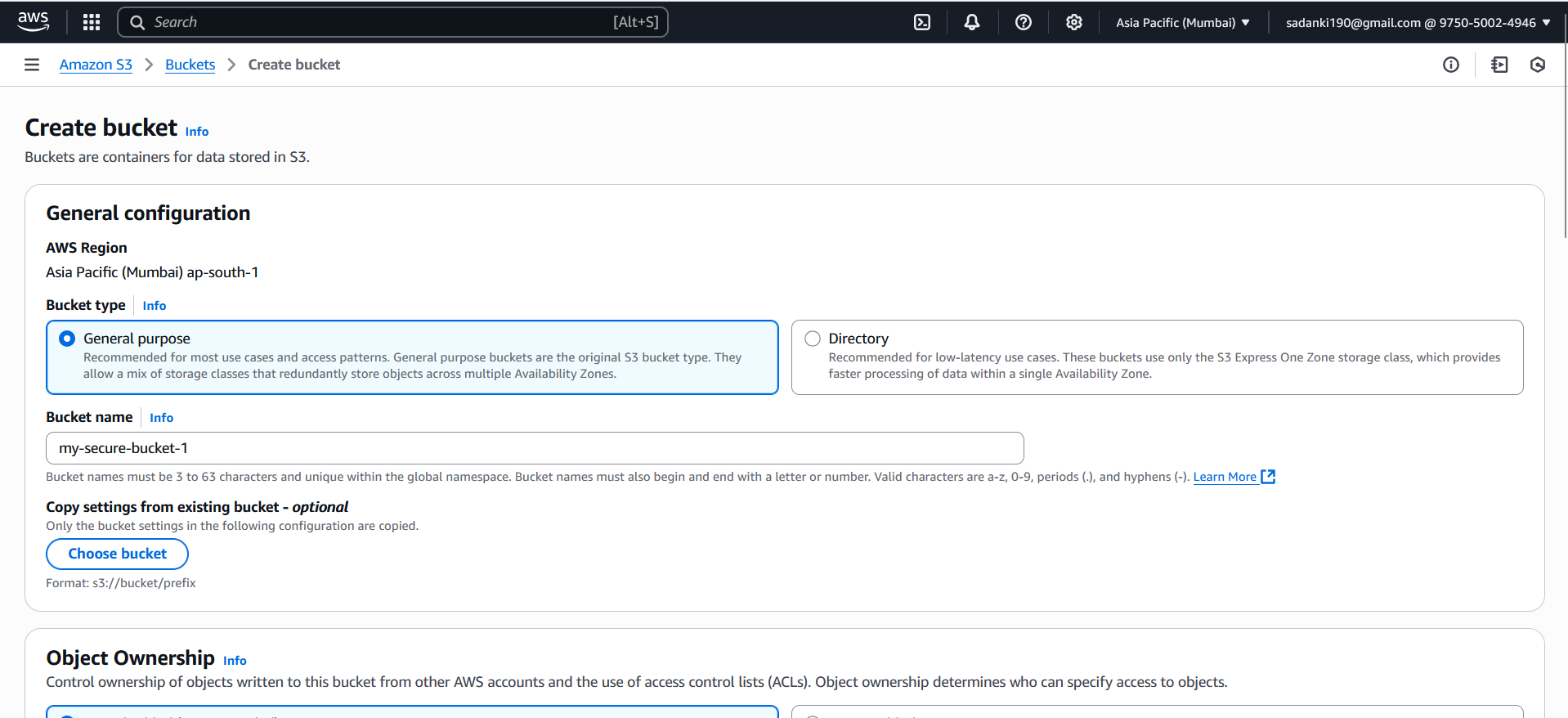
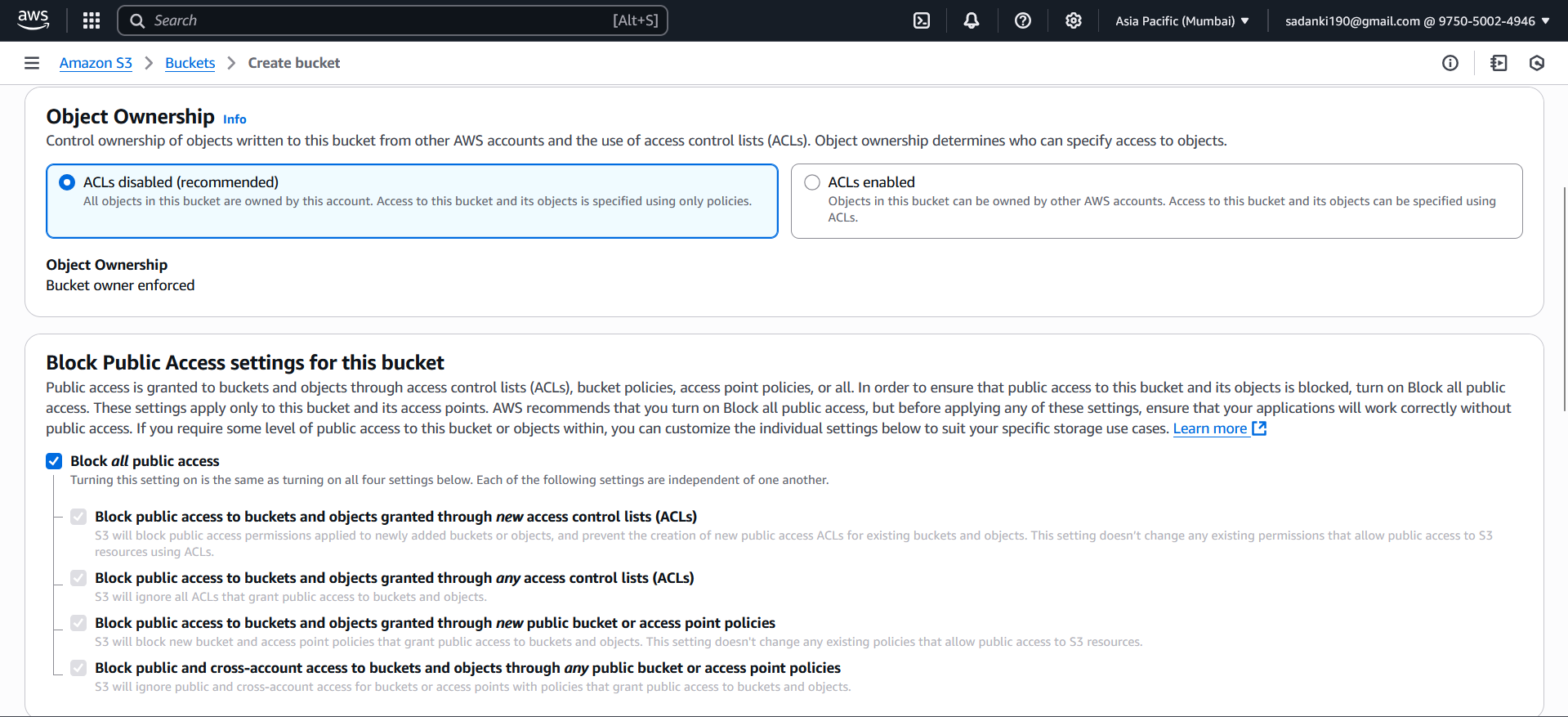
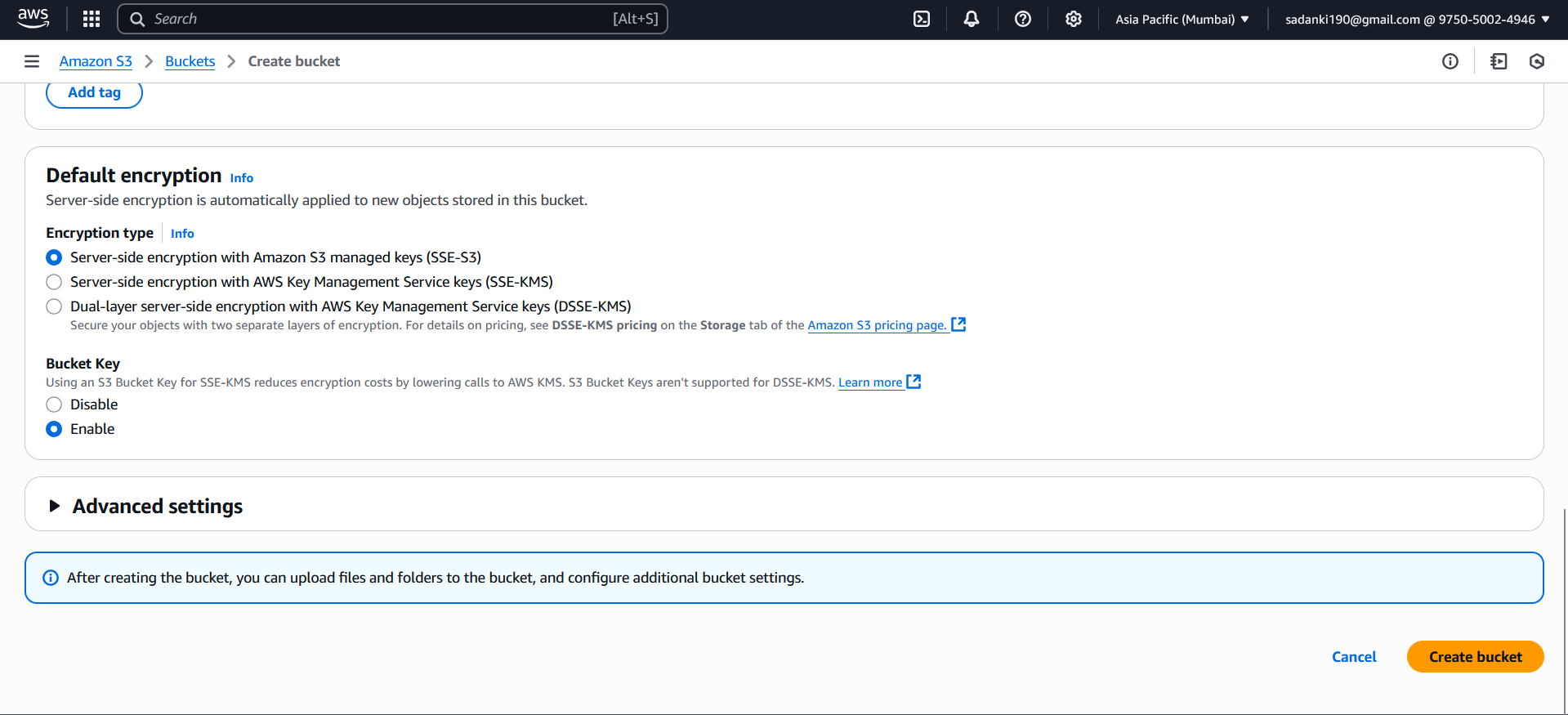
Assignment 3: Monitor Unencrypted S3 Buckets Using AWS Lambda and Boto3

