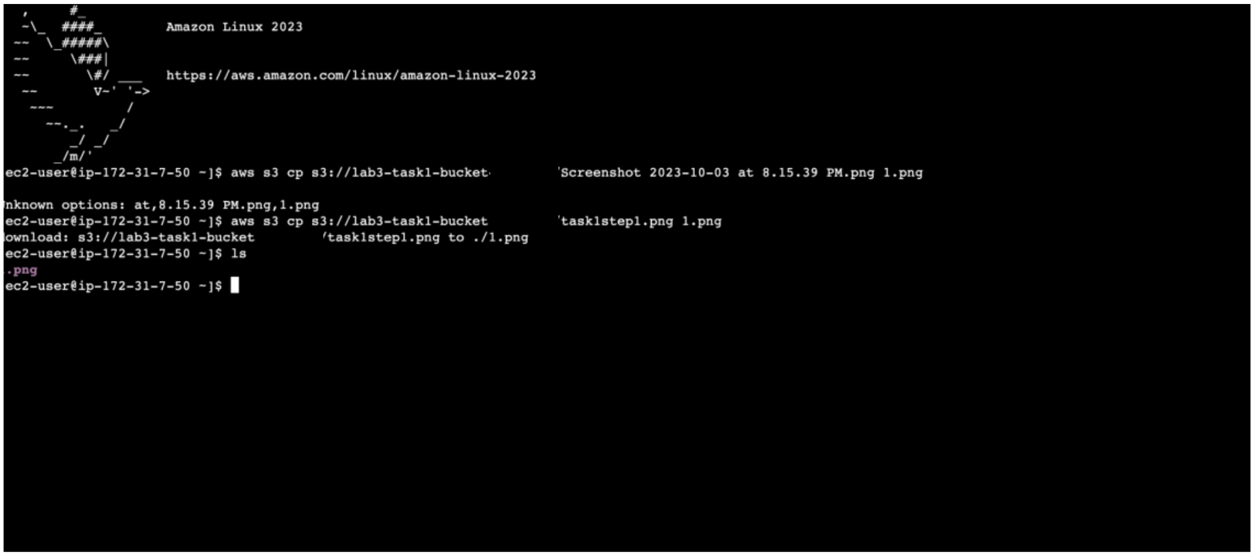
Assignment 3 – S3

- [task 1] Screen shot of the file name after downloading it from the S3 bucket in EC2. Something like below



```
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
ec2-user@ip-172-31-7-50 ~]$ aws s3 cp s3://lab3-task1-bucket        'Screenshot 2023-10-03 at 8.15.39 PM.png 1.png

nknown options: at,8.15.39 PM.png,1.png
ec2-user@ip-172-31-7-50 ~]$ aws s3 cp s3://lab3-task1-bucket        'task1step1.png 1.png
ownload: s3://lab3-task1-bucket        'task1step1.png to ./1.png
ec2-user@ip-172-31-7-50 ~]$ ls
.png
ec2-user@ip-172-31-7-50 ~]$ 
```

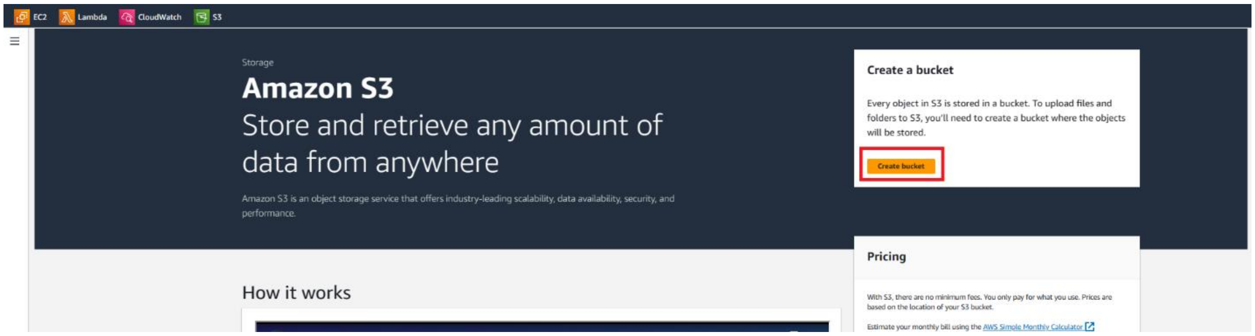
- [task 2] SNS topic.
- [task 2] Event notification in S3 properties.
- [task 2] The email from S3 Event notification.
- [task 3] The signed URL.
- [task 3] A Lambda code that generates a signed URL.

