Serverless Architecture Graded Assignment

Out of the 20 assignments mentioned on the portal, 4 are to be attempted.

The GitHub repository for the assignment has been attached with the link - <https://github.com/shuklakash1/Serverless_Architcture>

Here are the 4 questions that have been attempted using boto3 as base library with the Lambda function. All the questions have been attempted on the personal AWS account:

**Question 1 – Automated S3 bucket cleanup using AWS Lambda and Boto3**

**Objective –** To gain experience with AWS Lambda and Boto3 by creating a lambda function that will automatically clean up old files in an S3 bucket.

**Task –** Automate the deletion of files older than 30 days in a specific S3 bucket.

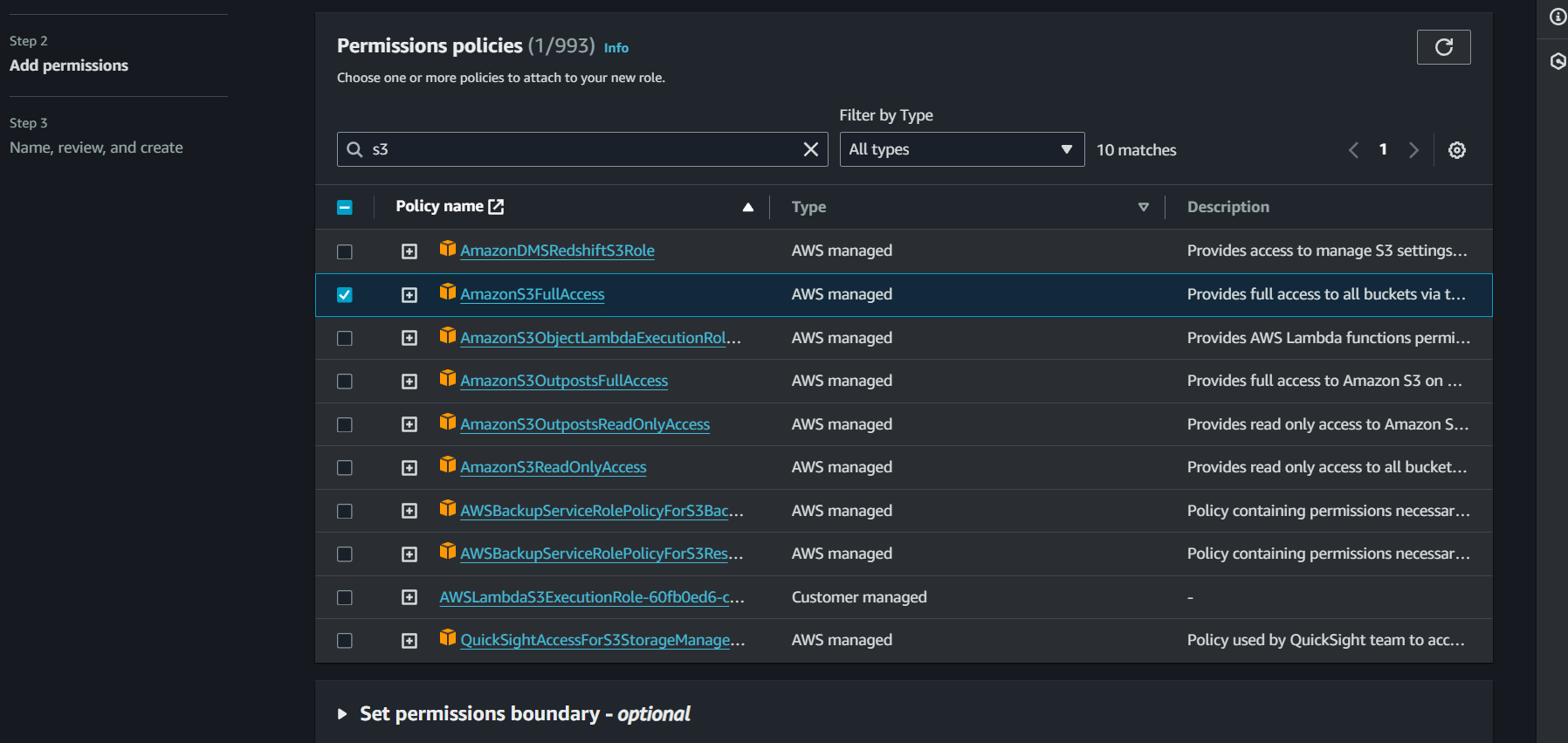
**Solution –**

For testing purposes, the time limit put by me was 1 day instead of 30 days. Added 2 files to the bucket between the duration of 1 hour each.

