analytics, mobile & gaming applications, content distribution…

S3 Storage Classes – Infrequent Access

- For data that is less frequently accessed, but requires rapid access when needed
- Lower cost than S3 Standard

- Amazon S3 Standard-Infrequent Access (S3 Standard-IA)
  - 99.9% Availability
  - Use cases: Disaster Recovery, backups

- Amazon S3 One Zone-Infrequent Access (S3 One Zone-IA)
  - High durability (99.999999999%) in a single AZ; data lost when AZ is destroyed
  - 99.5% Availability
  - Use Cases: Storing secondary backup copies of on-premise data, or data you can recreate

S3 Glacier Storage Classes

- Low-cost object storage meant for archiving / backup
- Pricing: price for storage + object retrieval cost

- Amazon S3 Glacier Instant Retrieval
  - Millisecond retrieval, great for data accessed once a quarter
  - Minimum storage duration of 90 days
- Amazon S3 Glacier Flexible Retrieval (formerly Amazon S3 Glacier):
  - Expedited (1 to 5 minutes), Standard (3 to 5 hours), Bulk (5 to 12 hours) – free
  - Minimum storage duration of 90 days
- Amazon S3 Glacier Deep Archive – for long term storage:
  - Standard (12 hours), Bulk (48 hours)
  - Minimum storage duration of 180 days

S3 Intelligent – Tiering

- Small monthly monitoring and auto-tiering fee
- Moves objects automatically between Access Tiers based on usage
- There are no retrieval charges in S3 Intelligent-Tiering

- *Frequent Access tier (automatic):* default tier
- *Infrequent Access tier (automatic):* objects not accessed for 30 days
- *Archive Instant Access tier (automatic):* objects not accessed for 90 days
- *Archive Access tier (optional):* configurable from 90 days to 700+ days
- *Deep Archive Access tier (optional):* config. from 180 days to 700+ days

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

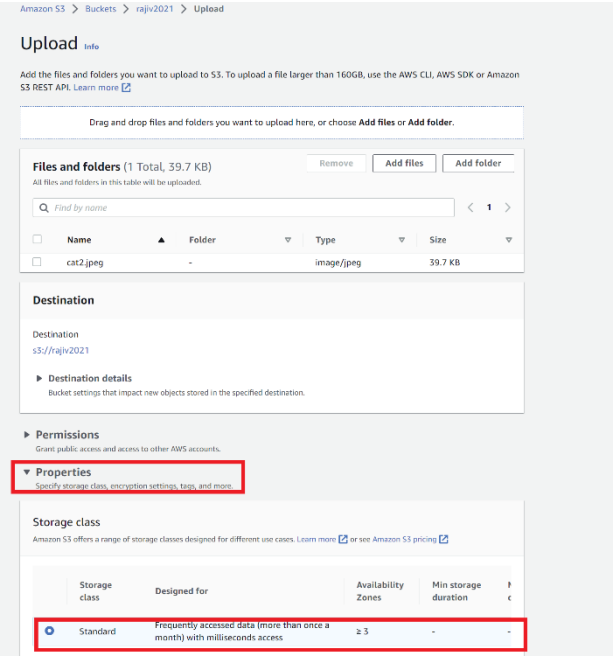
	Standard	Intelligent-Tiering	Standard-IA	One Zone-IA	Glacier Instant Retrieval	Glacier Flexible Retrieval	Glacier Deep Archive
Storage Cost (per GB per month)	$0.023	$0.0025 - $0.023	%0.0125	$0.01	$0.004	$0.0036	$0.00099
Retrieval Cost (per 1000 request)	**GET:** $0.0004 **POST:** $0.005	**GET:** $0.0004 **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 (pet 1000 objects)		$0.0025					

Pricing URL:
https://aws.amazon.com/s3/pricing/

Storage class details
https://aws.amazon.com/s3/storage-classes/

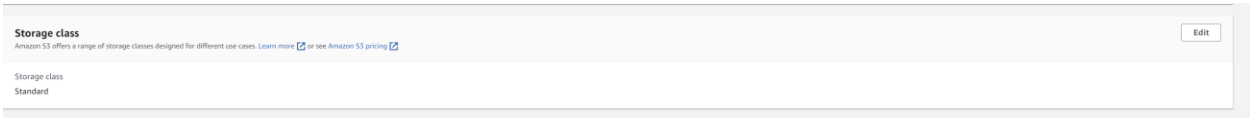
Lab: storage class

Create a bucket and then upload image
or
Choose storage class when upload an image

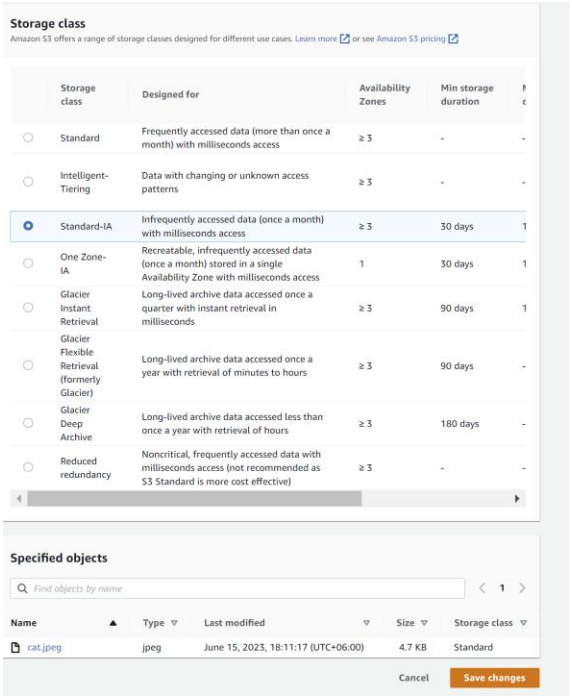


Edit the storage class

click the image>scroll down>click Edit button of Storage class

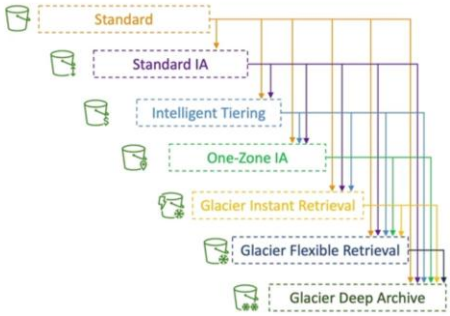


Now select any class and click save changes.



