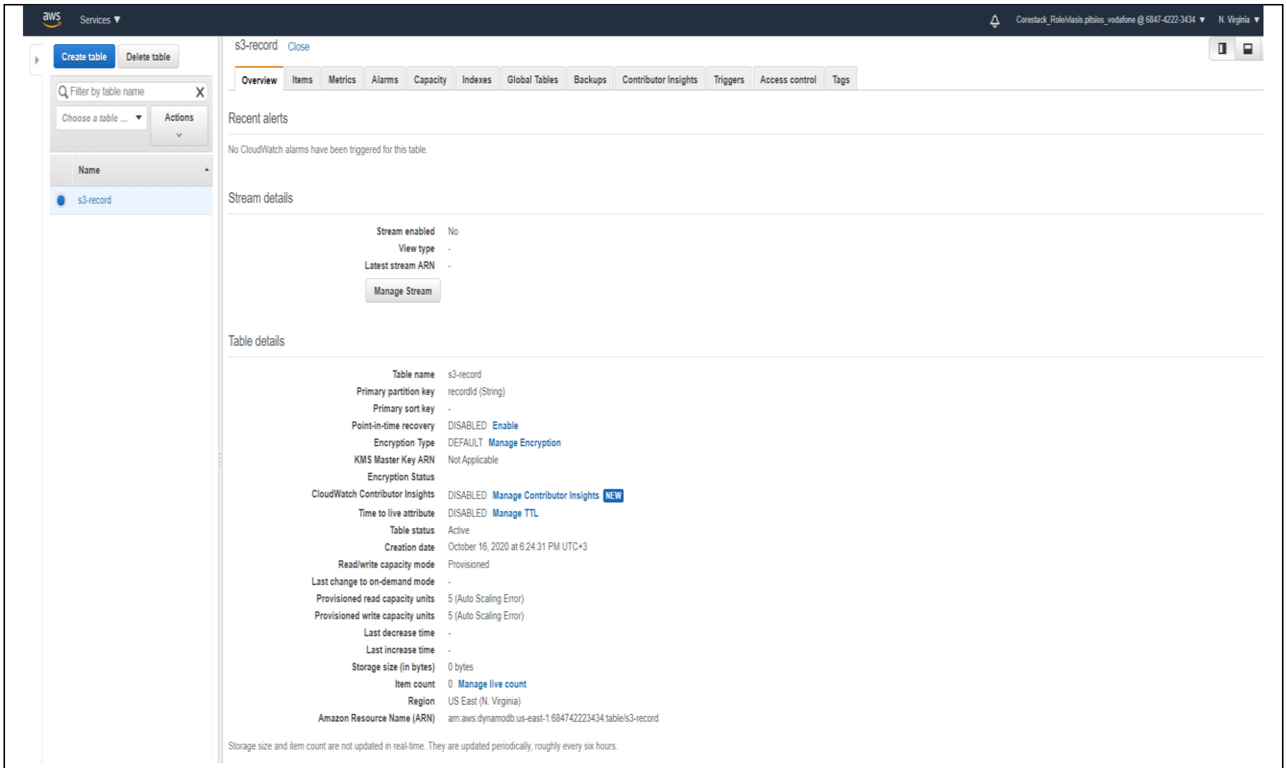
AWS Developer Associate Project

Name: Vlasis Pitsios

Email: vlasis.pitsios@vodafone.com

Company: Vodafone

Main project: Create the Amazon Dynamo DB Table, Add Items in to Table, Query it, Delete the table



Recent alerts

No CloudWatch alarms have been triggered for this table.