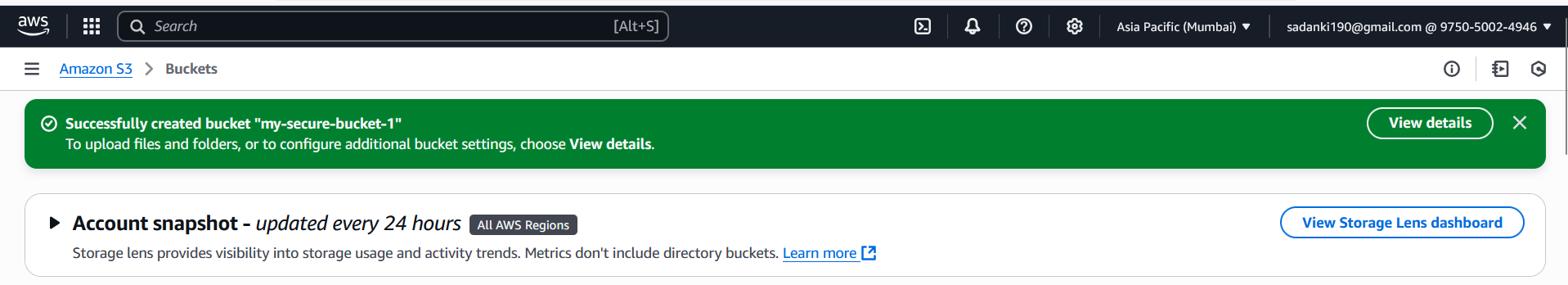
**Step 1: Create S3 Buckets**

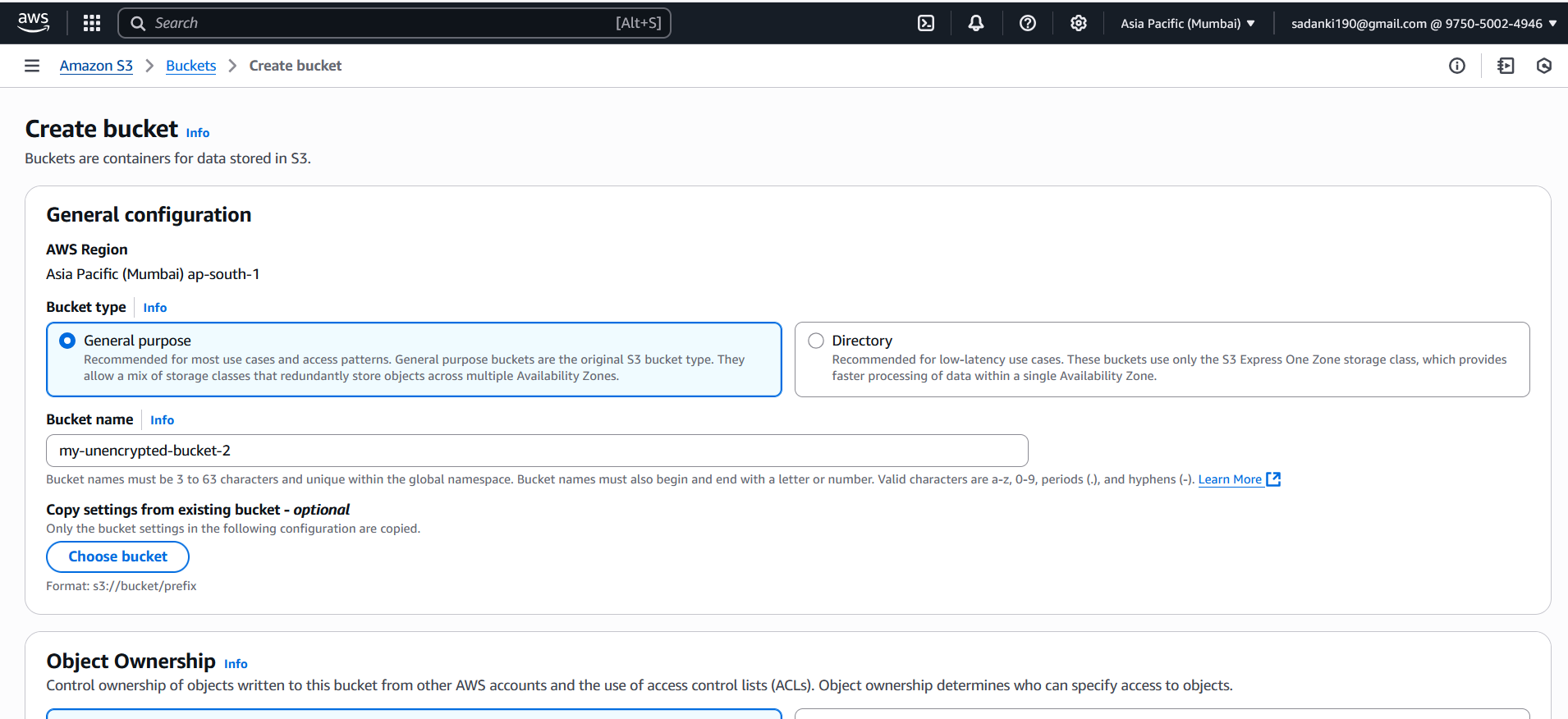
1. **Login to AWS Console.**
2. Navigate to **S3** service dashboard.
3. Click **Create bucket**.
4. Create at least 3 buckets:
   * For example: my-secure-bucket-1, my-unencrypted-bucket-1, my-unencrypted-bucket-2.
5. **For some buckets, enable Server-Side Encryption (SSE):**
   * While creating or in the bucket properties, go to **Default encryption**.
   * Enable **AES-256** or **AWS-KMS** encryption.
6. Leave at least one or two buckets **without SSE enabled** (to test detection).

