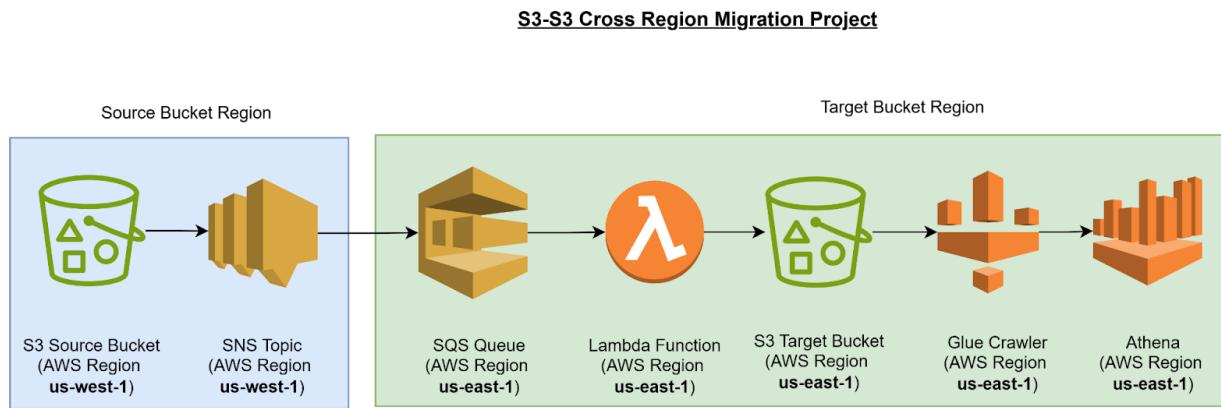


Architecture Diagram



Project Overview

1. In this project, we are using AWS S3, SNS, SQS, and Lambda to transfer a data file from an S3 bucket in the source region to an S3 bucket in the target region (your default region).
2. Once the data is loaded into the target region's S3 bucket, we will use SQS to send a notification to trigger Glue crawler to generate a table in the Glue catalog.
3. After the table is created in the Glue catalog, we will leverage AWS Athena to query the data and analyze its contents.

After completing this project in AWS, someone will learn how to:

1. Migrate data between S3 buckets in different regions using services like SNS, SQS, and Lambda.
2. Automate workflows with AWS Lambda, responding to events and triggering actions like data transfer.
3. Catalog data in AWS Glue by using Glue crawlers to create tables in the Glue Data Catalog.
4. Query data stored in S3 using AWS Athena with the Glue catalog for schema management.
5. Understand and work with cross-region data management and serverless architectures in AWS.

Overall, they will gain experience in building serverless, automated, and scalable data processing workflows in the cloud.

Project Steps

1. Create S3 Buckets

Set up two S3 buckets: one in the source region to hold the data file and one in the target region (your default region) for storing the transferred data.

2. Set Up SNS (Simple Notification Service)

Create an SNS topic that will be used to notify when a new file is uploaded to the source S3 bucket, which will trigger downstream actions.

3. Create SQS Queue

Set up an SQS queue that will receive notifications from SNS and trigger the Lambda function for processing the data file.

4. Develop Lambda Function

Create a Lambda function that listens to the SQS queue, moves the data file from the source S3 bucket to the target S3 bucket, and sends a notification to trigger the Glue crawler after the transfer.

5. Set Up Glue Crawler

Configure and run the Glue crawler to scan the target S3 bucket, detect the schema, and create a table in the Glue Data Catalog.

6. Configure S3 Event Notification

Configure an S3 event in the source bucket to publish notifications to the SQS topic whenever a new file is uploaded..

7. Set Up Athena for Querying

Configure Athena to use the Glue Data Catalog and run SQL queries to analyze the data stored in the target S3 bucket.

8. Test the Workflow

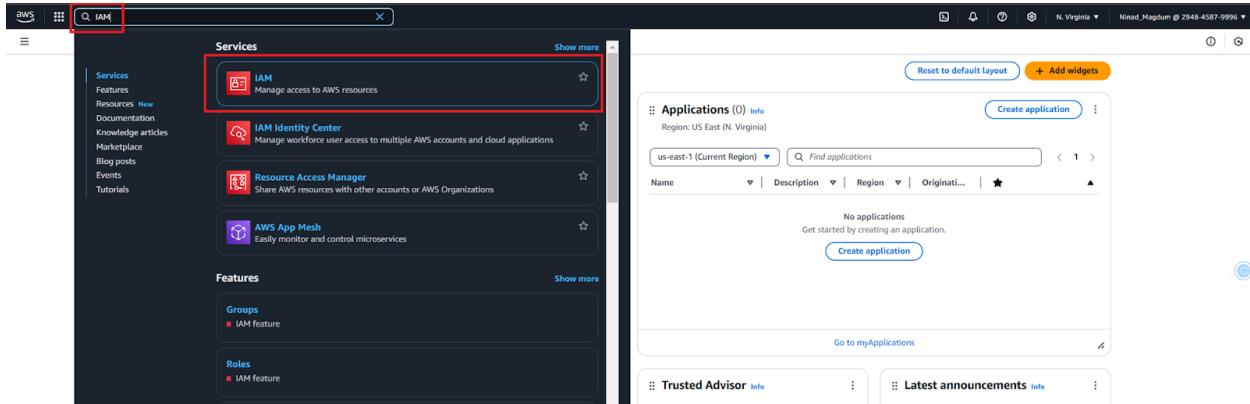
Upload a test file to the source S3 bucket, check if the data is transferred correctly, and ensure that the Glue crawler runs and Athena can query the data.

AWS IAM Role Creation guide

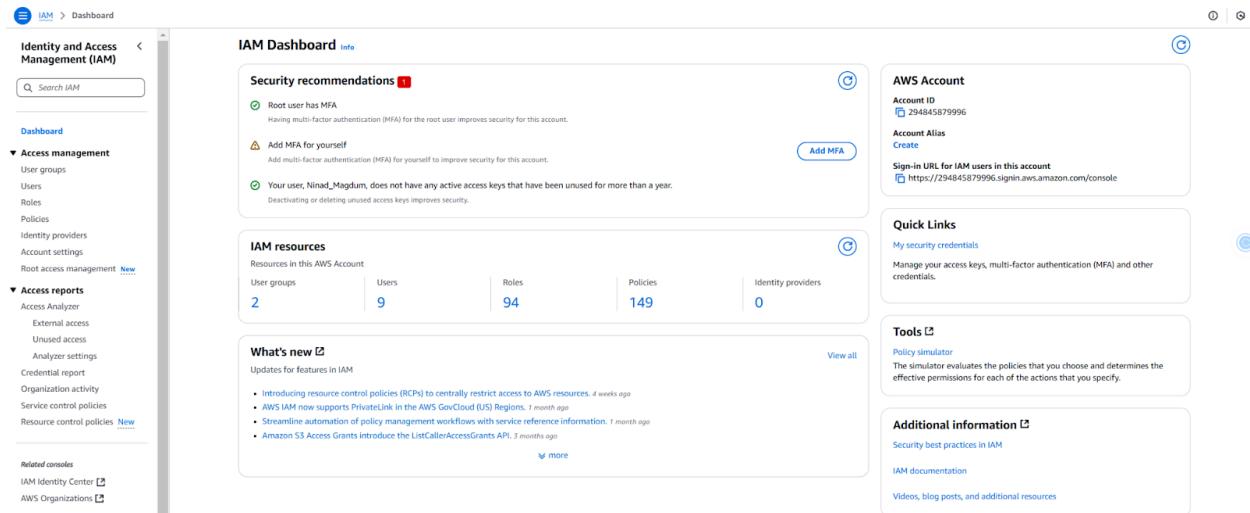
- Let us understand and walk through the process of creation of AWS IAM Role and attach specific Policies

Role Creation

- Open your AWS Management Console and navigate to AWS IAM (you can search for "IAM" in the search bar).



- Navigate to IAM, where you'll land on the page below. Here, you'll create the necessary role and policy that will be utilized in AWS Glue to write data to S3 buckets.



Role Creation

- Navigate to the “Roles” button located on the left-hand side of the page, which will direct us to the following page shown below.

The screenshot shows the AWS IAM Roles list page. On the left, there's a navigation sidebar with 'Identity and Access Management (IAM)' selected. Under 'Access management', 'Roles' is also selected and highlighted with a red box. The main area displays a table of 94 roles, each with a checkbox, role name, trusted entities, and last activity. The 'Create role' button at the top right is also highlighted with a red box.

| Role name | Trusted entities | Last activity |
|---|--|----------------|
| AmazonEC2RoleForLaunchWizard | AWS Service: ec2 | - |
| api-gateway-upload-to-s3 | AWS Service: apigateway | - |
| apigateway_s3_role | AWS Service: apigateway | 159 days ago |
| AWSAthenaSparkExecutionRole-mpzkrmehi2s | AWS Service: athena | - |
| AWSCodePipelineServiceRole-us-east-1-airflowdags-githubcode-pip | AWS Service: codepipeline | 254 days ago |
| AWSGlueServiceRole-default | AWS Service: glue | 36 days ago |
| AWSGLUESERVICEROLEDEFULT | AWS Service: glue | 262 days ago |
| AWSServiceRoleForAmazonMWA | AWS Service: airflow (Service-Linked) | 252 days ago |
| AWSServiceRoleForAPIGateway | AWS Service: ops.apigateway (Service-Linked) | - |
| AWSServiceRoleForApplicationAutoScaling_DynamoDBTable | AWS Service: dynamodb.application | 144 days ago |
| AWSServiceRoleForBatch | AWS Service: batch (Service-Linked) | 225 days ago |
| AWSServiceRoleForDevOpsGuru | AWS Service: devops-guru (Service-Linked) | 42 minutes ago |
| AWSServiceRoleForEC2Spot | AWS Service: spot (Service-Linked) | - |
| AWSServiceRoleForECS | AWS Service: ecs (Service-Linked) | 232 days ago |
| AWSServiceRoleForEMRCleanup | AWS Service: elasticmapreduce (Service-Linked) | 375 days ago |

- Next, click on the "Create role" button.

This screenshot is identical to the one above, showing the AWS IAM Roles list page with 94 roles listed. The 'Create role' button at the top right is again highlighted with a red box.

- On the next page, choose Custom trust policy option as indicated below. Then, paste the following JSON into the Custom trust policy box and click Next.

{

"Version": "2012-10-17",

"Statement": [

{

 "Effect": "Allow",

```

"Principal": {

    "Service": [
        "glue.amazonaws.com",
        "lambda.amazonaws.com"
    ],
    "Action": "sts:AssumeRole"
}

}
]
}

```

Step 1
 Select trusted entity info

Step 2
 Add permissions
 Step 3
 Name, review, and create

Select trusted entity type

AWS service Allow AWS services like EC2, Lambda, or others to perform actions in this account.

AWS account Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

Web identity Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.

Custom trust policy Create a custom trust policy to enable others to perform actions in this account.

Custom trust policy

Create a custom trust policy to enable others to perform actions in this account.

```

1 * {
2     "Version": "2012-10-17",
3     "Statement": [
4         {
5             "Effect": "Allow",
6             "Principal": "*",
7             "Service": [
8                 "glue.amazonaws.com",
9                 "lambda.amazonaws.com"
10            ],
11            "Action": "sts:AssumeRole"
12        }
13    ]
14 }
15
16

```

+ Add new statement

JSON | Line 16, Col 0

Preview external access

Cancel Next Step

- On the following page for ADD permissions, please locate the below listed policies individually and select each one to include them in your role.

- AmazonS3FullAccess

Screenshot of the AWS IAM 'Add permissions' step for creating a new role. The search bar shows 'AmazonS3FullAccess'. The policy 'AmazonS3FullAccess' is selected and highlighted with a red box.

- AmazonSQSFullAccess

Screenshot of the AWS IAM 'Add permissions' step for creating a new role. The search bar shows 'AmazonSQSFullAccess'. The policy 'AmazonSQSFullAccess' is selected and highlighted with a red box.

- AWSGlueConsoleFullAccess

Screenshot of the AWS IAM 'Add permissions' step for creating a new role. The search bar shows 'AWSGlueConsoleFullAccess'. The policy 'AWSGlueConsoleFullAccess' is selected and highlighted with a red box.

- CloudWatchLogsFullAccess

Screenshot of the AWS IAM 'Add permissions' step for creating a new role. The search bar shows 'CloudWatchLogsFullAccess'. The policy 'CloudWatchLogsFullAccess' is selected and highlighted with a red box.

- After selecting all the policies, click on the "Next" button.

Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create

Add permissions Info

Permissions policies (4/1166) Info

Choose one or more policies to attach to your new role.

Filter by Type
Q CloudWatchLogsFullAccess All types 1 match

Policy name CloudWatchLogsFullAccess

AWS managed

Set permissions boundary - optional

Cancel Previous Next

- On the following page, provide an appropriate name for the role <s3-to-s3-role>, then click the "Create role" button.

Step 1: Select trusted entity
Step 2: Add permissions
Step 3: Name, review, and create

Name, review, and create

Role details

Role name Enter a meaningful name to identify this role. Maximum 2000 characters. Use letters (a-z and A-Z), numbers (0-9), underscores (_), hyphens (-), periods (.), and/or dots (.) characters.

Description Add a short description for this role.