Always follow the least privilege principle for the IAM policies and security groups.

Task 1 – Download/Upload a file from S3 in EC2

a. Create an **S3 bucket** and put a file in it.

   i. Click on **Create bucket** with any globally unique name**.**



Storage

Amazon S3

Store and retrieve any amount of data from anywhere

Amazon S3 is an object storage service that offers industry-leading scalability, data availability, security, and performance.

Create a bucket

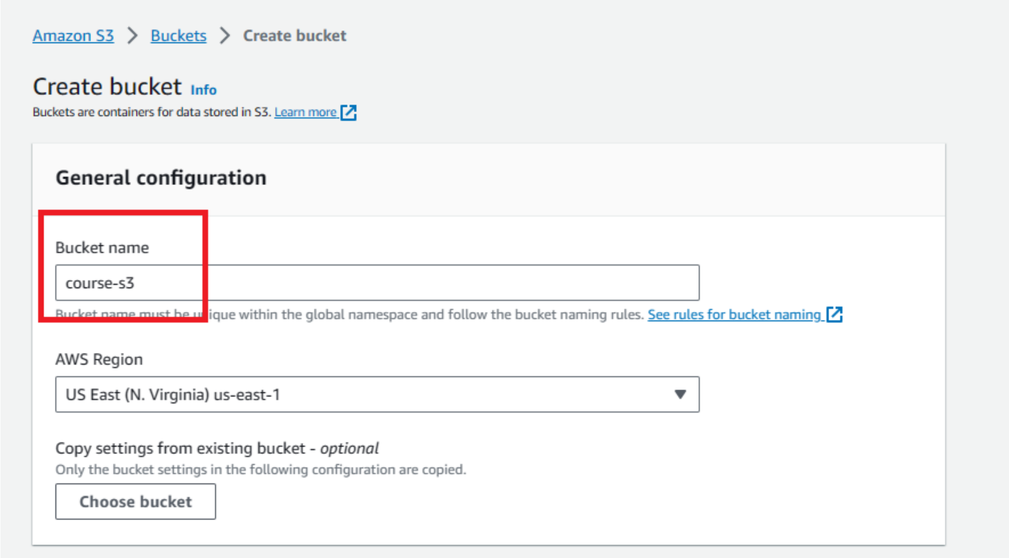
Every object in S3 is stored in a bucket. To upload files and folders to S3, you'll need to create a bucket where the objects will be stored.

Pricing

With S3, there are no minimum fees. You only pay for what you use. Prices are based on the location of your S3 bucket.