Stream details

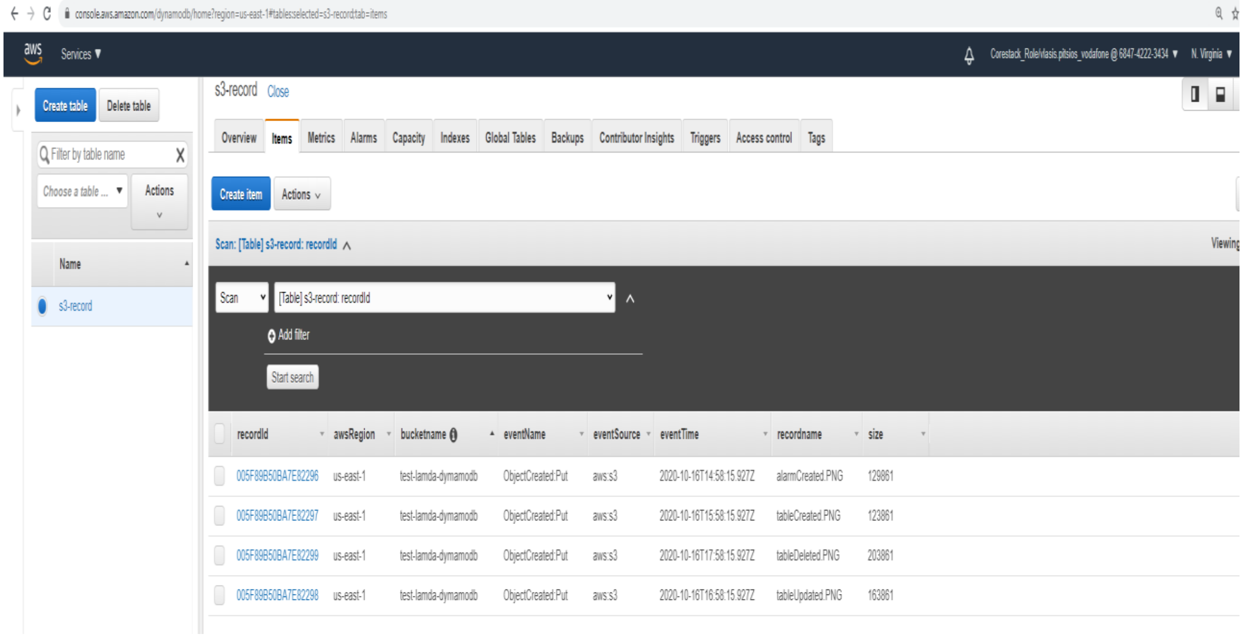
Stream enabled	No
View type	-
Latest stream ARN	-

Table details

Table name	s3-record
Primary partition key	recordId (String)
Primary sort key	-
Point-in-time recovery	DISABLED Enable
Encryption Type	DEFAULT Manage Encryption
KMS Master Key ARN	Not Applicable
Encryption Status	
CloudWatch Contributor Insights	DISABLED Manage Contributor Insights NEW
Time to live attribute	DISABLED Manage TTL
Table status	Active
Creation date	October 16, 2020 at 6:24:31 PM UTC+3
Read/write capacity mode	Provisioned
Last change to on-demand mode	-
Provisioned read capacity units	5 (Auto Scaling Error)
Provisioned write capacity units	5 (Auto Scaling Error)
Last decrease time	-
Last increase time	-
Storage size (in bytes)	0 bytes
Item count	0 Manage live count
Region	US East (N. Virginia)
Amazon Resource Name (ARN)	arn:aws:dynamodb:us-east-1:684742223434:table/s3-record

Storage size and item count are not updated in real-time. They are updated periodically, roughly every six hours.

Screenshot 1: DynamoDB table “s3-record” is created with partition key name as “recordId”