Step 1: Select trusted entities

Trust policy

```
1: {  
2:   "version": "2012-10-17",  
3:   "statement": [  
4:     {  
5:       "principal": "*",  
6:       "action": "sts:AssumeRole",  
7:       "resource": "arn:aws:iam::123456789012:role/s3-to-s3-role",  
8:       "condition": {}  
9:     }  
10:   ]  
11: }  
12:  
13:  
14: }
```

Step 2: Add permissions

Permissions policy summary

| Policy name | Type | Attached as |
|---------------------------|-------------|--------------------|
| AmazonS3FullAccess | AWS managed | Permissions policy |
| AmazonSQSFullAccess | AWS managed | Permissions policy |
| AWSLambdaVPCExecutionRole | AWS managed | Permissions policy |
| CloudWatchLogsFullAccess | AWS managed | Permissions policy |

Step 3: Add tags

Add tags - optional Info

Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources. No tags associated with this resource.

Add tag Info

How to add tags to your AWS resources

Cancel Previous Create role

- The required role is created successfully

Identity and Access Management (IAM)

Roles (95) Info

Role s3-to-s3-role created.

View role Delete Create role

Role name s3-to-s3-role

Trusted entities Last activity

AWS Service: lambda, and 1 more. E

Roles Anywhere Info

Authenticate your non AWS workloads and securely provide access to AWS services.

X.509 Standard Info

Use your own existing PKI infrastructure or use AWS Certificate Manager Private Certificate Authority E to authenticate identities.

Access AWS from your non AWS workloads Info

Operate your non AWS workloads using the same authentication and authorization strategy that you use within AWS.

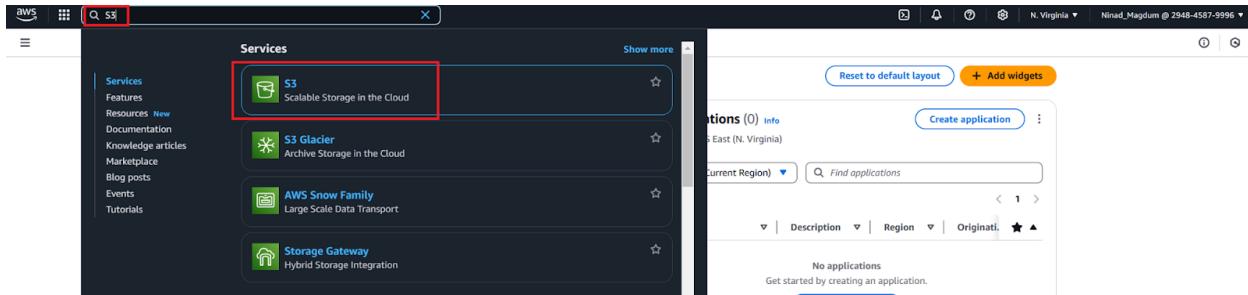
Temporary credentials Info

Use temporary credentials with ease and benefit from the enhanced security they provide.

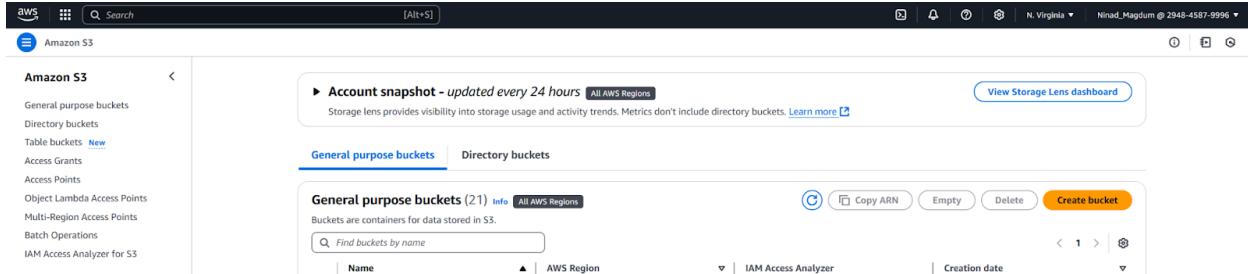
VOILA! YOUR SUCCESSFULLY CREATED YOUR S3 TO S3 IAM ROLE AND ATTACHED SPECIFIC POLICIES

AWS S3 Buckets Creation guide

- Access your AWS Management Console and find AWS S3 by entering "S3" in the search bar.

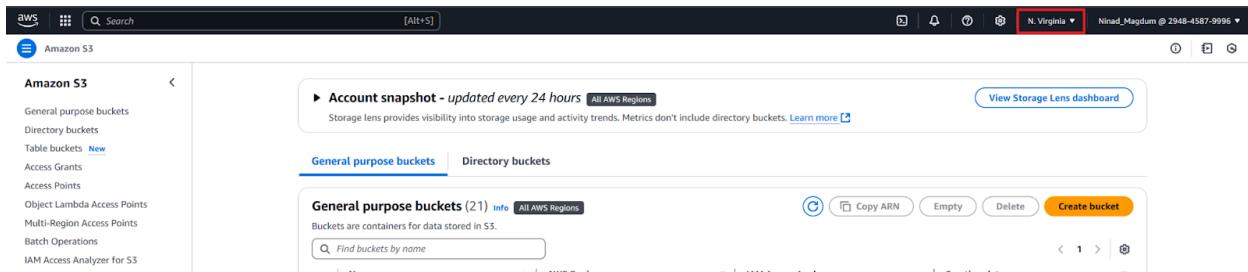


- Select S3 from the menu, which will direct you to the page where you can create an S3 Bucket.



Source Bucket Creation

- Now let's create the source S3 bucket in the region other than your AWS default region. Click on the Region button on the top right hand corner of your AWS Management Console. Whatever is the existing region is your default region



- Select any region other than your already existing AWS default region to create your Source bucket. I am selecting us-west-1 (my default AWS region is us-east-1 in your case it might be different)

The screenshot shows the AWS S3 console. On the right, there is a dropdown menu for selecting a region. The options listed are:

- United States
 - N. Virginia (selected, highlighted with a red box)
 - Ohio
 - N. California (highlighted with a red box)
 - Oregon
- Asia Pacific
 - Mumbai
 - Osaka
 - Seoul
 - Singapore
 - Sydney
 - Tokyo
- Canada
 - Central

- Once selected validate your AWS region changes to the new region

The screenshot shows the AWS S3 console with the 'N. California' region selected in the dropdown menu. The rest of the interface remains the same as the previous screenshot.

- Click on the Create bucket button to create the source bucket

The screenshot shows the AWS S3 console with the 'Create bucket' button highlighted with a red box. The rest of the interface remains the same as the previous screenshots.

- Once you click Create bucket, we will be landing onto the below page where first validate the region and then we need to fill the source bucket name <de-s3-to-s3-source-bucket> and click Create bucket

Screenshot of the AWS S3 Create Bucket wizard.

General configuration

AWS Region: US West (N. California) us-west-1

Bucket name: `de-s3-to-s3-source-bucket`

Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)

Copy settings from existing bucket - optional: Only the bucket settings in the following configuration are copied.

Choose bucket

Format: `s3://bucket/prefix`

Object Ownership

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

ACLs disabled (recommended): All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

ACLs enabled: Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership: Bucket owner enforced

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access: Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

- Block public access to buckets and objects granted through new access control lists (ACLs)**: S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- Block public access to buckets and objects granted through any access control lists (ACLs)**: S3 will ignore all ACLs that grant public access to buckets and objects.
- Block public access to buckets and objects granted through new public bucket or access point policies**: S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- Block public and cross-account access to buckets and objects through any public bucket or access point policies**: S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

Tags - optional (0)

You can use bucket tags to track storage costs and organize buckets. [Learn more](#)

No tags associated with this bucket.

Add tag

Default encryption

Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type: [Info](#)

Server-side encryption with Amazon S3 managed keys (SSE-S3)

Server-side encryption with AWS Key Management Service keys (SSE-KMS)

Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)

Secure your objects with two separate layers of encryption. For details on pricing, see [DSSE-KMS pricing](#) on the Storage tab of the [Amazon S3 pricing page](#).

Bucket Key: Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. [Learn more](#)

Disable

Enable

Advanced settings

Note: After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

Create bucket

- After clicking Create bucket, you will be directed to the following page confirming the successful creation of your S3 source bucket.

The screenshot shows the AWS S3 Buckets page. At the top, there's a search bar and a 'Create bucket' button. Below it, a table lists buckets under the 'General purpose buckets' tab. One row is highlighted with a red box around the bucket name 'de-s3-to-s3-source-bucket'. The table includes columns for Name, AWS Region, IAM Access Analyzer, and Creation date. The bucket was created in the US West (N. California) region on December 9, 2024, at 16:43:29 (UTC+05:30).

Target Bucket Creation

- Now let's create the target S3 bucket in your AWS default region. Click on the Region button on the top right hand corner of your AWS Management Console.

The screenshot shows the AWS S3 Buckets page with the 'Region' dropdown menu open. The menu lists regions under 'United States' (N. Virginia, Ohio, N. California, Oregon) and 'Asia Pacific' (Mumbai, Osaka, Seoul, Singapore). The 'N. Virginia' option is highlighted with a red box.

- Select your default AWS region. Mine is us-east-1, it will be different for you.

The screenshot shows the AWS S3 Buckets page with the 'Region' dropdown menu closed. The 'us-east-1' region is highlighted with a red box in the dropdown menu. The main interface shows the 'General purpose buckets' table.

- Once selected validate your AWS region changes to the default AWS region

The screenshot shows the AWS S3 Buckets page. On the left, there's a sidebar with links like 'General purpose buckets', 'Table buckets New', 'Access Grants', etc. The main area has a heading 'Account snapshot - updated every 24 hours' and a 'General purpose buckets' section. It displays 21 buckets, with a search bar and buttons for 'Copy ARN', 'Empty', 'Delete', and 'Create bucket'. The 'Create bucket' button is highlighted with a red box.

- Click on the Create bucket button to create the target bucket

This screenshot is similar to the previous one, but the 'Create bucket' button is explicitly highlighted with a red box, indicating it is the next step to take.

- Once you click Create bucket, we will be landing onto the below page where first validate the region and then we need to fill the target bucket name <de-s3-to-s3-target-bucket> and click Create bucket

The screenshot shows the 'Create bucket' configuration page. It has sections for 'General configuration' (AWS Region set to 'US East (N. Virginia) us-east-1', Bucket type set to 'General purpose'), 'Bucket name' (set to 'de-s3-to-s3-target-bucket'), 'Copy settings from existing bucket - optional' (with a 'Choose bucket' button), and 'Object Ownership' (set to 'ACLS disabled (recommended)'). The 'Bucket name' field is highlighted with a red box.

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

- Block public access to buckets and objects granted through new access control lists (ACLs)** This setting applies only to new buckets and objects. It does not change existing permissions that allow public access to S3 resources using ACLs.
- Block public access to buckets and objects granted through any access control lists (ACLs)** S3 will ignore all ACLs that grant public access to buckets and objects.
- Block public access to buckets and objects granted through new public bucket or access point policies** This setting applies only to new buckets and objects. It does not change any existing policies that allow public access to S3 resources.
- Block public and cross-account access to buckets and objects through any public bucket or access point policies** S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

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](#)

Bucket Versioning

Disable

Enable

Tags - optional (0)

You can use bucket tags to track storage costs and organize buckets. [Learn more](#)

No tags associated with this bucket.

[Add tag](#)

Default encryption [Info](#)

Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type [Info](#)

Server-side encryption with Amazon S3 managed keys (SSE-S3)

Server-side encryption with AWS Key Management Service keys (SSE-KMS)

Dual-layer server-side encryption with AWS Key Management Service keys (D SSE-KMS)

Secure your objects with two separate layers of encryption. For details on pricing, see [D SSE-KMS pricing](#) on the Storage tab of the [Amazon S3 pricing page](#).

Bucket Key

Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for D SSE-KMS. [Learn more](#)

Disable

Enable

Advanced settings

After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

[Cancel](#) [Create bucket](#)

- After clicking Create bucket, you will be directed to the following page confirming the successful creation of your s3 target bucket.

Account snapshot - updated every 24 hours [All AWS Regions](#) [View Storage Lens dashboard](#)

Storage lens provides visibility into storage usage and activity trends. Metrics don't include directory buckets. [Learn more](#)

General purpose buckets [Directory buckets](#)

General purpose buckets (21) [Info](#) All AWS Regions

Buckets are containers for data stored in S3.

| Name | AWS Region | IAM Access Analyzer | Creation date |
|---------------------------|---------------------------------|---|--|
| de-s3-to-s3-target-bucket | US East (N. Virginia) us-east-1 | View analyzer for us-east-1 | December 9, 2024, 16:44:43 (UTC+05:30) |

[Copy ARN](#) [Empty](#) [Delete](#) [Create bucket](#)

VOILA! YOU HAVE SUCCESSFULLY CREATED YOUR BOTH SOURCE AND TARGET S3 BUCKETS

AWS SNS Topic Creation guide

- Let us understand and walk through the process of creation of AWS SNS (Simple Notification Service) Topic

Identify the SNS Topic AWS Region

First we need to identify the AWS region in which we need to create the required SNS (Simple Notification Service) Topic

- Access your AWS Management Console and find AWS S3 by entering "S3" in the search bar.

The screenshot shows the AWS Management Console search results for 'S3'. The search bar at the top contains 'S3'. The left sidebar has a 'Services' section with various categories like Features, Resources, and Blogs. The main search results show 'S3' as the top result under 'Services', with a sub-section 'Scalable Storage in the Cloud'. Below it are other services like S3 Glacier, AWS Snow Family, and AWS Transfer Family. The right side of the screen shows the 'Welcome to AWS' dashboard with sections for AWS Health, Getting started with AWS, Training and certification, and What's new with AWS?.

- Select S3 from the menu, which will direct you to the page where you can see all of your buckets.

The screenshot shows the AWS S3 service page. The left sidebar lists options like General purpose buckets, Directory buckets, Table buckets, Access Grants, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, and IAM Access Analyzer for S3. The main area features an 'Account snapshot - updated every 24 hours' card with metrics for storage usage and activity trends. Below it is a table for 'General purpose buckets' with columns for Name, AWS Region, IAM Access Analyzer, and Creation date. A 'Create bucket' button is located at the top right of the table.