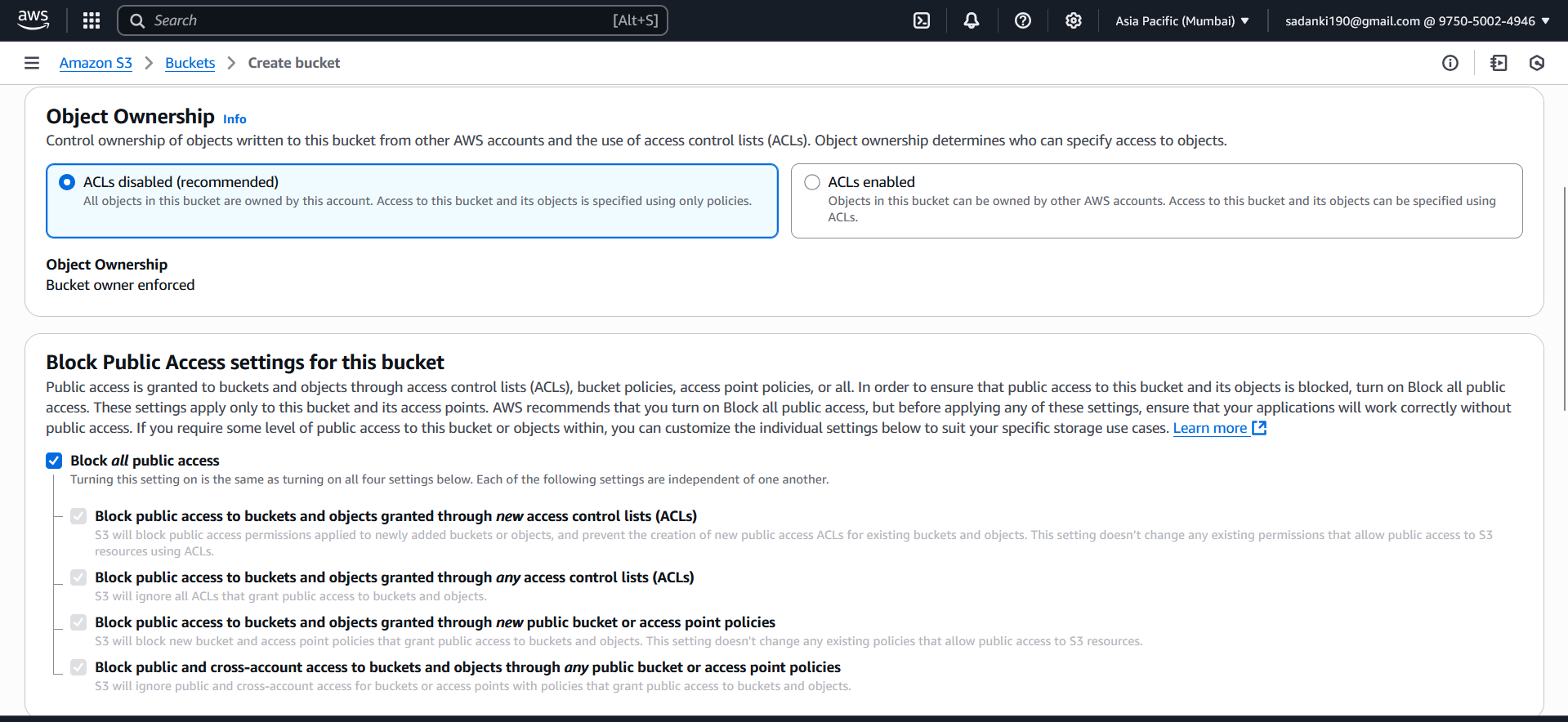


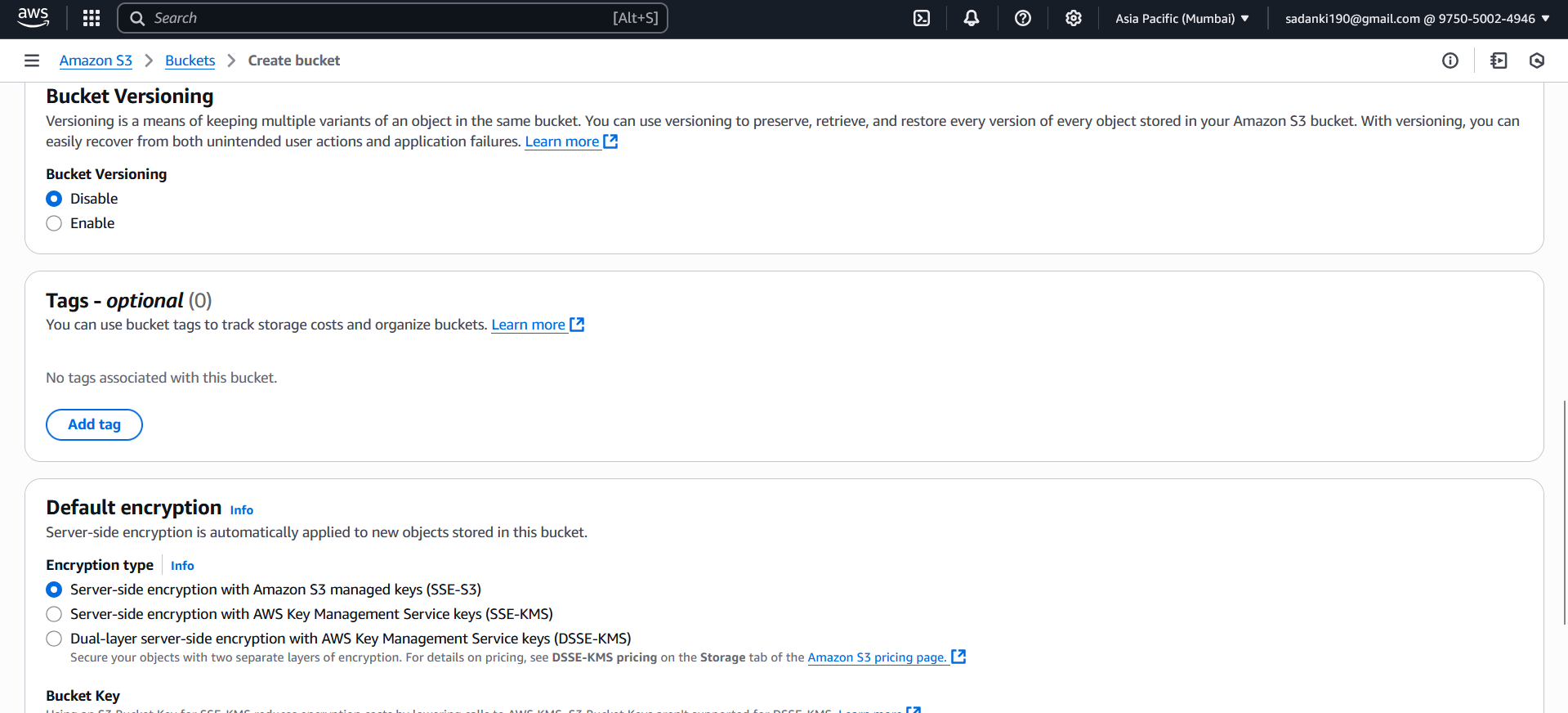


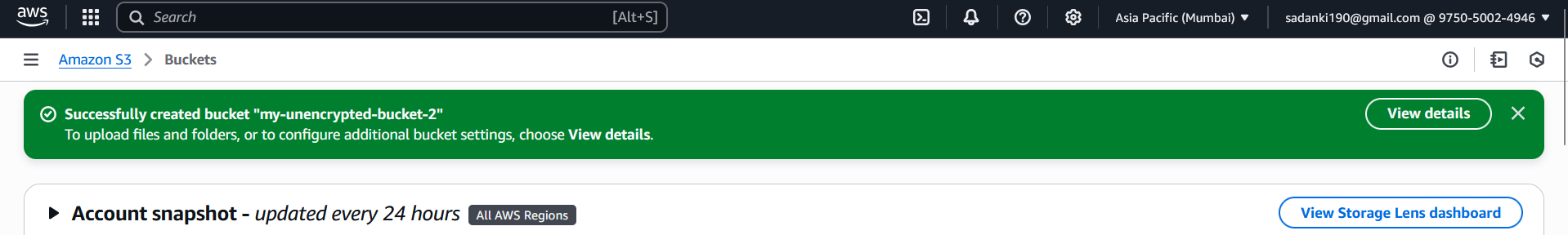












Disable Encryption **After** Bucket Creation.

Go to the **AWS S3 Console**.

Click on the **bucket name** (e.g., my-secure-bucket).

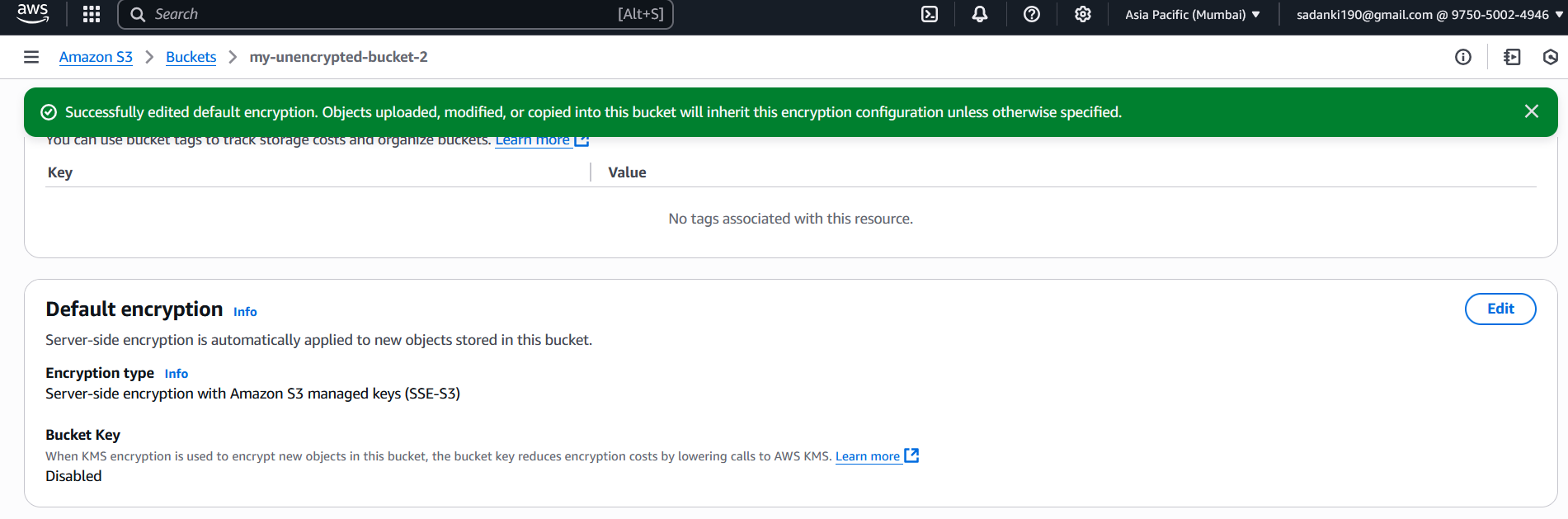
Navigate to the **Properties** tab.

