AWS PROJECT

This is step to step guide to building Blog/Audio Converter using Amazon Polly.

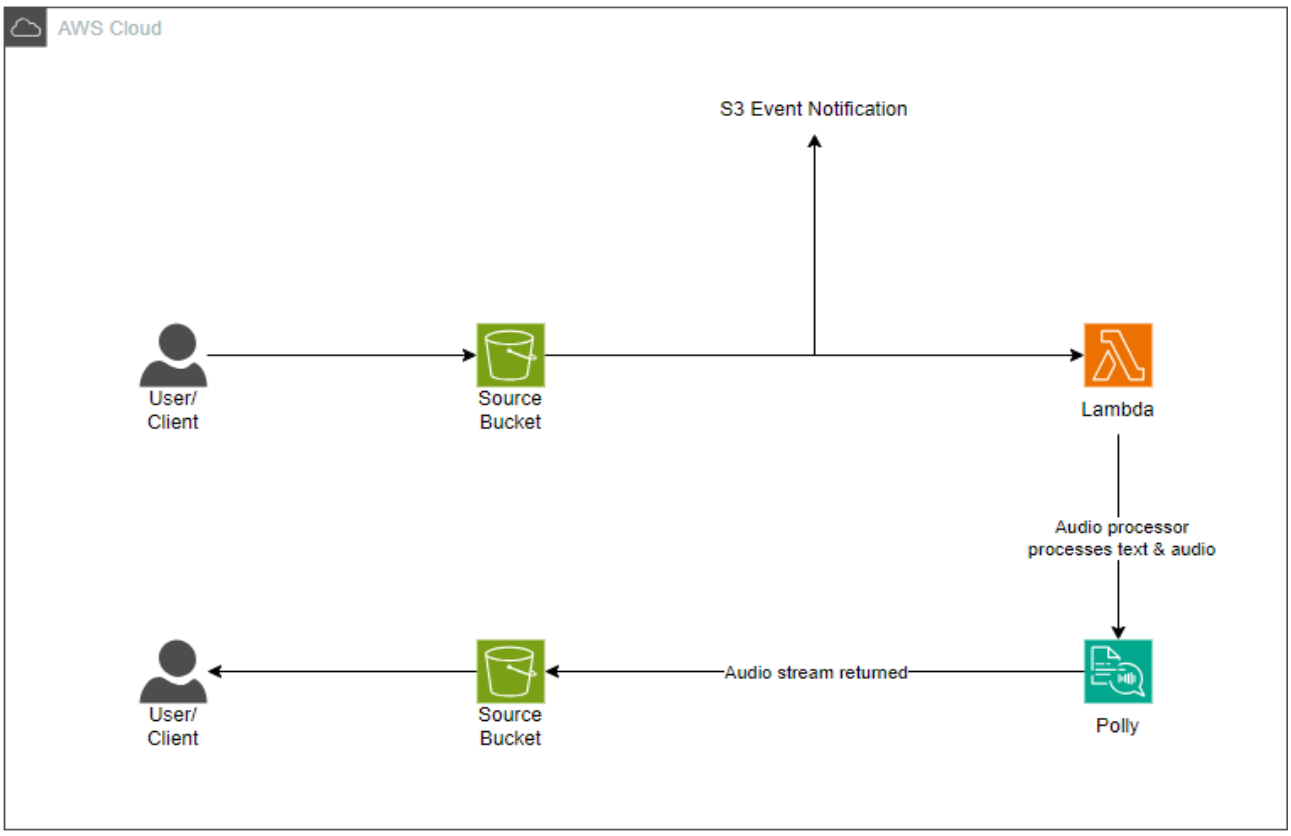
Project Description.

This project leverages some AWS services to convert text files such as blogs, articles, newsletters or book excerpts into speech. This is particularly useful for creating audio versions of written content, making it accessible to a wider audience, including those who prefer listening over reading.

Amazon Polly has the following use cases;

- Learning: Enables users to listen to educational materials, enhancing learning experiences.
- Content Accessibility: Provides audio versions of written content for visually impaired users.
- Content Distribution: Offers an additional medium for content consumption, increasing engagement.
- Convenience: Allows users to listen to articles or books while multitasking, such as during commutes or workouts.

Project Architecture.



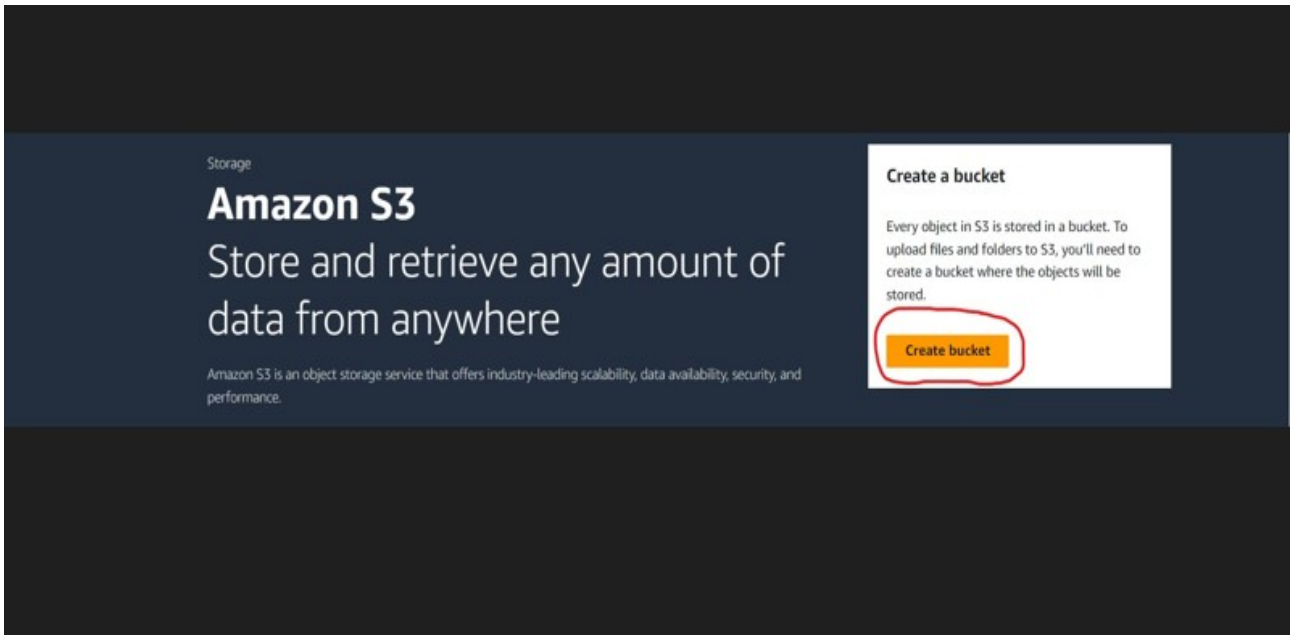
Steps

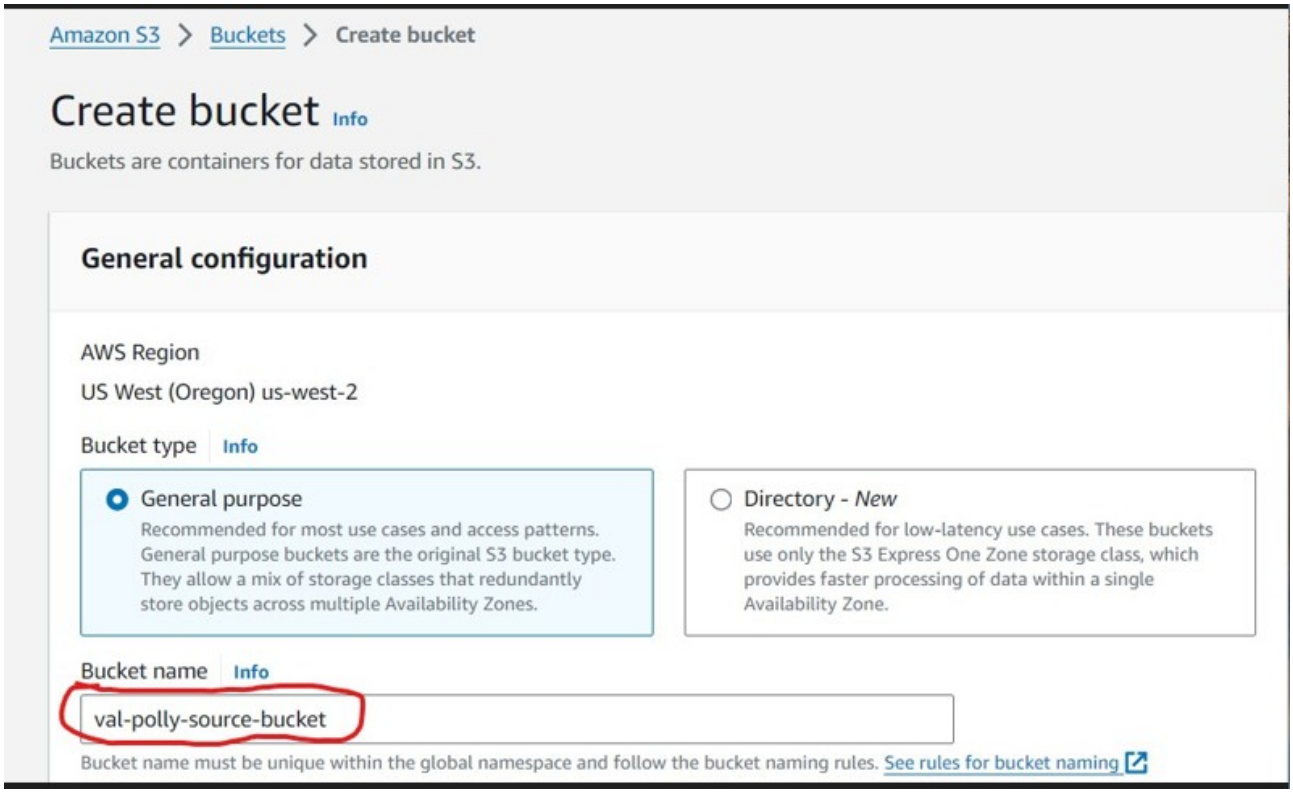
In my AWS account, I have to create two S3 Buckets;
Source Bucket (Bucket name: val-polly-source-bucket) and Destination Bucket (Bucket name: val-polly-destination-bucket).