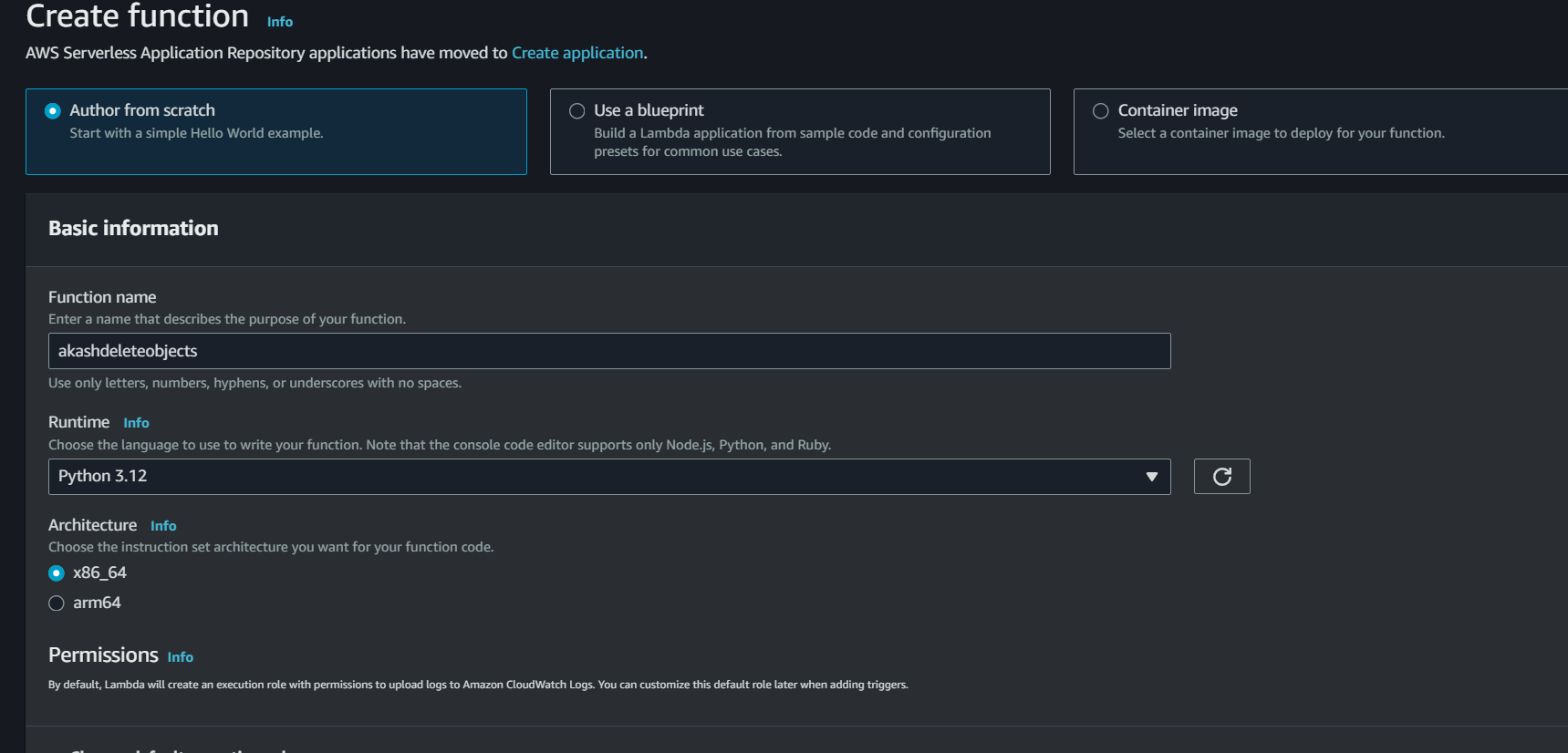
A screenshot of a computer

Description automatically generated

Create an IAM role for the Lambda function that gives full access for the S3 bucket



Create a lambda function using a python as the source code for implementing boto3.



The lambda code here shows how the datetime was converted to offset aware for easier comparison regarding the UTC timelines. If the time passed is greater than a day, the older will be removed.