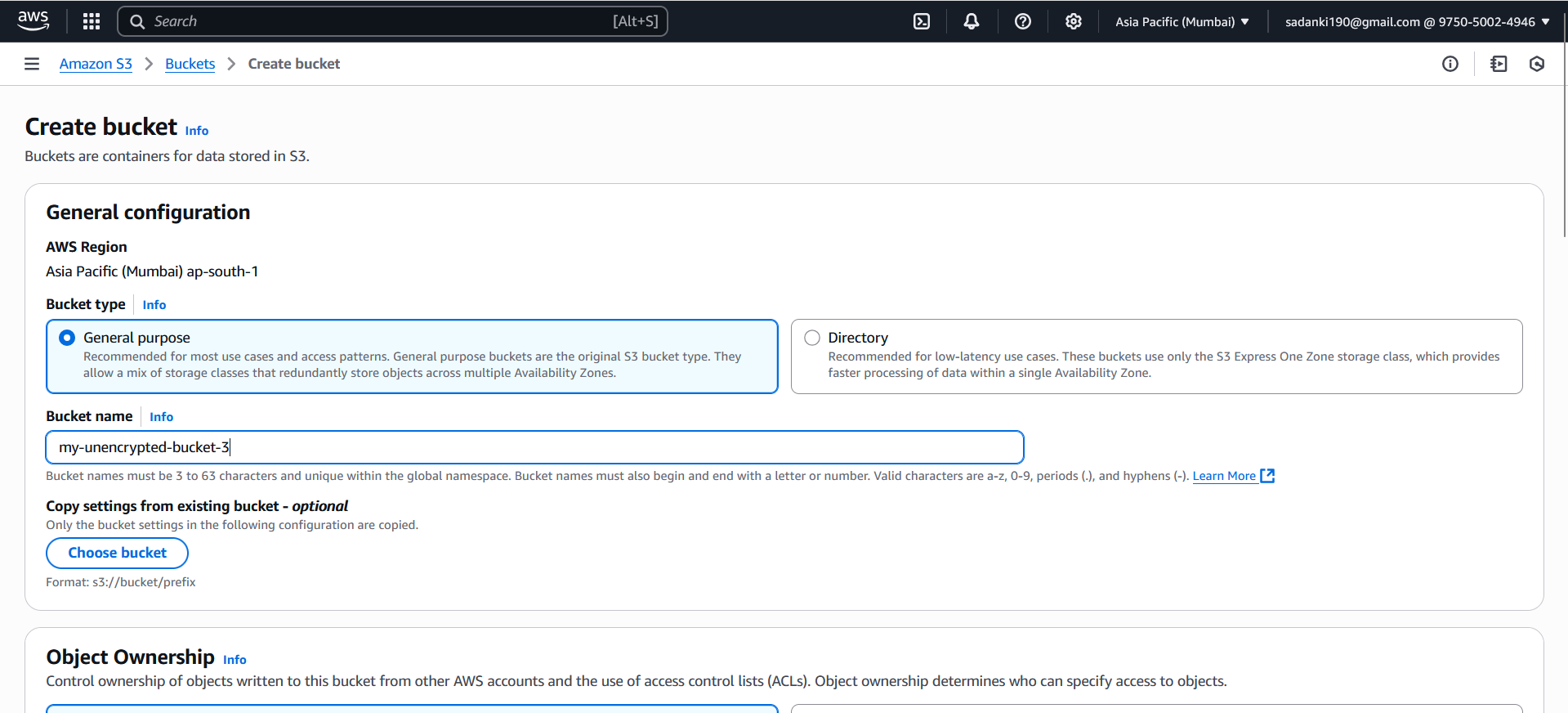
Scroll down to the **Default encryption** section.

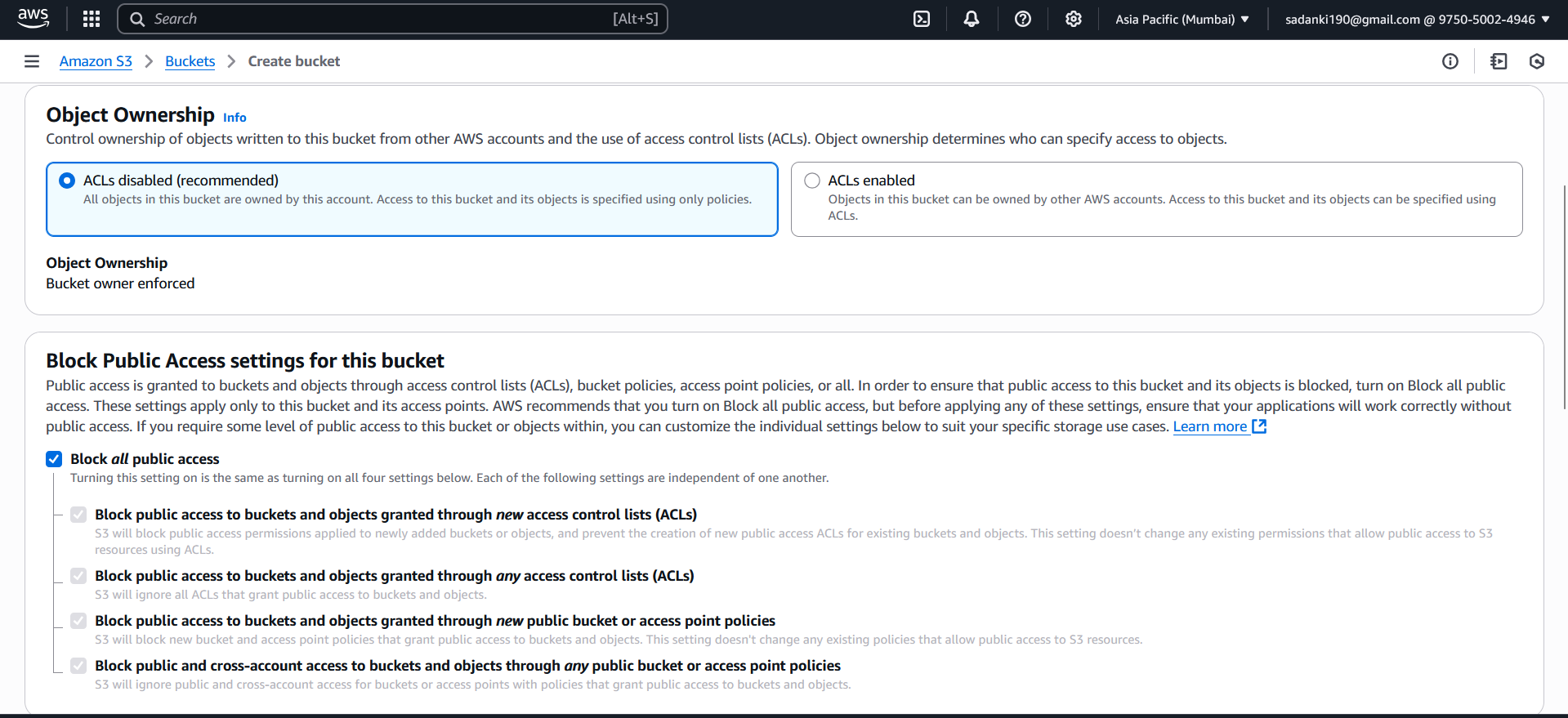
Click **Edit**.