Estimate your monthly bill using the AWS Simple Monthly Calculator

ii. Name your bucket and click **Create Bucket**



Amazon S3 > Buckets > Create bucket

Create bucket Info

Buckets are containers for data stored in S3. Learn more

General configuration

Bucket name

course-s3

Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming

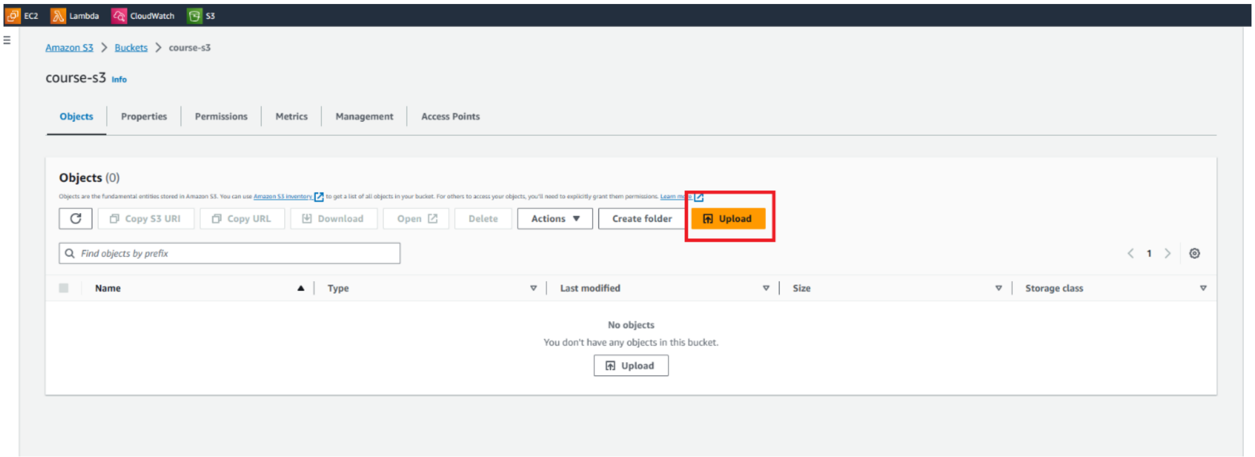
AWS Region

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

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

iii. Go to your bucket and **upload** any file.



Amazon S3 > Buckets > course-s3

course-s3 Info

Objects (0)

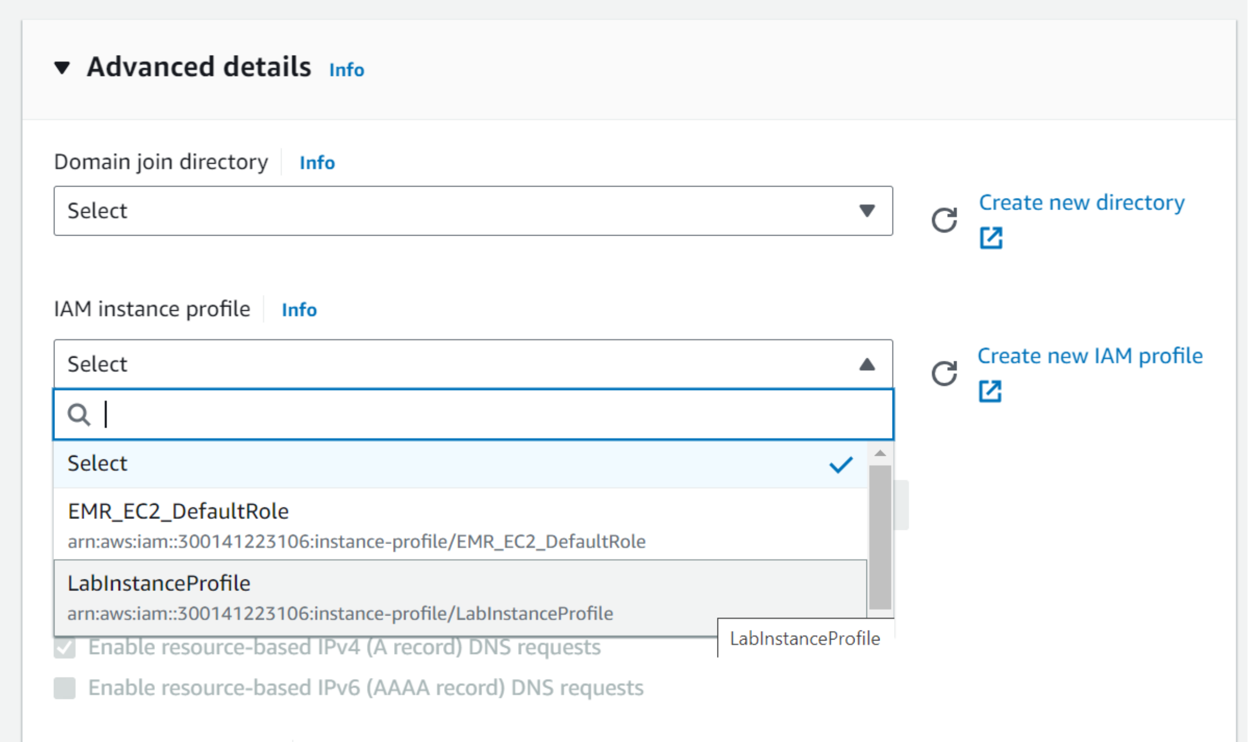
Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