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**

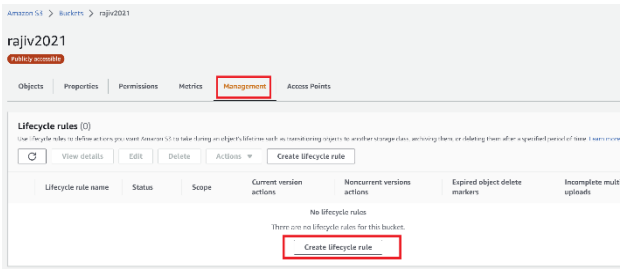


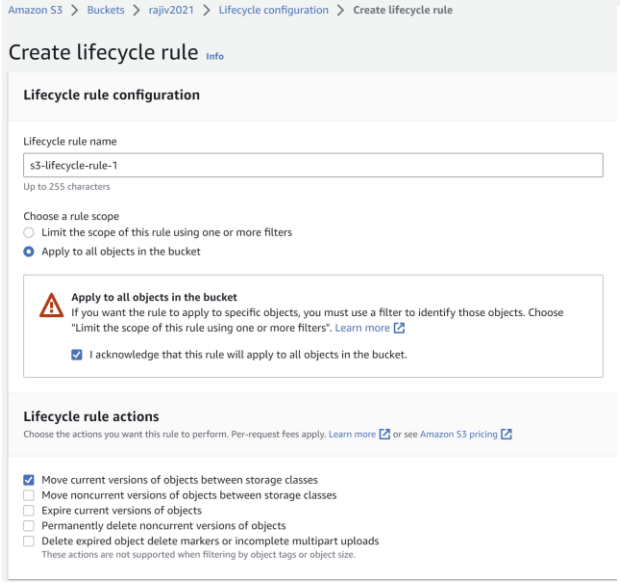
Amazon S3 – Life cycle rule

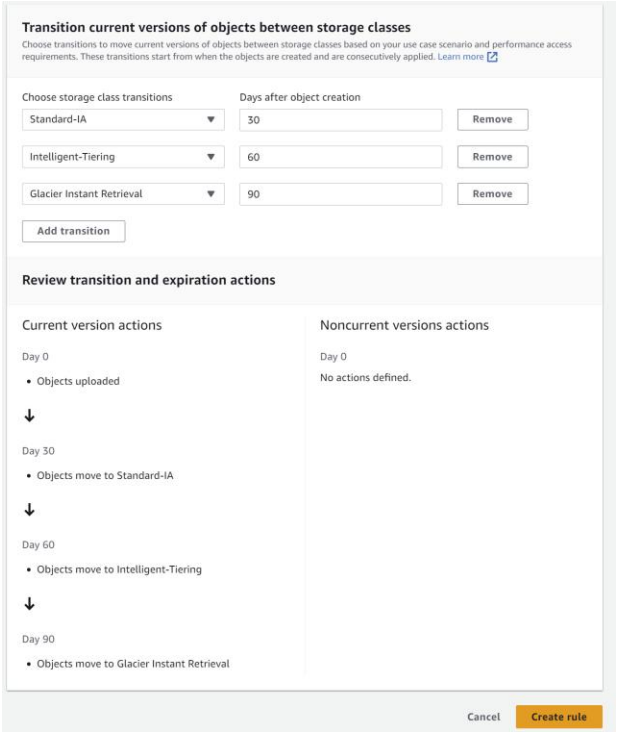
- **Transition Actions** – configure objects to transition to another storage class
  - Move objects to Standard IA class 60 days after creation
  - Move to Glacier for archiving after 6 months
- **Expiration actions** – configure objects to expire (delete) after some time
  - Access log files can be 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*)

LAB: Create life cycle rule:

Select or get in the bucket > Management>crate lifecycle rule:


rajiv2021
Publicly accessible
Objects
Properties
Permissions
Metrics
Management
Access Points
Lifecycle rules (0)
View details
Edit
Delete
Actions
Create lifecycle rule
Lifecycle rule name
Status
Scope
Current version actions
Noncurrent versions actions
Expired object delete markers
No lifecycle rules
There are no lifecycle rules for this bucket.
Create lifecycle rule


Amazon S3 > Buckets > rajiv2021 > Lifecycle configuration > Create lifecycle rule
Create lifecycle rule Info
Lifecycle rule configuration
Lifecycle rule name
s3-lifecycle-rule-1
Up to 255 characters
Choose a rule scope
Limit the scope of this rule using one or more filters
Apply to all objects in the bucket
Apply to all objects in the bucket
If you want the rule to apply to specific objects, you must use a filter to identify those objects. Choose "Limit the scope of this rule using one or more filters". Learn more
I acknowledge that this rule will apply to all objects in the bucket.
Lifecycle rule actions
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
Glacier Instant Retrieval
90
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 90
Objects move to Glacier Instant Retrieval
Noncurrent versions actions
Day 0
No actions defined.
Cancel
Create rule

Amazon S3 – Life cycle 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 – Life cycle rules (Scenario-2)

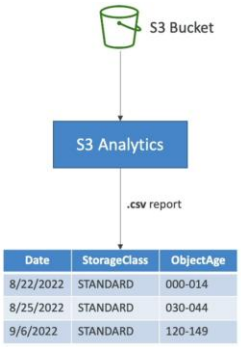
- A rule in your company states that you should be able to recover your deleted S3 objects immediately for 30 days, although this may happen rarely. After this time, and for up to 365 days, deleted objects should be recoverable within 48 hours.

- 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**

NB: non-current version means when we delete any object that object is save with a delete marker this deleted objects are non-current version.

Amazon S3 Analytics – Storage Class Analysis

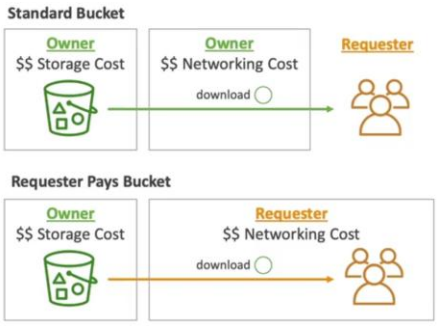
- Help you decide when to transition objects to the right storage class
- Recommendations for **Standard** and **Standard IA**
  - Does NOT work for One-Zone IA or Glacier
- Report is updated daily
- 24 to 48 hours to start seeing data analysis



Date	StorageClass	ObjectAge
8/22/2022	STANDARD	000-014
8/25/2022	STANDARD	030-044
9/6/2022	STANDARD	120-149

S3 Requester pay.

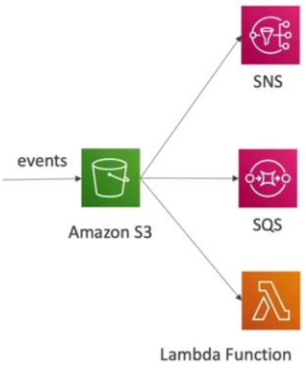
- In general, bucket owners pay for all Amazon S3 storage and data transfer costs associated with their bucket
- **With Requester Pays buckets**, the requester instead of the bucket owner pays the cost of the request and the data download from the bucket
- Helpful when you want to share large datasets with other accounts
- The requester must be authenticated in AWS (cannot be anonymous)



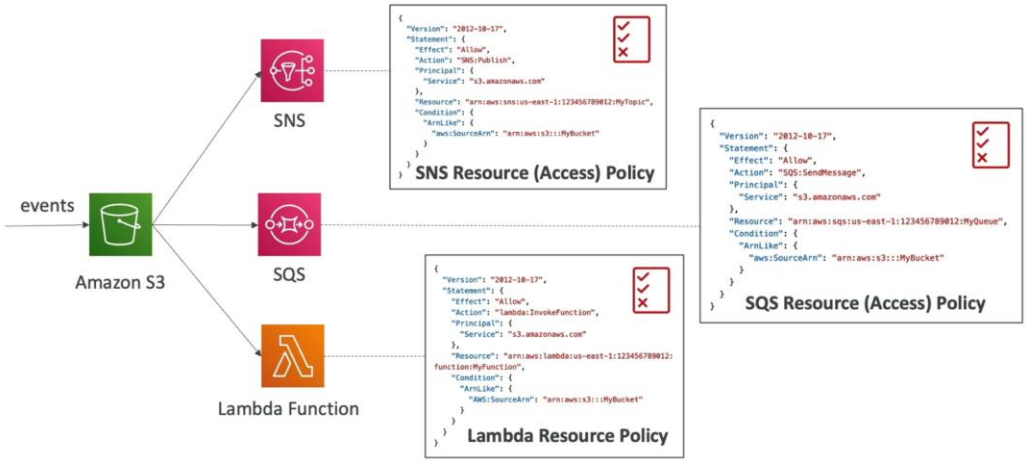
S3 Event Notifications

- 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

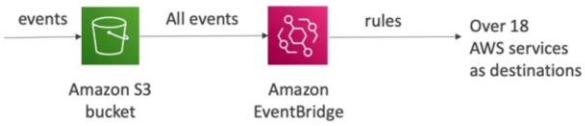
- S3 event notifications typically deliver events in seconds but can sometimes take a minute or longer