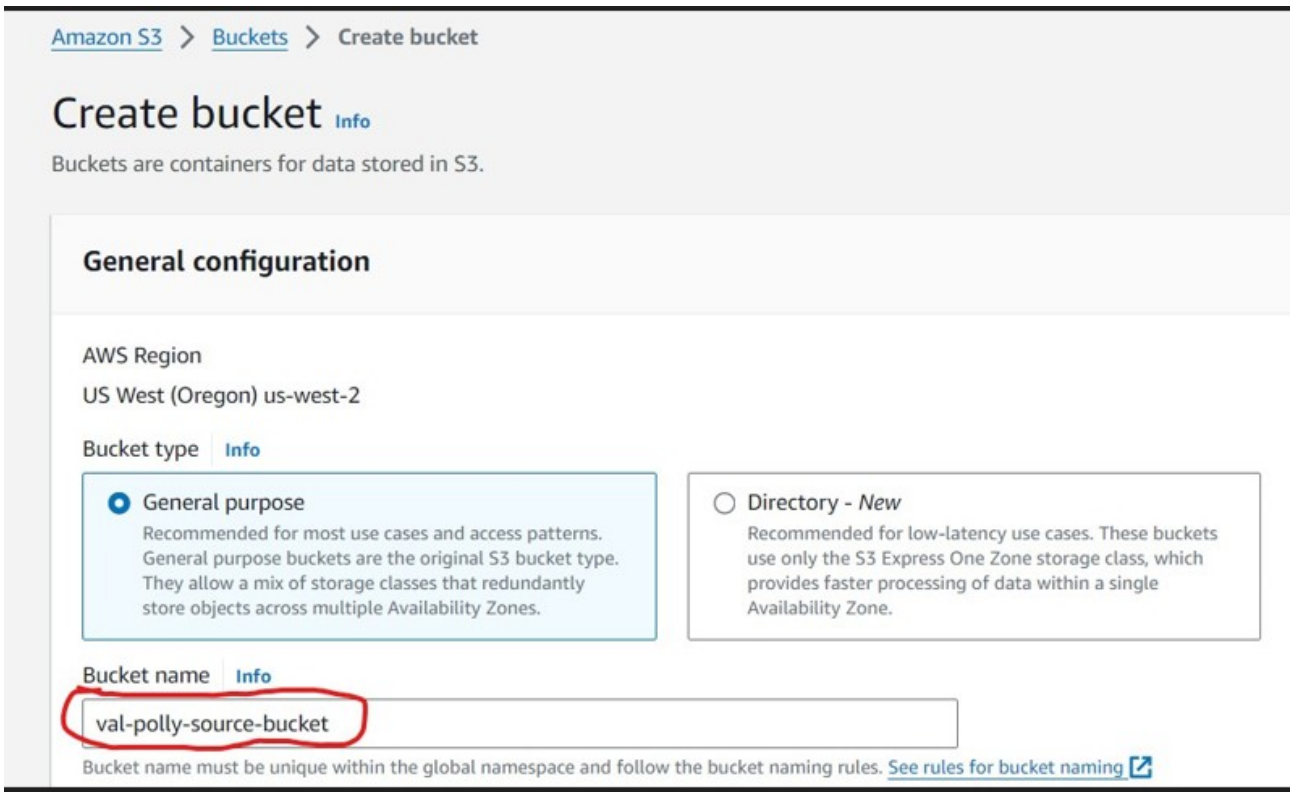
Process

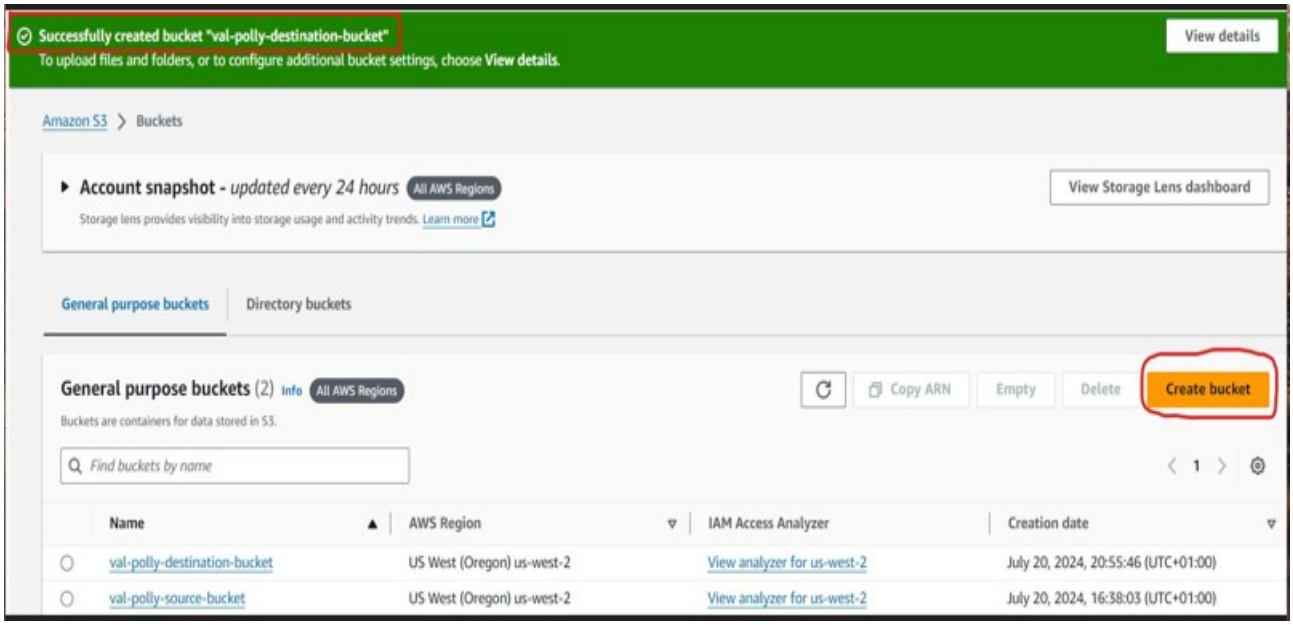
- Create a Source and Destination Bucket.
- The Source Bucket name will be (val-polly-source-bucket).
- Go to (Create Bucket).
- Enter the name (val-polly-source-bucket).
- Leave everything the same then click (Create Bucket).
- Then go back to the (S3 console), and repeat the same procedure above to create another Bucket and call it (Destination Bucket).
- Enter the name (val-polly-destination-bucket).
- Scroll down and click on (Create Bucket).
- Destination Bucket is created.

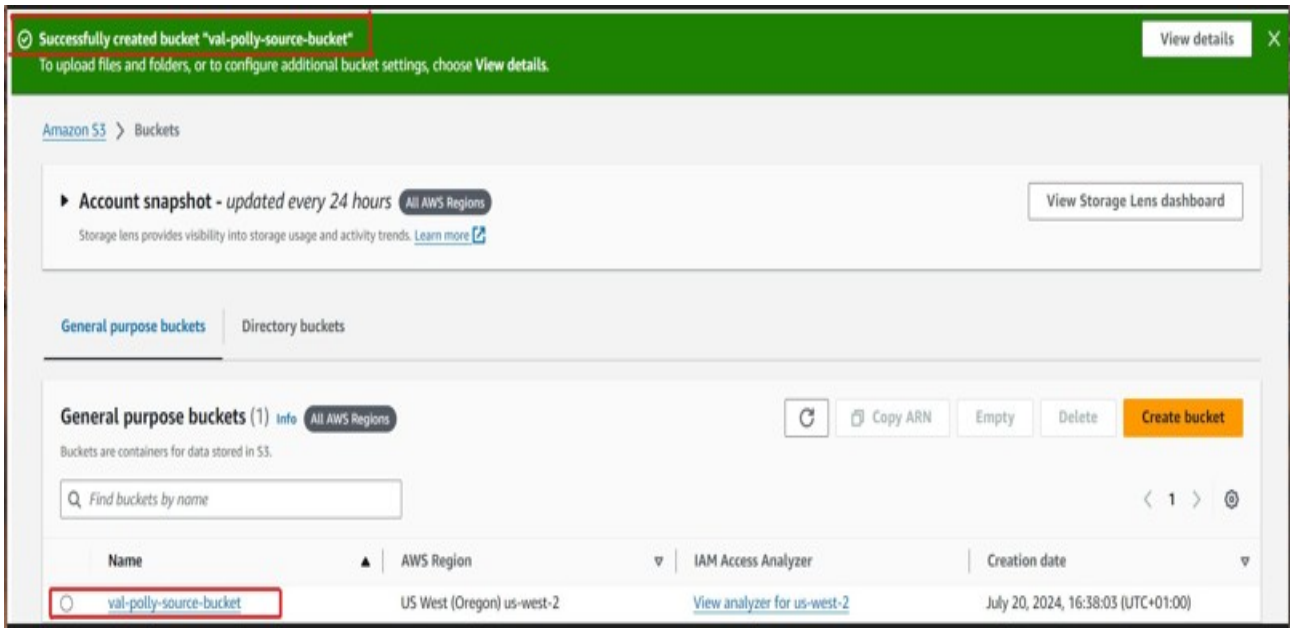
These are the shown in the following pictures below;