Name	Type	Last modified	Size	Storage class

No objects

You don't have any objects in this bucket.

b. Go to EC2 console. Hit Launch instances. Expand the “Advanced details” and select the **LabInstanceProfile** (it is the LabRole). And create the instance. SSH into it like you did in assignment 1.



▼ Advanced details Info

Domain join directory Info

Select ▼

IAM instance profile Info

Select ▲

Select ✓

EMR_EC2_DefaultRole
arn:aws:iam::300141223106:instance-profile/EMR_EC2_DefaultRole

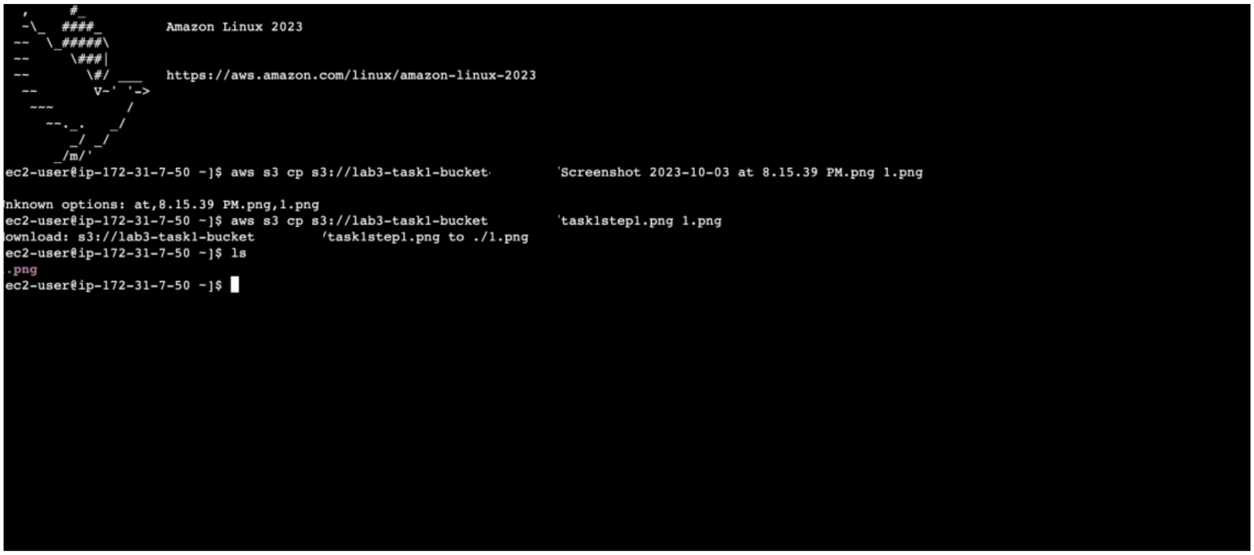
LabInstanceProfile
arn:aws:iam::300141223106:instance-profile/LabInstanceProfile

☑ Enable resource-based IPv4 (A record) DNS requests

□ Enable resource-based IPv6 (AAAA record) DNS requests

c. Download and update the file with the following CLI command.

