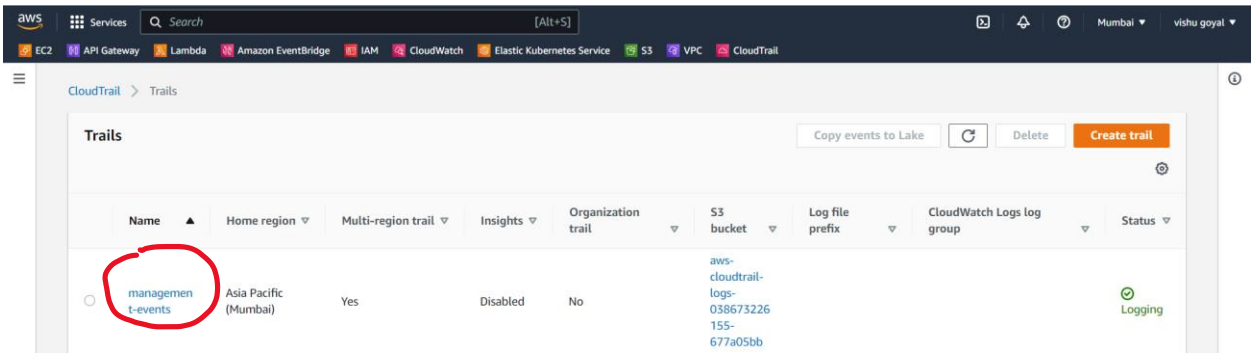
Notify When S3 bucket made public and private in nature after created

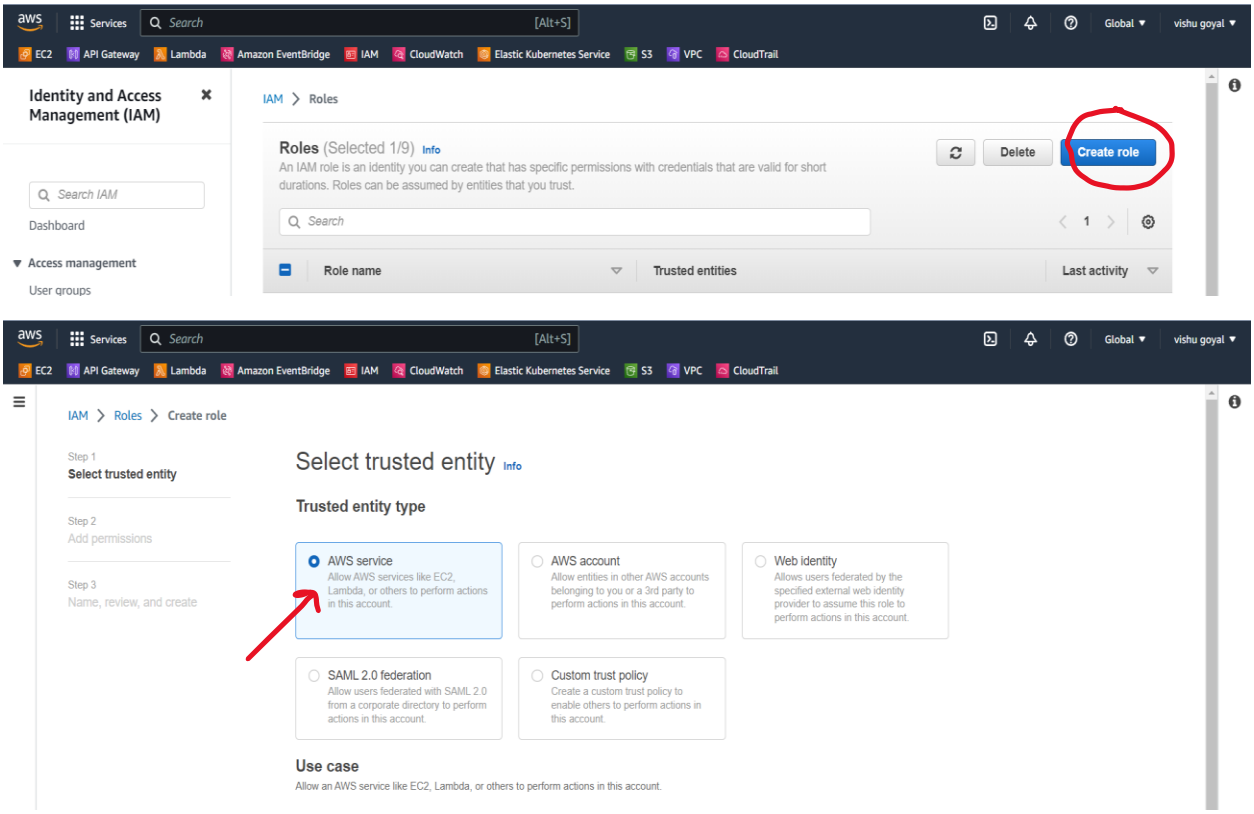
Step 1: Make sure you have create a cloud Trail bucket for storing a logs whatever you occur in aws account.