- Search your source bucket created for this project in the search bar and note down the AWS region of the same.

The screenshot shows the AWS S3 console with the search bar containing 'de-s3-to-s3-source-bucket'. The search results show one match: 'de-s3-to-s3-source-bucket' located in 'US West (N. California) us-west-1'. The 'us-west-1' region is highlighted with a red box.

In my case the region is us-west-1, in your case it might be different

- Click on the Region button on the top right hand corner of your AWS Management Console. Change the AWS region to the source bucket region.

The screenshot shows the AWS S3 console with the region dropdown menu open. The 'N. California' region is selected and highlighted with a red box.

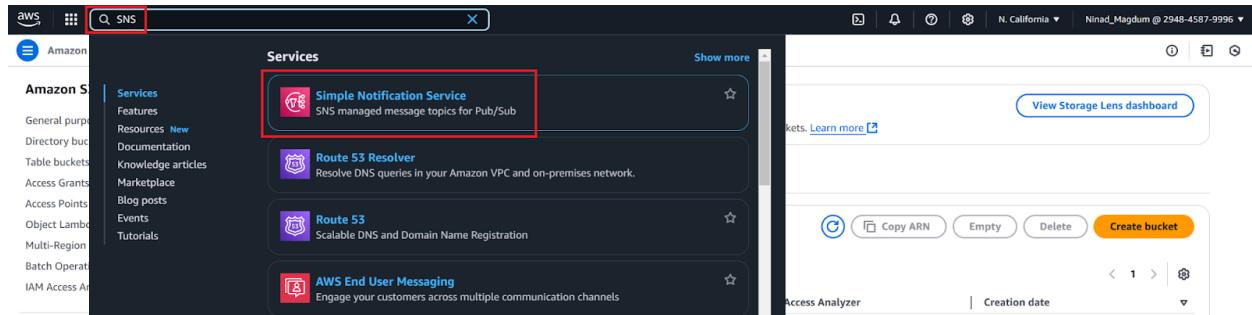
- Once selected validate your AWS region changed to the source bucket region.

The screenshot shows the AWS S3 console with the search bar containing 'de-s3-to-s3-source-bucket'. The search results show one match: 'de-s3-to-s3-source-bucket' located in 'US West (N. California) us-west-1'. The 'us-west-1' region is highlighted with a red box.

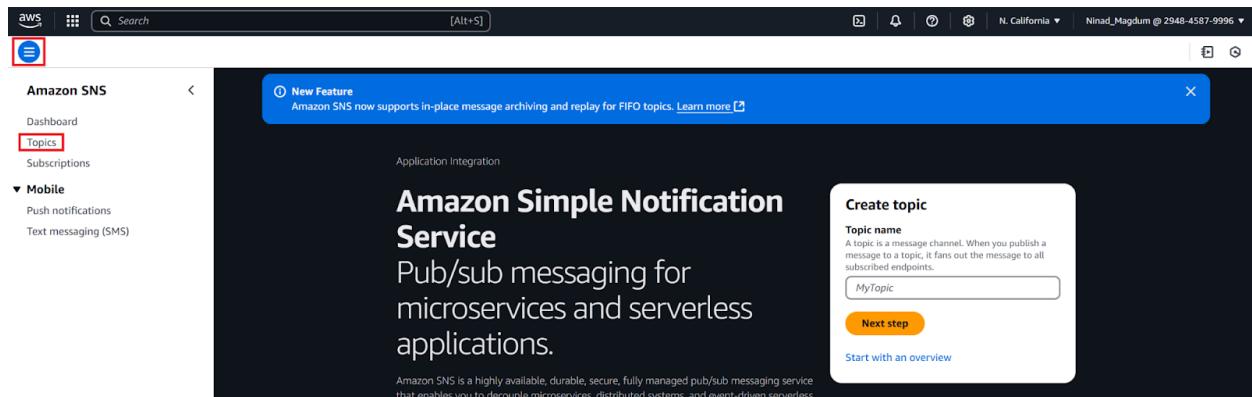
This is the region where we are going to create the SNS topic required for this project

Creation of SNS Topic

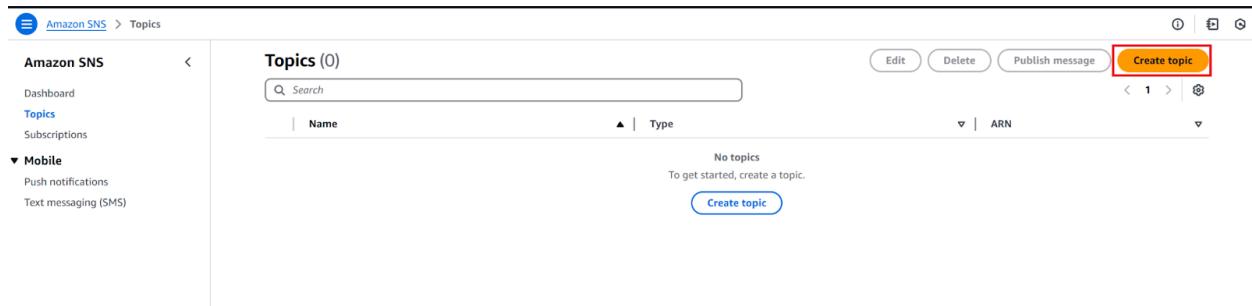
- In your AWS Management Console, search for SNS by entering "SNS" in the search bar and select Simple Notification Service



- On the SNS page , click the three line menu button on the left top corner and then click on the Topics button



- Once in the Topics page, click on the Create topic button to create a new SNS topic



- Under the Create topic page, select the Type as Standard and provide the name of the topic <s3-to-s3-sns-topic>

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
- Subscription protocols: SQS, Lambda, Data Firehose, HTTP, SMS, email, mobile application endpoints

Standard

- Best-effort message ordering
- At-least once message delivery
- Subscription protocols: SQS, Lambda, Data Firehose, HTTP, SMS, email, mobile application endpoints

Name
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.

Maximum 100 characters.

- On the same page click on the Access policy section and choose the method as Basic.

▼ Access policy - optional [Info](#)
This policy defines who can access your topic. By default, only the topic owner can publish or subscribe to the topic.

Choose method
 Basic
Use simple criteria to define a basic access policy.
 Advanced
Use a JSON object to define an advanced access policy.

Publishers
Specify who can publish messages to the topic.

Only the topic owner
Only the owner of the topic can publish to the topic.

Subscribers
Specify who can subscribe to this topic.

Only the topic owner
Only the owner of the topic can subscribe to the topic.

JSON preview

```
{
    "Version": "2008-10-17",
    "Id": "__default_policy_ID",
    "Statement": [
        {
            "Sid": "__default_statement_ID",
            "Effect": "Allow",
            "Principal": {
                "AWS": "*"
            },
            "Action": [
                "SNS:Publish",
                "SNS:RemovePermission",
                "SNS:SetTopicAttributes",
                "SNS:DeleteTopic"
            ]
        }
    ]
}
```

- Select Everyone in the both Publishers and Subscribers section

▼ Access policy - optional [Info](#)
This policy defines who can access your topic. By default, only the topic owner can publish or subscribe to the topic.

Choose method
 Basic
Use simple criteria to define a basic access policy.
 Advanced
Use a JSON object to define an advanced access policy.

Publishers
Specify who can publish messages to the topic.

Everyone
Anybody can publish.

Subscribers
Specify who can subscribe to this topic.

Everyone
Any AWS account can subscribe to the topic.

JSON preview

```
{
    "Version": "2008-10-17",
    "Id": "__default_policy_ID",
    "Statement": [
        {
            "Sid": "__default_statement_ID",
            "Effect": "Allow",
            "Principal": {
                "AWS": "*"
            },
            "Action": [
                "SNS:Publish",
                "SNS:RemovePermission",
                "SNS:SetTopicAttributes",
                "SNS:DeleteTopic"
            ]
        }
    ]
}
```

- Scroll down and click on the Create topic button to create your SNS topic.

Amazon SNS > Topics > Create topic

- Encryption - optional**
Amazon SNS provides in-transit encryption by default. Enabling server-side encryption adds at-rest encryption to your topic.
- Access policy - optional**
This policy defines who can access your topic. By default, only the topic owner can publish or subscribe to the topic.
- Data protection policy - optional**
This policy defines which sensitive data to monitor and to prevent from being exchanged via your topic.
- Delivery policy (HTTP/S) - optional**
The policy defines how Amazon SNS retries failed deliveries to HTTP/S endpoints. To modify the default settings, expand this section.
- Delivery status logging - optional**
These settings configure the logging of message delivery status to CloudWatch Logs.
- Tags - optional**
A tag is a metadata label that you can assign to an Amazon SNS topic. Each tag consists of a key and an optional value. You can use tags to search and filter your topics and track your costs.
[Learn more](#)
- Active tracing - optional**
Use AWS X-Ray active tracing for this topic to view its traces and service map in Amazon CloudWatch. Additional costs apply.

Cancel **Create topic**

- Once clicked your SNS topic will be created successfully.

Amazon SNS > Topics > s3-to-s3-sns-topic

s3-to-s3-sns-topic

Details

| | | | |
|------|---|--------------|--------------|
| Name | s3-to-s3-sns-topic | Display name | - |
| ARN | arn:aws:sns:us-west-1:294845879996:s3-to-s3-sns-topic | Topic owner | 294845879996 |
| Type | Standard | | |

Subscriptions | Access policy | Data protection policy | Delivery policy (HTTP/S) | Delivery status logging | Encryption | Tags | Integrations

Subscriptions (0)

| ID | Endpoint | Status | Protocol |
|----|----------|--------|----------|
|----|----------|--------|----------|

Create subscription

VOILA! YOU HAVE SUCCESSFULLY CREATED YOUR SNS TOPIC IN YOUR SOURCE S3 BUCKET REGION

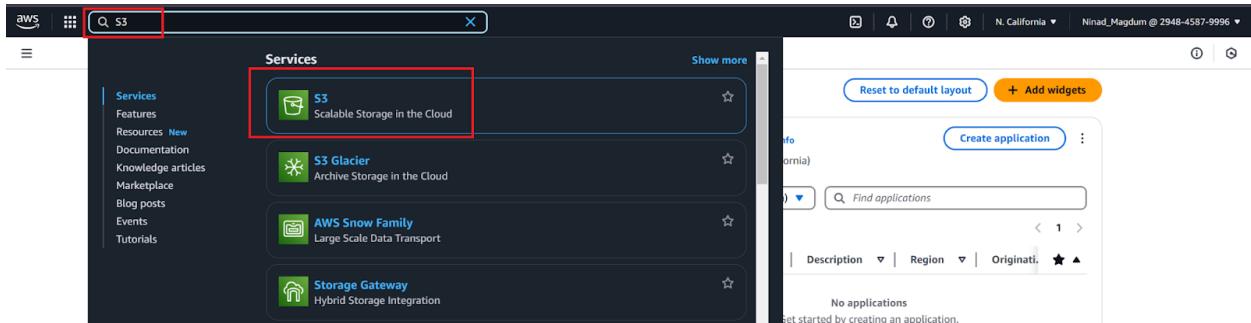
AWS SQS Queue Creation guide

- Let us understand and walk through the process of creation of AWS SQS (Simple Queue Service) Queue required for the lambda function

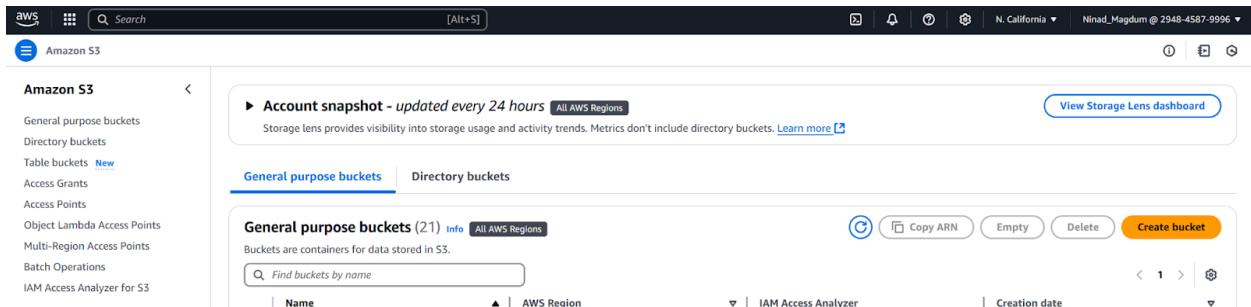
Identify the SQS Queue AWS Region

First we need to identify the AWS region in which we need to create the required AWS SQS (Simple Queue Service) Queue required for the lambda function

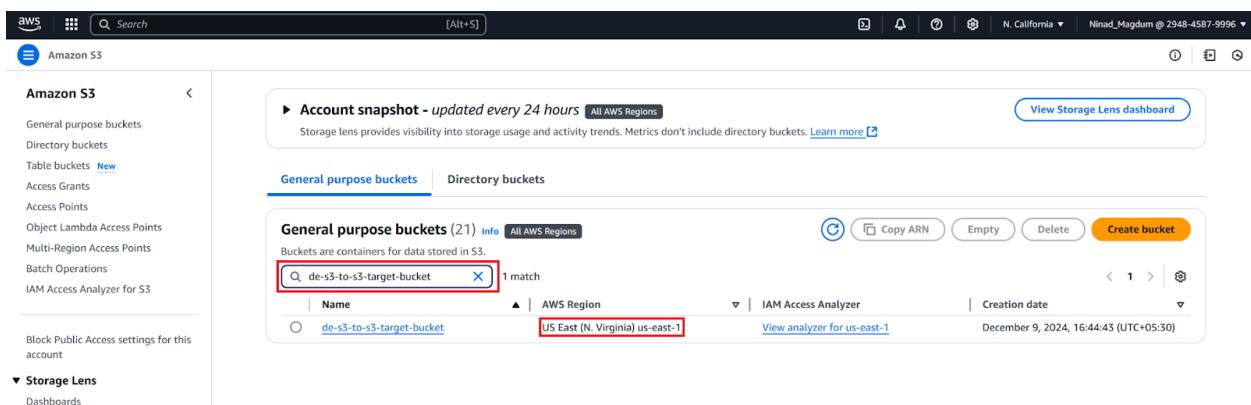
- Access your AWS Management Console and find AWS S3 by entering "S3" in the search bar.