aws s3 cp s3://<bucket_name>/<file_name_in_s3> <new_file_name_in_s3>



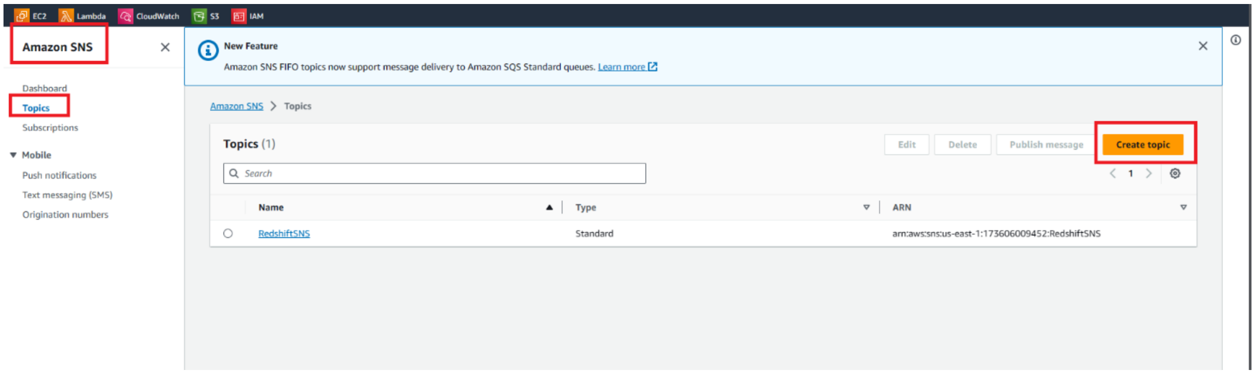
```
 ,     #_
 ~\_  ####_        Amazon Linux 2023
~~  \_#####\
~~     \###|
~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
 ~~       V~' '->
  ~~~         /
    ~~._.   _/
       _/ _/
      _/m/'
ec2-user@ip-172-31-7-50 ~]$ aws s3 cp s3://lab3-task1-bucket         'Screenshot 2023-10-03 at 8.15.39 PM.png 1.png