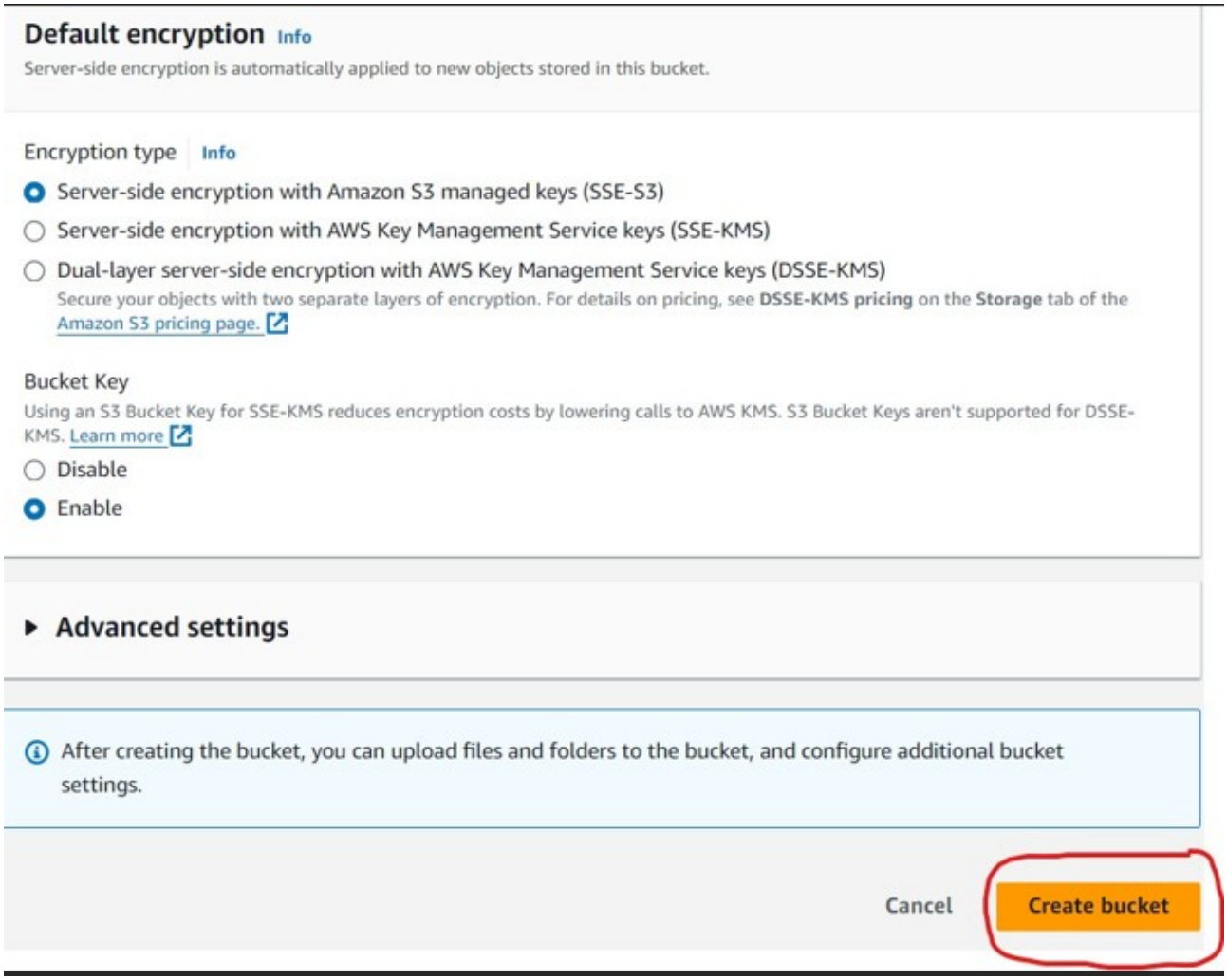



Amazon S3 > Buckets > Create bucket
Create bucket Info
Buckets are containers for data stored in S3.
General configuration
AWS Region
US West (Oregon) us-west-2
Bucket type Info
General purpose
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.
Directory - New
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.
Bucket name Info
val-polly-source-bucket
Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming


Amazon S3 > Buckets > Create bucket
Create bucket Info
Buckets are containers for data stored in S3.
General configuration
AWS Region
US West (Oregon) us-west-2
Bucket type Info
General purpose
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.
Directory - New
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.
Bucket name Info
val-polly-source-bucket
Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming


Successfully created bucket "val-polly-destination-bucket"
To upload files and folders, or to configure additional bucket settings, choose View details.
View details
Amazon S3 > Buckets
Account snapshot - updated every 24 hours
All AWS Regions
Storage lens provides visibility into storage usage and activity trends. Learn more
View Storage Lens dashboard
General purpose buckets
Directory buckets
General purpose buckets (2) Info
All AWS Regions
Buckets are containers for data stored in S3.
Copy ARN
Empty
Delete
Create bucket
Find buckets by name
1
Name
AWS Region
IAM Access Analyzer
Creation date
val-polly-destination-bucket
US West (Oregon) us-west-2
View analyzer for us-west-2
July 20, 2024, 20:55:46 (UTC+01:00)
val-polly-source-bucket
US West (Oregon) us-west-2
View analyzer for us-west-2
July 20, 2024, 16:38:03 (UTC+01:00)


Successfully created bucket "val-polly-source-bucket"
To upload files and folders, or to configure additional bucket settings, choose View details.
View details
Amazon S3 > Buckets
Account snapshot - updated every 24 hours
All AWS Regions
Storage lens provides visibility into storage usage and activity trends. Learn more
View Storage Lens dashboard
General purpose buckets
Directory buckets
General purpose buckets (1) Info
All AWS Regions
Buckets are containers for data stored in S3.
Copy ARN
Empty
Delete
Create bucket
Find buckets by name
1
Name
AWS Region
IAM Access Analyzer
Creation date
val-polly-source-bucket
US West (Oregon) us-west-2
View analyzer for us-west-2
July 20, 2024, 16:38:03 (UTC+01:00)



The next step is to create an IAM Policy. Going back to the architecture, the Lambda function requires access to the (Source Bucket), (Destination Bucket) and (Amazon Polly). It needs permission to access these resources.

The next step is to create an IAM Role for the Lambda, this Role should have access to my (Source Bucket), (Amazon Polly) and my (Destination Bucket).

Steps to create IAM Roles

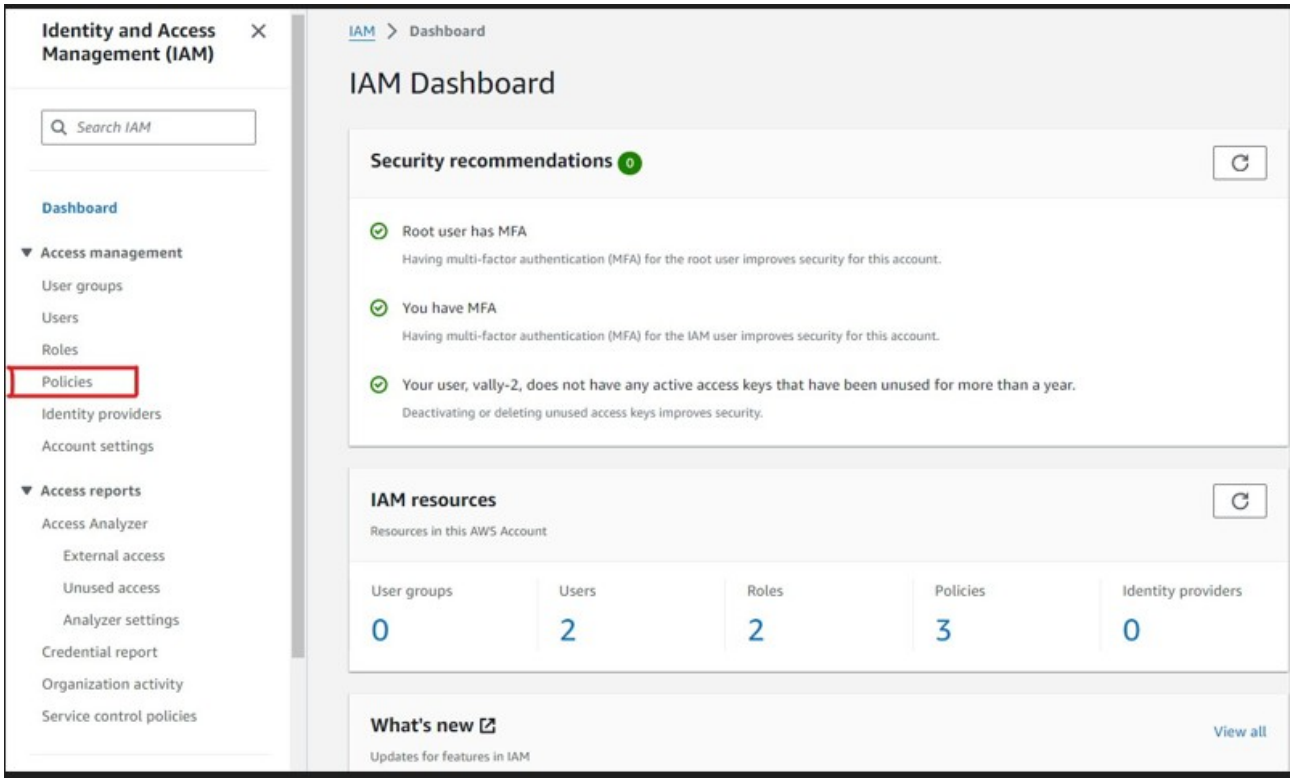
- First create an (IAM Role).
- Click on Roles.
- Click on (Create Role).
- Search for (Lambda) and click (Next).
- On the (permissions) search for (AWS Lambda) and click (AWSLambda-FullAccess).
- Click (Next).
- Role Name (lambdarole).