- Select S3 from the menu, which will direct you to the page where you can see all of your buckets.



- Search your target bucket created for this project in the search bar and note down the AWS region of the same.



In my case the region is us-east-1, in your case it might be different.

- Click on the Region button on the top right hand corner of your AWS Management Console. Change the AWS region to the target bucket region.

The screenshot shows the AWS S3 console. On the left, there's a sidebar with options like General purpose buckets, Directory buckets, Table buckets (New), Access Grants, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, IAM Access Analyzer for S3, and Block Public Access settings. The main area displays an account snapshot and a list of general purpose buckets. A dropdown menu for selecting an AWS Region is open, with 'N. California' highlighted by a red box. Other regions listed include N. Virginia, Ohio, N. California, Oregon, Mumbai, Osaka, Seoul, Singapore, Sydney, Tokyo, Canada, and Central.

- Once selected validate your AWS region changed to the target bucket region.

This screenshot is similar to the previous one but shows the region dropdown menu again. The 'N. Virginia' region is now highlighted with a red box. The rest of the interface is identical, showing the account snapshot and the list of general purpose buckets.

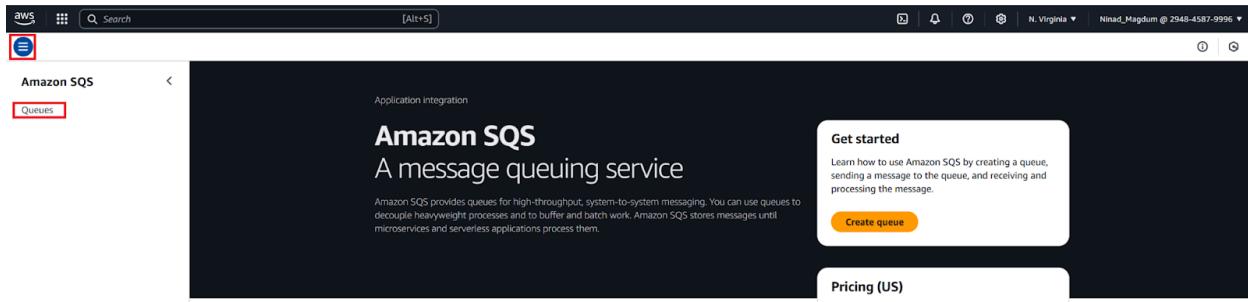
This is the region where we are going to create the SQS Queue required for this project lambda function

Creation of SQS Queue for the lambda

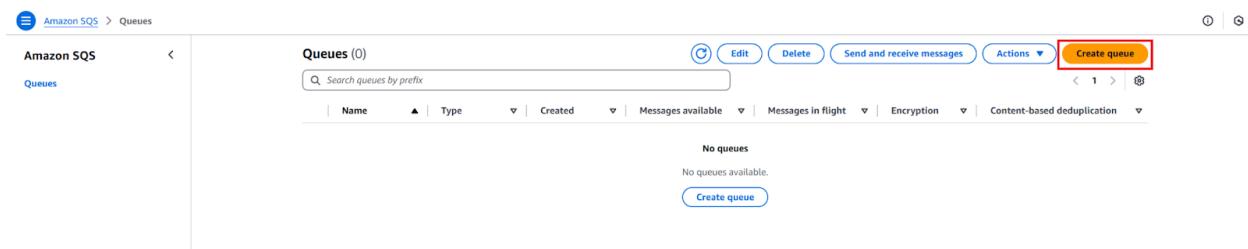
- In your AWS Management Console, search for SQS by entering "SQS" in the search bar and select Simple Queue Service

The screenshot shows the AWS Management Console search results for 'SQS'. The 'Simple Queue Service' option is highlighted with a red box. Other services listed include Service Quotas, Simple Notification Service, and RDS. The right side of the screen shows the AWS S3 console with the 'N. Virginia' region selected, displaying the account snapshot and a list of general purpose buckets.

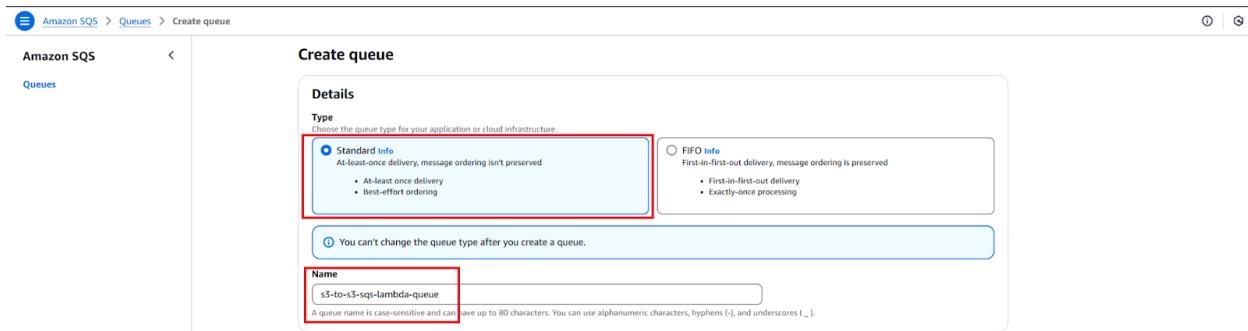
- On the SQS page , click the three line menu button on the left top corner and then click on the Queues button



- Once in the Queues page, click on the Create queue button to create a new SQS queue



- Under the Create queue page, select the Type as Standard and provide the name of the queue <s3-to-s3-sqs-lambda-queue>



- Scroll down and click on the Create queue button to create your SQS queue for the lambda function.

Redrive allow policy - Optional [Info](#)
Identify which source queues can use this queue as the dead-letter queue.

Select which source queues can use this queue as the dead-letter queue.

Disabled
 Enabled

Dead-letter queue - Optional [Info](#)
Send undeliverable messages to a dead-letter queue.

Set this queue to receive undeliverable messages.

Disabled
 Enabled

Tags - Optional [Info](#)
A tag is a label assigned to an AWS resource. Use tags to search and filter your resources or track your AWS costs.

| Key | Value - optional |
|--|--|
| <input type="text" value="Enter key"/> | <input type="text" value="Enter value"/> |
| Add new tag | |
| You can add 49 more tags. | |

[Cancel](#) [Create queue](#)

- Your SQS queue for the lambda function will be created successfully.

Amazon SQS > Queues > s3-to-s3-sqs-lambda-queue

s3-to-s3-sqs-lambda-queue

Details [Info](#)

| | | |
|---|--|---|
| Name s3-to-s3-sqs-lambda-queue | Type Standard | ARN arn:aws:sqs:us-east-1:294845879996:s3-to-s3-sqs-lambda-queue |
| Encryption Amazon SQS key (SSE-SQS) | URL https://sqs.us-east-1.amazonaws.com/294845879996/s3-to-s3-sqs-lambda-queue | Dead-letter queue - |

[Edit](#) [Delete](#) [Purge](#) [Send and receive messages](#) [Start DLQ redive](#)

SNS subscriptions Lambda triggers EventBridge Pipes Dead-letter queue Monitoring Tagging Queue policies Encryption

Subscription region: us-east-1

SNS subscriptions (0) [Info](#)

| Subscription ARN | Topic ARN |
|------------------|-----------|
| | |

[Subscribe to Amazon SNS topic](#)

Creation of SNS subscription for the lambda queue

Once the lambda SQS queue is created successfully, we need to create the SNS subscription to receive notification from SNS topic created in the previous step

- On the lambda SQS queue page, click the SNS subscriptions tab, and then click the Subscribe to Amazon SNS topic button

The screenshot shows the AWS SQS 'Queues' section with the 's3-to-s3-sqs-lambda-queue' selected. The 'Details' tab is open, displaying basic queue information like Name, Type, ARN, and URL. Below it, the 'SNS subscriptions' tab is active, showing a list of existing subscriptions. A red box highlights the 'Subscribe to Amazon SNS topic' button at the top right of the subscription list.

- Select the SNS topic created for the project on the next page

This screenshot shows the 'Subscribe to Amazon SNS topic' dialog. It includes instructions to choose an SNS topic and a dropdown menu where a specific topic ARN is selected. A red box highlights the selected topic ARN: 'arn:aws:sns:us-west-1:294845879996:s3-to-s3-sns-topic'.

- Click the Save button after selecting the SNS topic

This screenshot shows the same 'Subscribe to Amazon SNS topic' dialog after the 'Save' button has been clicked. The 'Save' button is highlighted with a red box, indicating the final step in the process.

- Once the subscription is created successfully, select the region where your SNS topic is created to confirm the subscription is created

The screenshot shows the AWS SQS Queue Details page for a queue named "s3-to-s3-sqs-lambda-queue". A green success message at the top states "Subscribed successfully to topic arn:aws:sns:us-west-1:294845879996:s3-to-s3-sns-topic." Below the message are buttons for Edit, Delete, Purge, Send and receive messages, and Start DLQ redrive. The main section displays queue details: Name (s3-to-s3-sqs-lambda-queue), Type (Standard), ARN (arn:aws:sqs:us-east-1:294845879996:s3-to-s3-sqs-lambda-queue), URL (https://sqs.us-east-1.amazonaws.com/294845879996/s3-to-s3-sqs-lambda-queue), and Dead-letter queue (-). A "More" link is also present. Below this, a navigation bar includes links for SNS subscriptions, Lambda triggers, EventBridge Pipes, Dead-letter queue, Monitoring, Tagging, Queue policies, and Encryption. A dropdown menu for "Subscription region" is set to "us-west-1". Under "SNS subscriptions", there is one entry: "arn:aws:sns:us-west-1:294845879996:s3-to-s3-sns-topic". A "Subscribe to Amazon SNS topic" button is visible.

VOILA! YOU HAVE SUCCESSFULLY CREATED YOUR SQS QUEUE FOR YOUR LAMBDA FUNCTION

AWS Lambda Function Creation guide

- Let us understand and walk through the process of creation of AWS lambda function

Identify the Lambda function AWS Region

First we need to identify the AWS region in which we need to create the required lambda function for this project

- Access your AWS Management Console and find AWS S3 by entering "S3" in the search bar.

The screenshot shows the AWS Management Console search results for "S3". The search bar at the top has "S3" entered. The left sidebar lists services like Services, Features, Resources, Documentation, Knowledge articles, Marketplace, Blog posts, Events, and Tutorials. The main area shows a list of services: "S3 Scalable Storage in the Cloud" (highlighted with a red box), "S3 Glacier Archive Storage in the Cloud", "AWS Snow Family Large Scale Data Transport", and "Storage Gateway Hybrid Storage Integration". On the right, there is a "Create application" button and a list of applications under "Find applications". The status bar at the bottom indicates "N. California" and "Ninad_Magnum @ 2948-4587-9996".

- Select S3 from the menu, which will direct you to the page where you can see all of your buckets.

- Search your target bucket created for this project in the search bar and note down the AWS region of the same.

In my case the region is us-east-1, in your case it might be different.

- Click on the Region button on the top right hand corner of your AWS Management Console. Change the AWS region to the target bucket region.

- Once selected validate your AWS region changed to the target bucket region.

The screenshot shows the AWS Storage Lens dashboard. At the top, there's a search bar and a region selector set to "N. Virginia". Below the header, there's a section titled "Account snapshot - updated every 24 hours" with a link to "All AWS Regions". A button "View Storage Lens dashboard" is also present. On the left, a sidebar titled "Amazon S3" lists various bucket types: General purpose buckets, Directory buckets, Table buckets (with a "New" link), Access Grants, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, and IAM Access Analyzer for S3. The main content area displays "General purpose buckets (21)" with a "Copy ARN" button, an "Empty" button, a "Delete" button, and a "Create bucket" button. There are filters for "Name", "AWS Region", "Creation date", and sorting options.

This is the region where we are going to create the lambda function required for this project

Creation of AWS Lambda Function

- Open your AWS Management Console and navigate to AWS Lambda by searching for "lambda" in the Search Bar.

The screenshot shows the AWS Management Console search results. The search bar at the top contains the text "lambda". Below the search bar, there's a sidebar with links like Services, Features, Resources, Documentation, Knowledge articles, Marketplace, Blog posts, Events, and Tutorials. The main content area shows several service cards: "Lambda" (highlighted with a red box), "CodeBuild", "AWS Signer", and "Amazon Inspector". To the right, there's a separate window for "Applications" which is currently empty.