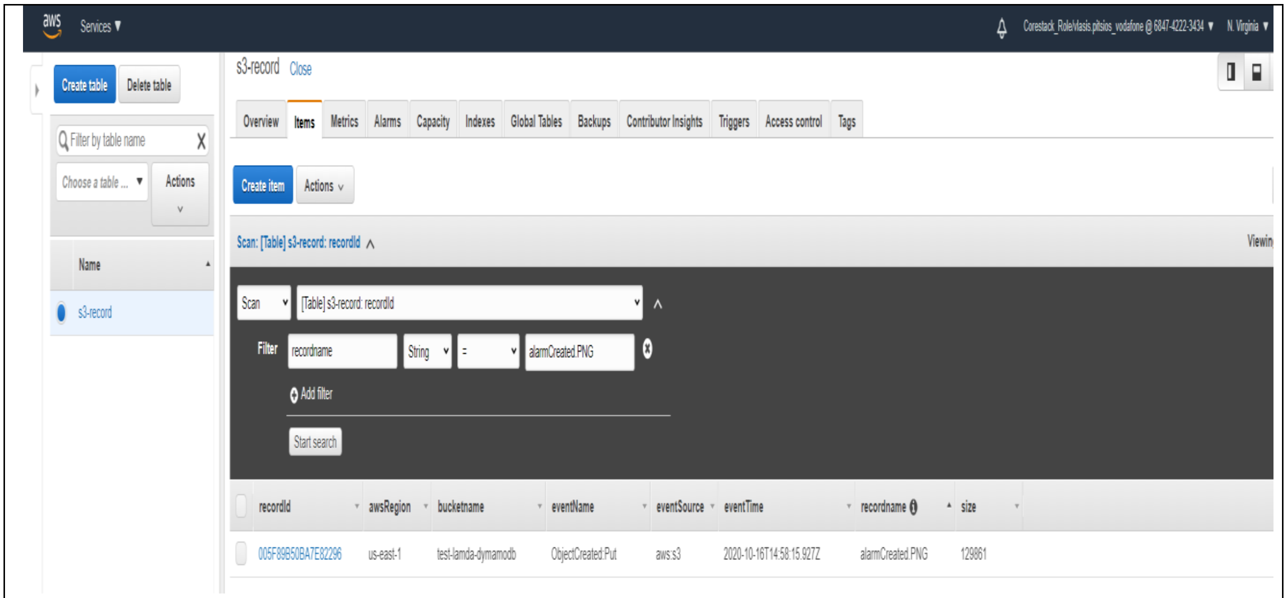
s3-record

Scan: [Table] s3-record: recordId

Scan | [Table] s3-record: recordId

recordId	awsRegion	bucketname	eventName	eventSource	eventTime	recordname	size
005F89B50BA7E82296	us-east-1	test-lamda-dymamodb	ObjectCreated:Put	aws:s3	2020-10-16T14:58:15.927Z	alarmCreated.PNG	129861
005F89B50BA7E82297	us-east-1	test-lamda-dymamodb	ObjectCreated:Put	aws:s3	2020-10-16T15:58:15.927Z	tableCreated.PNG	123861
005F89B50BA7E82299	us-east-1	test-lamda-dymamodb	ObjectCreated:Put	aws:s3	2020-10-16T17:58:15.927Z	tableDeleted.PNG	203861
005F89B50BA7E82298	us-east-1	test-lamda-dymamodb	ObjectCreated:Put	aws:s3	2020-10-16T16:58:15.927Z	tableUpdated.PNG	163861

Screenshot 2: Four items are inserted into the table with attributes (recordId, awsRegion, bucketname, eventName, eventSource, eventTime, recordname, size)