A screenshot of a computer program

Description automatically generated

The output of the code being run successfully is also attached beneath.

A screenshot of a computer

Description automatically generated

After the code was successfully run, the latest file was the only one that remained.

A screenshot of a computer

Description automatically generated

**Question 2 – Automated instance management using AWS lambda and boto3**

**Objective –** In this, gain hands-on exp with AWS lambda and boto3, Amazon’s SDK for Python. You will create a lambda function that automatically manages instances based on tags.

**Task –** Automate the starting and stopping of EC2 instances based on tags.

**Solution –**

2 Instances were created using key pair and value tags named – Auto Start and Auto Stop each.

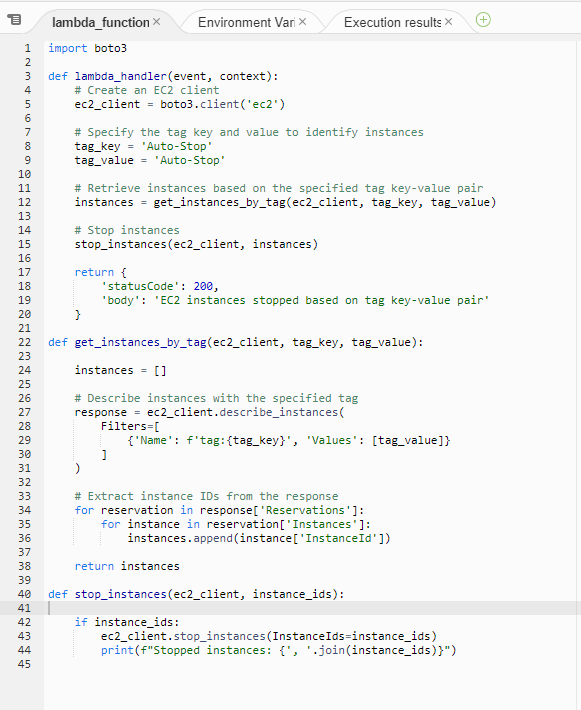
A screenshot of a computer

Description automatically generated

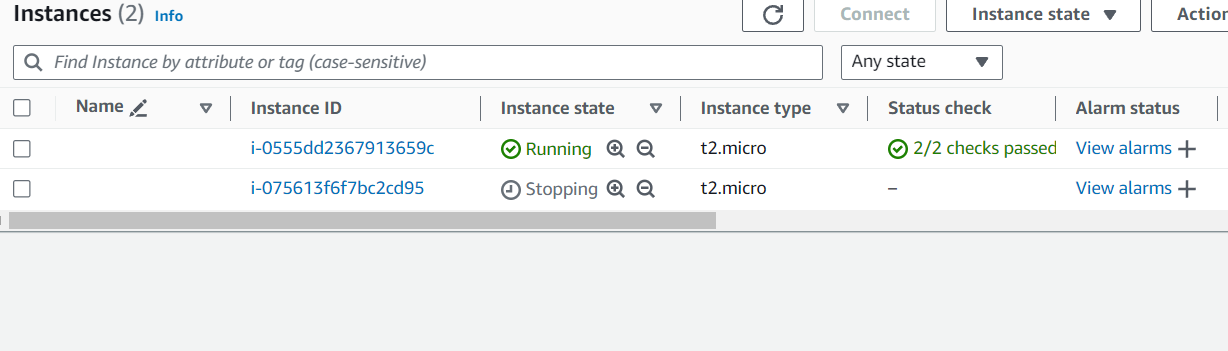
A white rectangular object with black lines

Description automatically generated

Then a lambda function was written for completing the task as mentioned.



The above function required me to mention the tag names of the ec2 instances. This checks the tag and calls out for the instance ID. Then, using the instance ID, the instance is either terminated or started.

****

The auto stop tag related instance was stopped via the Lambda code when it was run.

**Question 3 – Monitor unencrypted S3 buckets using AWS Lambda and Boto3**

**Objective –** To enhance your AWS Security posture by setting up a lambda function that detects any S3 bucket without server-side encryption.

**Task –** Automate the detection of S3 buckets that don’t have server-side encryption.

**Solution –**

Creating IAM roles and policies is the first step. Similar was done for the above questions as you can see in the screenshot below.