S3 Event Notifications – IAM Permissions



S3 Event Notifications with Amazon EventBridge

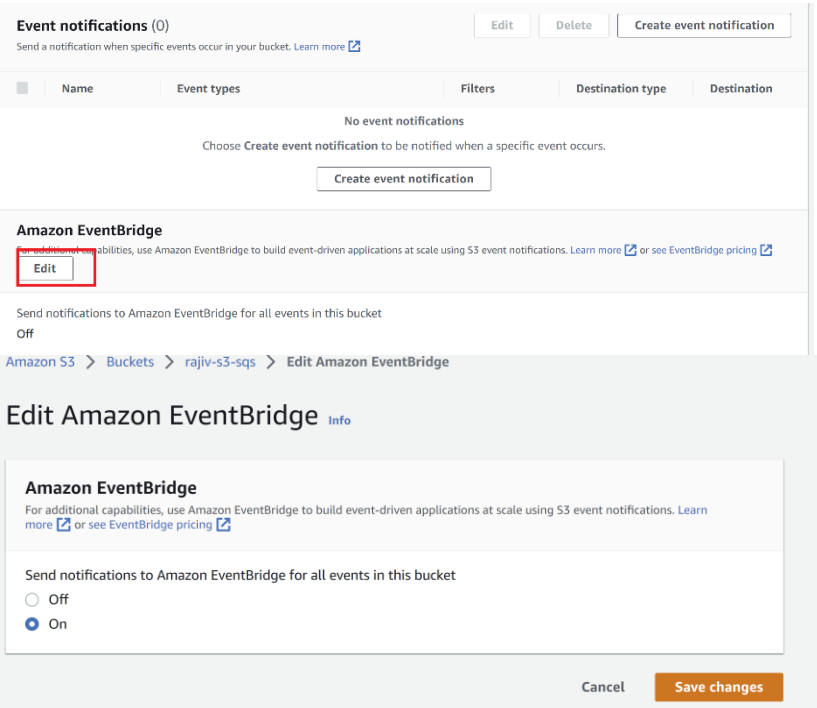


- **Advanced filtering** options with JSON rules (metadata, object size, name...)
- **Multiple Destinations** – ex Step Functions, Kinesis Streams / Firehose…
- **EventBridge Capabilities** – Archive, Replay Events, Reliable delivery

Lab: s3 event notification

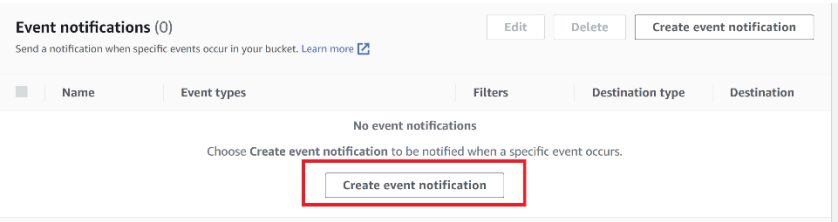
Create a bucket

select the bucket>properties>scroll down>edit amazon event bridge>select ON>click save changes.



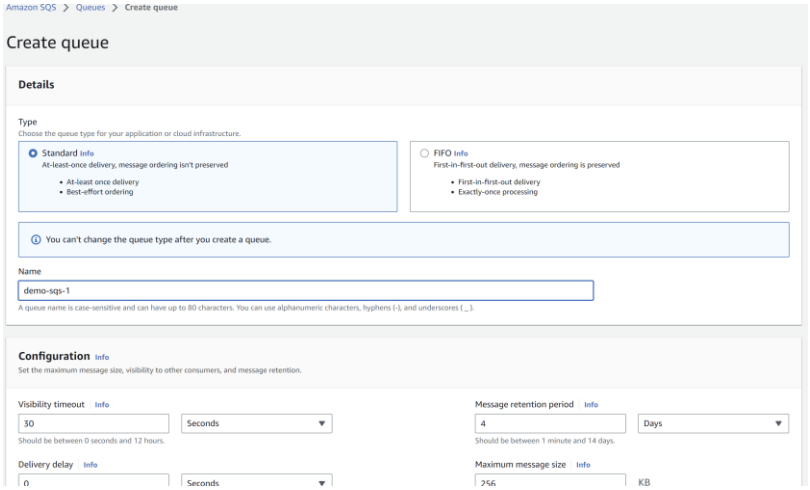
Event bridge is complicated so we will only see event notifications.

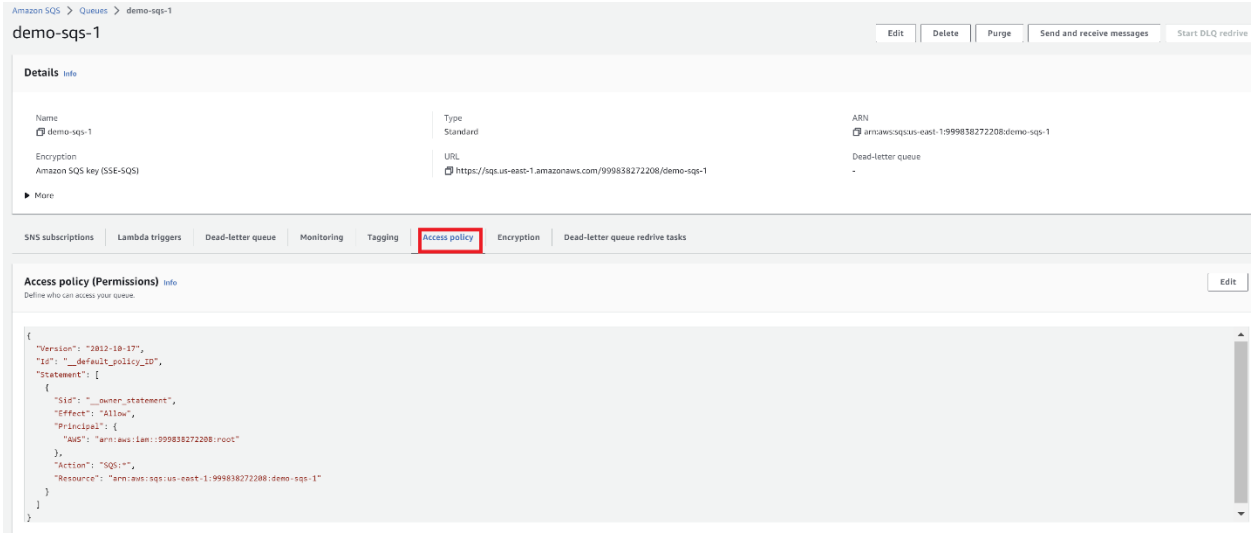
select the bucket>properties>scroll down>click event notification