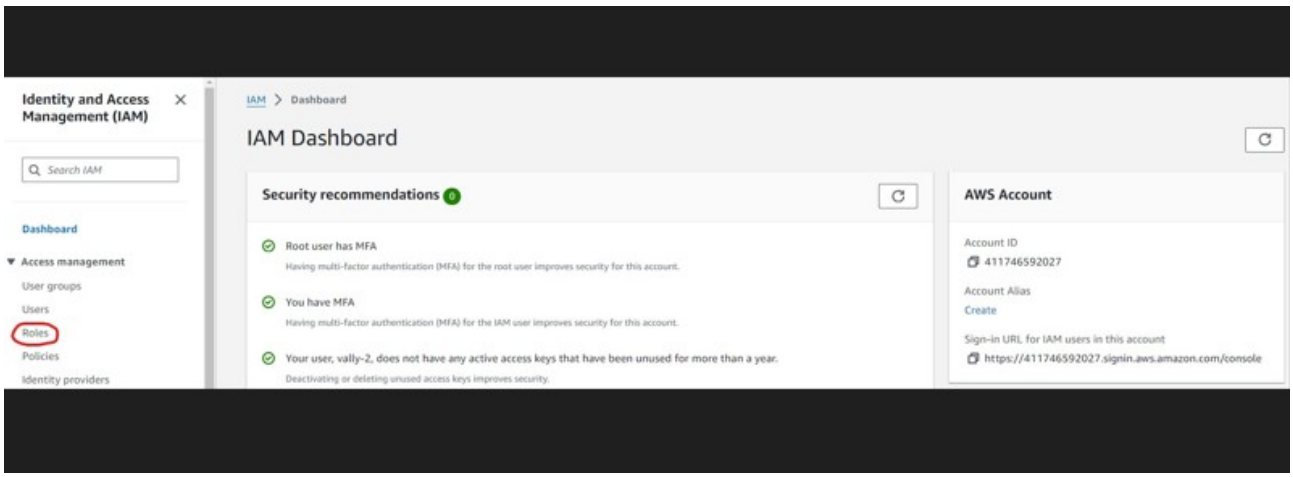
Steps to create IAM Policies

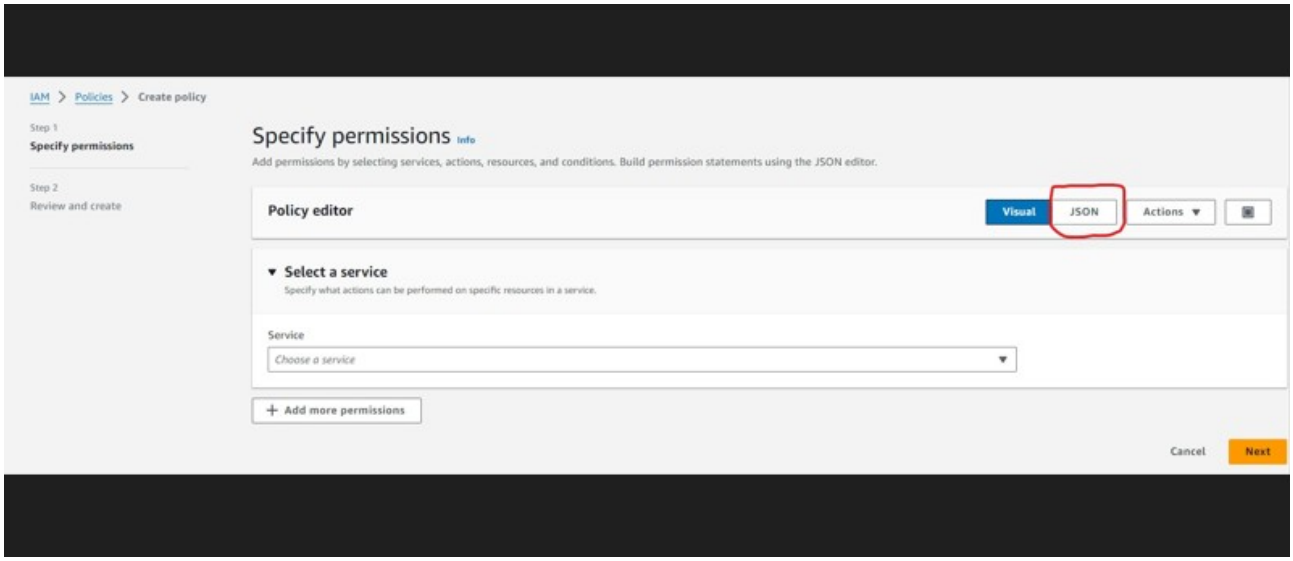
- Go to (IAM) in the console.
- Click on (Policies).
- Search for (AmazonS3fullaccess).
- Click on it.
- Click (Create Policy).
- Then click on (JSON) and copy the (JSON).
- IAM Policy created.

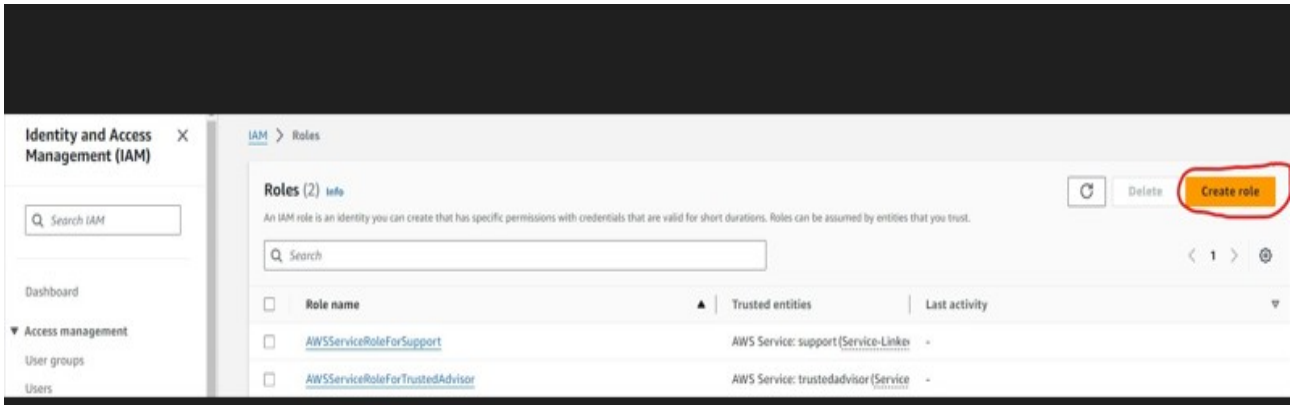
I attached the (polly-lambda-policy) to the Lambdarole

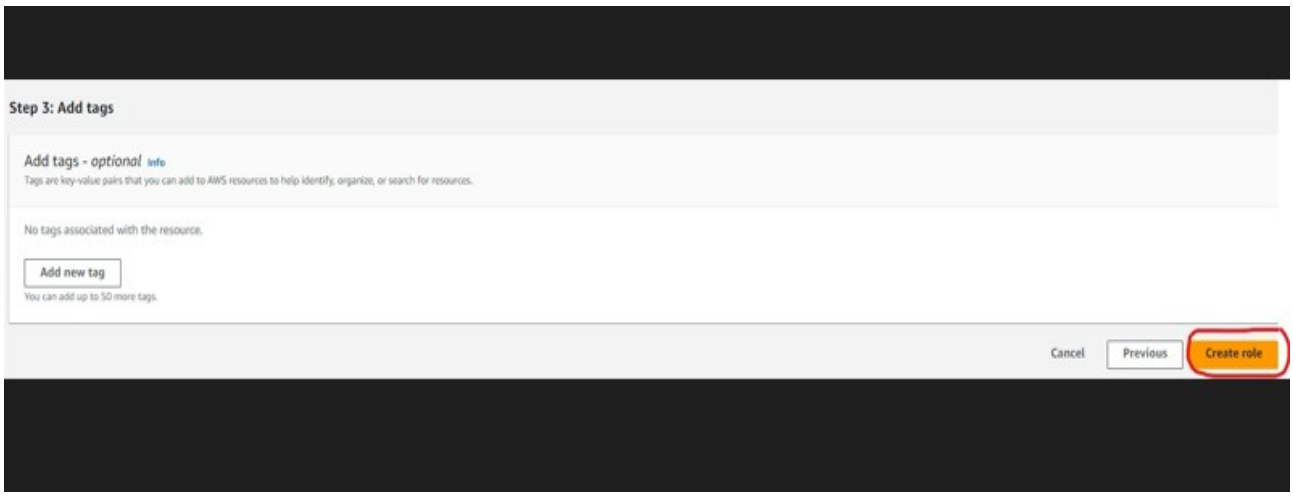
These are shown in the pictures below;

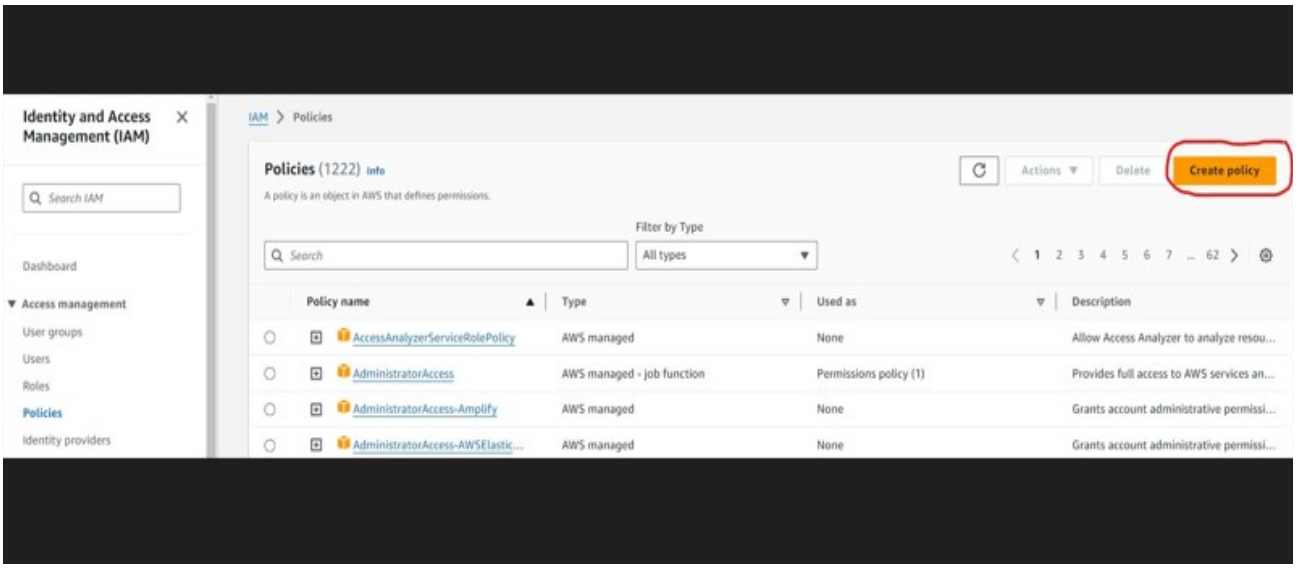


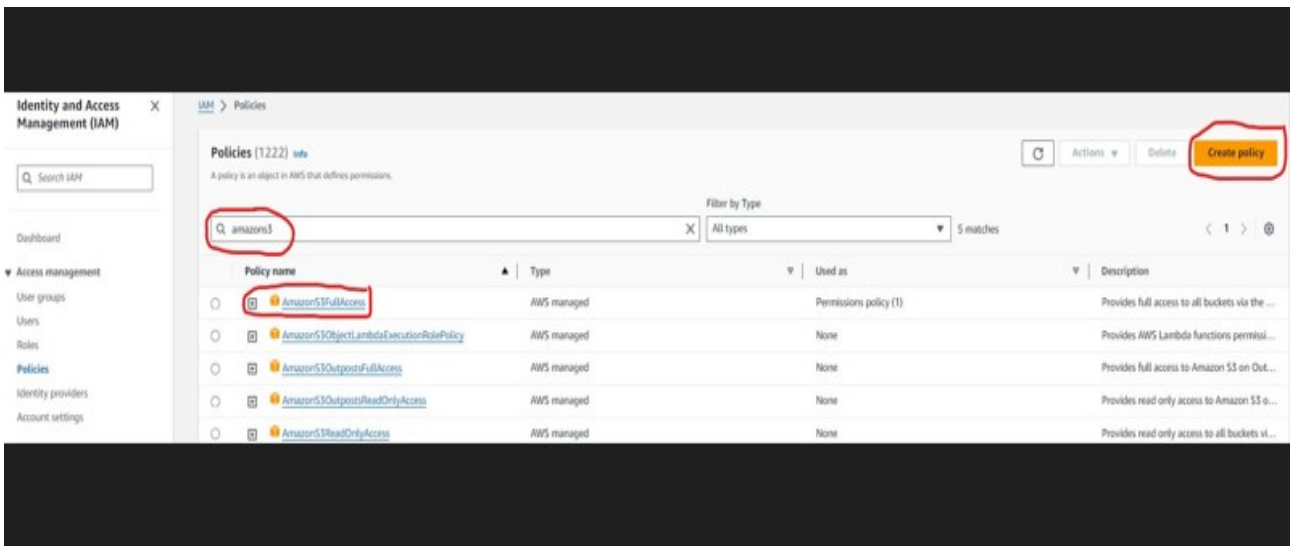

Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
IAM > Dashboard
IAM Dashboard
Security recommendations 0
Root user has MFA
Having multi-factor authentication (MFA) for the root user improves security for this account.
You have MFA
Having multi-factor authentication (MFA) for the IAM user improves security for this account.
Your user, vally-2, does not have any active access keys that have been unused for more than a year.
Deactivating or deleting unused access keys improves security.
AWS Account
Account ID
411746592027
Account Alias
Create
Sign-in URL for IAM users in this account
https://411746592027.signin.aws.amazon.com/console