nknown options: at,8.15.39 PM.png,1.png
ec2-user@ip-172-31-7-50 ~]$ aws s3 cp s3://lab3-task1-bucket         'task1step1.png 1.png
ownload: s3://lab3-task1-bucket         'task1step1.png to ./1.png
ec2-user@ip-172-31-7-50 ~]$ ls
.png
ec2-user@ip-172-31-7-50 ~]$ █
```

Task 2 – S3 Event Notification

Send an email to yourself when the object is created in the bucket.

1. You need to create an SNS topic.
   a. Go to **SNS → Topics → Create topic**.



New Feature

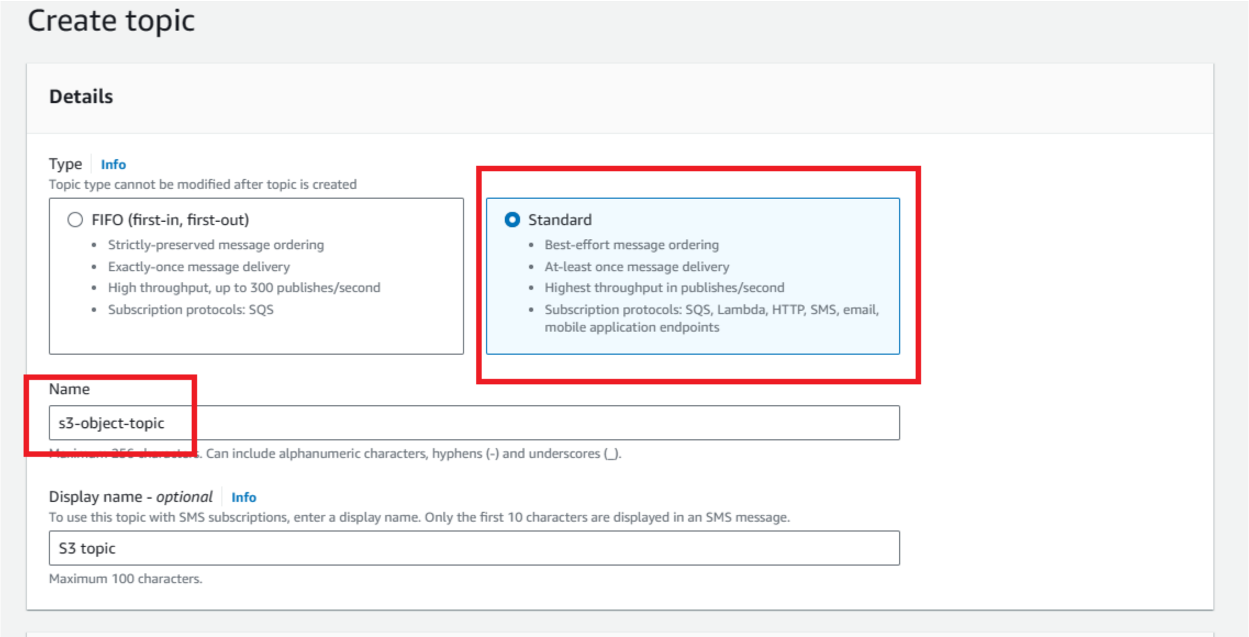
Amazon SNS FIFO topics now support message delivery to Amazon SQS Standard queues. Learn more

Amazon SNS > Topics

Topics (1)

	Name	Type	ARN
○	RedshiftSNS	Standard	arn:aws:sns:us-east-1:173606009452:RedshiftSNS

   b. Name your topic → **Create Topic.**
   c. Create a topic by providing a name and selecting **Standard**.


Create topic

Details

Type Info

Topic type cannot be modified after topic is created

○ FIFO (first-in, first-out)
- Strictly-preserved message ordering
- Exactly-once message delivery
- High throughput, up to 300 publishes/second
- Subscription protocols: SQS

● Standard
- Best-effort message ordering
- At-least once message delivery
- Highest throughput in publishes/second
- Subscription protocols: SQS, Lambda, HTTP, SMS, email, mobile application endpoints

Name

s3-object-topic

Maximum 256 characters. Can include alphanumeric characters, hyphens (-) and underscores (_).

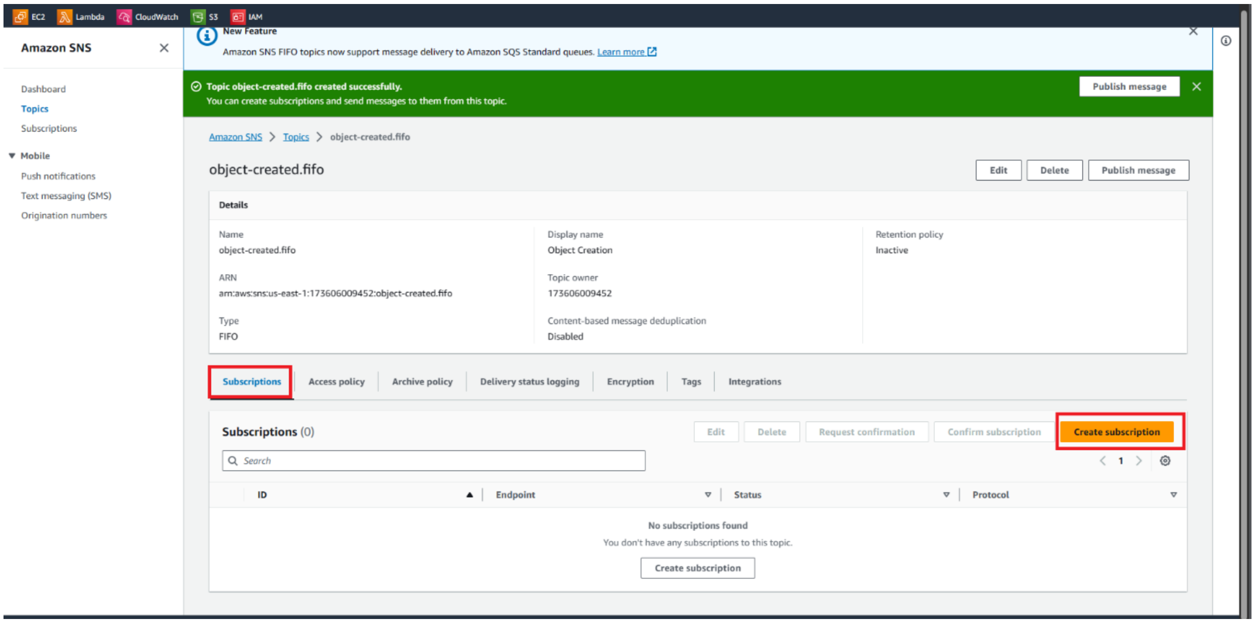
Display name - *optional* Info

To use this topic with SMS subscriptions, enter a display name. Only the first 10 characters are displayed in an SMS message.

S3 topic

Maximum 100 characters.