- Click on "Lambda," and you'll land on the page where click the three line menu button on the left top corner and then click on the Functions button

The screenshot shows the AWS Lambda service page. The sidebar on the left has a "Functions" button highlighted with a red box. The main content area features the heading "AWS Lambda" with the tagline "lets you run code without thinking about servers." It includes a "Get started" button and a "How it works" section with a code editor showing Node.js code:

```

    .NET | Java | Node.js | Python | Ruby | Custom runtime
    1* exports.handler = async (event) => {
    2   console.log(event);
    3   return 'Hello from Lambda!';
    4 };
    5
  
```

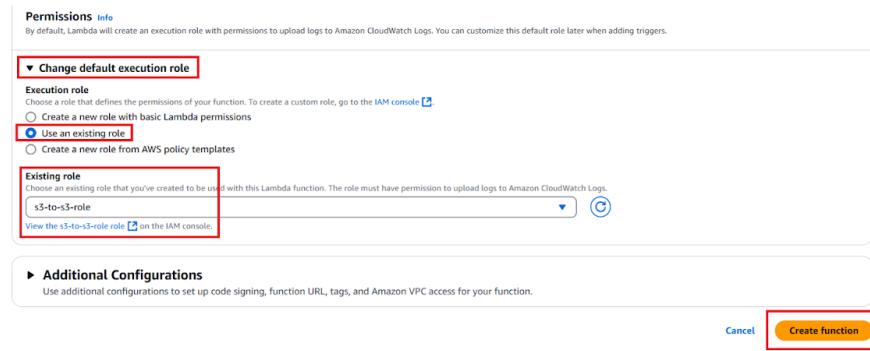
- On the functions page, click on the "Create function" button to create a new lambda function.

The screenshot shows the AWS Lambda Functions page. On the left, there's a sidebar with 'Lambda' selected. The main area is titled 'Functions (0)' with a search bar. Below it is a table header with columns for 'Function name', 'Description', 'Package type', 'Runtime', and 'Last modified'. A message says 'There is no data to display.' At the top right, there are buttons for 'Actions' and 'Create function' (which is highlighted with a red box).

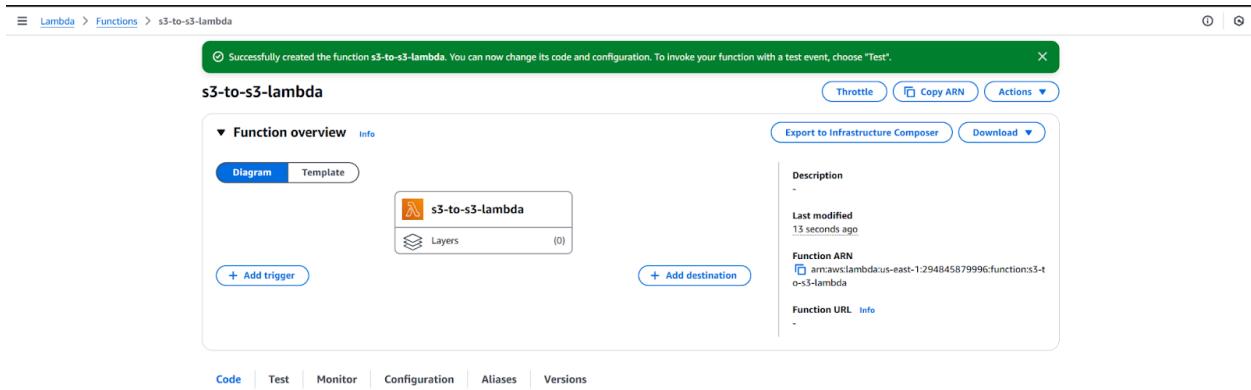
- Once you click "Create function," you'll be directed to the following page where you need to fill in the following information accordingly.
 - Select Author from scratch option
 - Provide name to your function under Function name section <s3-to-s3-lambda>
 - Select runtime as the latest python version
 - Keep the architecture as default x86_64

The screenshot shows the 'Create function' wizard. It has three tabs: 'Create function' (selected), 'Basic information', and 'Permissions'. In the 'Create function' tab, there are three options: 'Author from scratch' (selected), 'Use a blueprint', and 'Container image'. The 'Basic information' tab is expanded, showing fields for 'Function name' (set to 's3-to-s3-lambda'), 'Runtime' (set to 'Python 3.13'), and 'Architecture' (set to 'x86_64'). The 'Permissions' tab is partially visible at the bottom.

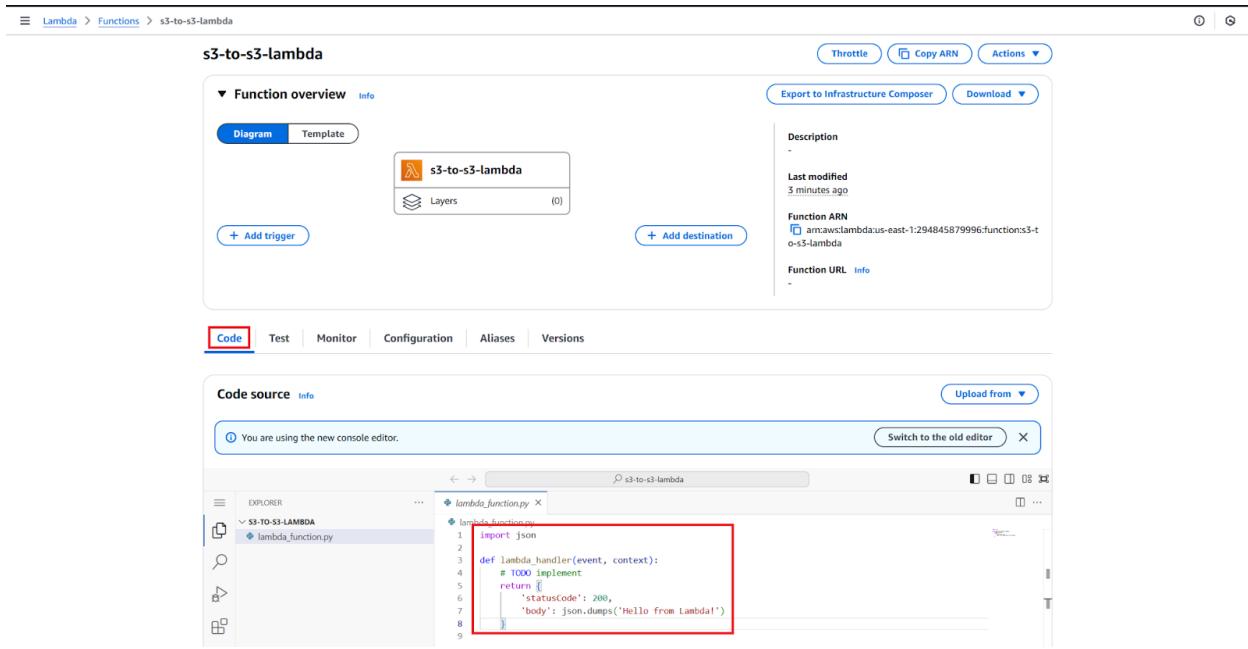
- Expand the "Change default execution role" section and select the option "Use an existing role." and select the already created role created previously for this project and click on "Create Function" button



- Once you click "Create function," you will land on the following page confirming that your Lambda function has been created successfully.



- Navigate to the "Code" section of the function and delete the existing code that is currently available.



- Paste the following code into the code editor and update the “Your-Target-S3-Bucket-Name” value with your actual target s3 bucket name.

```
import boto3
```

```
import json
```

```
# Get the S3 client
```

```
s3 = boto3.client('s3')
```

```
def lambda_handler(event, context):
```

```
    # Get the list of records from the event
```

```
    print(event)
```

```
    records = event['Records']
```

```
# Iterate through the records and copy the files from the source bucket to the destination bucket
```

```
for record in records:
```

```
    di = record['body']
```

```
    result = json.loads(di)
```

```
    message=json.loads(result['Message'])
```

```
    source_bucket = message['Records'][0]['s3']['bucket']['name']
```

```
    key = message['Records'][0]['s3']['object']['key']
```

```
    print('key',key)
```

```
    copy_source = {'Bucket': source_bucket, 'Key': key}
```

```
s3.copy_object(Bucket='Your-Target-S3-Bucket-Name', CopySource=copy_source,  
Key=key)
```

```
print('File moved successfully')
```

The screenshot shows the AWS Lambda Function Editor interface. At the top, it displays the function name 's3-to-s3-lambda'. Below the title bar are tabs for 'Code', 'Test', 'Monitor', 'Configuration', 'Aliases', and 'Versions'. The 'Code' tab is selected. The main area is titled 'Code source' and contains a file named 'lambda_function.py'. The code within the file is as follows:

```
import boto3
import json

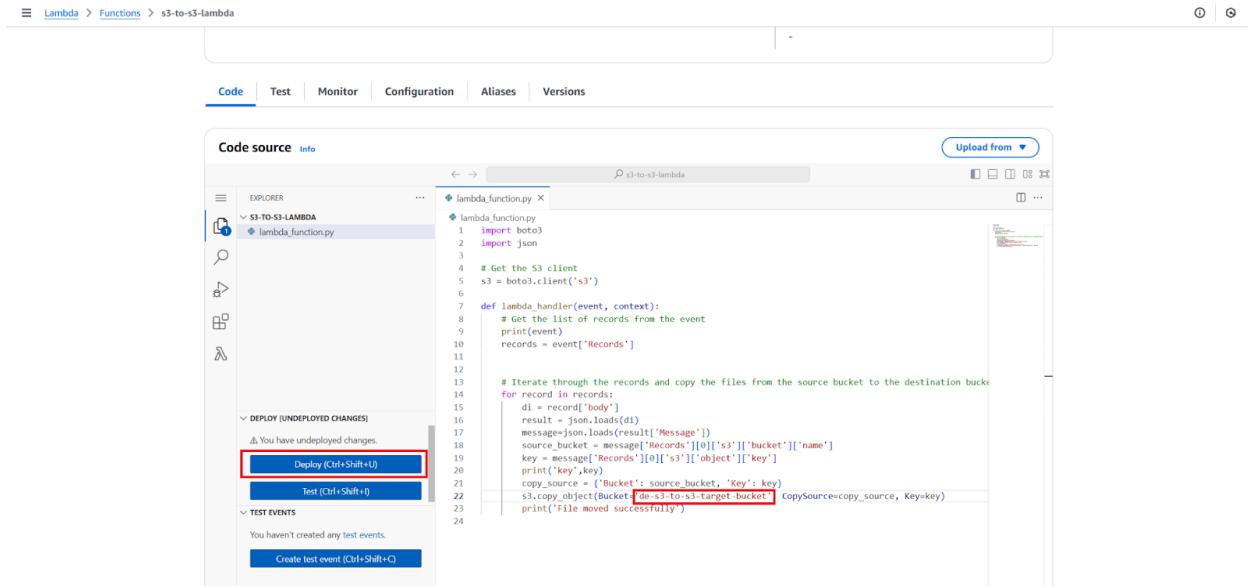
# Get the S3 client
s3 = boto3.client('s3')

def lambda_handler(event, context):
    # Get the list of records from the event
    print(event)
    records = event['Records']

    # Iterate through the records and copy the files from the source bucket to the destination bucket
    for record in records:
        di = record['body']
        result = json.loads(di)
        message=json.loads(result['Message'])
        source_bucket = message['Records'][0]['s3']['bucket']['name']
        key = message['Records'][0]['s3']['object']['key']
        print('key',key)
        copy_source = {'Bucket': source_bucket, 'Key': key}
        s3.copy_object(Bucket='Your-Target-S3-Bucket-Name', CopySource=copy_source, Key=key)
        print('File moved successfully')
```

The code is annotated with several red boxes highlighting specific lines of code. One box covers the entire first section of the code (lines 1-13). Another box highlights the 'CopySource=copy_source, Key=key' part of the 's3.copy_object' call (line 23). A third box highlights the final 'print' statement (line 24).

- Once you update the Your-Target-S3-Bucket-Name value with your actual target s3 bucket name, click on the "Deploy" button to update your Lambda function



The screenshot shows the AWS Lambda Functions interface. In the top navigation bar, it says "Lambda > Functions > s3-to-s3-lambda". Below the navigation, there are tabs: Code (which is selected), Test, Monitor, Configuration, Aliases, and Versions. The main area is titled "Code source" with an "Info" link. On the left, there's an "EXPLORER" sidebar showing a folder named "S3-TO-S3-LAMBDA" containing a file "lambda_function.py". The code editor window contains the following Python script:

```

# S3-TO-S3-LAMBDA
# lambda_function.py

# Import required libraries
import boto3
import json

# Get the S3 client
s3 = boto3.client('s3')

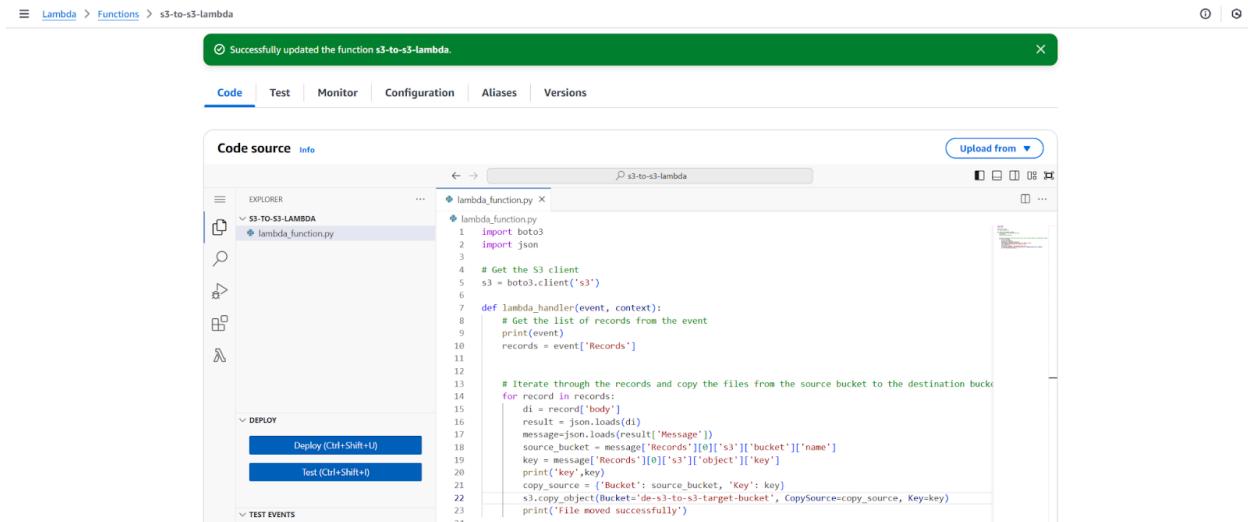
# Lambda handler function
def lambda_handler(event, context):
    # Get the list of records from the event
    print(event)
    records = event['Records']

    # Iterate through the records and copy the files from the source bucket to the destination bucket
    for record in records:
        # Get the object from the record
        di = record['body']
        result = json.loads(di)
        message=json.loads(result['Message'])
        source_bucket = message['Records'][0]['s3']['bucket']['name']
        key = message['Records'][0]['s3']['object']['key']
        print("key:",key)
        copy_source = {'Bucket': source_bucket, 'Key': key}
        s3.copy_object(Bucket='de-s3-to-s3-target-bucket', CopySource=copy_source, Key=key)
        print("File moved successfully")

```

In the bottom left of the code editor, there are two buttons: "Deploy (Ctrl+Shift+U)" and "Test (Ctrl+Shift+T)". The "Deploy" button is highlighted with a red box.