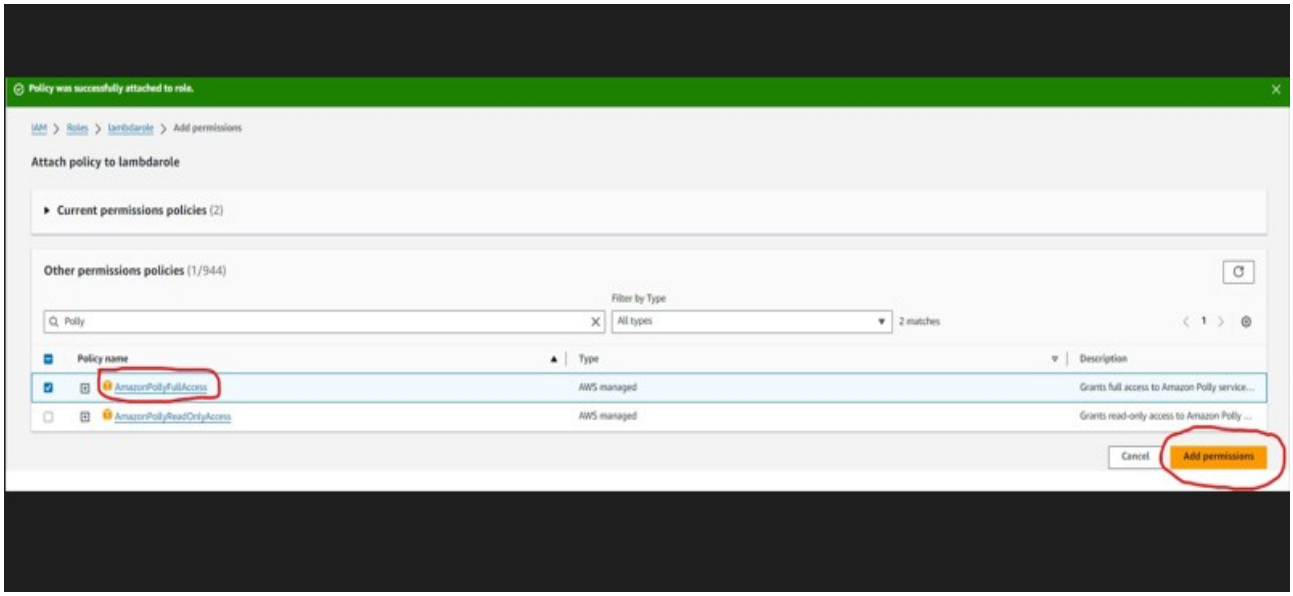

IAM > Policies > Create policy
Step 1
Specify permissions
Step 2
Review and create
Specify permissions Info
Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.
Policy editor
Visual
JSON
Actions
Select a service
Specify what actions can be performed on specific resources in a service.
Service
Choose a service
Add more permissions
Cancel
Next


Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
IAM > Roles
Roles (2) Info
An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.
Delete
Create role
Search
Role name
Trusted entities
Last activity
AWSServiceRoleForSupport
AWS Service: support (Service-Linke
AWSServiceRoleForTrustedAdvisor
AWS Service: trustedadvisor (Service


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 the resource.
Add new tag
You can add up to 50 more tags.
Cancel
Previous
Create role


Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
IAM > Policies
Policies (1222) Info
A policy is an object in AWS that defines permissions.
Actions
Delete
Create policy
Filter by Type
Search
All types
1 2 3 4 5 6 7 ... 62
Policy name
Type
Used as
Description
AccessAnalyzerServiceRolePolicy
AWS managed
None
Allow Access Analyzer to analyze resou...
AdministratorAccess
AWS managed - job function
Permissions policy (1)
Provides full access to AWS services an...
AdministratorAccess-Amplify
AWS managed
None
Grants account administrative permissi...
AdministratorAccess-AWSElastic...
AWS managed
None
Grants account administrative permissi...


Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
IAM > Policies
Policies (1222) Info
A policy is an object in AWS that defines permissions.
Actions
Delete
Create policy
Filter by Type
amazons3
All types
5 matches
Policy name
Type
Used as
Description
AmazonS3FullAccess
AWS managed
Permissions policy (1)
Provides full access to all buckets via the ...
AmazonS3ObjectLambdaExecutionRolePolicy
AWS managed
None
Provides AWS Lambda functions permissi...
AmazonS3OutpostsFullAccess
AWS managed
None
Provides full access to Amazon S3 on Out...
AmazonS3OutpostsReadOnlyAccess
AWS managed
None
Provides read only access to Amazon S3 o...
AmazonS3ReadOnlyAccess
AWS managed
None
Provides read only access to all buckets vi...

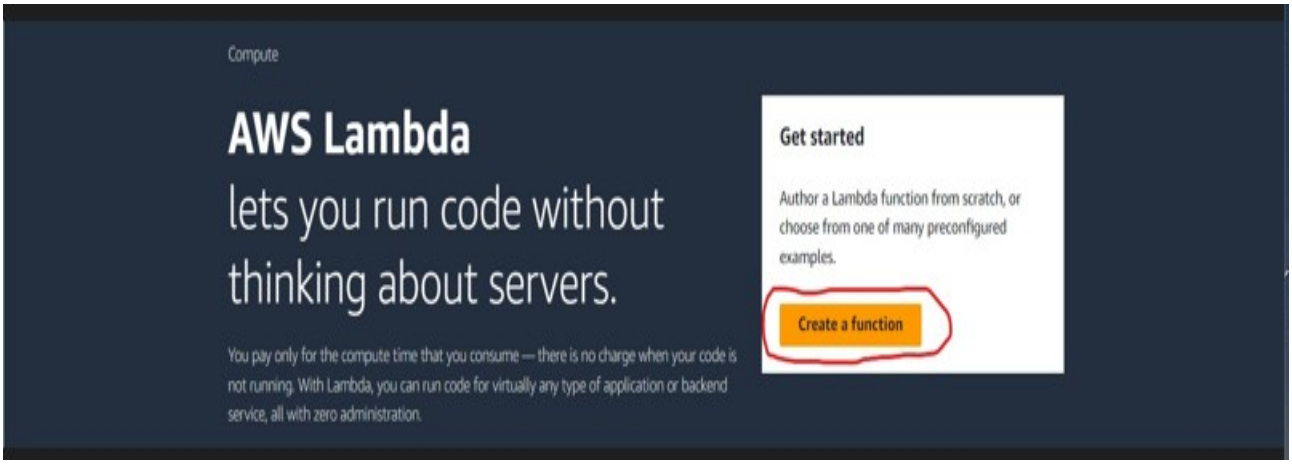


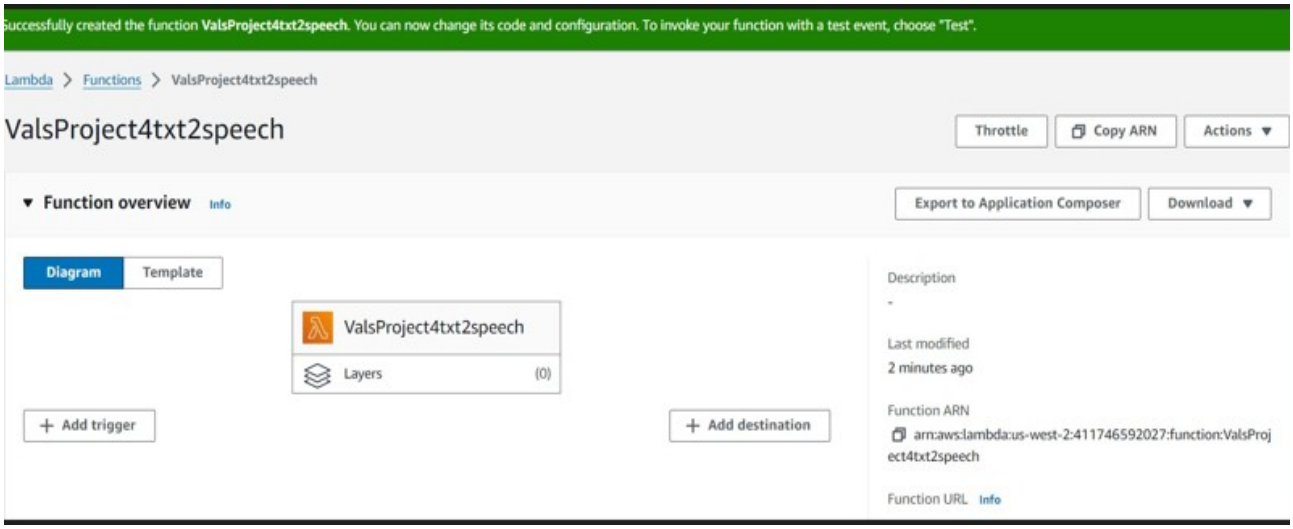
The next step is to configure the Lambda function with the following steps.