e. **Subscribe** to it using your email.



New Feature

Amazon SNS FIFO topics now support message delivery to Amazon SQS Standard queues. Learn more

Topic object-created.fifo created successfully.
You can create subscriptions and send messages to them from this topic.

Amazon SNS > Topics > object-created.fifo

object-created.fifo

Details

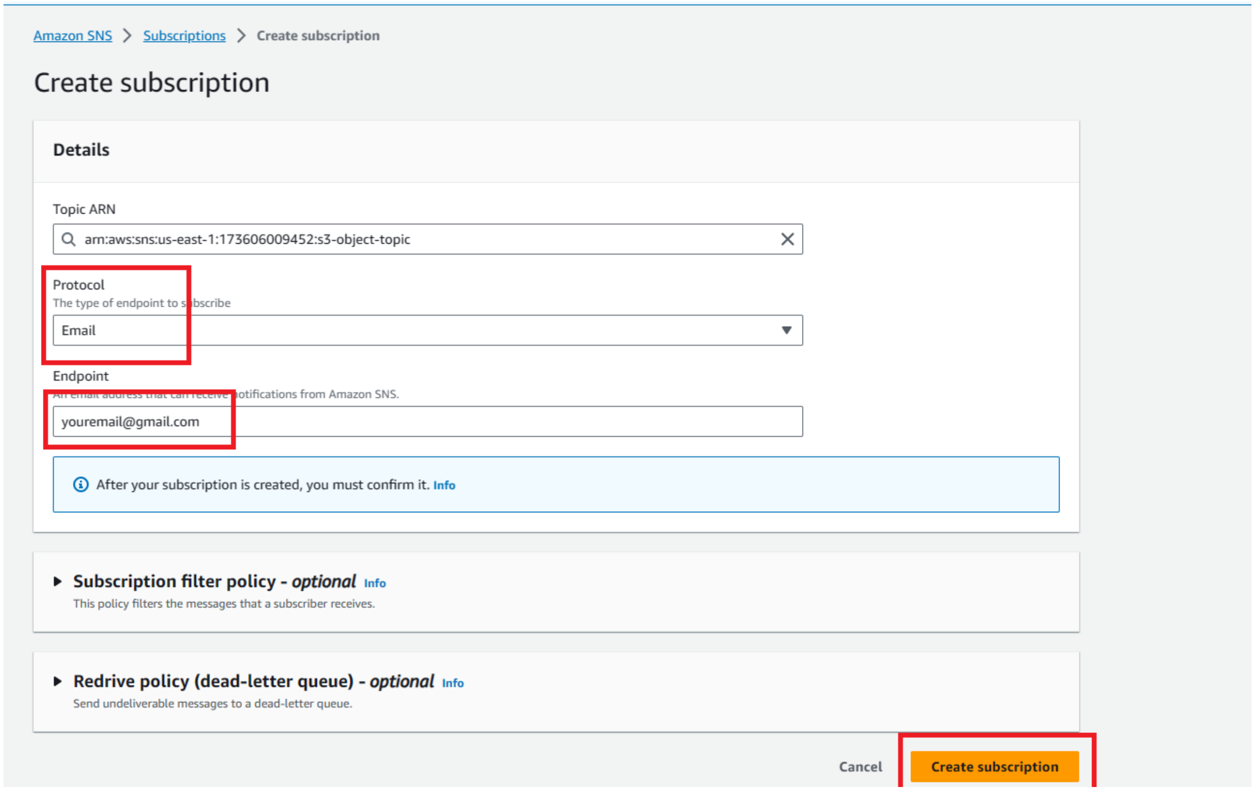
Name object-created.fifo	Display name Object Creation	Retention policy Inactive
ARN arn:aws:sns:us-east-1:173606009452:object-created.fifo	Topic owner 173606009452	
Type FIFO	Content-based message deduplication Disabled	

Subscriptions (0)

ID	Endpoint	Status	Protocol

No subscriptions found
You don't have any subscriptions to this topic.

f. Select email and provide your email. Click **Create subscription**.


Subscription filter policy - *optional* Info
Redrive policy (dead-letter queue) - *optional* Info
Cancel
Create subscription

Amazon SNS > Subscriptions > Create subscription

Create subscription

Details

Topic ARN
arn:aws:sns:us-east-1:173606009452:s3-object-topic

Protocol
The type of endpoint to subscribe
Email

Endpoint
An email address that can receive notifications from Amazon SNS.
youremail@gmail.com

After your subscription is created, you must confirm it. Info

This policy filters the messages that a subscriber receives.

Send undeliverable messages to a dead-letter queue.

3. You must write a resource-based policy in the **SNS topic** after creation. Remember, if you are connecting 2 different AWS services, you must write an IAM policy. That allows S3 to publish messages on the topic. **[Include it in the PDF].**