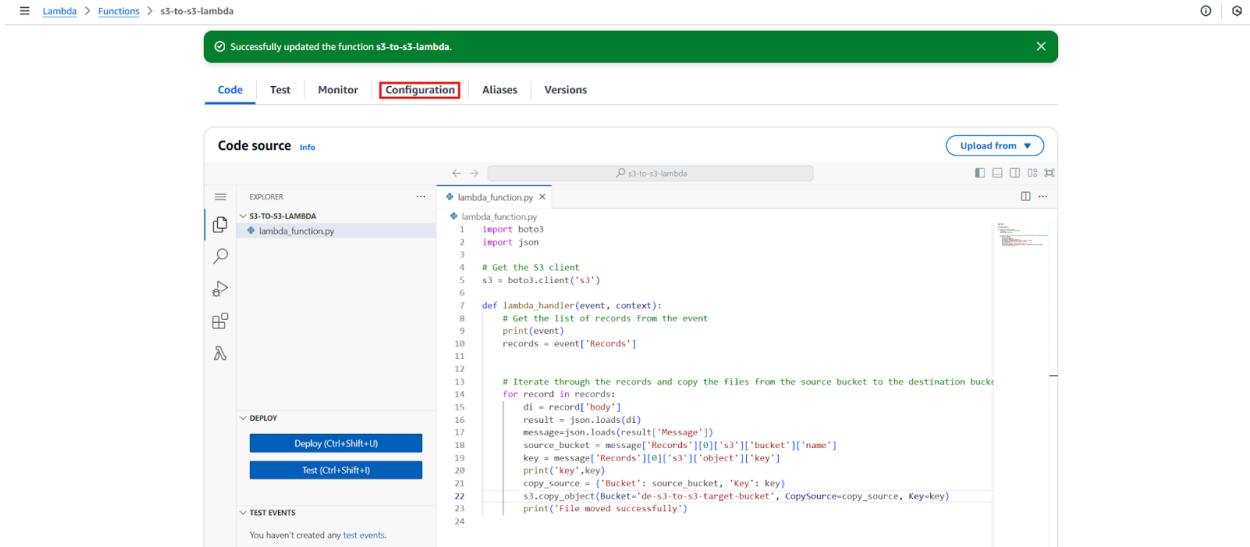
- One clicked Deploy button, your lambda function will be created with the updated code



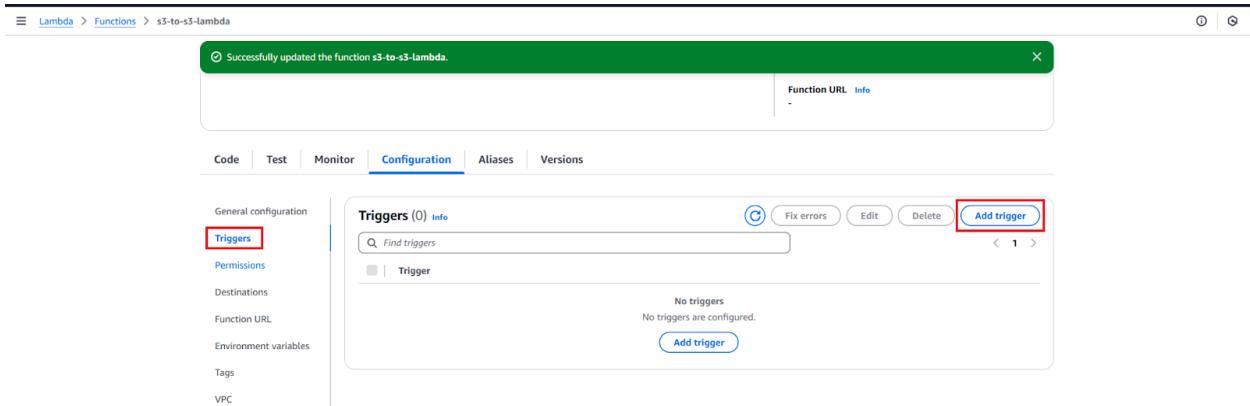
The screenshot shows the AWS Lambda Functions interface after deployment. The top navigation bar now says "Lambda > Functions > s3-to-s3-lambda". A green success message box at the top center says "Successfully updated the function s3-to-s3-lambda." Below the message, the interface looks identical to the previous screenshot, with the "Code" tab selected, the "EXPLORER" sidebar, and the code editor showing the same Python script. The "Deploy" and "Test" buttons are still present at the bottom left of the code editor.

Setup the SQS Trigger to AWS Lambda Function

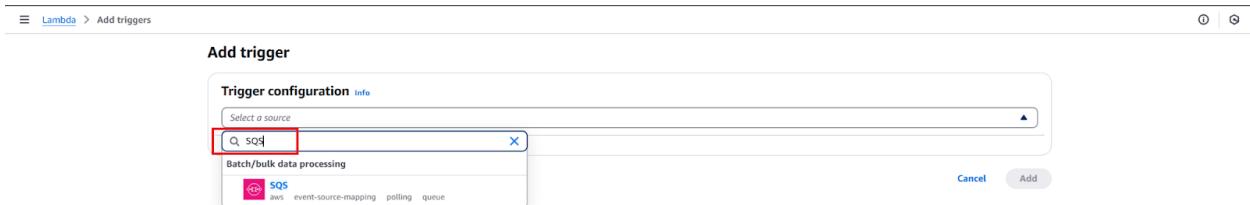
- Once the lambda function is deployed with the updated code, click on the Configuration tab



- Under the Configuration tab, click on the Trigger button and then click Add trigger button to create a new trigger



- On the Trigger configuration page, select the source as SQS and select the SQS from the below list



- Once you select SQS from the list, select the previously created lambda SQS queue in the SQS queue section and then click the Add button.

[Lambda](#) > Add triggers

Add trigger

Trigger configuration [Info](#)

SQS [aws](#) [event-source-mapping](#) [polling](#) [queue](#)

SQS queue Choose or enter the ARN of an SQS queue.

[X](#) [C](#)

Use: "arn:aws:sqs:us-east-1:294845879996:s3-to-s3-sqs-lambda-queue"

[s3-to-s3-sqs-crawler-queue](#)

[s3-to-s3-sqs-lambda-queue](#) Enabled state for testing (recommended).

[Enable metrics](#) Monitor your event source with metrics. You can view those metrics in CloudWatch console. Enabling this feature incurs additional costs. [Learn more](#)

Batch size - optional The number of records in each batch to send to the function.

The maximum is 10,000 for standard queues and 10 for FIFO queues.

Batch window - optional The maximum amount of time to gather records before invoking the function, in seconds.

When the batch size is greater than 10, set the batch window to at least 1 second.

Maximum concurrency - optional The maximum number of concurrent function instances that the SQS event source can invoke.

Specify a value between 2 and 1000. To deactivate, leave the box empty.

Report batch item failures - optional Allow your function to return a partial successful response for a batch of records.

Filter criteria - optional Define the filtering criteria to determine whether or not to process an event. Each filter must be in a valid JSON format in filter rule syntax. Lambda processes an event if any one of the filters are met. Otherwise, Lambda discards the event. [Learn more](#)

[Add](#)

During trigger creation, Lambda translates your filter(s) into a single JSON structure containing all your filtering criteria.

[Encrypt filter criteria with customer managed KMS key](#) Lambda will use this key to encrypt and decrypt your resources. [Learn more](#)

Additional settings

In order to read from the SQS trigger, your execution role must have proper permissions.

[Cancel](#) [Add](#)

- Once created you should be able to see the SQS trigger setup for your lambda function and it's in Enable state

[Lambda](#) > Functions > s3-to-s3-lambda

s3-to-s3-lambda

[Throttle](#) [Copy ARN](#) [Actions ▾](#)

The trigger s3-to-s3-sqs-lambda-queue was successfully added to function s3-to-s3-lambda. The trigger is in a disabled state.

Function overview [Info](#) [Export to Infrastructure Composer](#) [Download](#)

Diagram [Template](#)

Triggers (1) [Info](#)

| Trigger | ARN | State |
|---|--|-------------------------|
| s3-to-s3-sqs-lambda-queue | arn:aws:lambda:us-east-1:294845879996:function:s3-to-s3-lambda | Enabled |

[Fix errors](#) [Edit](#) [Delete](#) [Add trigger](#)

Code [Test](#) [Monitor](#) [Configuration](#) [Aliases](#) [Versions](#)

General configuration

Triggers

Permissions

Destinations

Function URL

Environment variables

Tags

VOILA! YOUR SUCCESSFULLY CREATED YOUR LAMBDA FUNCTION AND SETUP THE SQS TRIGGER

1. [aws_module](#)
2. [Cross region S3-S3 migration using Lambda](#)
3. [AWS Glue Crawler](#)
4. [Documentation](#)

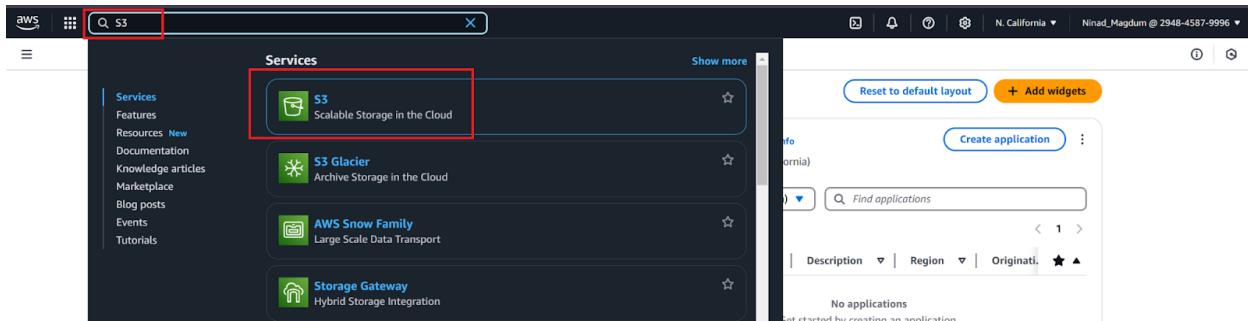
AWS Glue Crawler Creation guide

- Let us understand and walk through the process of creation of Crawler in the AWS Glue

Identify the Glue Crawler AWS Region

First we need to identify the AWS region in which we need to setup the required Glue crawler for this project

- Access your AWS Management Console and find AWS S3 by entering "S3" in the search bar.



- Select S3 from the menu, which will direct you to the page where you can see all of your buckets.

- Search your target bucket created for this project in the search bar and note down the AWS region of the same.

In my case the region is us-east-1, in your case it might be different.

- Click on the Region button on the top right hand corner of your AWS Management Console. Change the AWS region to the target bucket region.

- Once selected validate your AWS region changed to the target bucket region.

The screenshot shows the AWS S3 console. On the left, there's a sidebar with options like 'General purpose buckets', 'Directory buckets', and 'Table buckets'. The main area is titled 'Account snapshot - updated every 24 hours' and shows 'General purpose buckets (21)'. A red box highlights the 'Create bucket' button at the top right of the list.

This is the region where we are going to create the lambda function required for this project
(In your case it might be different)

Creation of AWS Glue Crawler Classifiers

- Open your AWS Management Console and navigate to AWS Glue by searching for "Glue" in the Search Bar.

The screenshot shows the AWS Management Console search results. The search bar at the top has 'glue' typed into it. In the results, 'AWS Glue' is highlighted with a red box. It is described as a 'serverless data integration service'. To the right, there's a separate window for 'AWS Glue Applications' which is currently empty.

- Click on "AWS Glue," and you'll land on the page where click the three line menu button on the left top corner and then click on the Classifier button under crawlers

The screenshot shows the 'Welcome to AWS Glue' page. On the left, there's a navigation sidebar with links like 'Getting started', 'Data Catalog', 'Data Integration and ETL', and 'Legacy pages'. Under 'Data Catalog', the 'Classifiers' link is highlighted with a red box. The main content area has sections for 'Prepare your account for AWS Glue', 'Catalog and search for datasets', 'Move and transform data', 'Resources and tutorials', 'Data integration and management', and 'What's new in Glue'. A red box also highlights the 'Classifiers' link in the 'Resources and tutorials' section.