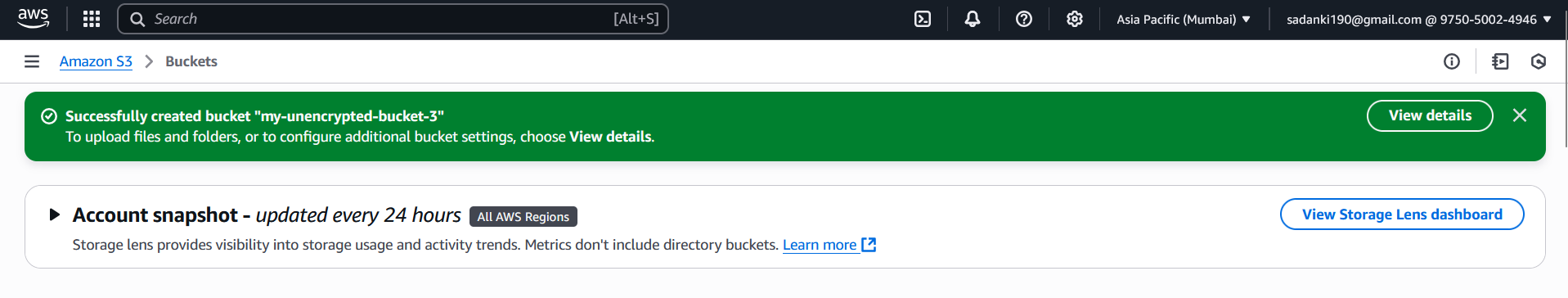
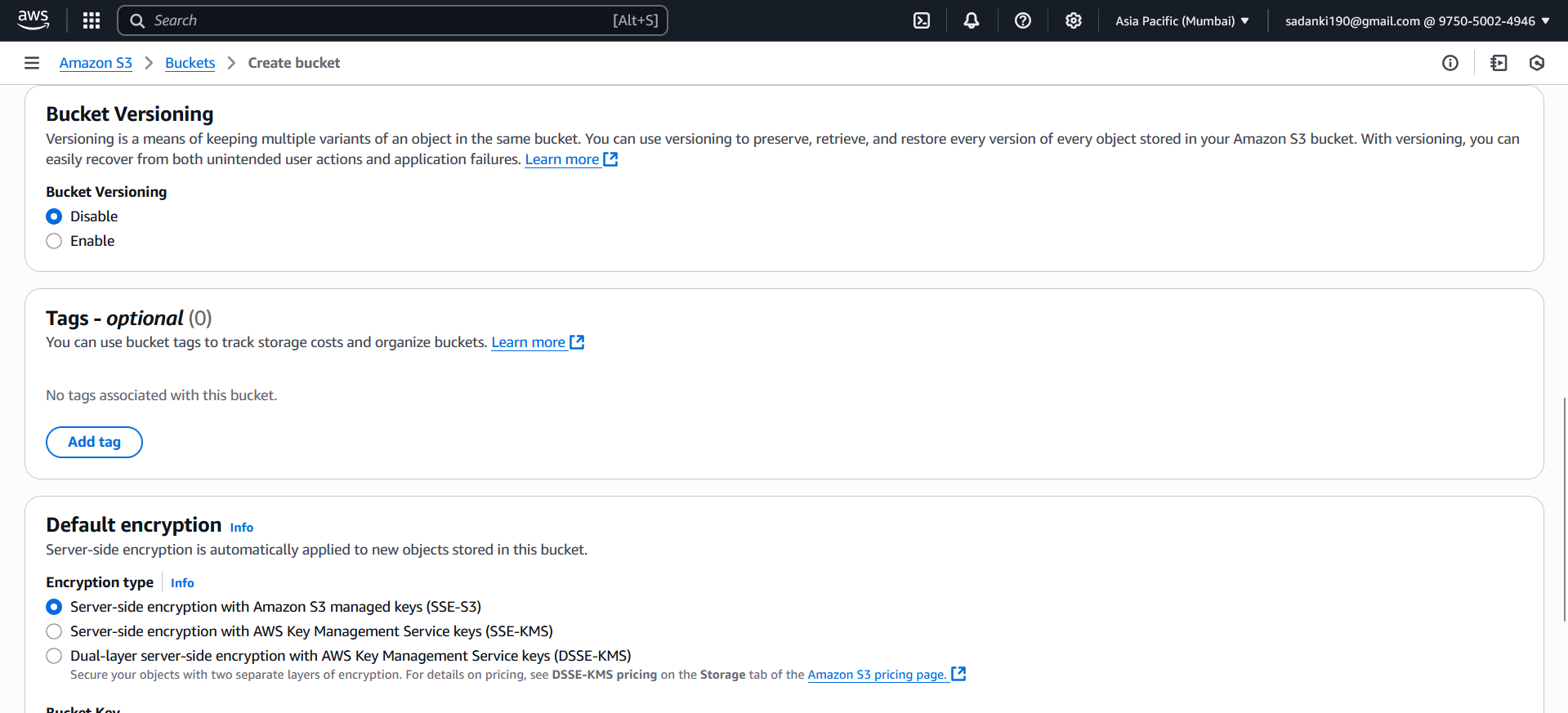
Choose **"Disable"** under "Default encryption".