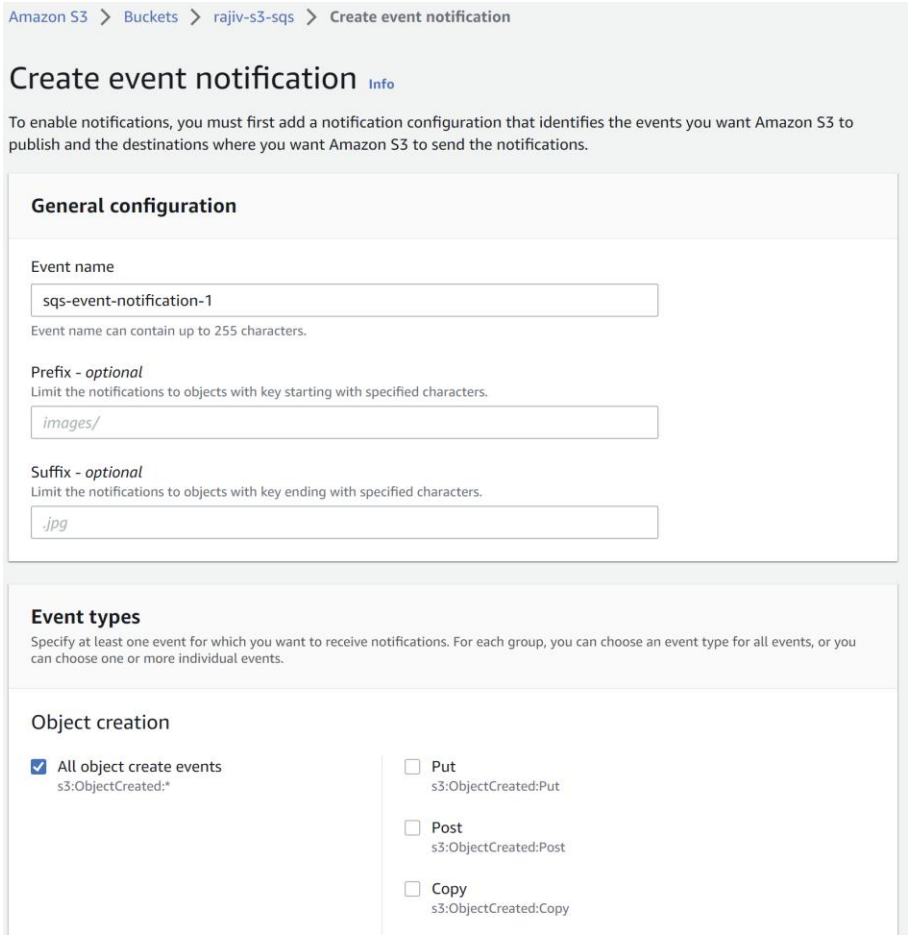
before doing anything create a SQS

give a SQS name and rest of thing keep same and click create que.

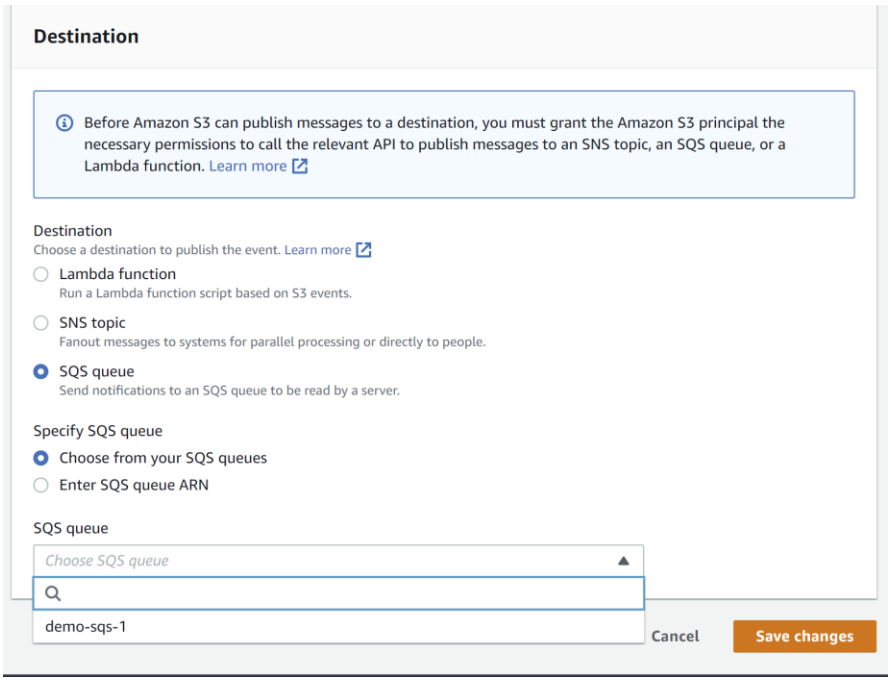




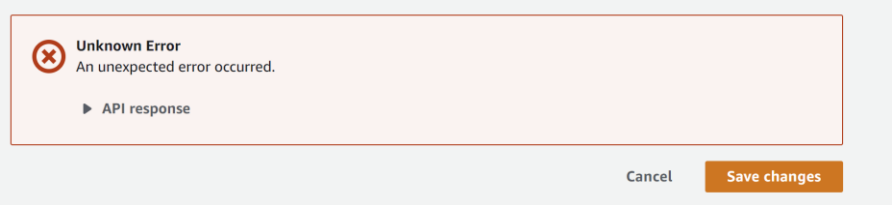
Now go to the event creation page and refresh it and give the name and select sqs and select the created sqs


Amazon S3 > Buckets > rajiv-s3-sqs > Create event notification
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
sqs-event-notification-1
Event name can contain up to 255 characters.
Prefix - optional
Limit the notifications to objects with key starting with specified characters.
images/
Suffix - optional
Limit the notifications to objects with key ending with specified characters.
.jpg
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

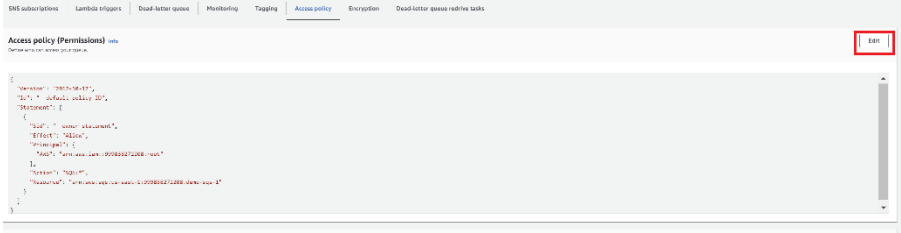
Scroll down


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
Fanout messages to systems for parallel processing or directly to people.
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
Choose SQS queue
demo-sqs-1
Cancel
Save changes

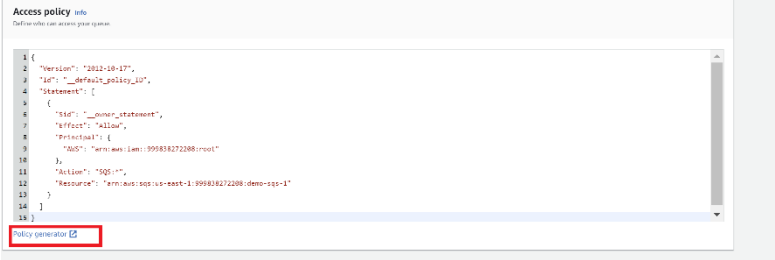
Click save changes and you will get and error