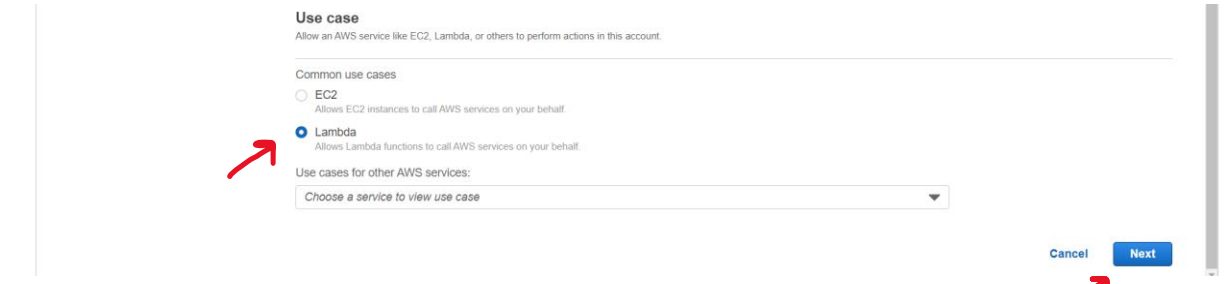


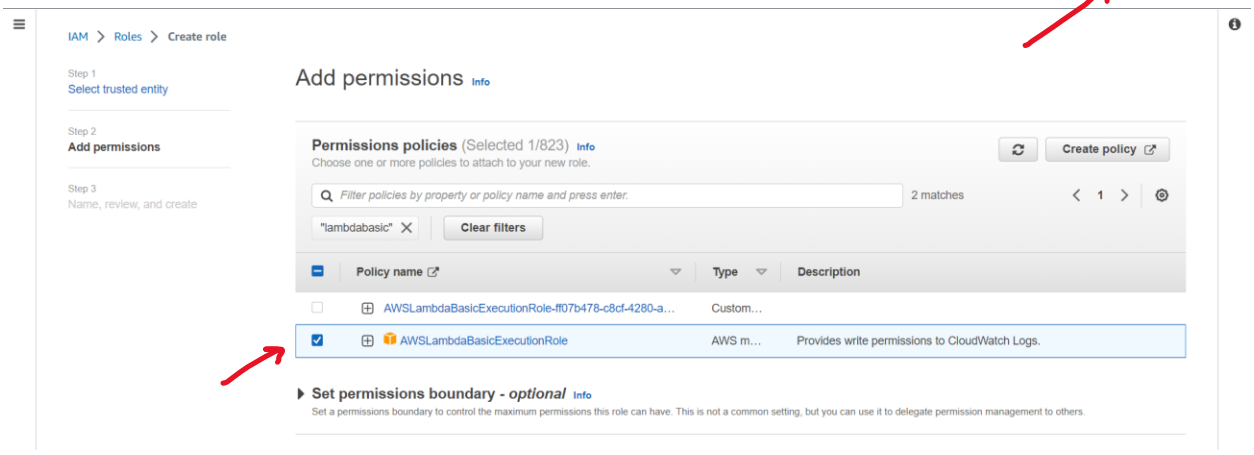
Its looks like that the above screenshot.

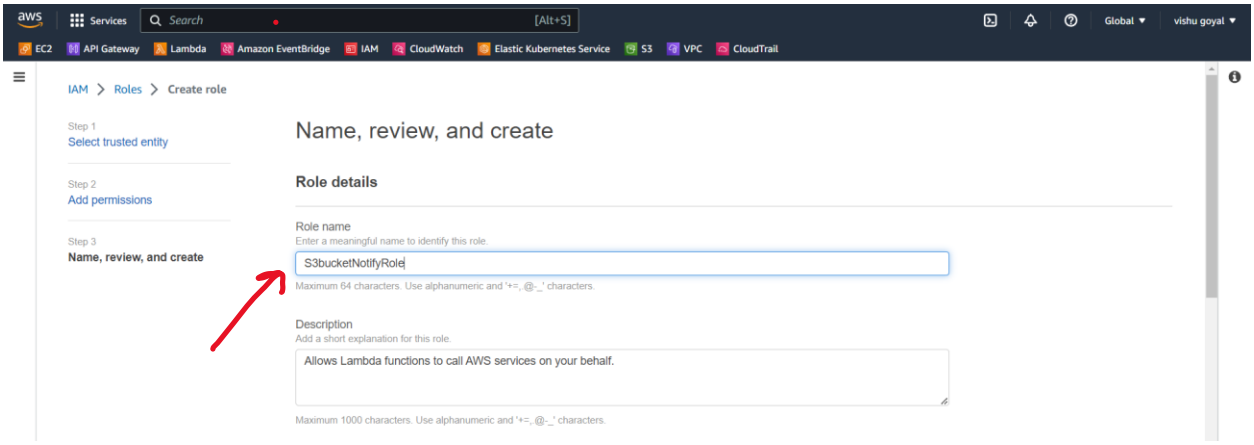
Step 2:

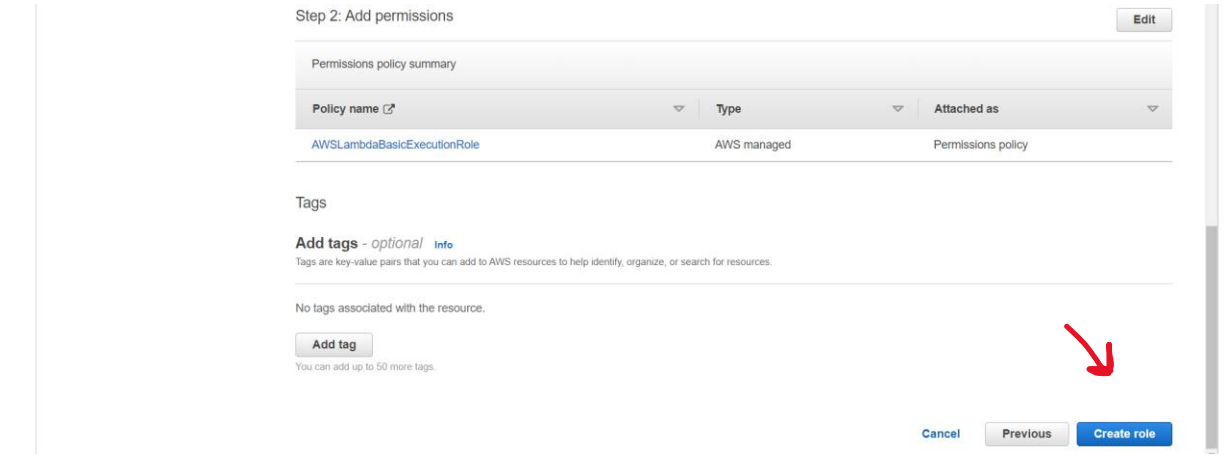
Now you have to create a IAM role for lambda function