Click **Save changes**



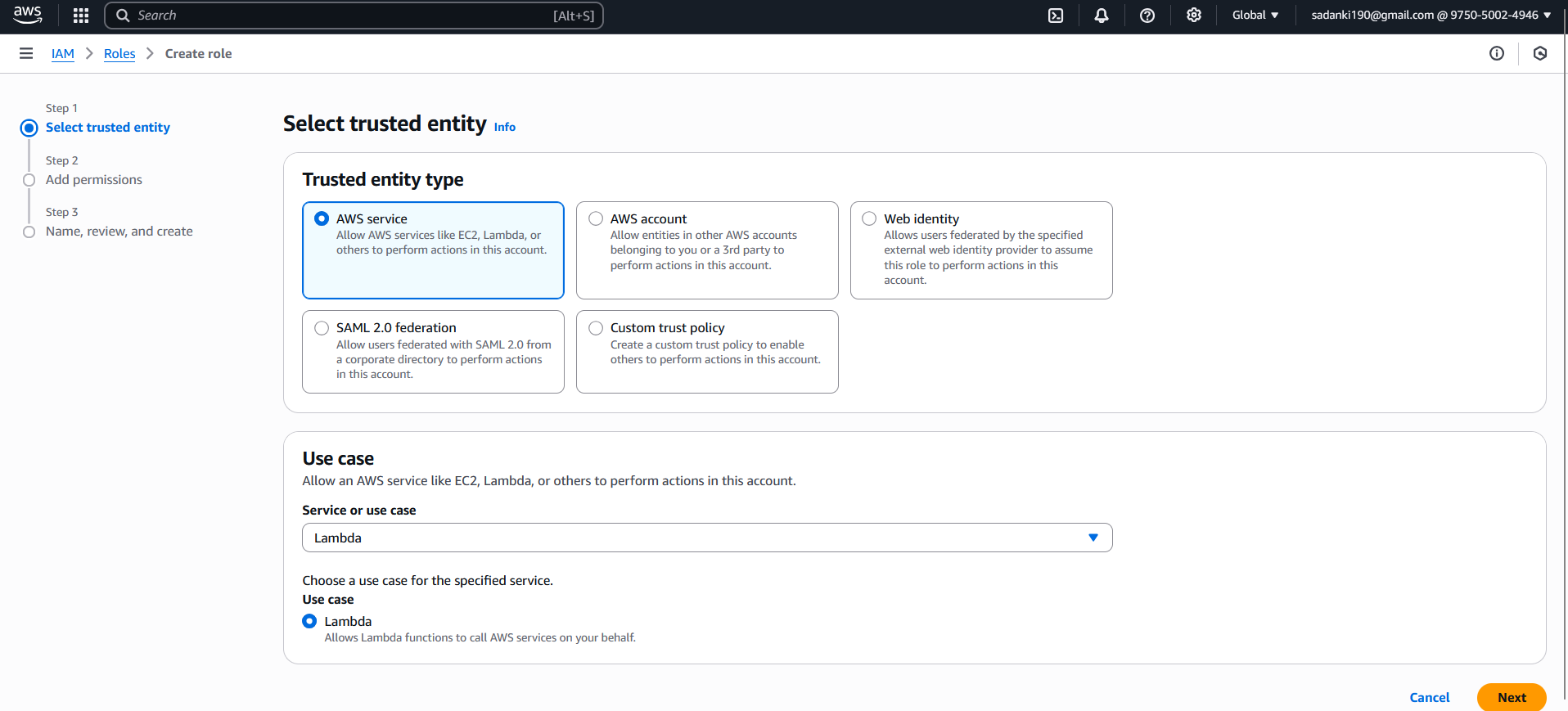


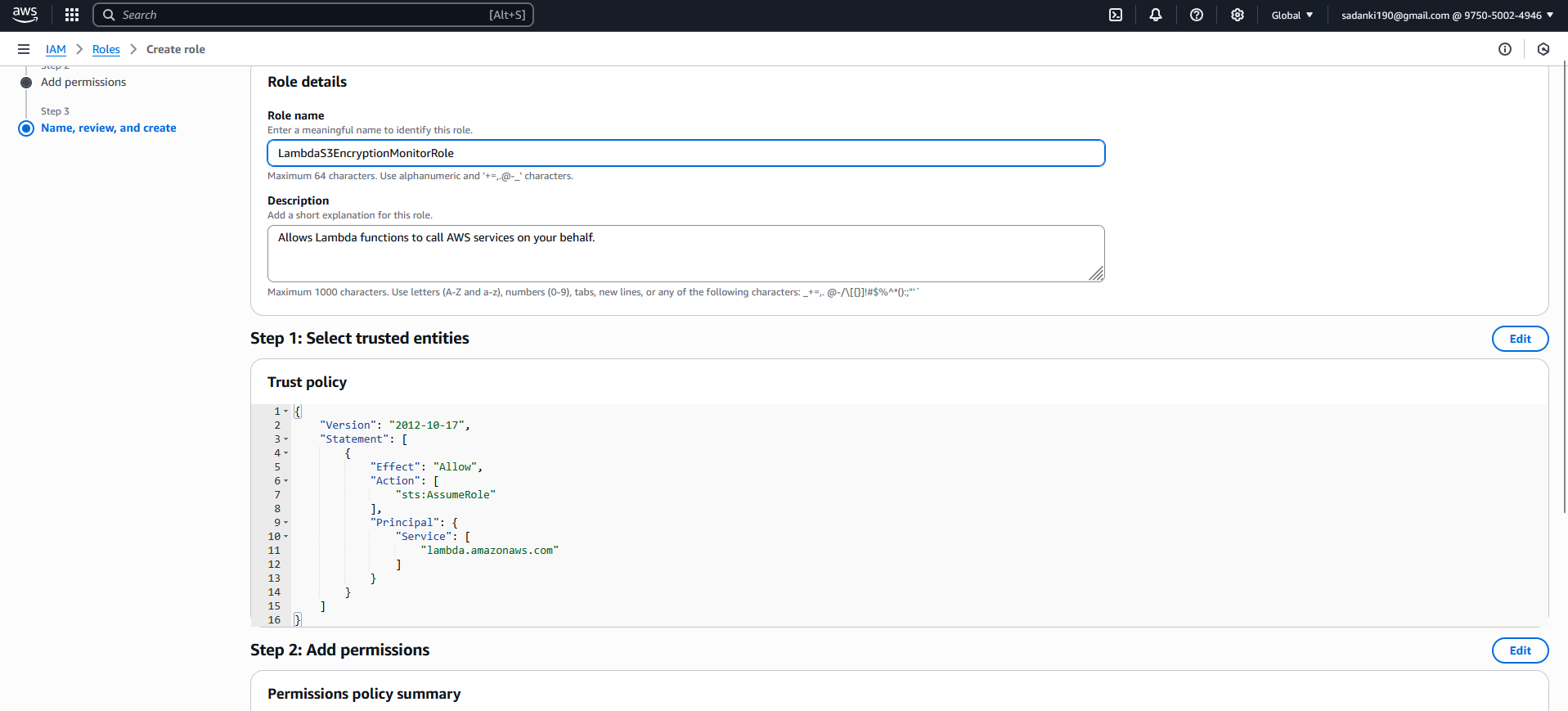


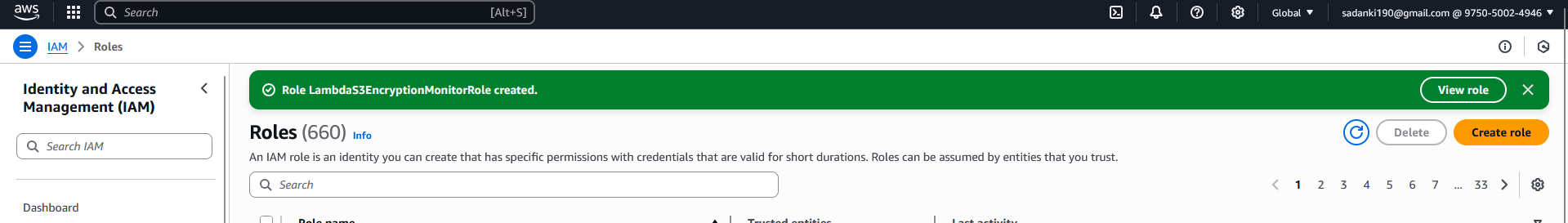


**Step 2: Create IAM Role for Lambda (if not done yet)**

1. Go to **IAM Console > Roles**
2. Click **Create role**
3. **Trusted entity**: AWS service
4. Use case: **Lambda**
5. Click **Next**
6. Attach policy:  
    AmazonS3ReadOnlyAccess
7. Click **Next**, name the role:  
   LambdaS3EncryptionMonitorRole
8. Click **Create Role**

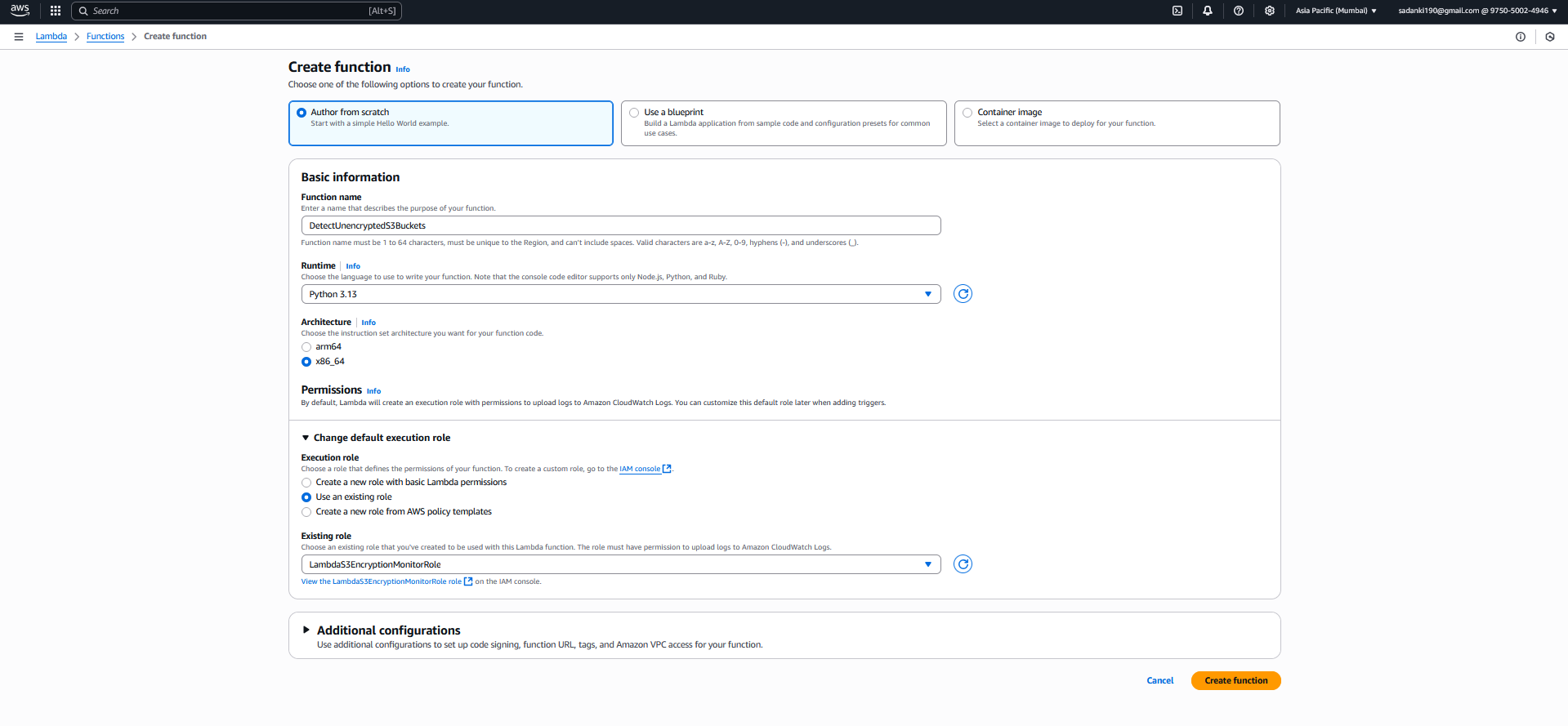






**Step 3: Create the Lambda Function**

1. Go to [Lambda Console](https://console.aws.amazon.com/lambda)
2. Click **Create function**
3. Use:
   * **Name:** DetectUnencryptedS3Buckets
   * **Runtime:** Python 3.12
   * **Permissions:** Use existing role → LambdaS3EncryptionMonitorRole
4. Click **Create function**



Step 4: Paste This Python Code

import boto3

def lambda\_handler(event, context):

    s3 = boto3.client('s3')

    unencrypted\_buckets = []

    # List all buckets

    response = s3.list\_buckets()

    for bucket in response['Buckets']:

        bucket\_name = bucket['Name']

        try:

            encryption = s3.get\_bucket\_encryption(Bucket=bucket\_name)

            rules = encryption['ServerSideEncryptionConfiguration']['Rules']

            print(f"✅ {bucket\_name} is encrypted with: {rules[0]['ApplyServerSideEncryptionByDefault']['SSEAlgorithm']}")

        except s3.exceptions.ClientError as e:

            error\_code = e.response['Error']['Code']

            if error\_code == 'ServerSideEncryptionConfigurationNotFoundError':

                print(f"❌ {bucket\_name} is NOT encrypted")

                unencrypted\_buckets.append(bucket\_name)

            else:

                print(f"⚠️ Error checking {bucket\_name}: {str(e)}")