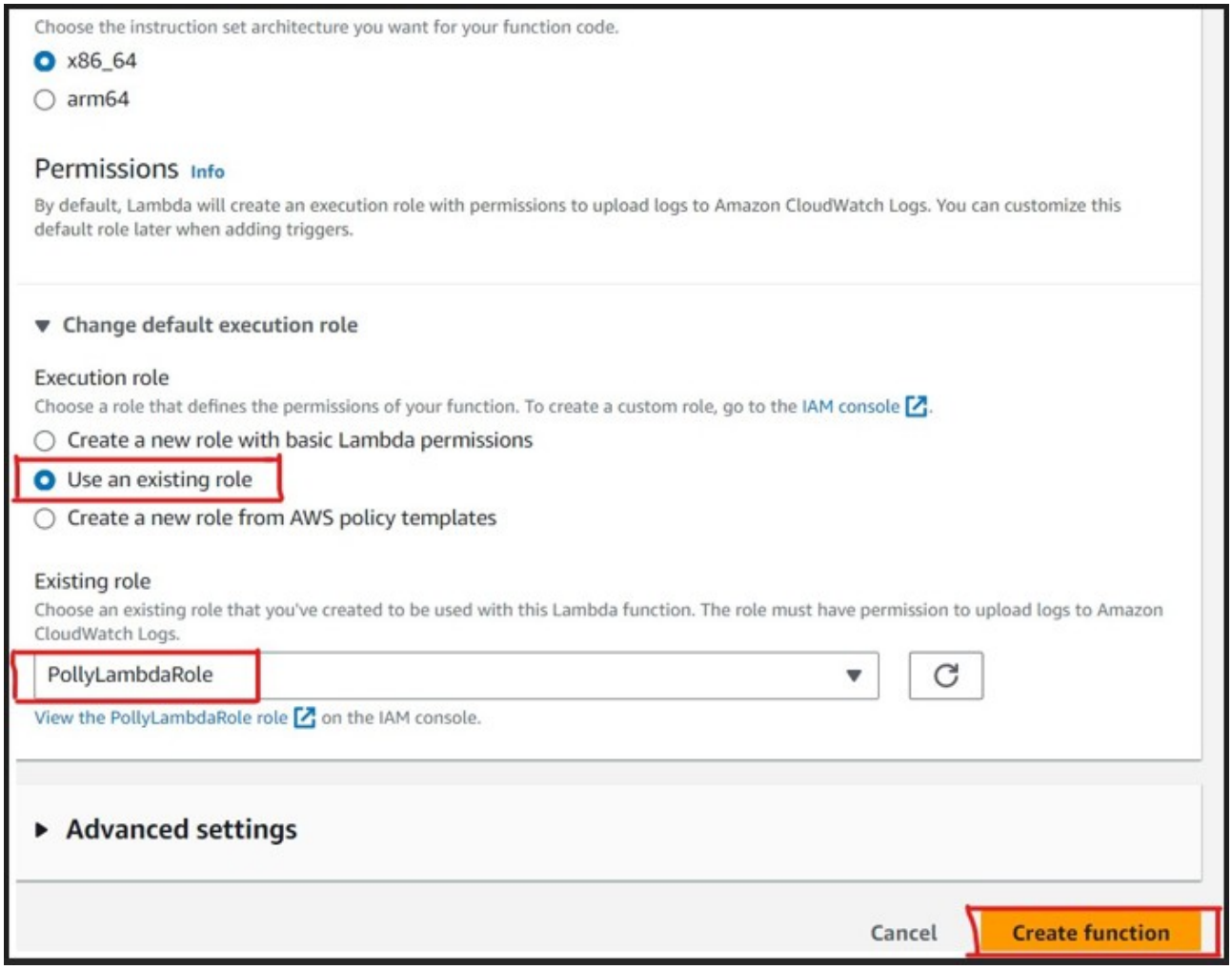
Steps

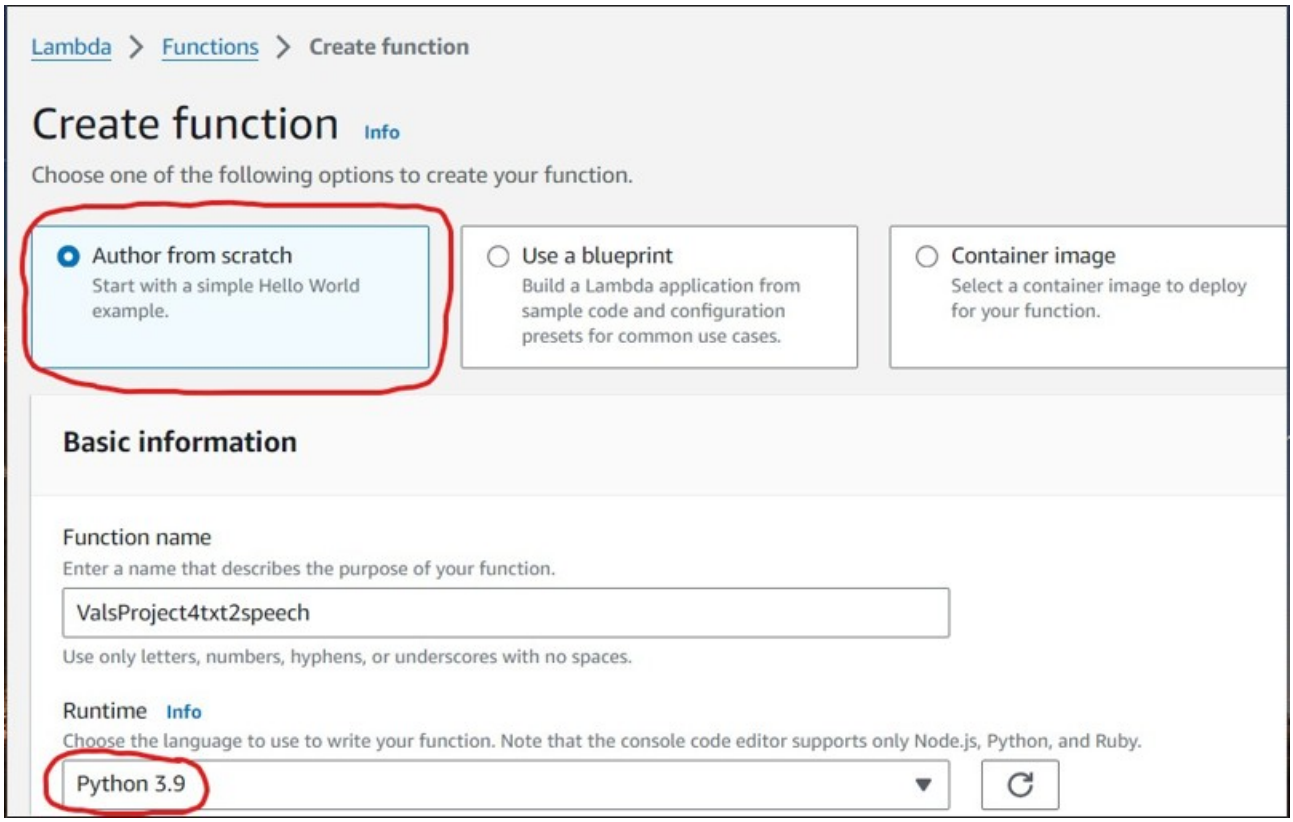
- Create and configure the Lambda function; Function name (Valsproject4txt2speech).
- Go to Lambda and click on (Create a function).
- Leave it on (Author from Scratch).
- Enter the function name as (Valsproject4txt2speech).
- Using (python 3.8) as the (Run time).
- Go to (Permissions) and expand (Change default, Execution Role).
- Then click on (Use an existing Role) which is (pollyLambdaRole).
- Then click on (Create function).

These are shown in the pictures below;




Successfully created the function ValsProject4txt2speech. You can now change its code and configuration. To invoke your function with a test event, choose "Test".
Lambda > Functions > ValsProject4txt2speech
ValsProject4txt2speech
Throttle
Copy ARN
Actions
Function overview Info
Export to Application Composer
Download
Diagram
Template
ValsProject4txt2speech
Layers (0)
Add trigger
Add destination
Description
-
Last modified
2 minutes ago
Function ARN
arn:aws:lambda:us-west-2:411746592027:function:ValsProject4txt2speech
Function URL Info


Choose the instruction set architecture you want for your function code.
x86_64
arm64
Permissions Info
By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.
Change default execution role
Execution role
Choose a role that defines the permissions of your function. To create a custom role, go to the IAM console.
Create a new role with basic Lambda permissions
Use an existing role
Create a new role from AWS policy templates
Existing role
Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs.
PollyLambdaRole
View the PollyLambdaRole role on the IAM console.
Advanced settings
Cancel
Create function