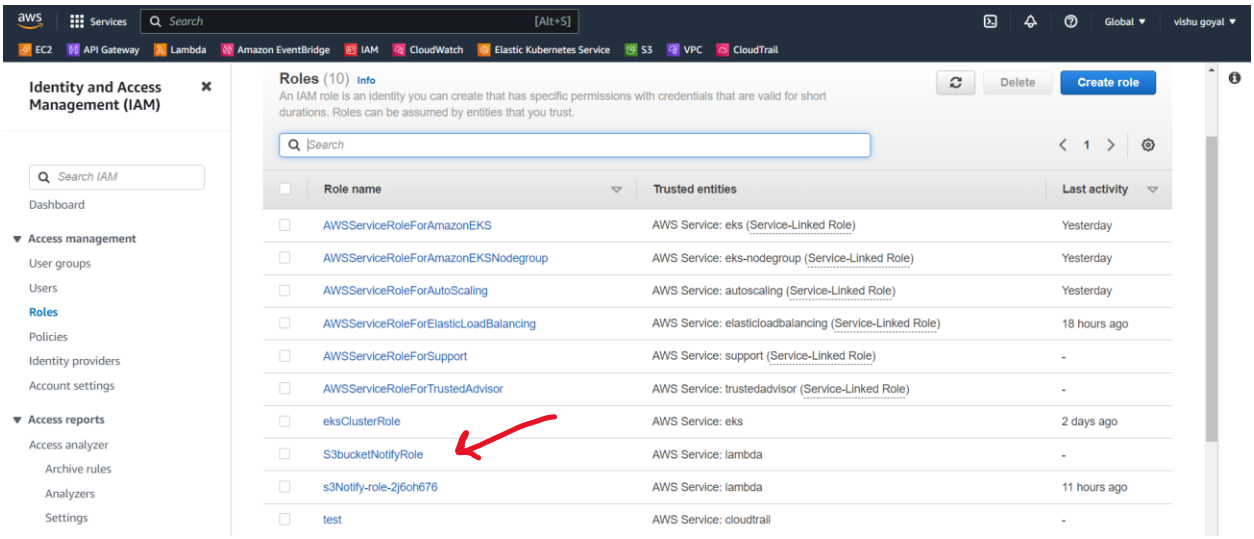

Use case
Allow an AWS service like EC2, Lambda, or others to perform actions in this account.
Common use cases
EC2
Allows EC2 instances to call AWS services on your behalf.
Lambda
Allows Lambda functions to call AWS services on your behalf.
Use cases for other AWS services:
Choose a service to view use case
Cancel
Next


IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Add permissions Info
Permissions policies (Selected 1/823) Info
Choose one or more policies to attach to your new role.
Create policy
Filter policies by property or policy name and press enter.
2 matches
"lambdabasic"
Clear filters
Policy name
Type
Description
AWSLambdaBasicExecutionRole-ff07b478-c8cf-4280-a…
Custom…
AWSLambdaBasicExecutionRole
AWS m…
Provides write permissions to CloudWatch Logs.
Set permissions boundary - optional Info
Set a permissions boundary to control the maximum permissions this role can have. This is not a common setting, but you can use it to delegate permission management to others.

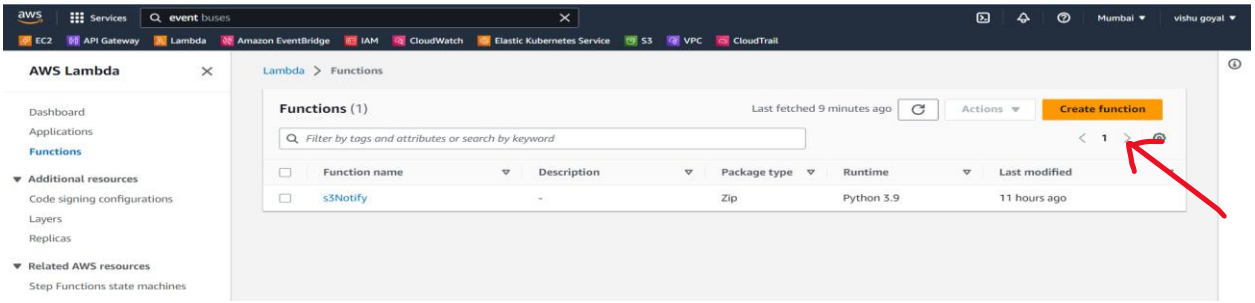

aws
Services
Search
[Alt+S]
Global
vishu goyal
EC2
API Gateway
Lambda
Amazon EventBridge
IAM
CloudWatch
Elastic Kubernetes Service
S3
VPC
CloudTrail
IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Name, review, and create
Role details
Role name
Enter a meaningful name to identify this role.
S3bucketNotifyRole
Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.
Description
Add a short explanation for this role.
Allows Lambda functions to call AWS services on your behalf.
Maximum 1000 characters. Use alphanumeric and '+=,.@-_' characters.

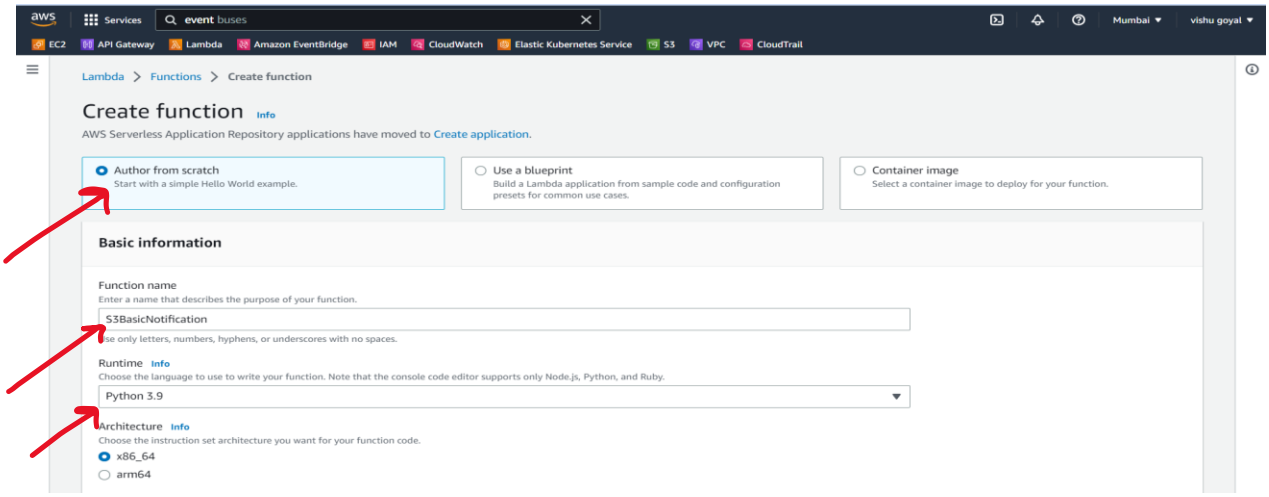

Step 2: Add permissions
Edit
Permissions policy summary
Policy name
Type
Attached as
AWSLambdaBasicExecutionRole
AWS managed
Permissions policy
Tags
Add tags - optional Info
Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.
No tags associated with the resource.
Add tag
You can add up to 50 more tags.
Cancel
Previous
Create role