s3-record

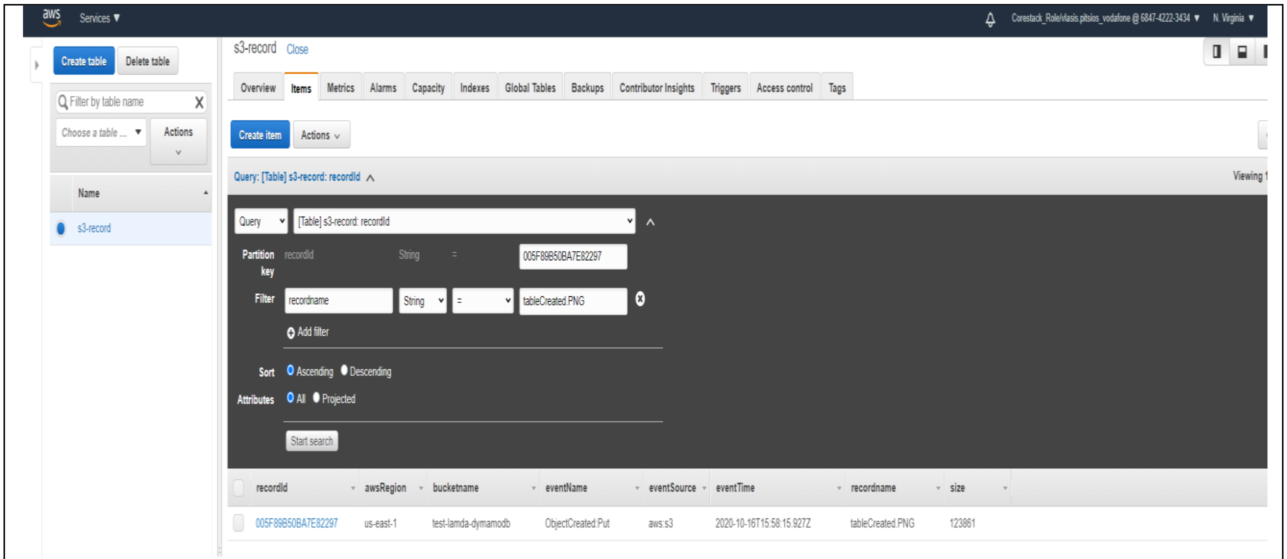
Scan: [Table] s3-record: recordId

Scan | [Table] s3-record: recordId

Filter recordname | String | = | alarmCreated.PNG

recordId	awsRegion	bucketname	eventName	eventSource	eventTime	recordname	size
005F89B50BA7E82296	us-east-1	test-lamda-dymamodb	ObjectCreated:Put	aws:s3	2020-10-16T14:58:15.927Z	alarmCreated.PNG	129861

Screenshot 3: Scan search on table “s3-record”. Scan search is a full table scan, so this is **not** efficient.



Query: [Table] s3-record: recordId

Query | [Table] s3-record: recordId

Partition key: recordId String = 005F89B50BA7E82297

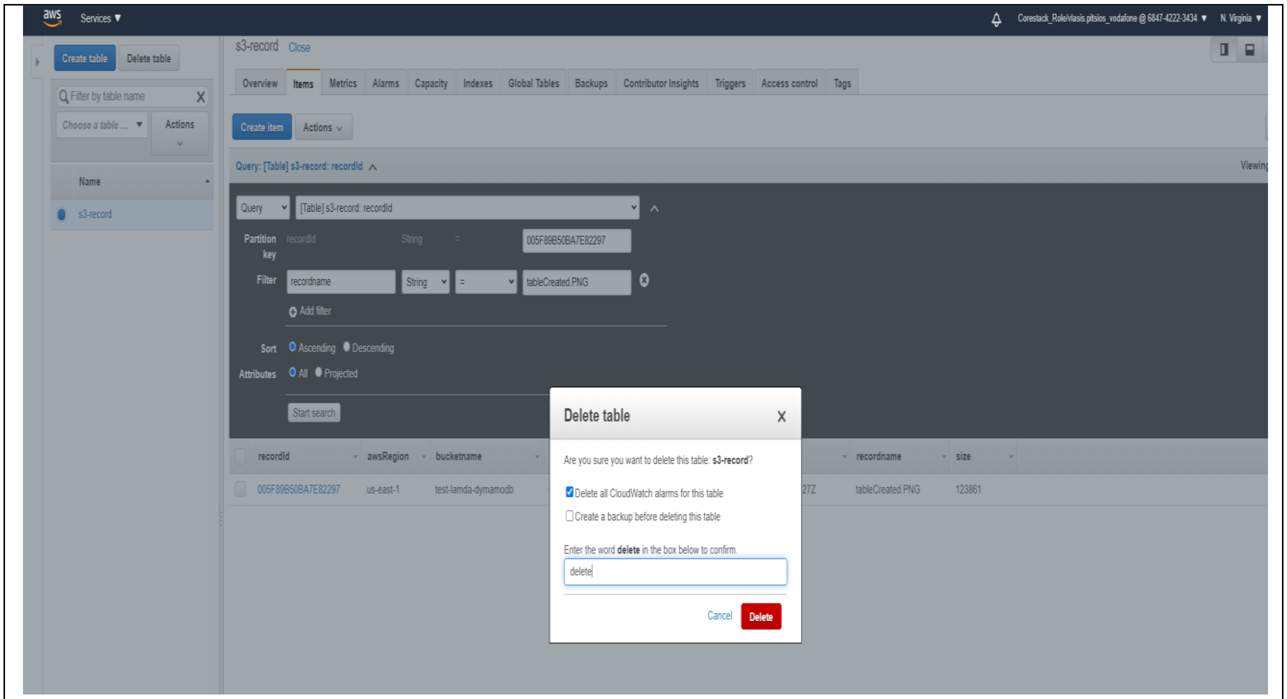
Filter: recordname String = tableCreated.PNG

Sort: ● Ascending ○ Descending

Attributes: ● All ○ Projected

recordId	awsRegion	bucketname	eventName	eventSource	eventTime	recordname	size
005F89B50BA7E82297	us-east-1	test-lamda-dymamodb	ObjectCreated:Put	aws:s3	2020-10-16T15:58:15.927Z	tableCreated.PNG	123861

Screenshot 4: Query search on table "s3-record". This is better in terms of performance as this search creates an index and is faster.