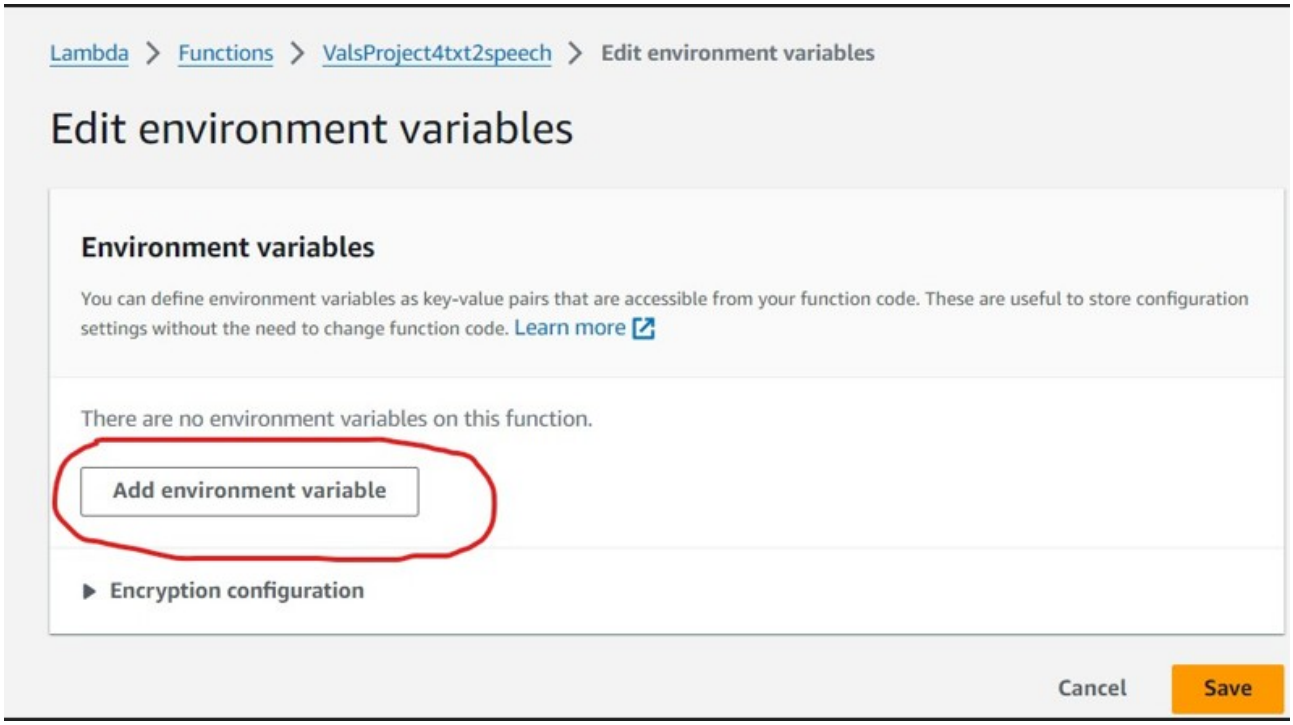


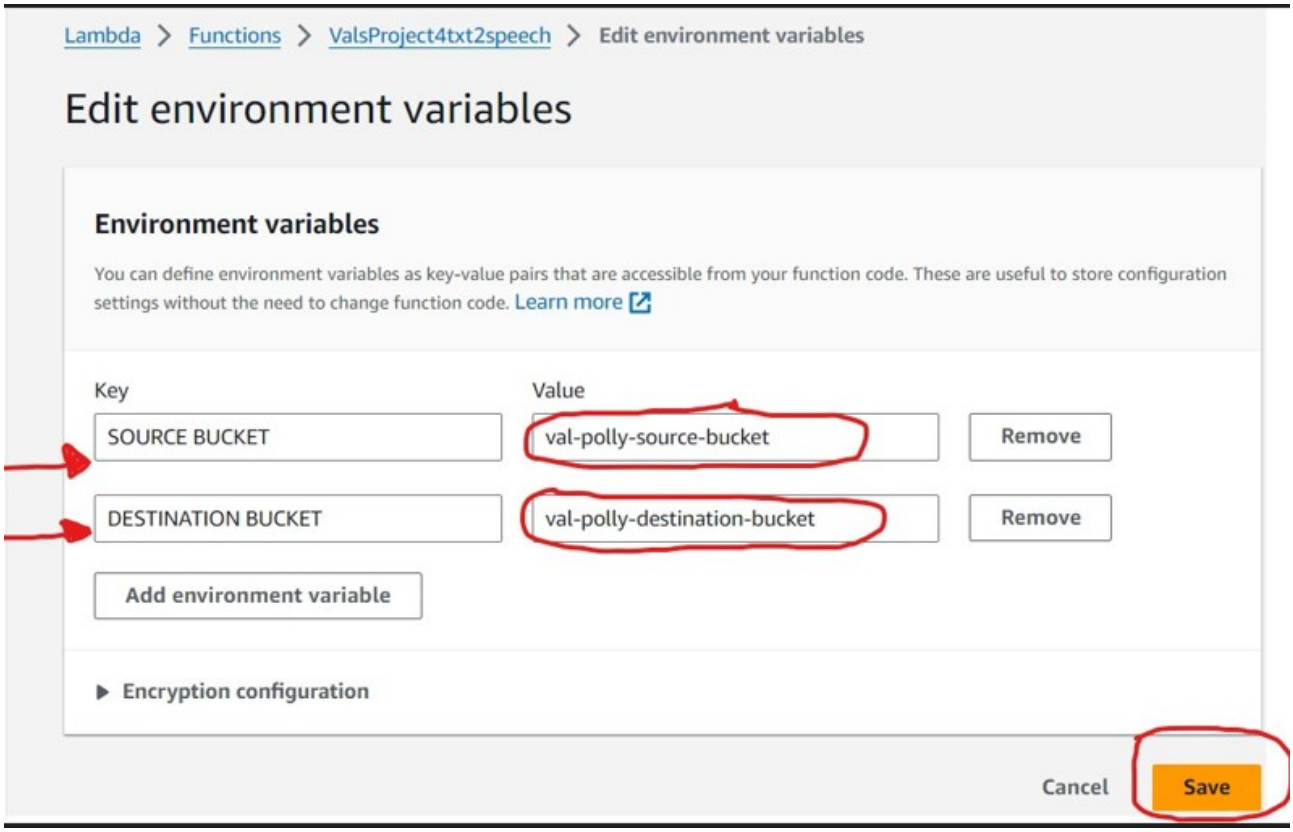
The next step is to configure the (Source Bucket) and the (Destination Bucket) as the (Environment variables).

Steps

- Go to (Configuration) under the (Lambda function overview).
- And to the left select (Environmental variables).
- Click on (Environmental variables).
- Then click on (Edit).
- Click on (Add environment variable).
- Enter the Source Bucket (val-polly-source-bucket) and the Destination Bucket (val-polly-destination-bucket) under value.
- Then click on (Save).

These are shown in the pictures below;





Moving on the next stage of this project which is to Configure an (S3 Event Notification).

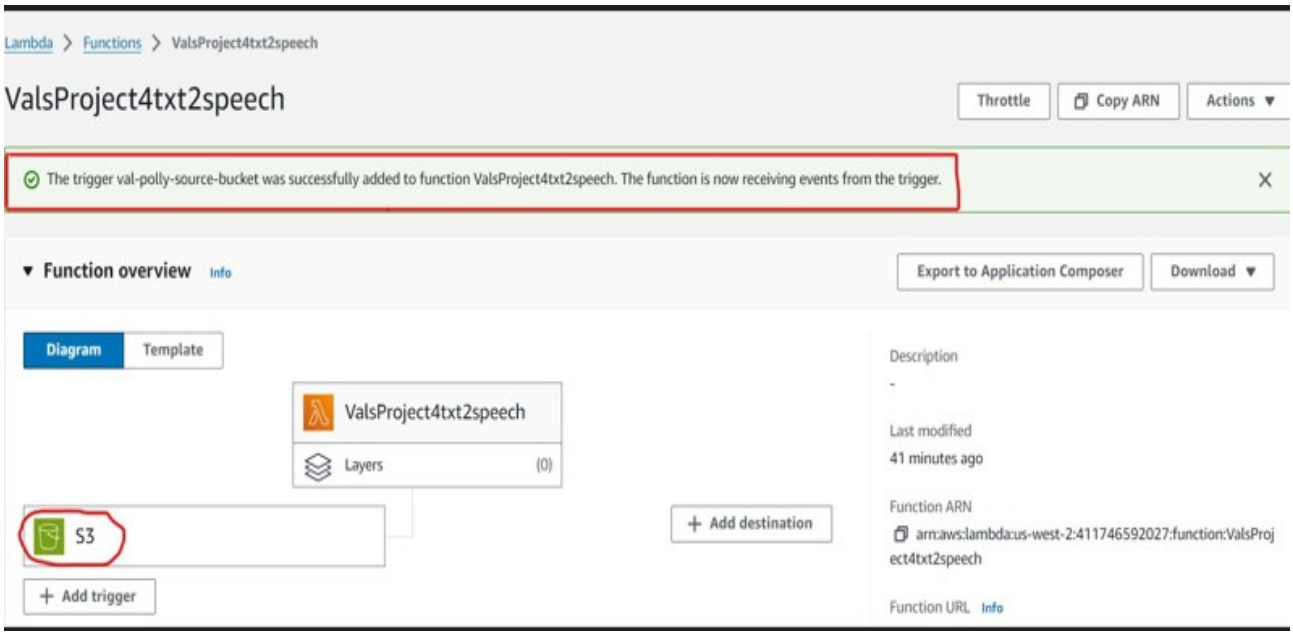
In this stage, I will set up an Event Notification in the (Source S3 Bucket) to trigger the (Lambda function) on new (Object creation events) with the (.txt) suffix.