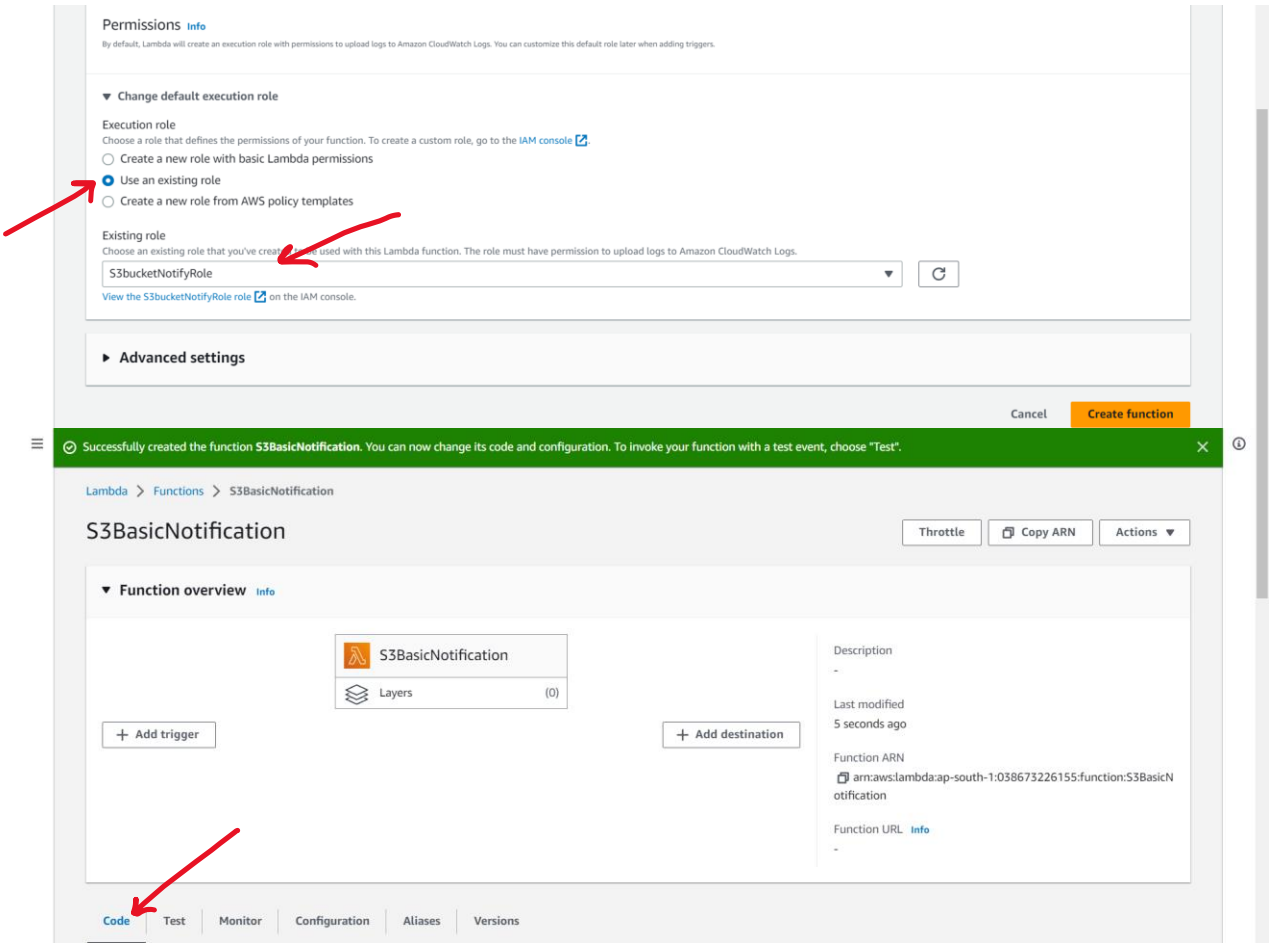
Just click on Next button it seem like the below step