- On the classifiers page, click on the "Add classifier" button to create a new crawler classifier.

The screenshot shows the AWS Glue interface with the 'Classifiers' page selected. The left sidebar includes sections for AWS Glue (Getting started, ETL jobs, Visual ETL, Notebooks, Job run monitoring), Data Catalog (Databases, Tables, Stream schema registries, Schemas, Connections, Crawlers, Classifiers, Catalog settings), and Data Integration and ETL (Legacy pages). The main content area is titled 'Classifiers' and contains a sub-section 'Classifiers (0)'. It includes a search bar 'Filter classifiers', columns for 'Name' and 'Type', and sorting options for 'Classification' and 'Last updated (UTC)'. A red box highlights the 'Add classifier' button at the top right of the page.

- Once you click "Add classifier", you'll be directed to the following page where you need to fill in the following information accordingly and then click on Create button to create your crawler classifier
 - Classifier name : CSV
 - Classifier Type : CSV
 - Column headings : Has headings

The screenshot shows the 'Create classifier' wizard. The left sidebar is identical to the previous screenshot. The main form has two tabs: 'Classifier details' and 'Classifier type and properties'. The 'Classifier details' tab shows 'Classifier name' set to 'CSV'. The 'Classifier type and properties' tab shows 'Classifier type' set to 'CSV' (radio button selected) and 'CSV Serde - optional' set to 'None'. Other fields include 'Column delimiter' (Comma (,), 'Quote symbol' (Double-quote (")), 'Column headings' (Has headings), and processing options like 'Allow files with single column' and 'Disable whitespace trimming before identifying column values'. A red box highlights the 'CSV' radio button and the 'CSV Serde' dropdown. The bottom right of the form has 'Cancel' and 'Create' buttons, with 'Create' also highlighted by a red box.

- Once you click "Create", your crawler classifier will be created successfully.

The screenshot shows the AWS Glue Classifiers page. On the left, there's a navigation sidebar with sections like AWS Glue, Data Catalog, Data Integration and ETL, and Legacy pages. The main area is titled 'Classifiers' and contains a table with one row. The row has a red border and shows a classifier named 'CSV' with a type of 'CSV'. The table includes columns for Name, Type, Classification, and Last updated (UTC). The last update was December 18, 2024, at 05:52:52.

Creation of AWS Glue Crawler Job

- Open your AWS Management Console and navigate to AWS Glue by searching for "Glue" in the Search Bar.

The screenshot shows the AWS Management Console search results for 'glue'. The search bar at the top has 'glue' typed in. Below it, the 'Services' section lists several AWS services, with 'AWS Glue' highlighted by a red box. To the right, there's a separate window or tab showing the AWS Glue Applications page, which is currently empty.

- Click on "AWS Glue," and you'll land on the page where click the three line menu button on the left top corner and then click on the Crawlers button

The screenshot shows the AWS Glue Welcome page. The sidebar on the left has a 'Crawlers' button, which is highlighted with a red box. The main content area includes sections for 'Prepare your account for AWS Glue', 'Catalog and search for datasets', 'Move and transform data', 'Resources and tutorials', 'What's new in Glue', and 'Data integration and management'. Each section contains various links and icons related to AWS Glue features.

- On the crawlers page, click on the "Create crawler" button to create a new crawler

The screenshot shows the AWS Glue interface with the 'Crawlers' section selected. On the right, there is a table header for 'Crawlers' with columns: Name, State, Schedule, Last run, Last run timestamp, Log, and Table changes from last run. A red box highlights the 'Create crawler' button at the top right of the table area.

- Once you click "Create crawler", you'll be directed to the following pages where you need to fill in the following
- Set crawler properties Page
 - On this page provide the name of your crawler <s3-to-s3-crawler> and click Next button

The screenshot shows the 'Set crawler properties' step of the 'Add crawler' wizard. The 'Name' field is highlighted with a red box and contains the value 's3-to-s3-crawler'. The 'Description - optional' and 'Tags - optional' sections are also visible. At the bottom right, the 'Cancel' and 'Next' buttons are shown, with 'Next' being highlighted by a red box.

- Choose data sources and classifiers Page
 - On this page select the Not yet option under the Data source configuration and then click on Add a Data Source button

The screenshot shows the 'Choose data sources and classifiers' step of the 'Add crawler' wizard. Under 'Data source configuration', the 'Not yet' option is selected and highlighted with a red box. Below it, the 'Yes' option is available. The 'Data sources' section shows a table with columns: Type, Data source, and Parameters. An 'Add a data source' button is located at the bottom right of this section, highlighted with a red box. The 'Custom classifiers - optional' section is also visible.

- On the Add data source page, fill the below details accordingly and click Add an S3 data source button
 - Data Source : S3
 - Location of S3 data : In this account
 - S3 Path : Your S3 Target bucket location

Add data source

Data source
Choose the source of data to be crawled.

S3

Network connection - optional
Optionaly include a Network connection to use with this S3 target. Note that each crawler is limited to one Network connection so any other S3 targets will also use the same connection (or none, if left blank).

Location of S3 data

In this account
 In a different account

S3 path
Browse for or enter an existing S3 path.

s3://de-s3-to-s3-target-bucket

All folders and files contained in the S3 path are crawled. For example, type s3://MyBucket/MyFolder/ to crawl all objects in MyFolder within MyBucket.

Subsequent crawler runs
This field is a global field that affects all S3 data sources.

Crawl all sub-folders
Crawl all folders again with every subsequent crawl.

Crawl new sub-folders only
Only Amazon S3 folders that were added since the last crawl will be crawled. If the schemas are compatible, new partitions will be added to existing tables.

Crawl based on events
Rely on Amazon S3 events to control what folders to crawl.

Sample only a subset of files

Exclude files matching pattern

- Choose the already existing CSV classifier under the Custom classifiers section and click on the Next button.

The screenshot shows the 'Add crawler' wizard in AWS Glue. The left sidebar shows navigation links for AWS Glue, Data Catalog, and Data Integration and ETL. The main panel is titled 'Choose data sources and classifiers'. It has sections for 'Data source configuration' (radio buttons for 'Not yet' or 'Yes'), 'Data sources (1) info' (listing 's3://de-s3-to-s3-target-bucket' with columns for Type, Data source, and Parameters), and 'Custom classifiers - optional' (a dropdown menu with 'CSV' selected). At the bottom right are 'Cancel', 'Previous', and 'Next' buttons, with 'Next' being highlighted.

- Configure security settings page
 - On this page, choose the already existing IAM role created for this project and click on the Next button.

The screenshot shows the 'Add crawler' wizard in AWS Glue. The left sidebar shows navigation links for AWS Glue, Data Catalog, and Data Integration and ETL. The main panel is titled 'Configure security settings'. It has sections for 'IAM role' (radio buttons for 'Existing IAM role' or 'Create new IAM role', with 'Existing IAM role' selected and 's3-to-s3-role' chosen), 'Lake Formation configuration - optional' (checkbox for 'Use Lake Formation credentials for crawling S3 data source'), and 'Security configuration - optional' (checkbox for 'Enable at-rest encryption with a security configuration'). At the bottom right are 'Cancel', 'Previous', and 'Next' buttons, with 'Next' being highlighted.

- Set output and scheduling page
 - On this page, choose the Target Database as default and the Frequency as On demand under Crawler Schedule and then click on the Next button.

AWS Glue > Crawlers > Add crawler

Set output and scheduling

Output configuration info
Target database
default

Table name prefix - optional
Type a prefix added to table names

Maximum table threshold - optional
This field sets the maximum number of tables the crawler is allowed to generate. In the event that this number is surpassed, the crawl will fail with an error. If not set, the crawler will automatically generate the number of tables depending on the data schema.

Crawler schedule
You can define a time-based schedule for your crawlers and jobs in AWS Glue. The definition of these schedules uses the Unix-like cron syntax. [Learn more](#)

Frequency
On demand

Cancel Previous Next

- On the Review and create page, click Create crawler to create your job

AWS Glue > Crawlers > Add crawler

Review and create

Step 1: Set crawler properties

Set crawler properties

| | | | | | |
|------|------------------|-------------|---|------|---|
| Name | s3-to-s3-crawler | Description | - | Tags | - |
|------|------------------|-------------|---|------|---|

Step 2: Choose data sources and classifiers

Data sources (1) info
The list of data sources to be scanned by the crawler.

| | | | | | |
|------|----|-------------|--------------------------------|------------|-------------|
| Type | S3 | Data source | s3://de-s3-to-s3-target-bucket | Parameters | Recrawl all |
|------|----|-------------|--------------------------------|------------|-------------|

Classifiers (1) info
A classifier can help determine the schema of your data.

| | | | | | | | |
|------|-----|------|-----|----------------|---|--------------------|-------------------------------|
| Name | CSV | Type | CSV | Classification | - | Last updated (UTC) | December 18, 2024 at 05:52:52 |
|------|-----|------|-----|----------------|---|--------------------|-------------------------------|

Step 3: Configure security settings

Configure security settings

| | | | | | |
|----------|---------------|------------------------|---|------------------------------|---|
| IAM role | s3-to-s3-role | Security configuration | - | Lake Formation configuration | - |
|----------|---------------|------------------------|---|------------------------------|---|

Step 4: Set output and scheduling

Set output and scheduling

| | | | | | | | |
|----------|---------|-------------------------|---|------------------------------------|---|----------|-----------|
| Database | default | Table prefix - optional | - | Maximum table threshold - optional | - | Schedule | On demand |
|----------|---------|-------------------------|---|------------------------------------|---|----------|-----------|

Cancel Previous Create crawler

- Once you click "Create crawler", your crawler job will be created successfully.

AWS Glue > Crawlers > s3-to-s3-crawler

Crawler properties

| | | | | | | | |
|-------------------------|------------------|------------------------|---------------|------------------------------|---------|--------------|-------|
| Name | s3-to-s3-crawler | IAM role | s3-to-s3-role | Database | default | State | READY |
| Description | - | Security configuration | - | Lake Formation configuration | - | Table prefix | - |
| Maximum table threshold | - | | | | | | |

Advanced settings

Crawler runs (0)
The list of crawler runs for this crawler.

| Start time (UTC) | End time (UTC) | Filter data | Stop run | View CloudWatch logs | View run details |
|------------------|----------------|---------------------------------|-----------------------|----------------------|------------------|
| Start time (UTC) | End time (UTC) | Filter by a date and time range | Current/last duration | Status | DPU hours |

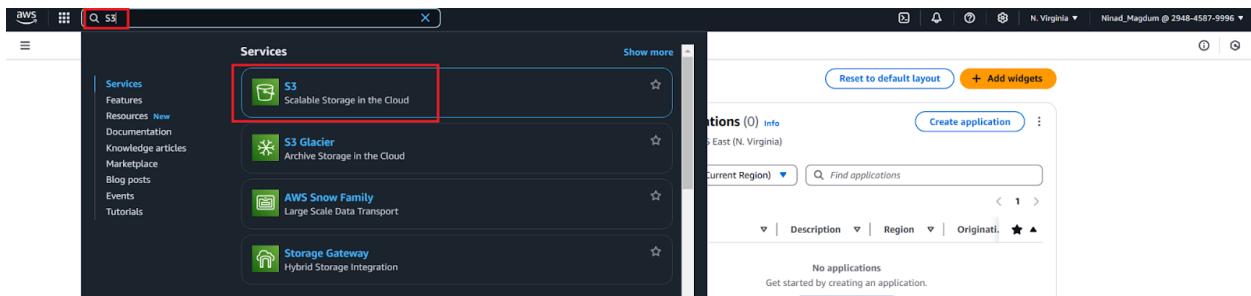
You don't have any crawler runs.
Run crawler

VOILA! YOUR SUCCESSFULLY CREATED YOUR AWS GLUE CRAWLER

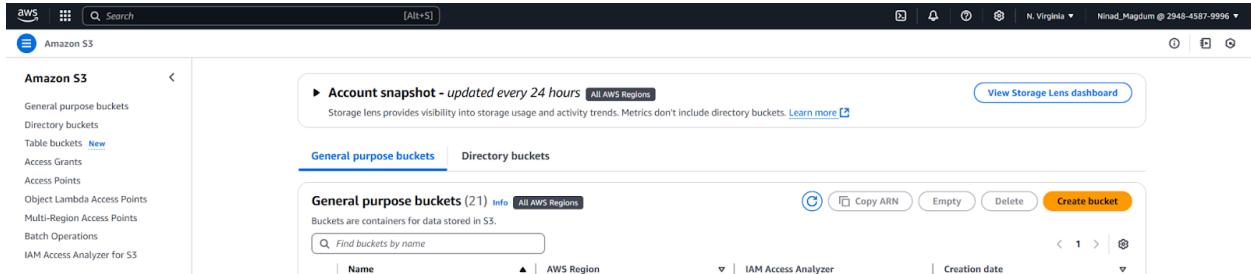
AWS S3 Source Bucket Event Notification Creation guide

Let us understand and walk through the process of Event notification creation on the S3 source bucket

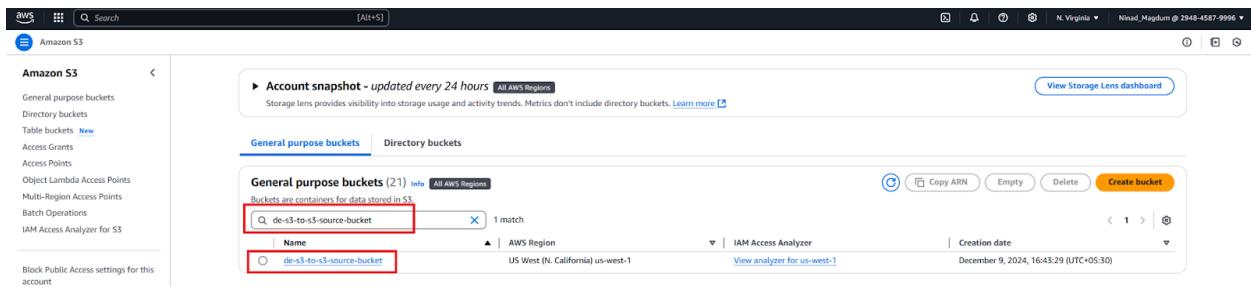
- Access your AWS Management Console and find AWS S3 by entering "S3" in the search bar.