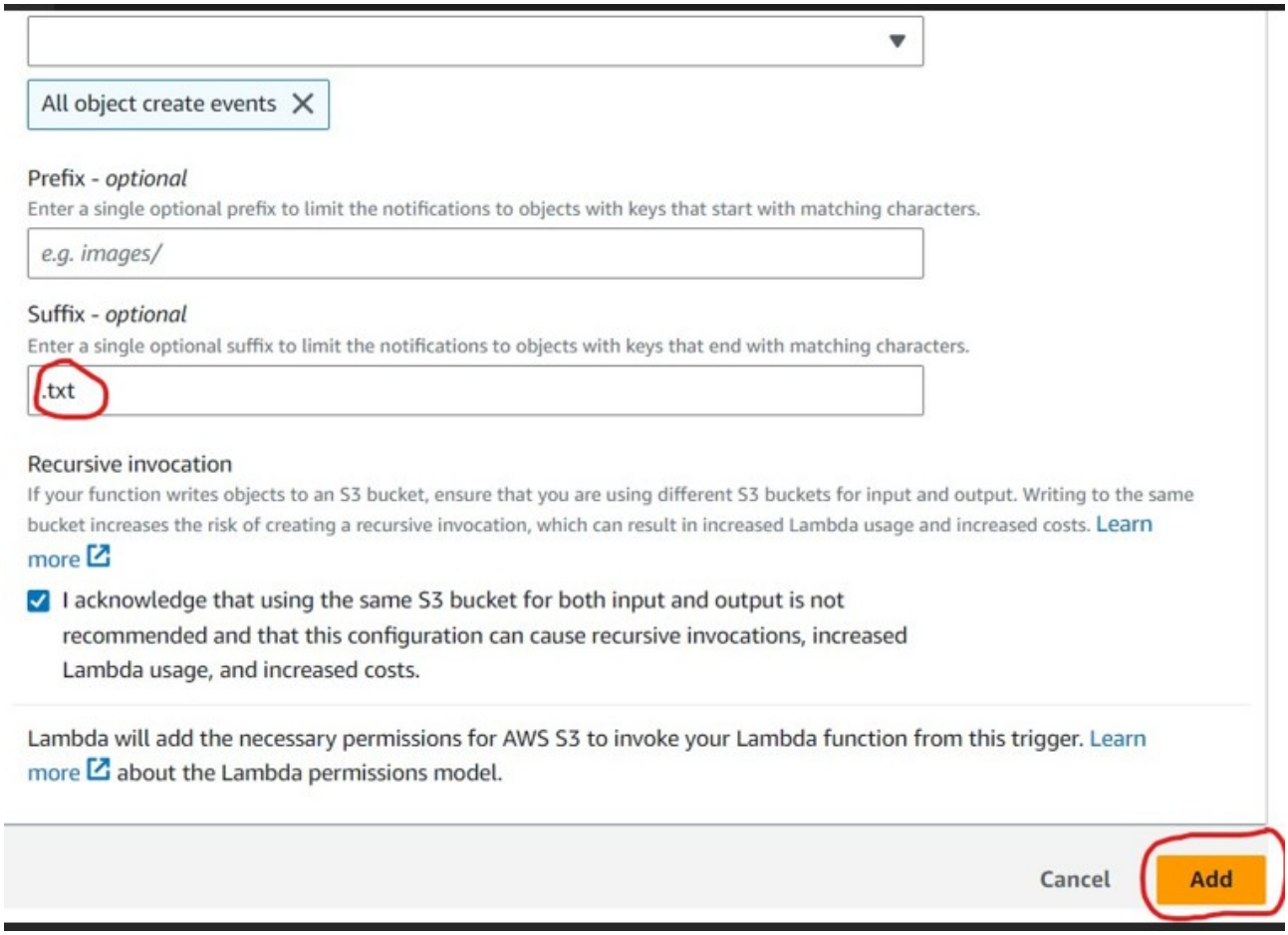
Steps

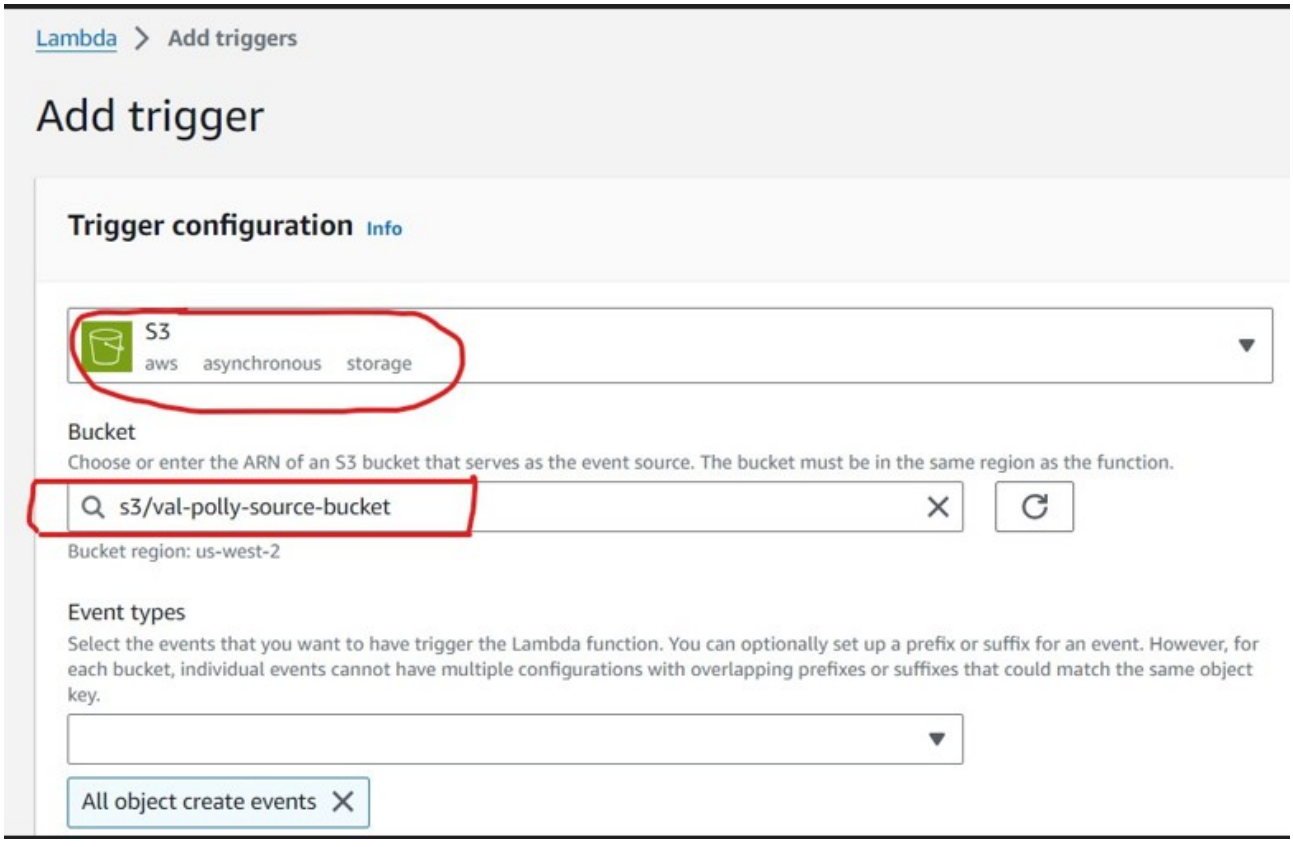
- Go to the Lambda function.
- Click on (Add trigger).
- Select a source, which will be the (S3 Bucket).
- Select the Source Bucket.
- Then leaving the (prefix) empty as I don't have any and for the (suffix), use a (.txt) file.
- Then click on the acknowledgement and click on (Add) at the bottom of the page.
- Then the S3 trigger will be automatically added.

This completes this process.

These are illustrated in the pictures below;




All object create events
Prefix - optional
Enter a single optional prefix to limit the notifications to objects with keys that start with matching characters.
e.g. images/
Suffix - optional
Enter a single optional suffix to limit the notifications to objects with keys that end with matching characters.
.txt
Recursive invocation
If your function writes objects to an S3 bucket, ensure that you are using different S3 buckets for input and output. Writing to the same bucket increases the risk of creating a recursive invocation, which can result in increased Lambda usage and increased costs. Learn more
I acknowledge that using the same S3 bucket for both input and output is not recommended and that this configuration can cause recursive invocations, increased Lambda usage, and increased costs.
Lambda will add the necessary permissions for AWS S3 to invoke your Lambda function from this trigger. Learn more about the Lambda permissions model.
Cancel
Add

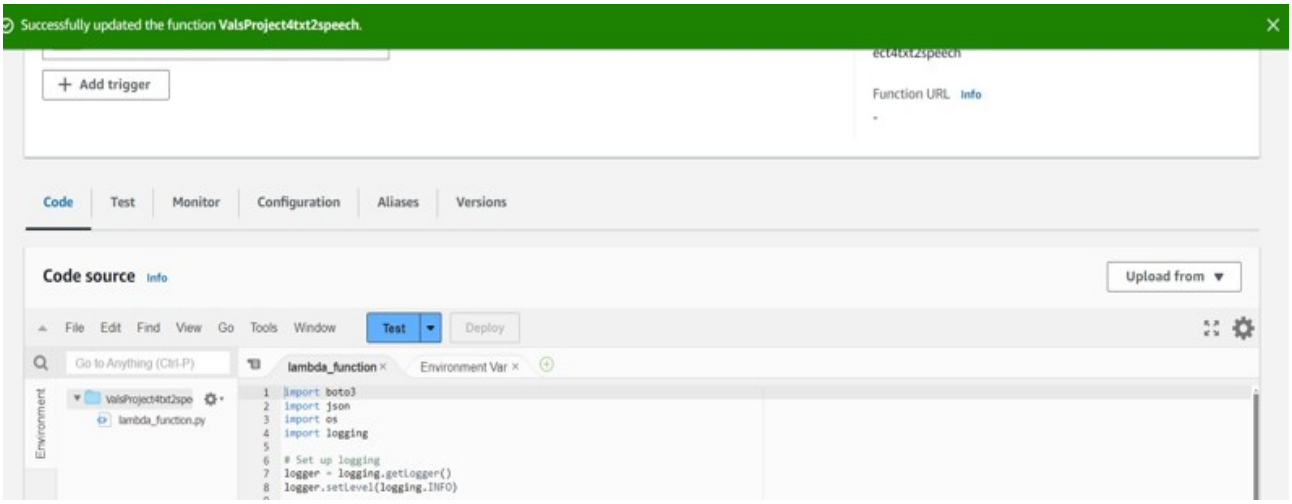

Lambda > Add triggers
Add trigger
Trigger configuration Info
S3
aws asynchronous storage
Bucket
Choose or enter the ARN of an S3 bucket that serves as the event source. The bucket must be in the same region as the function.
s3/val-polly-source-bucket
Bucket region: us-west-2
Event types
Select the events that you want to have trigger the Lambda function. You can optionally set up a prefix or suffix for an event. However, for each bucket, individual events cannot have multiple configurations with overlapping prefixes or suffixes that could match the same object key.
All object create events

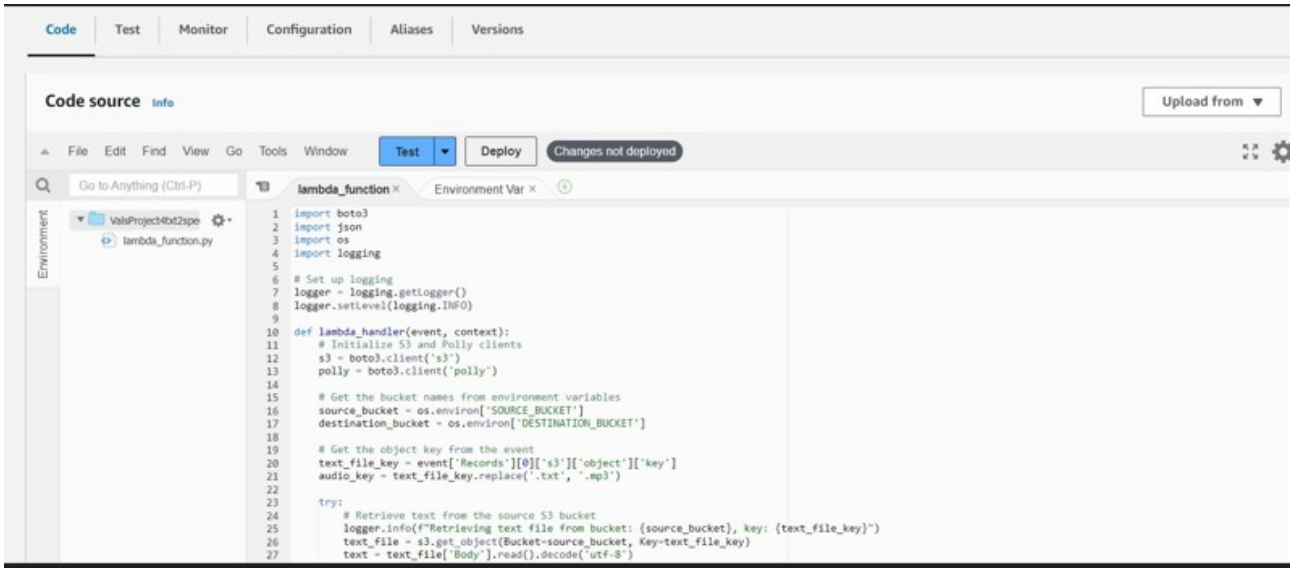
The next step is to write the Lambda code.

Steps

- Using the function (Text to speech function.py).
- Using (boto 3) which is the (SDK) for AWS.
- Go back to the function
- And Enter/(Paste) the new code from the repository (Text to speech.py) on GitHub and (Deploy) it.
- Next the (Lambda Code) is successfully deployed.
- Going back to the (Source S3 Bucket), and add a (.txt) file.

These are shown in the pictures below;





The last thing is to test the system.

Testing the System

- Now going back to the S3 Source Bucket (val-polly-source-bucket), opening it to show that it's empty.
- Also, open the Destination Bucket (val-polly-destination-bucket) to show that it's empty.
- I have to (Upload) a (.txt file) into the Source Bucket.
- Go to the (vals-polly-source-bucket) and click (Upload).
- Go back to the lambda function and (Test) the Code.
- Then go to the Destination bucket to check to see the MP3 file transferred into it through the Code after testing it.
- Test the MP3 file to listen to it through (Amazon Polly).
- Test competed.

The picture below shows it;