**A screenshot of a computer

Description automatically generated**

**This is the rule that is being used by the lambda that has certain permission policies attached. For this question, for easier access, I have provided the AmazonS3FullAccess.**

**According to recent policy changes by AWS, all the S3 buckets will be created with a minimum level of encryption hence no bucket will be without the server-side encryption.**

****

**Due to this, I created a lambda function that will check the encryption status of the buckets and list them each out.**

**A screenshot of a computer

Description automatically generated**

**This above is the lambda code that only monitors the encryption and no bucket can be terminated since all the buckets have an encryption level enabled.**

**A screenshot of a computer

Description automatically generated**

**Both the buckets that I had created are visible in the output and shows that the encryption status of both is true.**

**Assignment 3 – Monitor and alert high AWS billing using Lambda, boto3 and SNS**

**Objective – Create an automated alerting mechanism for when your AWS billing exceeds a certain threshold.**

**Task – Setup a lambda function to check your AWS billing amount daily, and if it exceeds a specified threshold, send an alert via SNS.**

**Solution –**

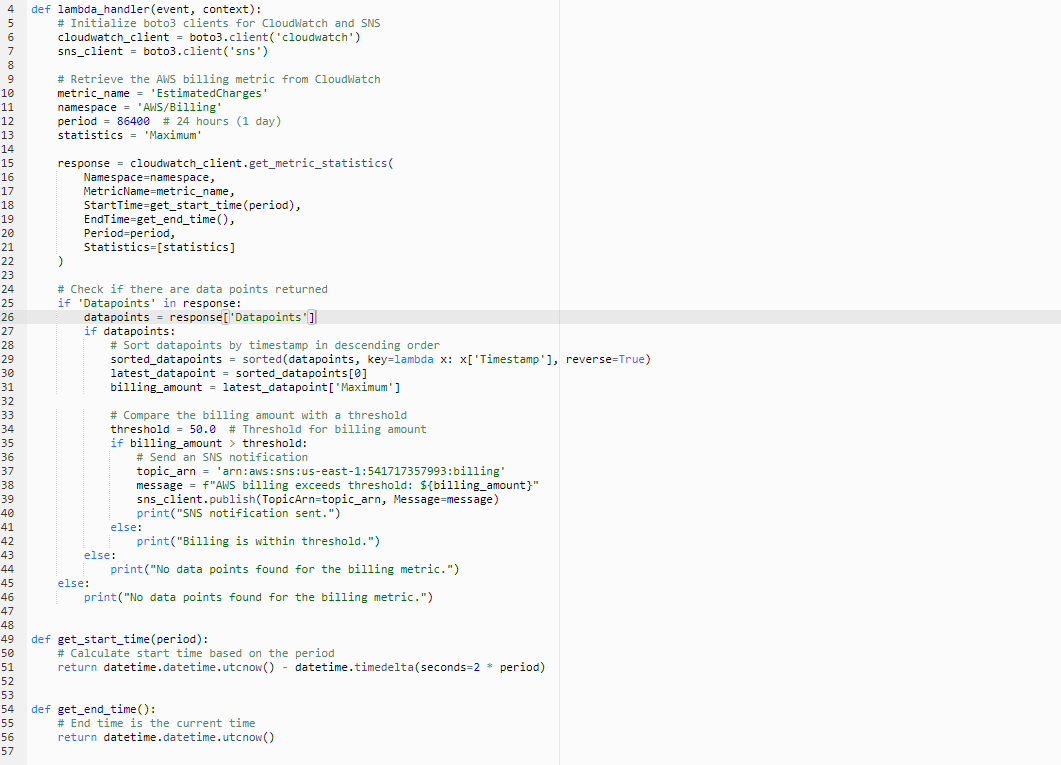
**Started off by creating an SNS topic to send a notification.**

**A screenshot of a computer

Description automatically generated**

**The topic was subscribed to my email address.**

**After the required IAM permissions were allotted, a lambda function was created.**

****

**The above lambda functions calls on the SNS topic that has been created. This also looks for the metric of the cloudwatch that is named “estimated billing”. Since I’ve been using free tier for my personal account, there are currently no billing metric or data that can be shown on the cloudwatch. Yet, the command was run.**

**A screenshot of a computer

Description automatically generated**

**As seen in the output, it says no data points were found for the billing metric.**

**The word document contains 4 questions completed for the serverless architecture assignment. The questions were chosen on a personal basis and does not follow the hierarchy of the assignment.**