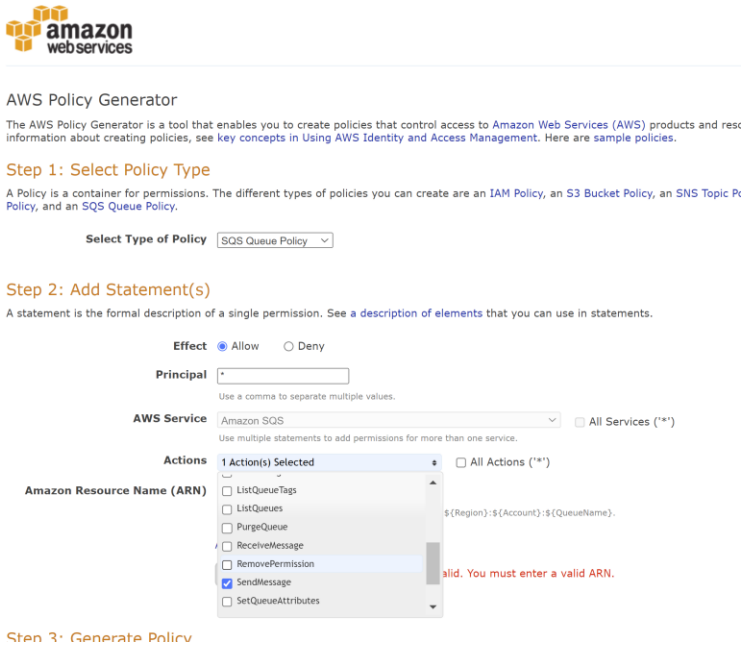


For fix this need to edit the access policy

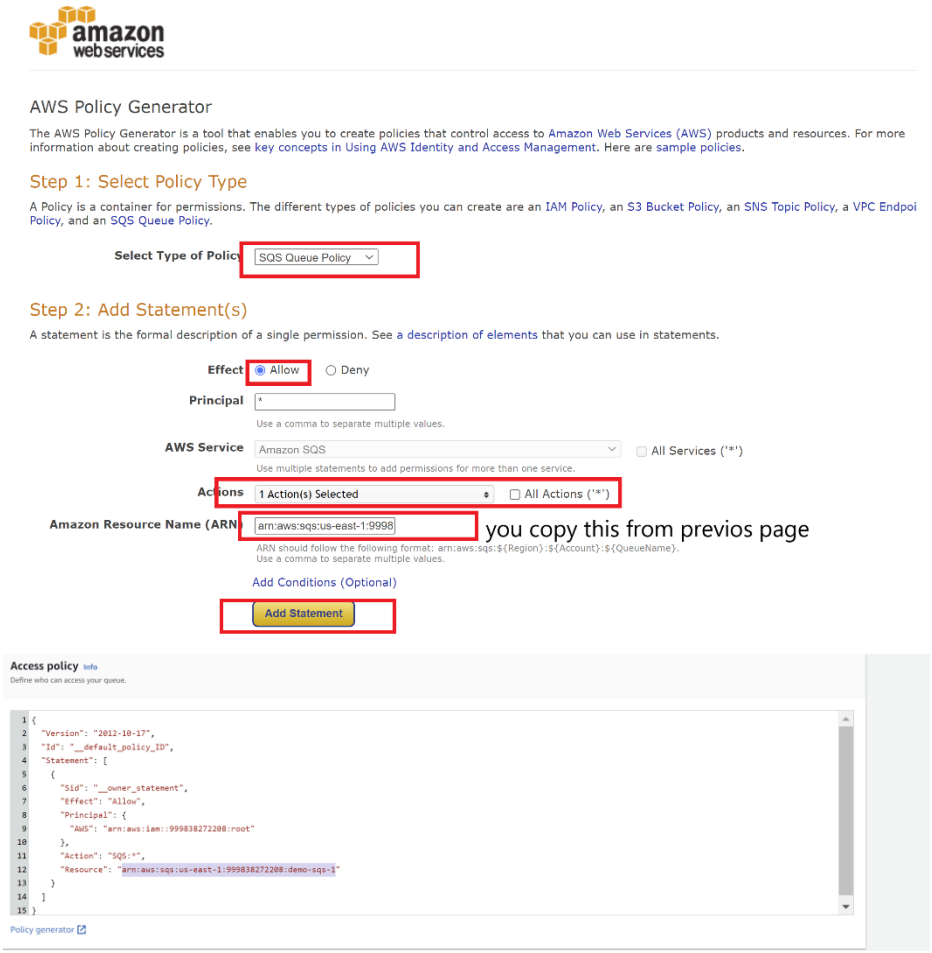


Scroll down and click policy generator

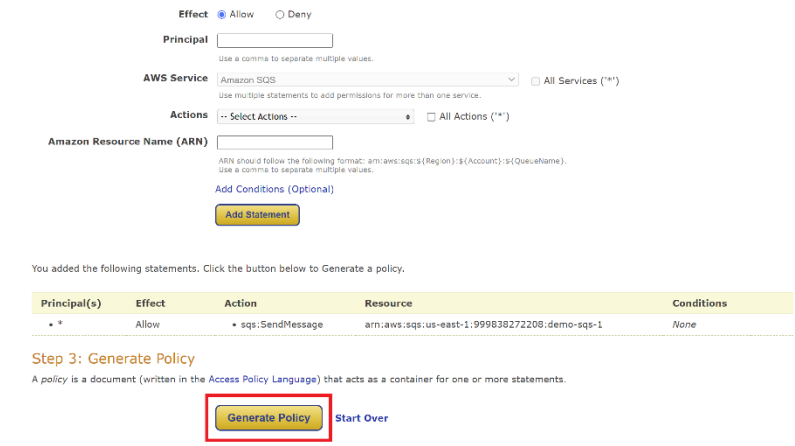


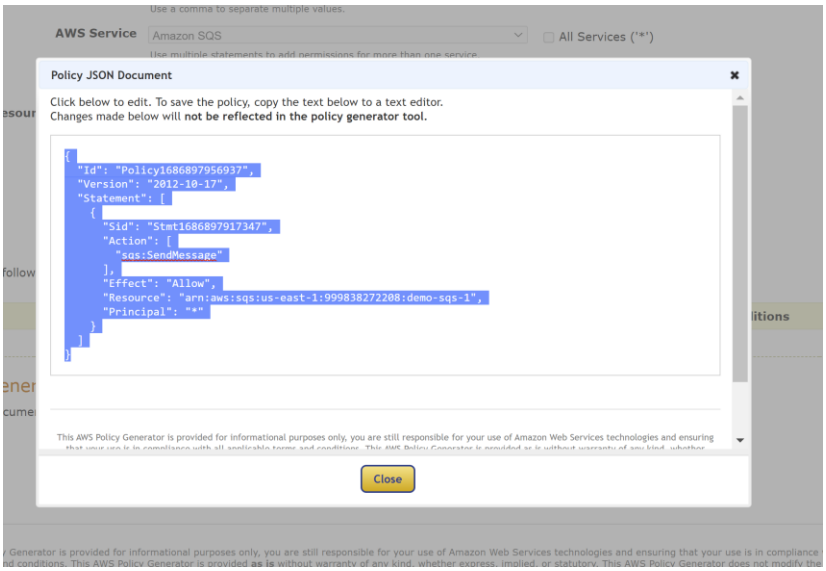


now go to the policy page and copy the arn and pest it here

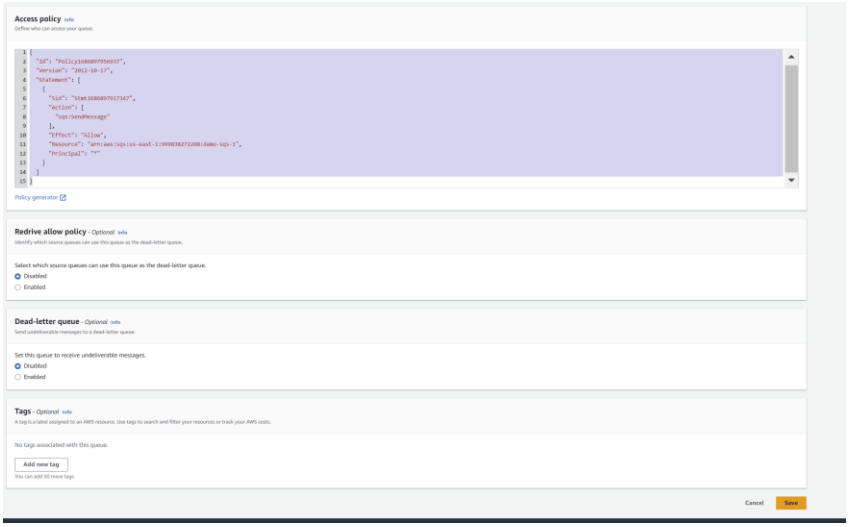


After clicking Add Statement

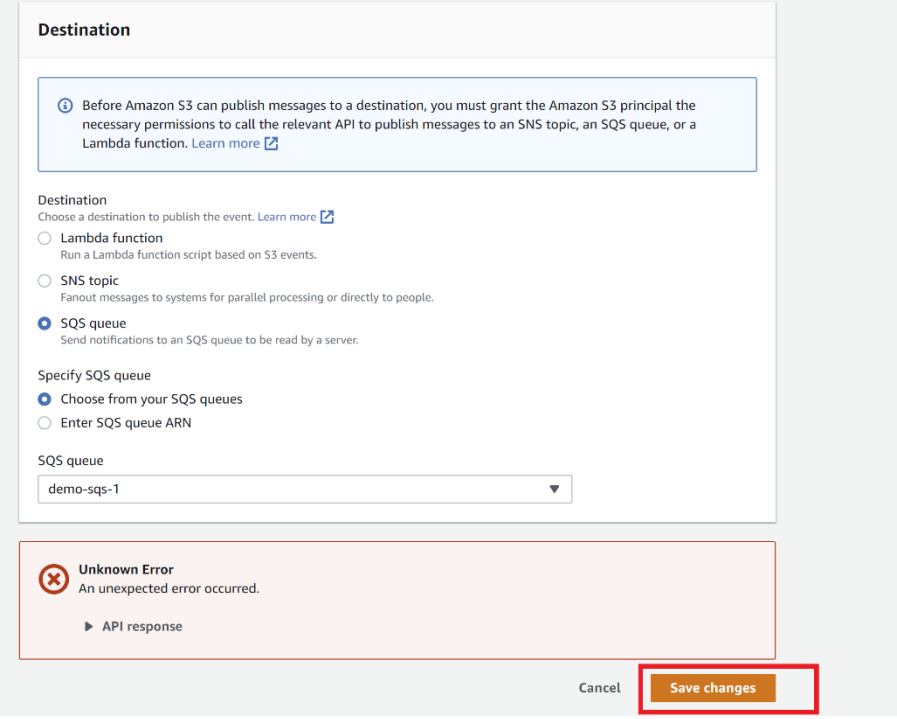