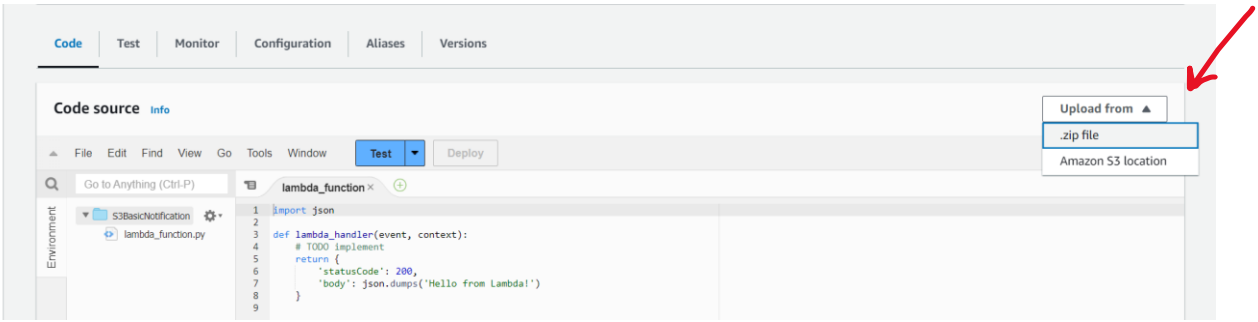
Now create a lambda function



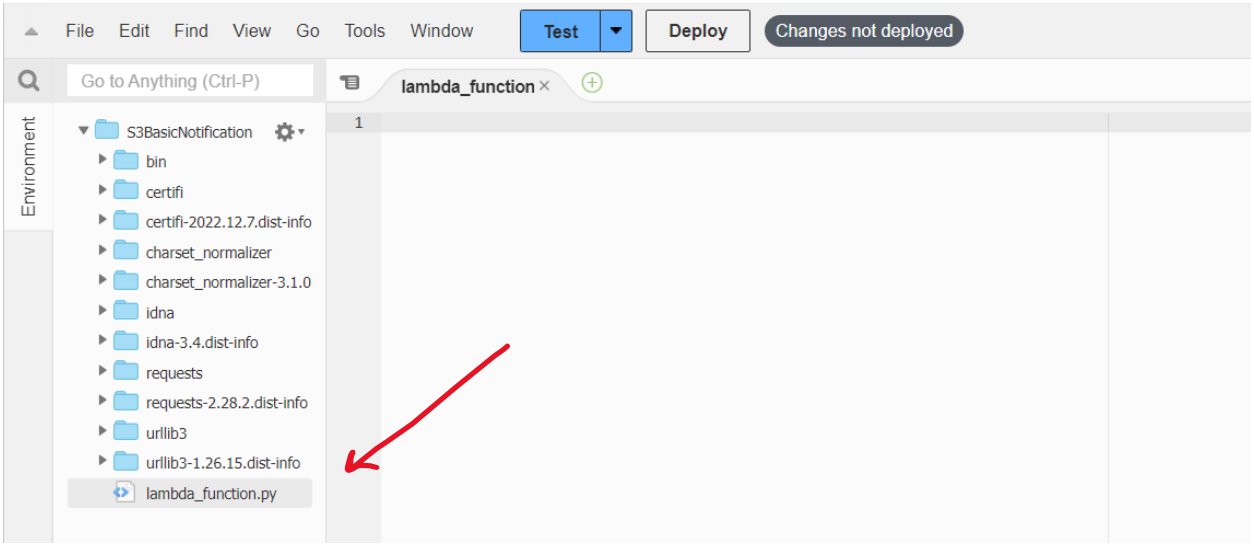




Now Scroll down and open the code Tab

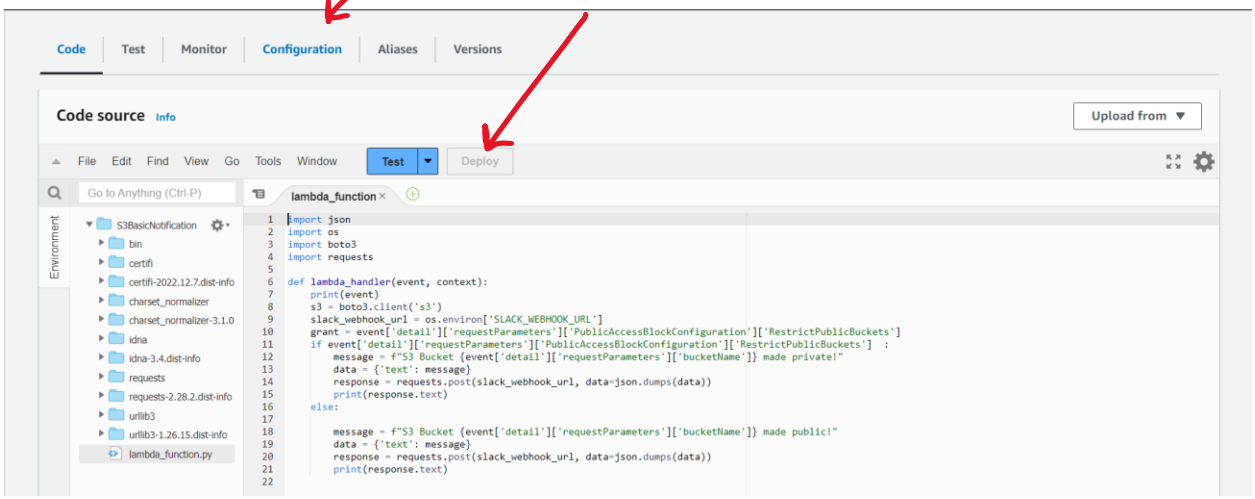


https://drive.google.com/drive/folders/1BiltuSGWxPeD0QDDTmNEI7vJ21p8fMdq?usp=sharing

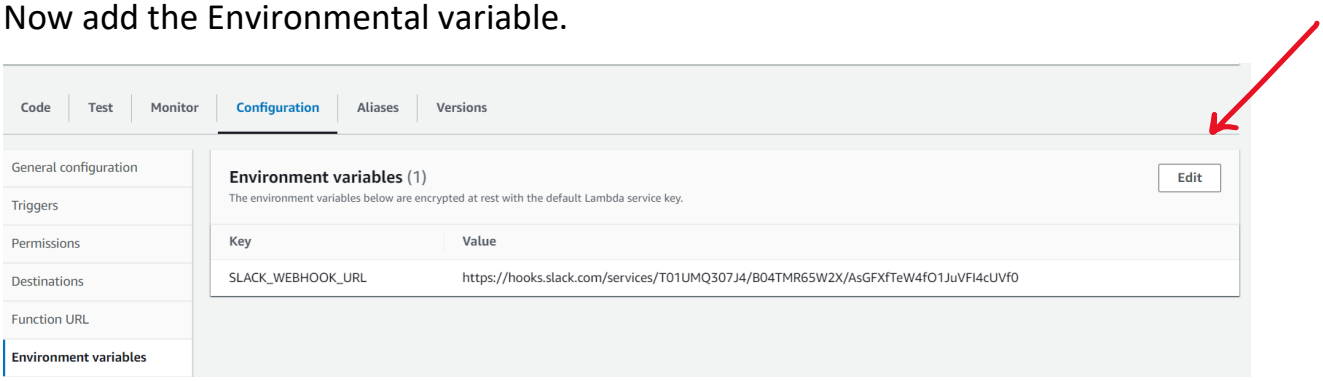


After created a lambda_function.py now just write a code or copy paste the given code below.