Delete table

Are you sure you want to delete this table: **s3-record**?

☑ Delete all CloudWatch alarms for this table

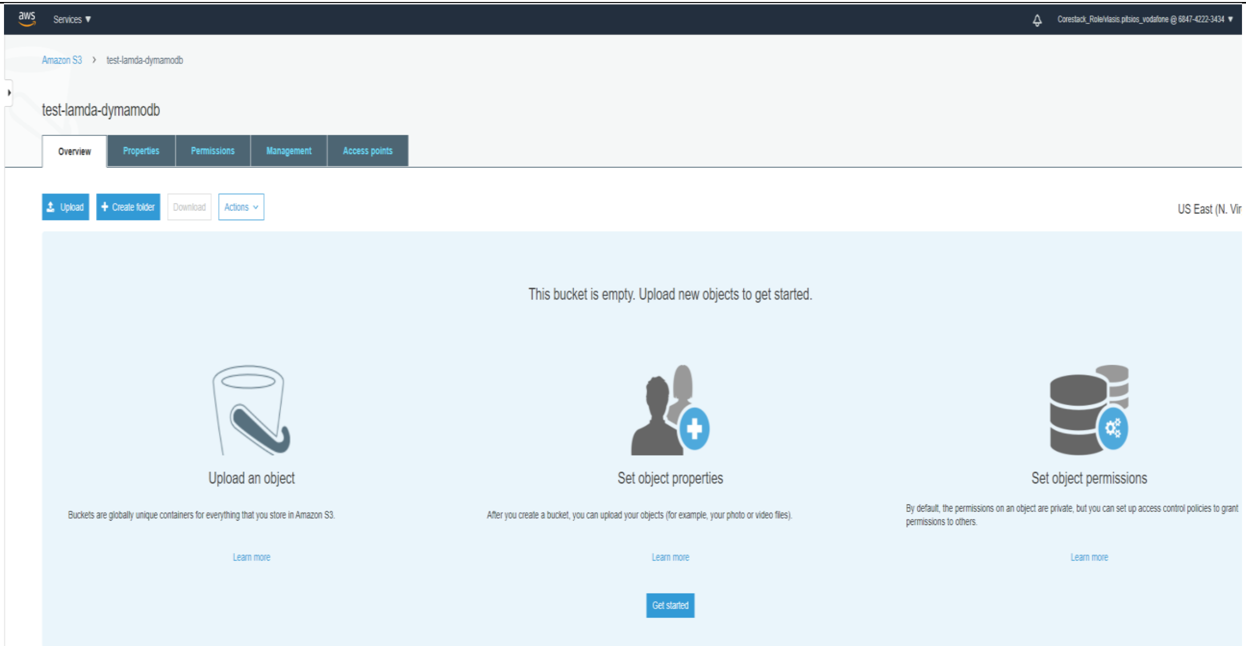
☐ Create a backup before deleting this table

Enter the word **delete** in the box below to confirm.

delete

Screenshot 5: Deletion of the table "s3-record".

Optional project: Upload a file on S3 and a lamda function should be triggered that will insert some data on a table in DynamoDB.



This bucket is empty. Upload new objects to get started.

Upload an object

Buckets are globally unique containers for everything that you store in Amazon S3.

Learn more

Set object properties

After you create a bucket, you can upload your objects (for example, your photo or video files).