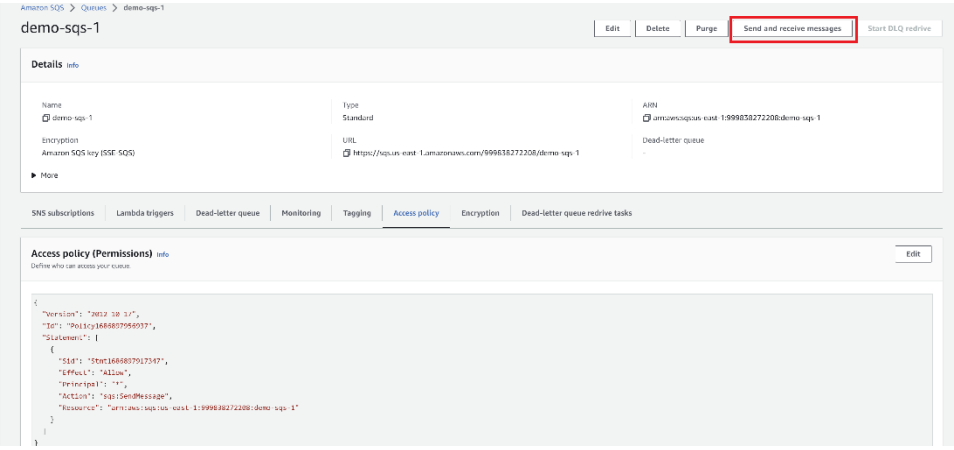
Now copy this and pest it in access policy which is in previous page and save it

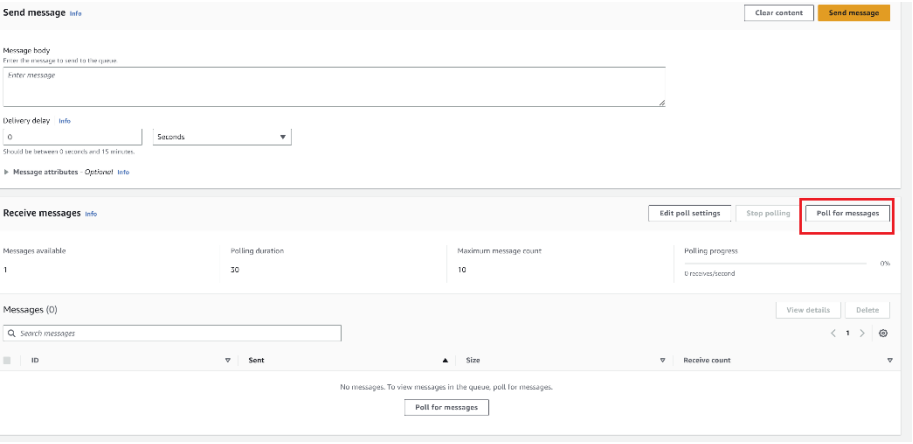


Now go to the S3 page and click save changes and now you will not get the error

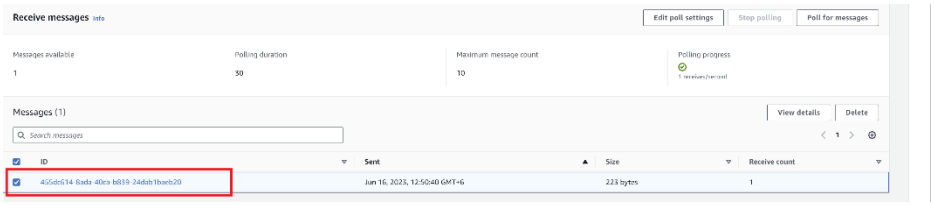


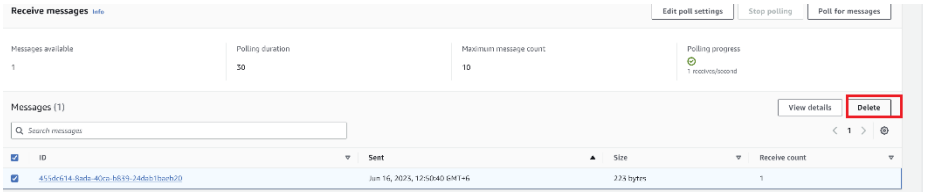
Now go to the sqs page and click send and receive messages then scroll down