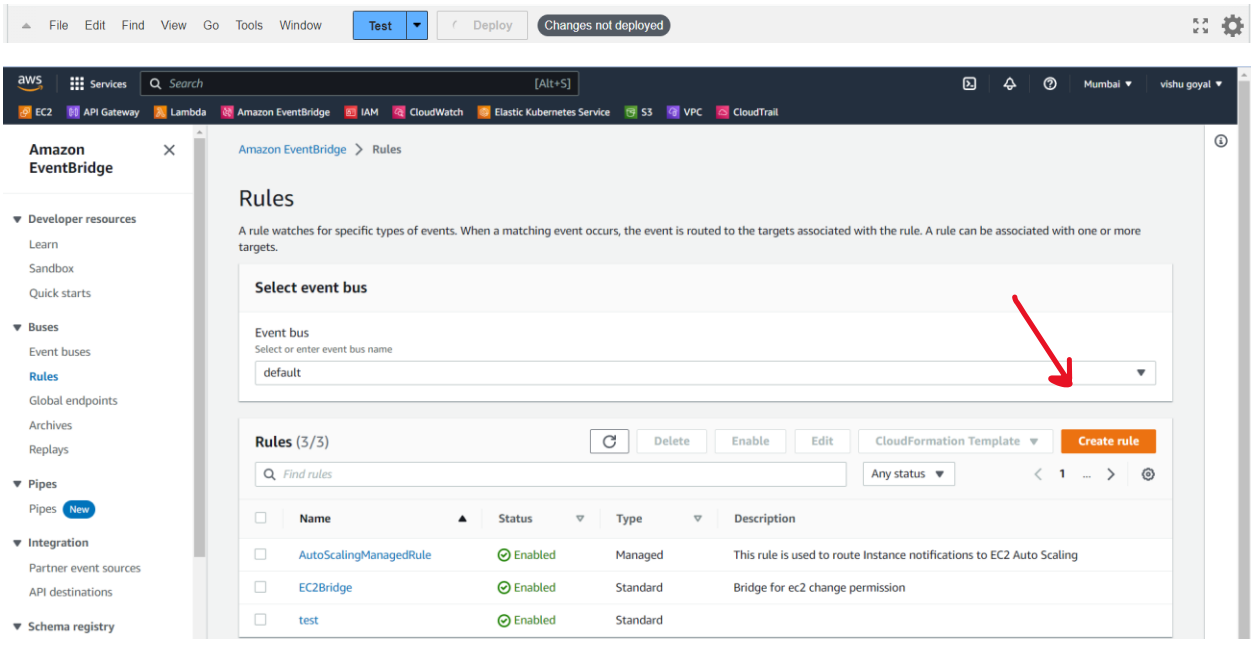
```
import json

import os

import boto3

import requests

def lambda_handler(event, context):

    print(event)

    s3 = boto3.client('s3')

    slack_webhook_url = os.environ['SLACK_WEBHOOK_URL']

    grant = event['detail']['requestParameters']['PublicAccessBlockConfiguration']['RestrictPublicBuckets']

    if event['detail']['requestParameters']['PublicAccessBlockConfiguration']['RestrictPublicBuckets']    :

        message = f"S3 Bucket {event['detail']['requestParameters']['bucketName']} made private!"

        data = {'text': message}

        response = requests.post(slack_webhook_url, data=json.dumps(data))

        print(response.text)

    else:

        message = f"S3 Bucket {event['detail']['requestParameters']['bucketName']} made public!"

        data = {'text': message}

        response = requests.post(slack_webhook_url, data=json.dumps(data))

        print(response.text)
```



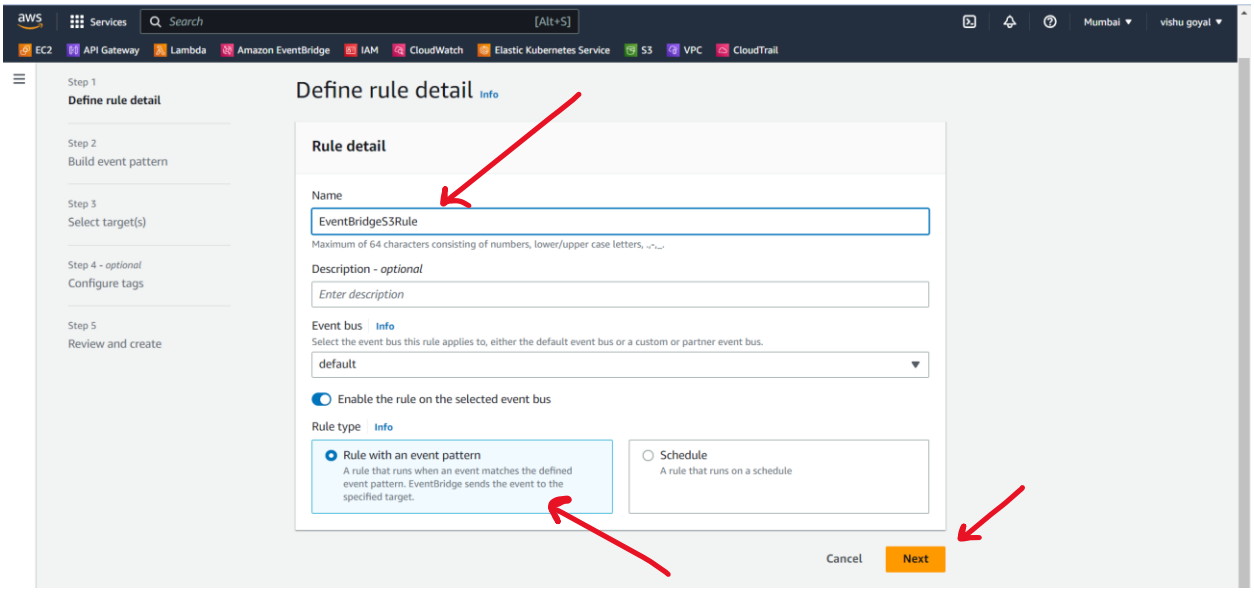
Now add the Environmental variable.