```
{
    "Version": "2008-10-17",
    "Statement": [{
      "Effect": "Allow",
      "Principal": {
        "Service": "s3.amazonaws.com"
      },
      "Action": "SNS:Publish",
      "Resource": "",
      "Condition": {
        "StringEquals": {
          "AWS:SourceAccount": ""
         },
         "ArnLike": {
          "aws:SourceArn": "arn:aws:s3:*:*:"
        }
      }
    }]
  }
```

4. Create an S3 bucket or use an existing one. Go to the Properties tab. In the “Event notifications” section, click on the “Create event notifications”.
   a. Give event name
   b. Select All object create events
   c. In Destination, select SNS topic then choose the SNS you created earlier. If the policy in the SNS is correct, you should be able to create. If you get an error, fix that and it will work!

Task 3 – S3 Signed URL

Write a lambda that returns a Signed URL of the object in S3. Make sure the LabRole has an inline policy that allows getting objects from the bucket. Create a “Test Event” to trigger the lambda. For more: https://docs.aws.amazon.com/AWSJavaScriptSDK/v3/latest/index.html Add the following to your lambda and run your test (replace the bucket and key). Key is your file name in S3.

```
import { S3Client } from '@aws-sdk/client-s3';
import { GetObjectCommand } from '@aws-sdk/client-s3';
import { getSignedUrl } from '@aws-sdk/s3-request-presigner';
const s3 = new S3Client({ region: 'us-east-1' });

export const handler = async (event) => {
    const params = { Bucket: '<bucket_name>', Key: '<object_key>' };
    const command = new GetObjectCommand(params);
    const url = await getSignedUrl(s3, command, { expiresIn: 3600 });
    return url;
  }
```