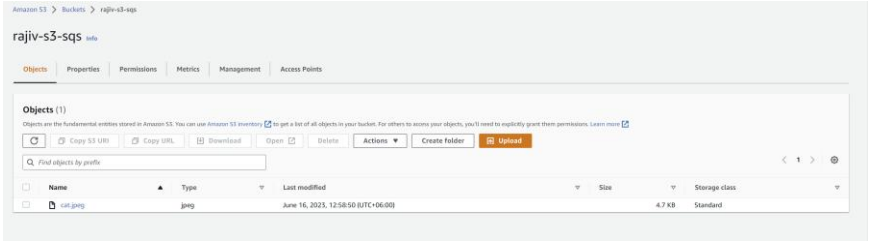


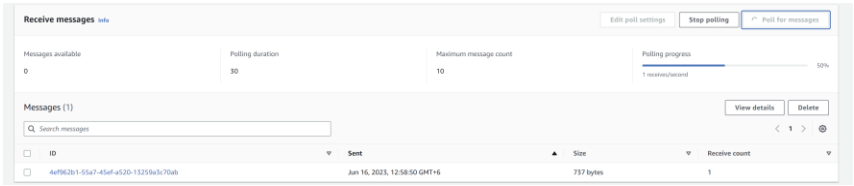
Now click the message and you can see the message after read click the done and delete the message

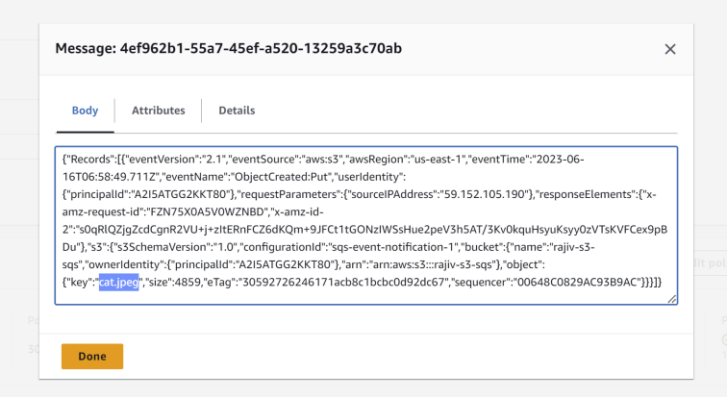




Now go to s3 page and upload an image then go to sql notification page and pull the message







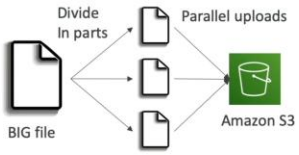
After reading the message delete it

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 or 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 performance

- Multi-Part upload:
  - recommended for files > 100MB, must use for files > 5GB
  - Can help parallelize uploads (speed up transfers)

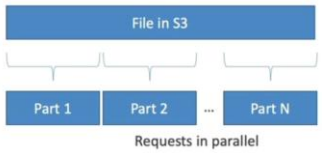


- S3 Transfer Acceleration
  - Increase transfer speed by transferring file to an AWS edge location which will forward the data to the S3 bucket in the target region
  - Compatible with multi-part upload

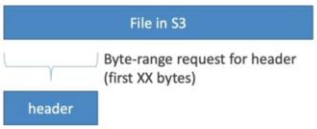
S3 performance – S3 Byte-Range Fetches

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

Can be used to speed up downloads

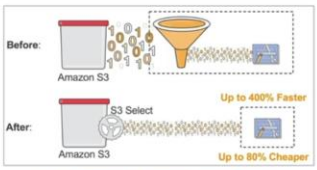


Can be used to retrieve only partial data (for example the head of a file)



S3 Select and Glacier Select

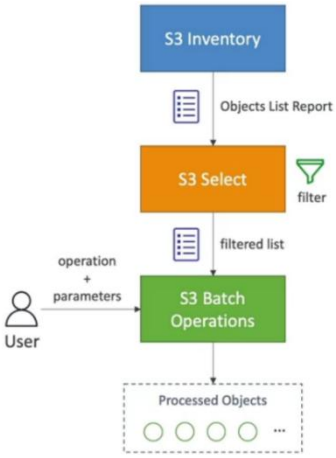
- Retrieve less data using SQL by performing server-side filtering
- Can filter by rows & columns (simple SQL statements)
- Less network transfer, less CPU cost client-side



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

S3 Batch Operations

- Perform bulk operations on existing S3 objects with a single request, example:
  - Modify object metadata & properties
  - Copy objects between S3 buckets
  - Encrypt un-encrypted objects
  - Modify ACLs, tags
  - Restore objects from S3 Glacier
  - Invoke Lambda function to perform custom action on each object
- A job consists of a list of objects, the action to perform, and optional parameters
- S3 Batch Operations manages retries, tracks progress, sends completion notifications, generate reports …
- You can use S3 Inventory to get object list and use S3 Select to filter your objects



S3 – Object Encryption

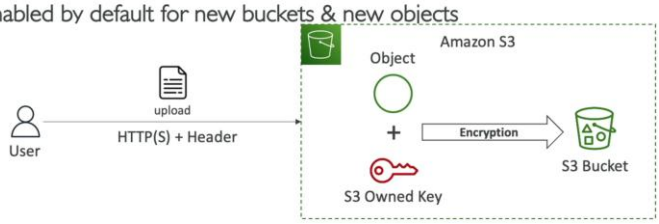
- You can encrypt objects in S3 buckets using one of 4 methods

- Server-Side Encryption (SSE)
  - Server-Side Encryption with Amazon S3-Managed Keys (SSE-S3) – Enabled by Default
    - 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

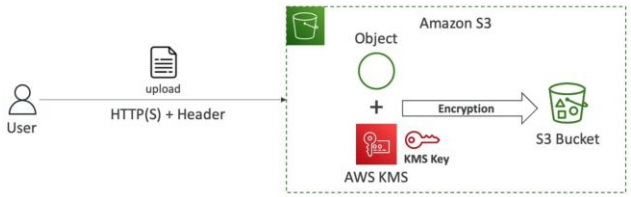
S3 Encryption – SSE-S3

- 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"
- Enabled by default for new buckets & new objects



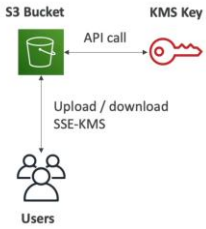
S3 Encryption – SSE-KMS

- 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"



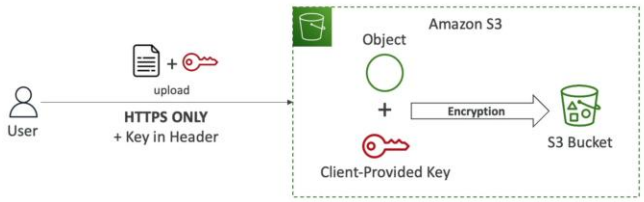
SSE-KMS Limitation

- 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



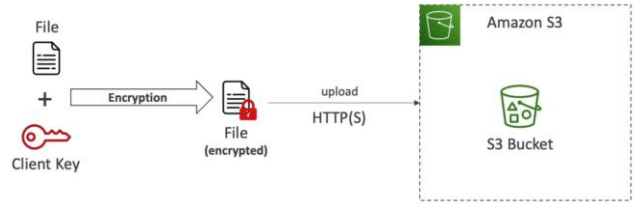
S3 Encryption – SSE-C

- 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



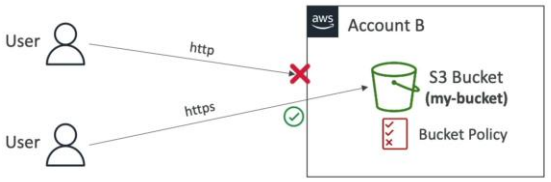
Amazon S3 Encryption in transit (SSL/TSL)

- 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

Amazon S3 – Force Encryption in Transit aws: Secure Transport




```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Principal": "*",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::my-bucket/*",
            "Condition": {
                "Bool": {
                    "aws:SecureTransport": "false"
                }
            }
        }
    ]
}
```