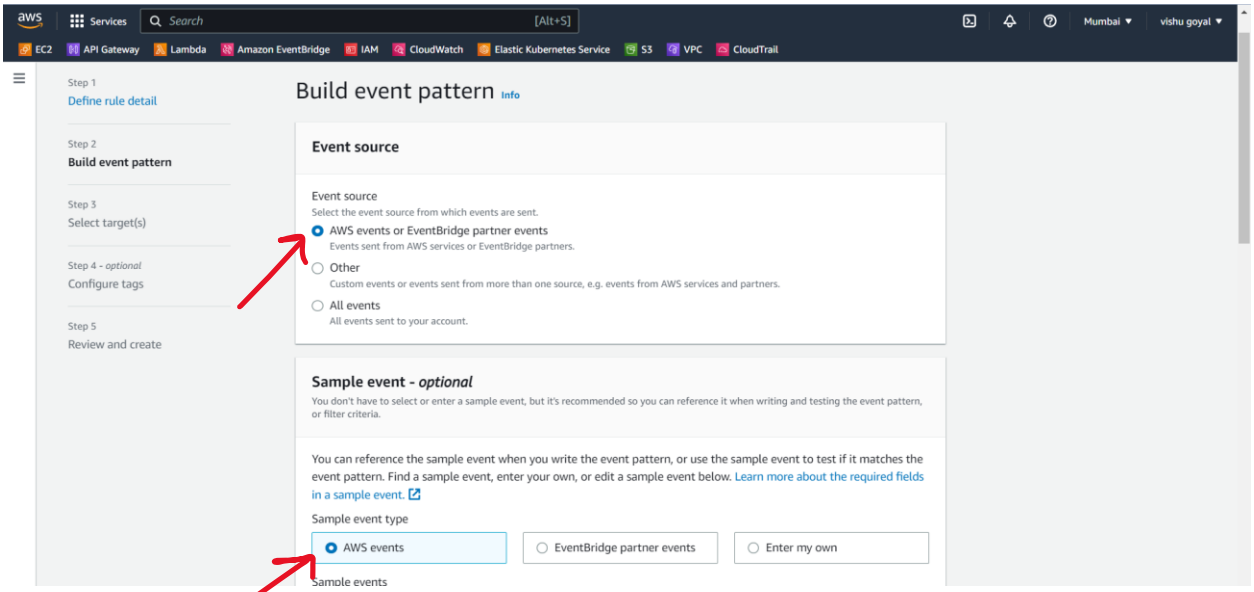


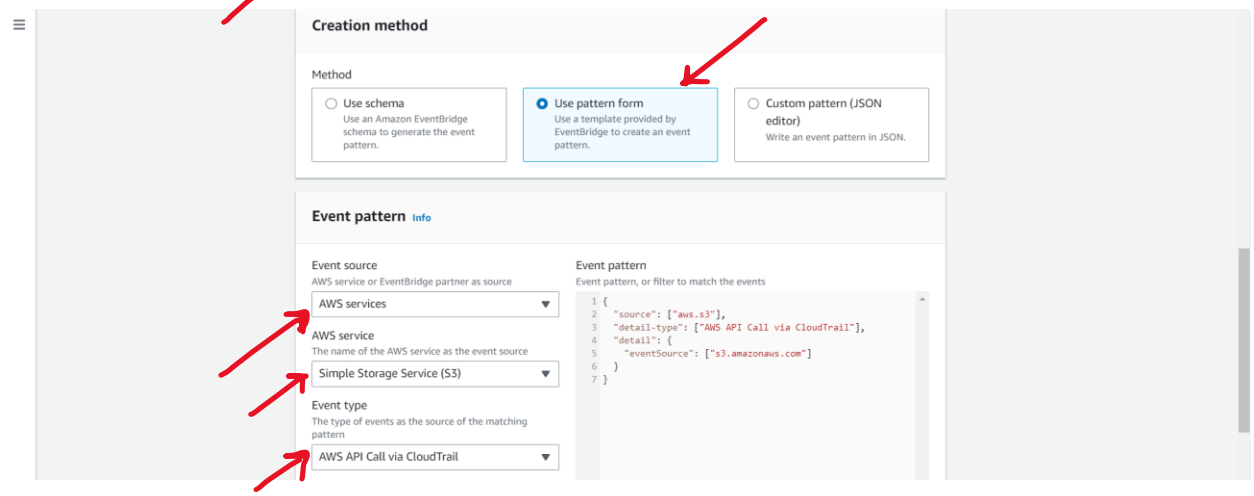
Now deploy the lambda function

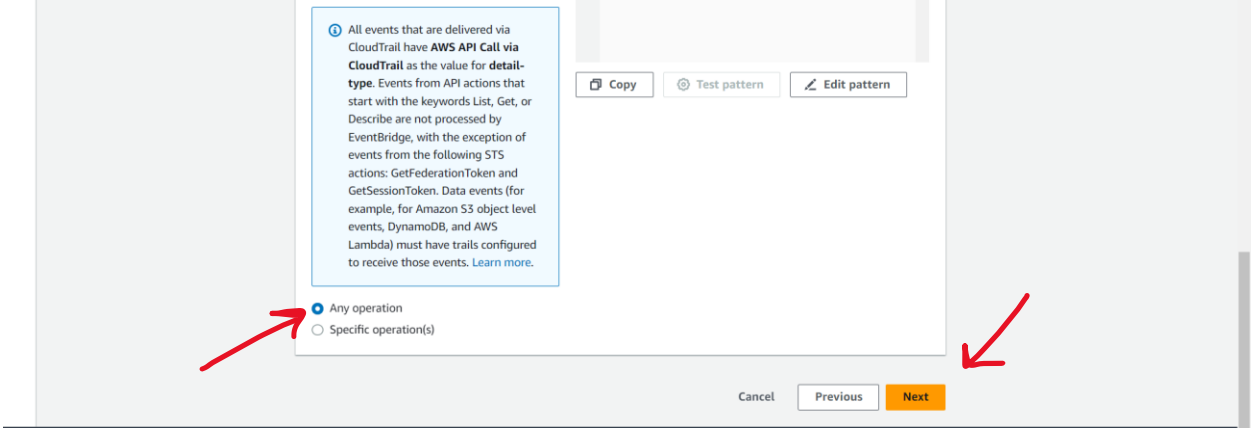


Now you have to create a eventBridge Rule

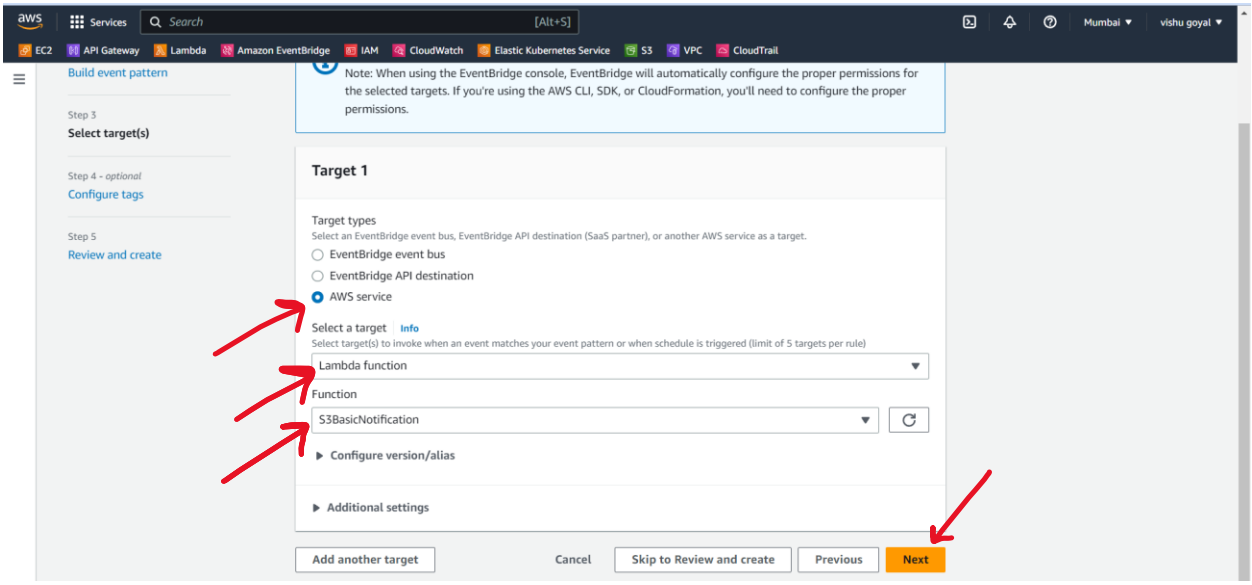

Services
Search
[Alt+S]
Mumbai
vishu goyal
EC2
API Gateway
Lambda
Amazon EventBridge
IAM
CloudWatch
Elastic Kubernetes Service
S3
VPC
CloudTrail
Step 1
Define rule detail
Step 2
Build event pattern
Step 3
Select target(s)
Step 4 - optional
Configure tags
Step 5
Review and create
Define rule detail Info
Rule detail
Name
EventBridgeS3Rule
Maximum of 64 characters consisting of numbers, lower/upper case letters, .,-,_.
Description - optional
Enter description
Event bus Info
Select the event bus this rule applies to, either the default event bus or a custom or partner event bus.
default
Enable the rule on the selected event bus
Rule type Info
Rule with an event pattern
A rule that runs when an event matches the defined event pattern. EventBridge sends the event to the specified target.
Schedule
A rule that runs on a schedule
Cancel
Next


Services
Search
[Alt+S]
Mumbai
vishu goyal
EC2
API Gateway
Lambda
Amazon EventBridge
IAM
CloudWatch
Elastic Kubernetes Service
S3
VPC
CloudTrail
Step 1
Define rule detail
Step 2
Build event pattern
Step 3
Select target(s)
Step 4 - optional
Configure tags
Step 5
Review and create
Build event pattern Info
Event source
Event source
Select the event source from which events are sent.
AWS events or EventBridge partner events
Events sent from AWS services or EventBridge partners.
Other
Custom events or events sent from more than one source, e.g. events from AWS services and partners.
All events
All events sent to your account.
Sample event - optional