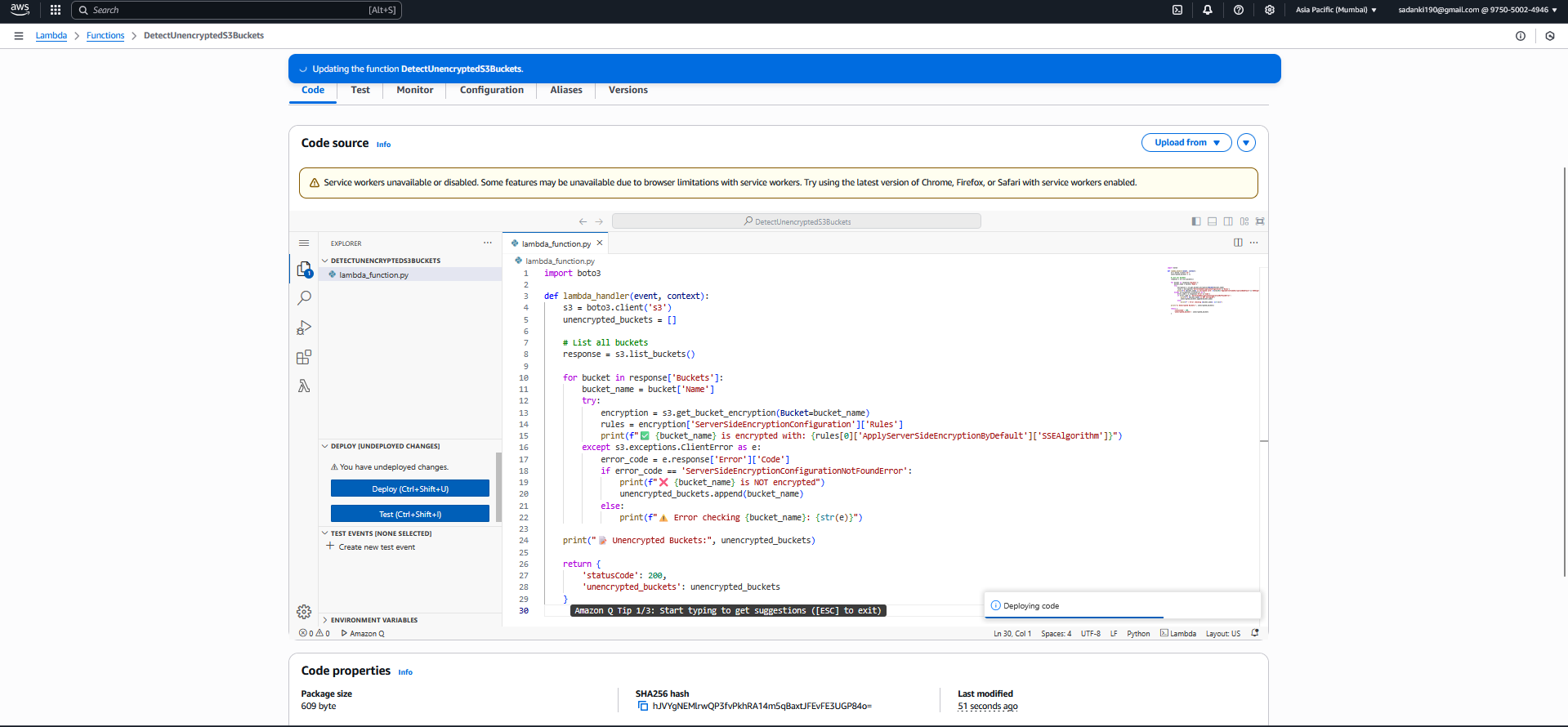
    print("📝 Unencrypted Buckets:", unencrypted\_buckets)