- Select S3 from the menu, which will direct you to the page where you can create an S3 Bucket.



- Search your source bucket created for this project in the search bar and click on the bucket to open it.



- On the source bucket page, click on the Properties tab

de-s3-to-s3-source-bucket

Objects **Properties** Permissions Metrics Management Access Points

Bucket overview

AWS Region: US West (N. California) us-west-1

Amazon Resource Name (ARN): arnaws3::de-s3-to-s3-source-bucket

Creation date: December 9, 2024, 16:43:29 (UTC+05:30)

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](#)

Bucket Versioning

Disabled

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

- On the Properties tab, Scroll down to the Event Notification section and click on Create event notification.

Event notifications (0)

Send a notification when specific events occur in your bucket. [Learn more](#)

| Name | Event types | Filters | Destination type | Destination |
|---|-------------|---------|------------------|-------------|
| No event notifications | | | | |
| Choose Create event notification to be notified when a specific event occurs. | | | | |
| Create event notification | | | | |

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](#)

Send notifications to Amazon EventBridge for all events in this bucket

Off

- Give a name to the Event notification and select All object create events under the Event types

Amazon S3 > Buckets > de-s3-to-s3-source-bucket > 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 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.

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

- Now scroll down to the Destination section and select SNS topic as destination, select Choose from your SNS topics and select the already created SNS topic for this project . Then click on Save changes.

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 SNS topic

Choose from your SNS topics

Enter SNS topic ARN

SNS topic

[Cancel](#) [Save changes](#)

- Your event notification for your source s3 bucket is now set up successfully under the Event notification section of S3

| Event notifications (1) | | | | |
|---|--------------------------|---------|------------------|--------------------|
| Edit Delete Create event notification | | | | |
| Name | Event types | Filters | Destination type | Destination |
| s3-to-s3-event | All object create events | - | SNS topic | s3-to-s3-sns-topic |
| 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 | | | | |
| Send notifications to Amazon EventBridge for all events in this bucket | | | | |
| Off | | | | |

VOILA! YOU HAVE SUCCESSFULLY CREATED THE EVENT NOTIFICATION ON THE SOURCE S3 BUCKETS

Test Data Flow

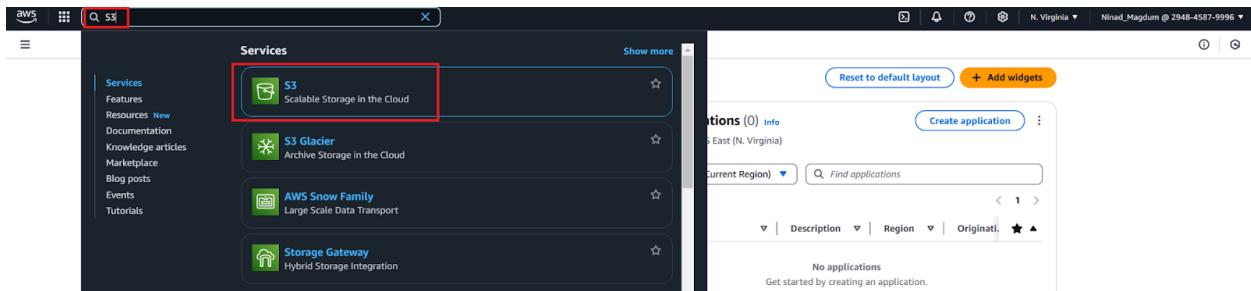
Step1: Download the customer Data File

- Download the customers.csv file shared along with the project to your local machine

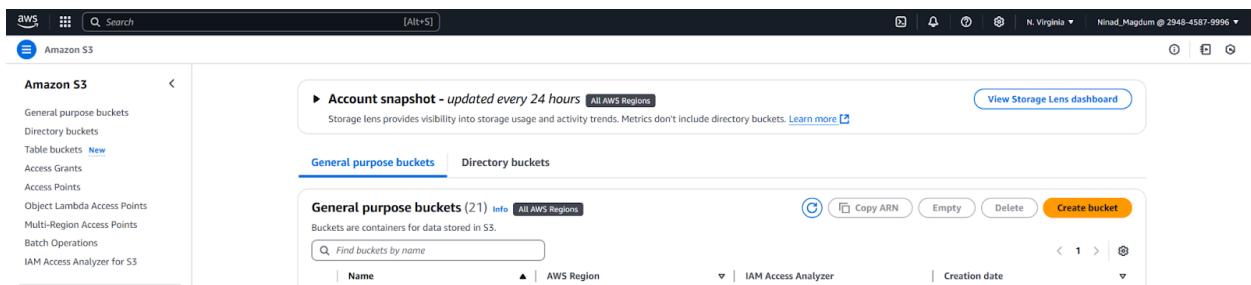
Step2: Upload file to S3 Source Bucket

In this step we will upload the downloaded customers.csv file to the source S3 bucket setup for this project

- Access your AWS Management Console and find AWS S3 by entering "S3" in the search bar.



- Select S3 from the menu, which will direct you to the page where you can create an S3 Bucket.



- Search your s3 source bucket and click it to open it

- On the s3 source bucket page, click on the Upload button to upload the customers.csv file

- On the Upload page, click the Add file button and select the customer.csv from your local machine

- Once the customer.csv is selected, click the Upload button to upload the file in the source S3 bucket

The screenshot shows the AWS Lambda console with a success message at the top: "Upload succeeded. For more information, see the Files and folders table." Below it, the "Upload: status" section indicates "Succeeded" with 1 file (128.1 KB) and "Failed" with 0 files (0 B). The "Files and folders" tab is selected, showing a table with one row: "customers.csv" (text/csv, 128.1 KB, Succeeded).

Step3: Validate if file is migrated to S3 Target Bucket

In this step we will validate if the customers.csv file is migrated to the target S3 bucket setup for this project

- Access your AWS Management Console and find AWS S3 by entering "S3" in the search bar.

The screenshot shows the AWS Management Console search results for "S3". The "Services" section is visible on the left, and the main area shows the "S3 Scalable Storage in the Cloud" service highlighted with a red box. To the right, there is a separate window or tab showing the AWS Application Discovery Service (ADS) interface.

- Select S3 from the menu, which will direct you to the page where you can create an S3 Bucket.

The screenshot shows the Amazon S3 service page. On the left, the navigation menu includes "Amazon S3" and "General purpose buckets". The main content area displays an "Account snapshot - updated every 24 hours" and a table for "General purpose buckets" (21). The table includes columns for Name, AWS Region, IAM Access Analyzer, and Creation date. A "Create bucket" button is located at the bottom right of the table.

- Search your s3 target bucket and click it to open it

- On the s3 target bucket page, you should be able to see the customers.csv file migrated successfully if all of the previous steps are set up correctly.

Step4: Trigger Crawler to create table in Glue Catalog

In this step we will execute the glue crawler to create a table in the Glue catalog for the data analysis

- Open your AWS Management Console and navigate to AWS Glue by searching for "Glue" in the Search Bar.

- Click on "AWS Glue," and you'll land on the page where click the three line menu button on the left top corner and then click on the Crawlers button

- On the crawlers page, select the already created crawler and click the Run button

- Once you click the Run button, your crawler will be in the running phase. Wait for the crawler to finish

- Click the crawler to check the status of the execution. If everything is setup correctly it should be completed successfully

The screenshot shows the AWS Glue Crawler properties page for a crawler named 's3-to-s3-crawler'. The crawler has the following configuration:

- Crawler properties:**
 - Name: s3-to-s3-crawler
 - IAM role: s3-to-s3-role
 - Description: -
 - Database: default
 - State: STOPPING
 - Table prefix: -
- Crawler runs (1):**
 - Start time (UTC): December 19, 2024 at 11:37:20
 - End time (UTC): December 19, 2024 at 11:38:20
 - Status: Completed (highlighted with a red box)
 - DPU hours: 01 min
 - Table changes: 1 table change, 0 partition changes

- Once the crawler completed successfully, Click on the Databases button on the left hand side menu

The screenshot shows the AWS Glue Databases page. The 'Tables' section is selected, and the 'default' database is highlighted with a red box. The page displays the following information:

- Databases (1):**
 - A database is a set of associated table definitions, organized into a logical group.
 - Filter databases: default
 - Location URI: -
 - Created on (UTC): September 8, 2023 at 15:32:56

- Once you are on the databases page, click on the already existing default database to open it

The screenshot shows the AWS Glue Databases page. The 'Tables' section is selected, and the 'default' table is highlighted with a red box. The page displays the following information:

- Databases (1):**
 - A database is a set of associated table definitions, organized into a logical group.
 - Filter databases: default
 - Location URI: -
 - Created on (UTC): September 8, 2023 at 15:32:56

- On the tables page, you should be able to see a table with your S3 target bucket name.

The screenshot shows the AWS Glue interface for managing databases. On the left, there's a navigation sidebar with options like 'AWS Glue', 'Getting started', 'ETL jobs', 'Visual ETL', 'Notebooks', 'Job run monitoring', 'Data Catalog tables', 'Data connections', 'Workflows (orchestration)', 'Zero-ETL integrations', 'Data Catalog', 'Tables', 'Stream schema registries', 'Schemas', 'Connections', 'Crawlers', 'Classifiers', 'Catalog settings', 'Data Integration and ETL', and 'Legacy pages'. The main area is titled 'default' and shows 'Database properties' for a database named 'default'. It includes fields for 'Name' (default), 'Description' (empty), 'Location' (empty), and 'Created on (UTC)' (September 8, 2023 at 15:32:56). Below this is a table section titled 'Tables (1)'. A table is listed with columns: Name, Database, Location, Classification, Deprecated, View data, Data quality, and Column statistics. The table row for 'de_s3_to_s3_target_bucket' is selected, highlighted with a red border. The 'View data' button in the 'Table data' column is also highlighted with a red border.

- Click on the Table data option to see this table data in the AWS Athena

This screenshot is similar to the previous one but focuses on the 'Table data' section. The 'Table data' button in the 'Table data' column of the 'Tables (1)' table is highlighted with a red border. The rest of the interface and table data are identical to the first screenshot.

- Click on the Proceed button to open this table in the AWS Athena

This screenshot shows a confirmation dialog box over the AWS Glue interface. The dialog says: 'You will be taken to Athena to preview data, and you will be charged separately for Athena queries.' It has two buttons: 'Close' and 'Proceed'. The 'Proceed' button is highlighted with a red border. The background shows the same 'Tables (1)' section as the previous screenshots, with the 'Table data' button also highlighted with a red border.

- Once you clicked on Proceed, AWS will open the Athena Query editor with a simple select statement with your table data

Amazon Athena > Query editor tabs

Editor Recent queries Saved queries Settings Workgroup primary

Data

Data source: AwsDataCatalog catalogue: None Database: default

Tables and views **Create**

Tables (1) de_s3_to_s3_target_bucket Views (0)

Query 7 | X | Query 8 | X | Query 9 | X | Query 10 | X | **Query 11 | X**

```
1 | SELECT * FROM "AwsDataCatalog"."default"."de_s3_to_s3_target_bucket" limit 10;
```

SQL Ln 1, Col 1 Run again Explain Cancel Clear Create Reuse query results up to 60 minutes ago

Query results Query status

Completed Time in queue: 112 ms Run time: 490 ms Data scanned: 111.48 KB

Results (10) Copy Download results

| # | customerid | firstname | lastname | company | city | country | phone | email |
|---|-------------------|-----------|----------|----------------------------|------------------|------------------|-----------------------|--------------------------------|
| 1 | dE014d010c7ab0c | Andrew | Goodman | Stewart-Flynn | Rowlandberg | Macao | 846-790-4623x4715 | marieyates@gomez-spencer.info |
| 2 | 2B54172c8b65eC3 | Alvin | Lane | Terry Proctor and Lawrence | Bethside | Papua New Guinea | 124-597-8652x05682 | alexandra86@mccoy.com |
| 3 | d794D484988d2e2 | Jenna | Harding | Bailey Group | Moniquemouth | China | (355)987-3085x3780 | justincurtis@spierce.org |
| 4 | 3b5Aa4aCc68f5Be | Fernando | Ford | Moss-Maxwell | Leeborough | Macao | (047)752-3122 | adeleon@hubbard.org |
| 5 | D60df52aa2ae41E | Kara | Wood | Mccarthy-Kelley | Port Jacksonland | Nepal | +1-360-693-4419x19272 | jesus90@robertson.info |
| 6 | 8aaa5d0CE9ee311 | Marissa | Gamble | Cherry and Sons | Webertown | Sudan | 001-645-334-5514x0786 | katieallison@leonard.com |
| 7 | 73822Ac8A43DD1A | Julie | Cooley | Yu Norman and Sharp | West Sandra | Japan | +1-675-243-7422x9177 | priscillas@stephens.info |
| 8 | DE94f4C099303311b | Lauren | Villa | French Travers and Hemley | New Volunda | Fiji | 081-226-17976x47 | coldumber@william-caldwell.com |

VOILA! YOU HAVE SUCCESSFULLY TESTED YOUR S3-S3 CROSS REGION MIGRATION PROJECT AND ABLE TO QUERY THE DATA IN THE AWS ATHENA