You don't have to select or enter a sample event, but it's recommended so you can reference it when writing and testing the event pattern, or filter criteria.
You can reference the sample event when you write the event pattern, or use the sample event to test if it matches the event pattern. Find a sample event, enter your own, or edit a sample event below. Learn more about the required fields in a sample event.
Sample event type
AWS events
EventBridge partner events
Enter my own
Sample events


Creation method
Method
Use schema
Use an Amazon EventBridge schema to generate the event pattern.
Use pattern form
Use a template provided by EventBridge to create an event pattern.
Custom pattern (JSON editor)
Write an event pattern in JSON.
Event pattern Info
Event source
AWS service or EventBridge partner as source
AWS services
AWS service
The name of the AWS service as the event source
Simple Storage Service (S3)
Event type
The type of events as the source of the matching pattern
AWS API Call via CloudTrail
Event pattern
Event pattern, or filter to match the events
{
"source": ["aws.s3"],
"detail-type": ["AWS API Call via CloudTrail"],
"detail": {
"eventSource": ["s3.amazonaws.com"]
}
}

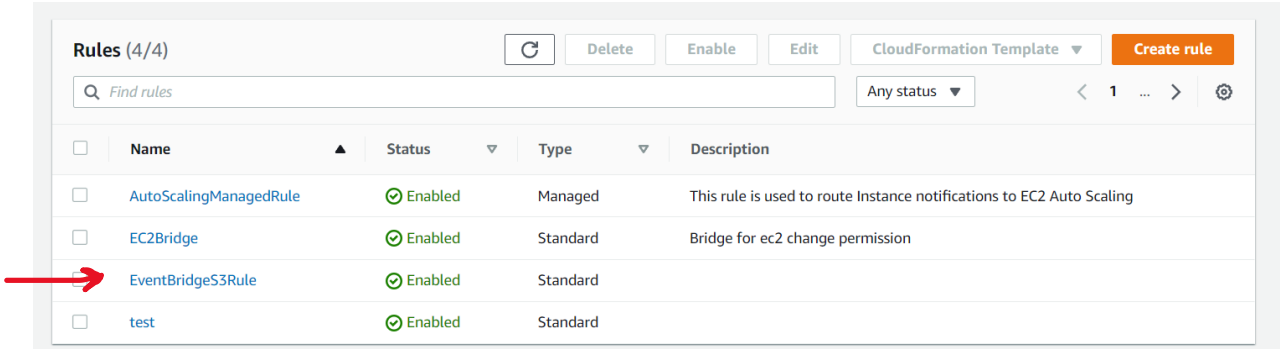


After the above steps just click on the Next button.



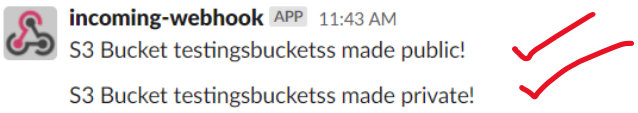
After the above steps just click on the next button to move forward steps.

Next → Next → createRule



The interface looks like that.

Now have to do a check to create a bucket with private permission it trigger the lambda function after when you have changed the permission of that bucket make public or private again it trigger notification on slack.



Everything is complete !!!!

Congratulations you have setup Successfully!!!!