    return {

        'statusCode': 200,

        'unencrypted\_buckets': unencrypted\_buckets

    }



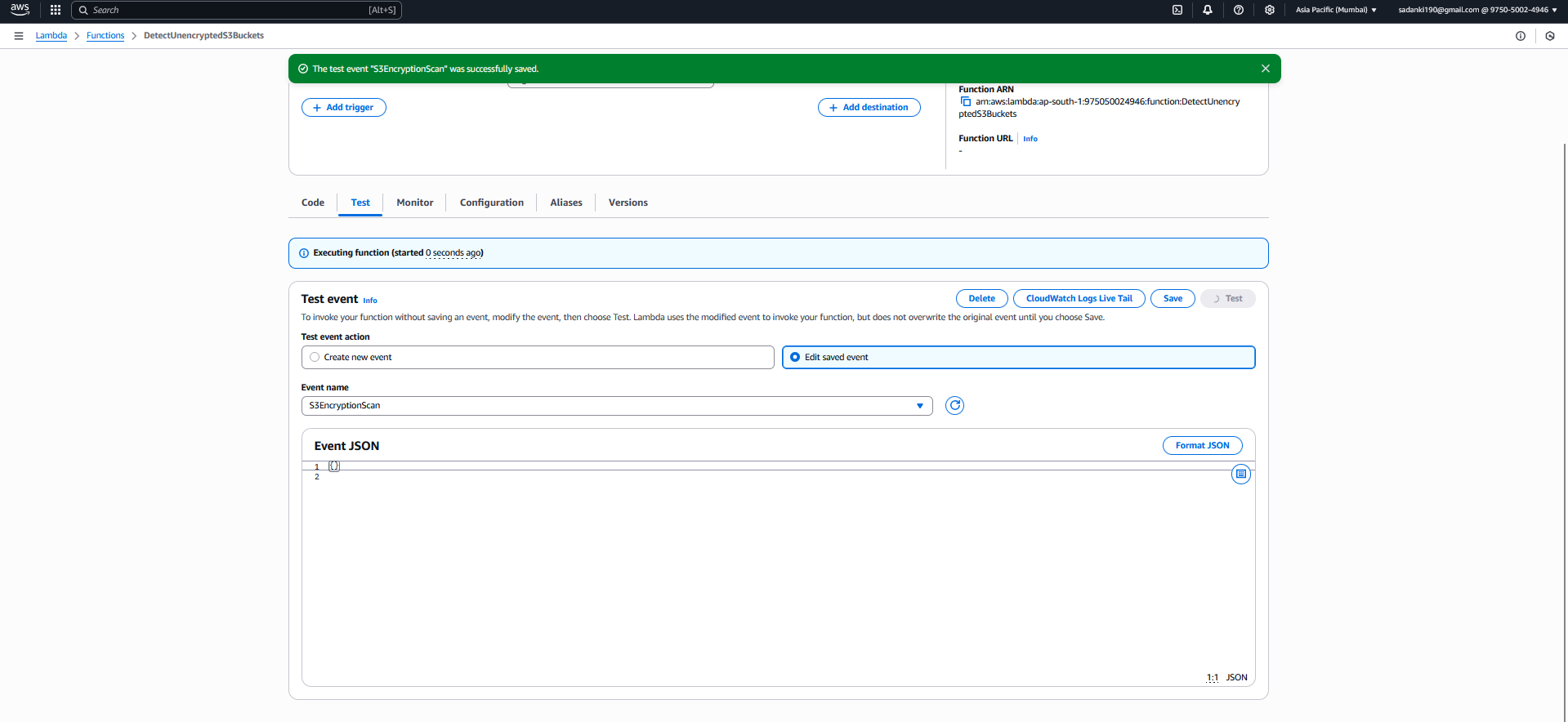
**Step 5: Test the Lambda**

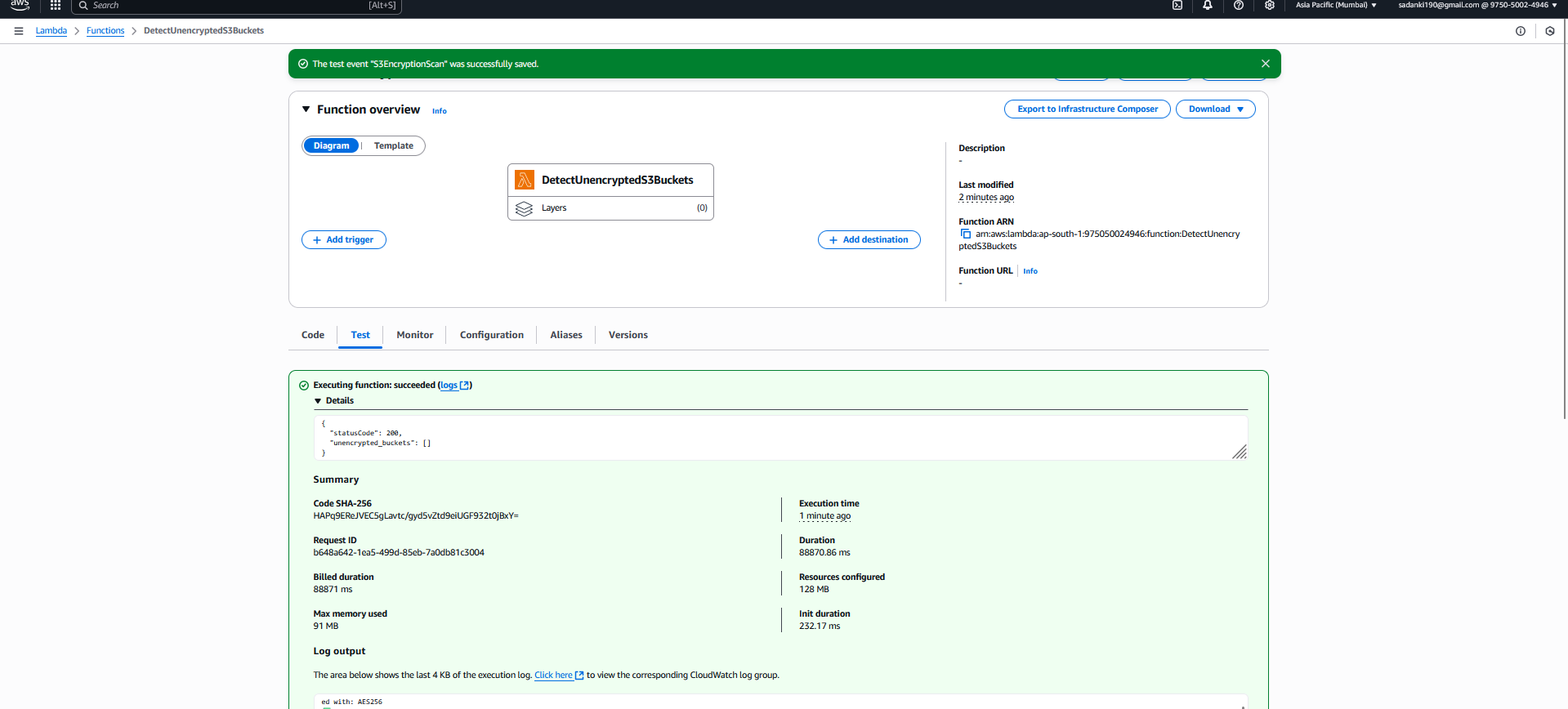
In the **AWS Lambda Console**, go to your function:  
DetectUnencryptedS3Buckets

Click the **Test** button (top-right)

In the pop-up:

* **Event name**: S3EncryptionScan (or anything you like)
* **Event JSON**: leave as just {}





**Server-side encryption (SSE-S3) is enabled**  
**Encryption type:** Server-side encryption with Amazon S3 managed keys (SSE-S3)

This confirms that **your buckets *are encrypted***, and that's exactly why your Lambda function returned:

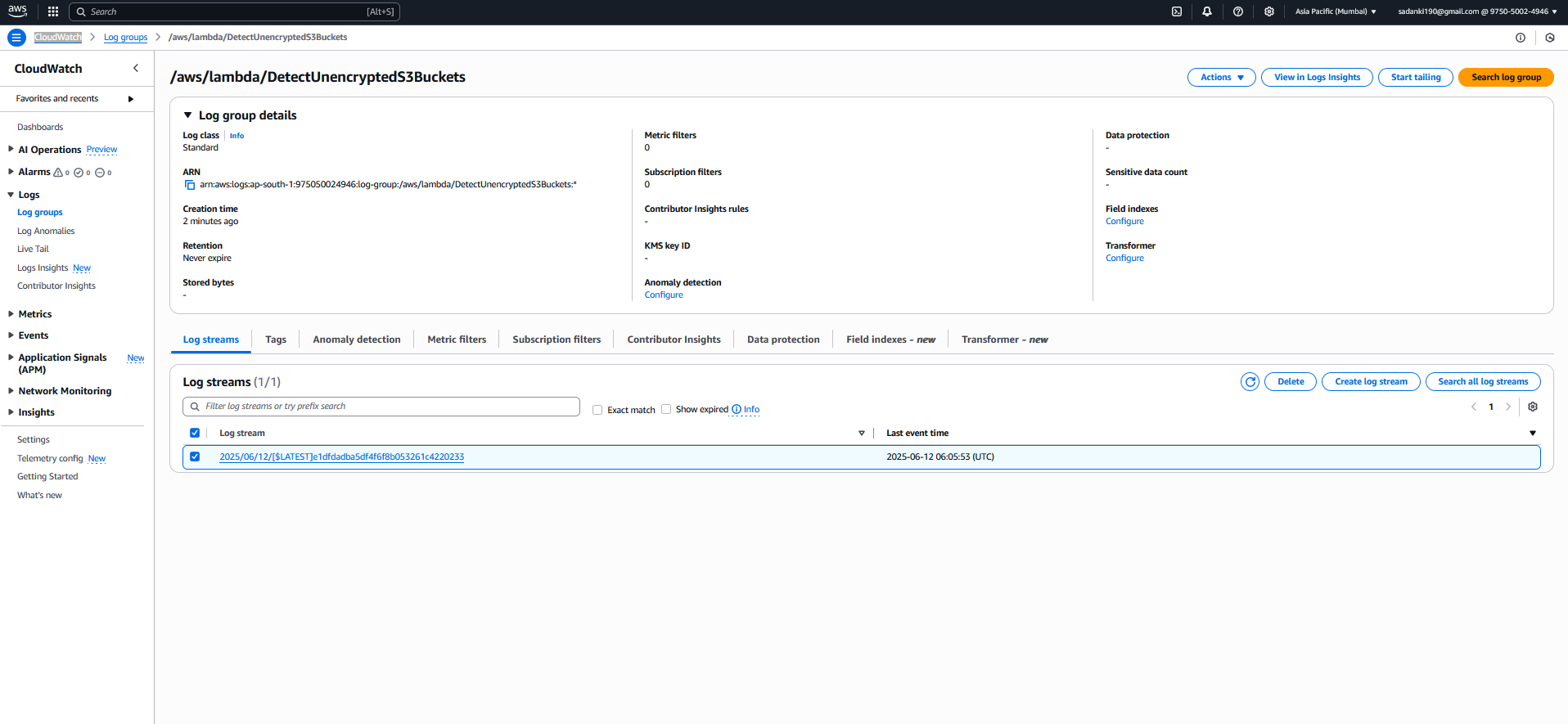
{

"statusCode": 200,

"unencrypted\_buckets": []

}

Verify in [CloudWatch](https://ap-south-1.console.aws.amazon.com/cloudwatch/home?region=ap-south-1):



**Assignment 3 Overview** you can use in your submission (Word file, GitHub README, or final report):

**📘 Assignment 3: Monitor Unencrypted S3 Buckets Using AWS Lambda and Boto3**

**🎯 Objective**

To enhance the security posture of AWS infrastructure by building a **Lambda function** that automatically scans all **S3 buckets** in an account and identifies buckets that **do not have server-side encryption (SSE)** enabled.

This helps ensure data stored in S3 is protected according to best practices and compliance standards.

**🛠️ What Was Done**

1. **S3 Bucket Setup**
   * Created 3 test buckets:
     + my-secure-bucket-1 – with encryption
     + my-unencrypted-bucket-2 – no encryption (later encrypted for testing)
     + my-unencrypted-bucket-3 – no encryption (optional)
   * Enabled SSE-S3 on some buckets
2. **IAM Role Setup**
   * Created IAM Role: LambdaS3EncryptionMonitorRole
   * Attached policy: AmazonS3ReadOnlyAccess
   * Attached policy: AWSLambdaBasicExecutionRole (for CloudWatch logs)
3. **Lambda Function Setup**
   * Created a function named CheckS3Encryption
   * Runtime: Python 3.x
   * Logic:
     + List all buckets using Boto3
     + Check for encryption using get\_bucket\_encryption()
     + Catch ServerSideEncryptionConfigurationNotFoundError if unencrypted
     + Log unencrypted bucket names using print()
4. **Logging & Output**
   * Configured Lambda to log results in **CloudWatch**
   * Example output:
   * {
   * "statusCode": 200,
   * "unencrypted\_buckets": ["my-unencrypted-bucket-2", "my-unencrypted-bucket-3"]
   * }
   * Verified log output and encryption status via screenshots

**✅ Why This Is Useful**

* Detects misconfigured S3 buckets lacking encryption
* Prevents unintentional data leaks
* Helps comply with data privacy regulations (e.g., GDPR, HIPAA, ISO 27001)
* Automates what would otherwise be a manual audit

**✅ Benefits**

| **Benefit** | **Description** |
| --- | --- |
| 🔒 Improves Security | Flags buckets without SSE |
| 🤖 Automation Ready | Can be scheduled to run regularly via CloudWatch Events |
| 📉 Cost-Effective | Serverless, no infrastructure to manage |
| 📊 Logs for Auditing | Logs can be used as audit trail in CloudWatch |

**⚠️ Limitations**

| **Limitation** | **Description** |
| --- | --- |
| ❌ Checks Bucket Config Only | Doesn’t validate encryption at the object level |
| 🌐 All Regions Not Scoped | Only scans buckets in the account, not across all regions |
| 🔐 No Remediation Yet | Detects but does not automatically enable encryption |