Learn more

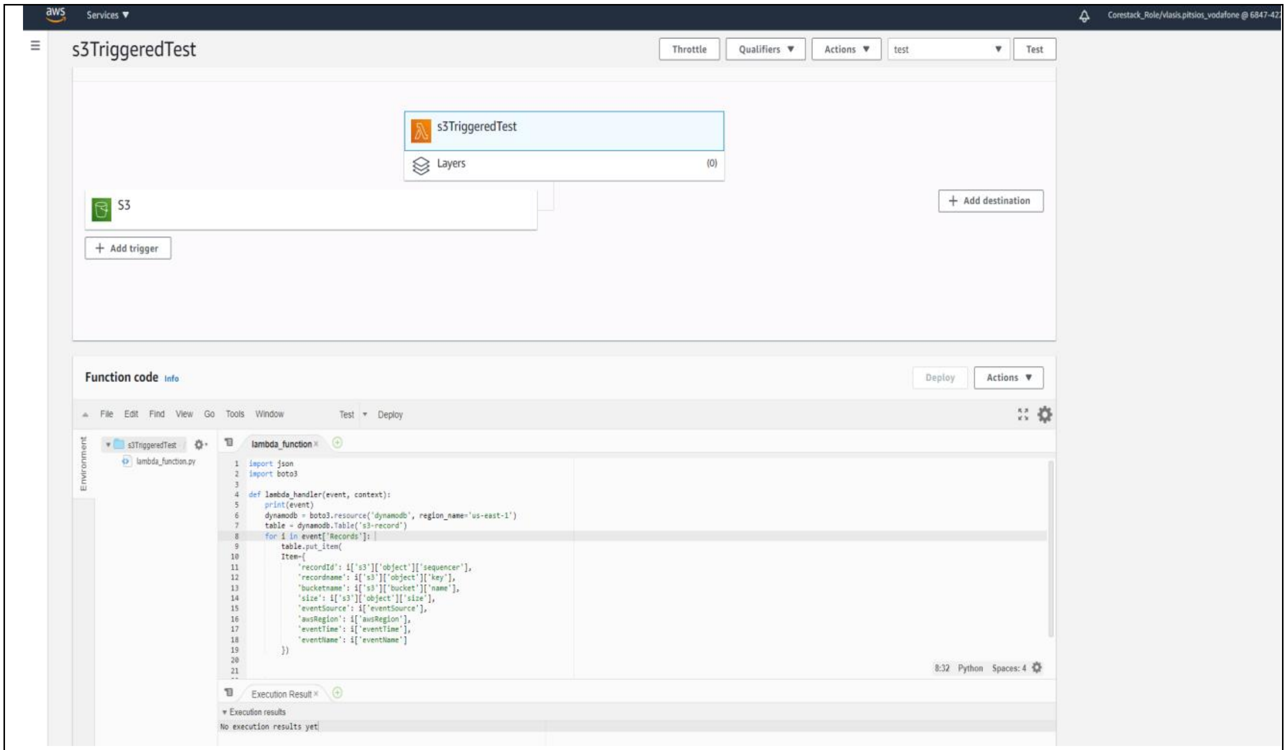
Set object permissions

By default, the permissions on an object are private, but you can set up access control policies to grant permissions to others.

Learn more

Get started

Screenshot 1: Create an s3 bucket called “test-lamda-dynamodb”



```
import json
import boto3

def lambda_handler(event, context):
    print(event)
    dynamodb = boto3.resource('dynamodb', region_name='us-east-1')
    table = dynamodb.Table('s3-record')
    for i in event['Records']:
        table.put_item(
        Item={
            'recordId': i['s3']['object']['sequencer'],
            'recordname': i['s3']['object']['key'],
            'bucketname': i['s3']['bucket']['name'],
            'size': i['s3']['object']['size'],
            'eventSource': i['eventSource'],
            'awsRegion': i['awsRegion'],
            'eventTime': i['eventTime'],
            'eventName': i['eventName']
        })
```

```
No execution results yet
```

Screenshot 2: Create a lamda function that will accept an event from the s3 bucket when a file is uploaded and will send some details from the event on a dynamoDB table.