Amazon S3 – Default Encryption vs Bucket Policies

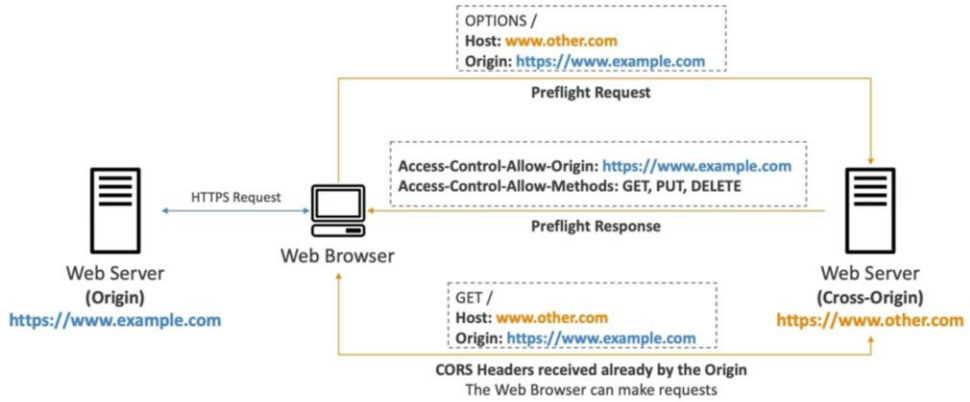
- SSE-S3 encryption is automatically applied to new objects stored in S3 bucket
- Optionally, you can "force encryption" using a bucket policy and refuse any API call to PUT an S3 object without encryption headers (SSE-KMS or SSE-C)

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

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

CORS (Cross Origin Resource Sharing)

- Cross-Origin Resource Sharing (CORS)
- Origin = scheme (protocol) + host (domain) + port
  - example: https://www.example.com (implied port is 443 for HTTPS, 80 for HTTP)
- Web Browser based mechanism to allow requests to other origins while visiting the main origin
- Same origin: http://example.com/app1 & http://example.com/app2
- Different origins: http://www.example.com & http://other.example.com
- The requests won't be fulfilled unless the other origin allows for the requests, using CORS Headers (example: Access-Control-Allow-Origin)



CORS use in S3

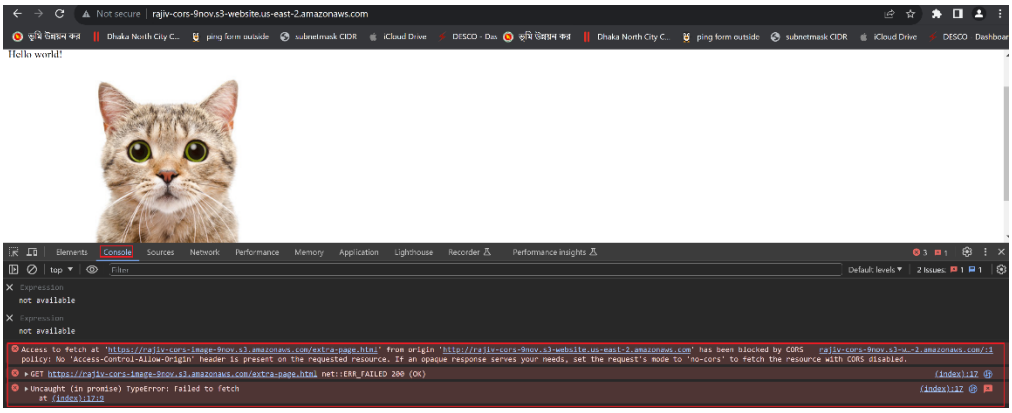
- If a client makes a cross-origin request on our S3 bucket, we need to enable the correct CORS headers
- It's a popular exam question
- You can allow for a specific origin or for * (all origins)

Lab: CORS

step: 1
first create a bucket with public enable then enable static web hosting and also add the Json script to access the bucket publicly
upload 3 filles index.html,extra-page.html,cat.jpg
now brows the static web page- extra-page bottom portion is coming

Step:2
Now, create another bucket with public enable then enable static web hosting and json script for access publicly
upload the extra-page.html and check the URL that this page is showing the page

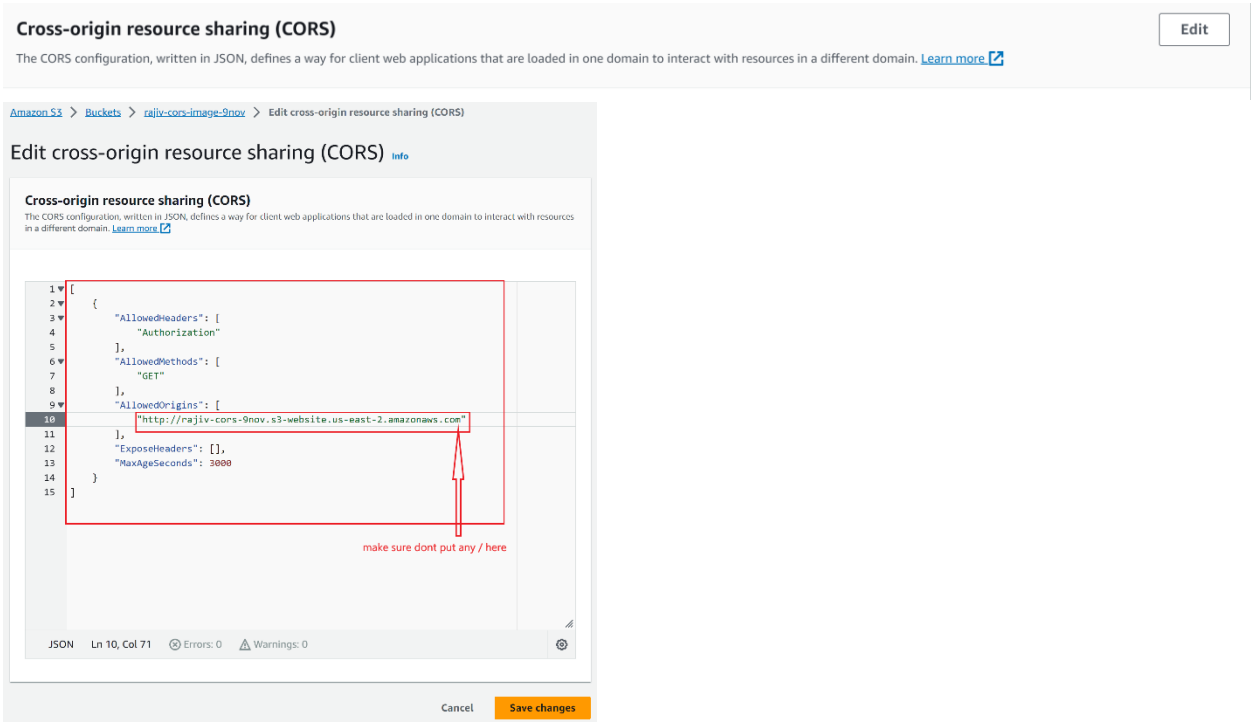
Step:3
now go to the previous bucket and delete the extra-page.html
now brose the bucket one URL and we can see the bottom extra page option is not coming
edit the index page add the extra page URL (https://rajiv-cors-image-9nov.s3.amazonaws.com/extra-page.html) to the index file which we get from image bucket. Now upload the index file
now brows the first bucket webpage with developer toll and we can see there is error come



Step:4

now in the 2nd bucket add the cross Json script and save

select the image bucket >permission >scroll down > click the edit and then pest the script and save changes as shown below



now browse the page and we can see page is showing properly and don't get any error in the console.