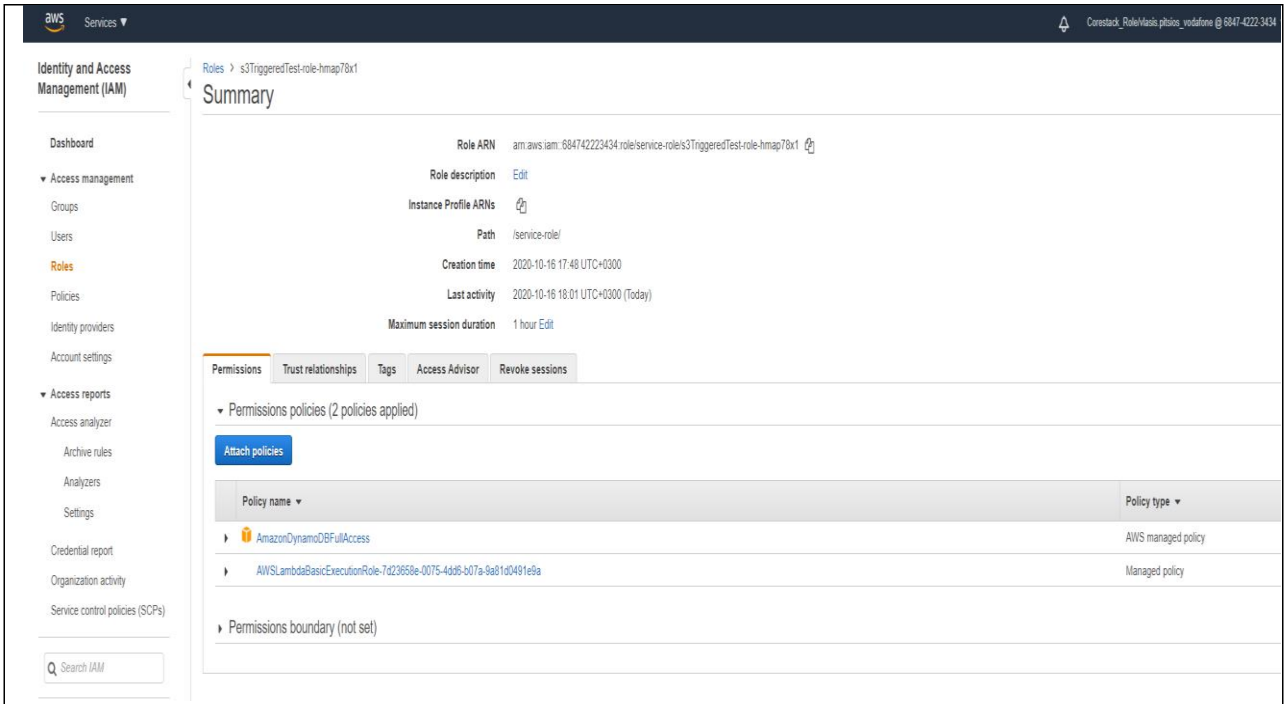
Python code:

```
import json
import boto3

def lambda_handler(event, context):
    print(event)
    dynamodb = boto3.resource('dynamodb', region_name='us-east-1')
    table = dynamodb.Table('s3-record')
    for i in event['Records']:
        table.put_item(
        Item={
            'recordId': i['s3']['object']['sequencer'],
            'recordname': i['s3']['object']['key'],
            'bucketname': i['s3']['bucket']['name'],
            'size': i['s3']['object']['size'],
            'eventSource': i['eventSource'],
            'awsRegion': i['awsRegion'],
            'eventTime': i['eventTime'],
            'eventName': i['eventName']
        })
```

S3 event which triggers the lamda function

```
{
 "Records": [
  {
   "eventVersion": "2.1",
   "eventSource": "aws:s3",
   "awsRegion": "us-east-1",
   "eventTime": "2020-10-16T14:58:15.927Z",
   "eventName": "ObjectCreated:Put",
   "userIdentity": {
    "principalId": "AWS:AROAZ63OREZFEFUGBGZCA:vlasis.pitsios_vodafone"
   },
   "requestParameters": {
    "sourceIPAddress": "91.140.88.253"
   },
   "responseElements": {
    "x-amz-request-id": "09F581FA2632810B",
    "x-amz-id-2": "192F98OdOzRZHDYSVHSsFNApzLeCSZHWEGlJeY8SNblpcKTtOV4SZ95GBN6Eh79cXZ4a+jAV910YqtYE0FW0iaChTM63IGjf"
   },
   "s3": {
    "s3SchemaVersion": "1.0",
    "configurationId": "OTUzMDc1NDQtMTBhMS00MTBhLTg0NjYtZDQ2YTM3YzM3MWMx",
    "bucket": {
     "name": "test-lamda-dymamodb",
     "ownerIdentity": {
      "principalId": "A3JTXMCNICSX2"
     },
     "arn": "arn:aws:s3:::test-lamda-dymamodb"
    },
    "object": {
     "key": "alarmCreated.PNG",
     "size": 129861,
     "eTag": "3bab12e1b1eac319317c43ba5035e26f",
     "sequencer": "005F89B50BA7E82296"
    }
   }
  }
 ]
}
```



Roles › s3TriggeredTest-role-hmap78x1

Summary

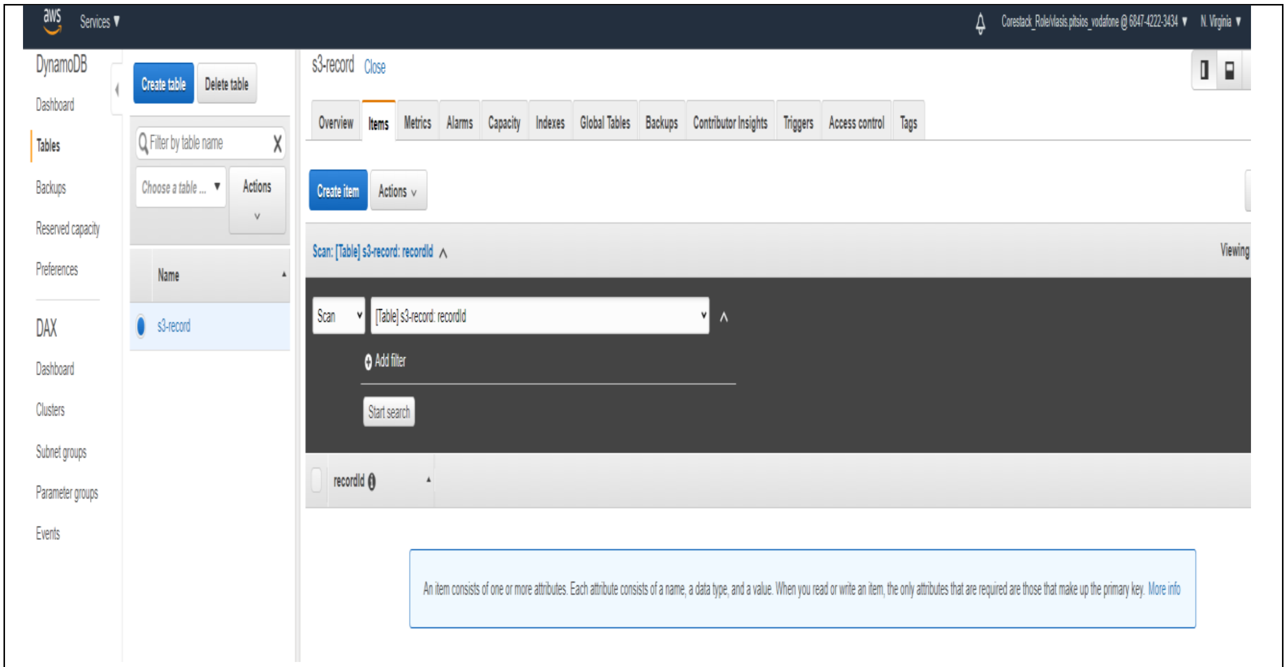
Role ARN	arn:aws:iam::684742223434:role/service-role/s3TriggeredTest-role-hmap78x1
Role description	Edit
Instance Profile ARNs	
Path	/service-role/
Creation time	2020-10-16 17:48 UTC+0300
Last activity	2020-10-16 18:01 UTC+0300 (Today)
Maximum session duration	1 hour Edit

▾ Permissions policies (2 policies applied)

Policy name ▾	Policy type ▾
AmazonDynamoDBFullAccess	AWS managed policy
AWSLambdaBasicExecutionRole-7d23658e-0075-4dd6-b07a-9a81d0491e9a	Managed policy

▸ Permissions boundary (not set)

Screenshot 3: Lamda function should have a “AmazonDynamoDBFullAccess” policy in order to access DynamoDB.



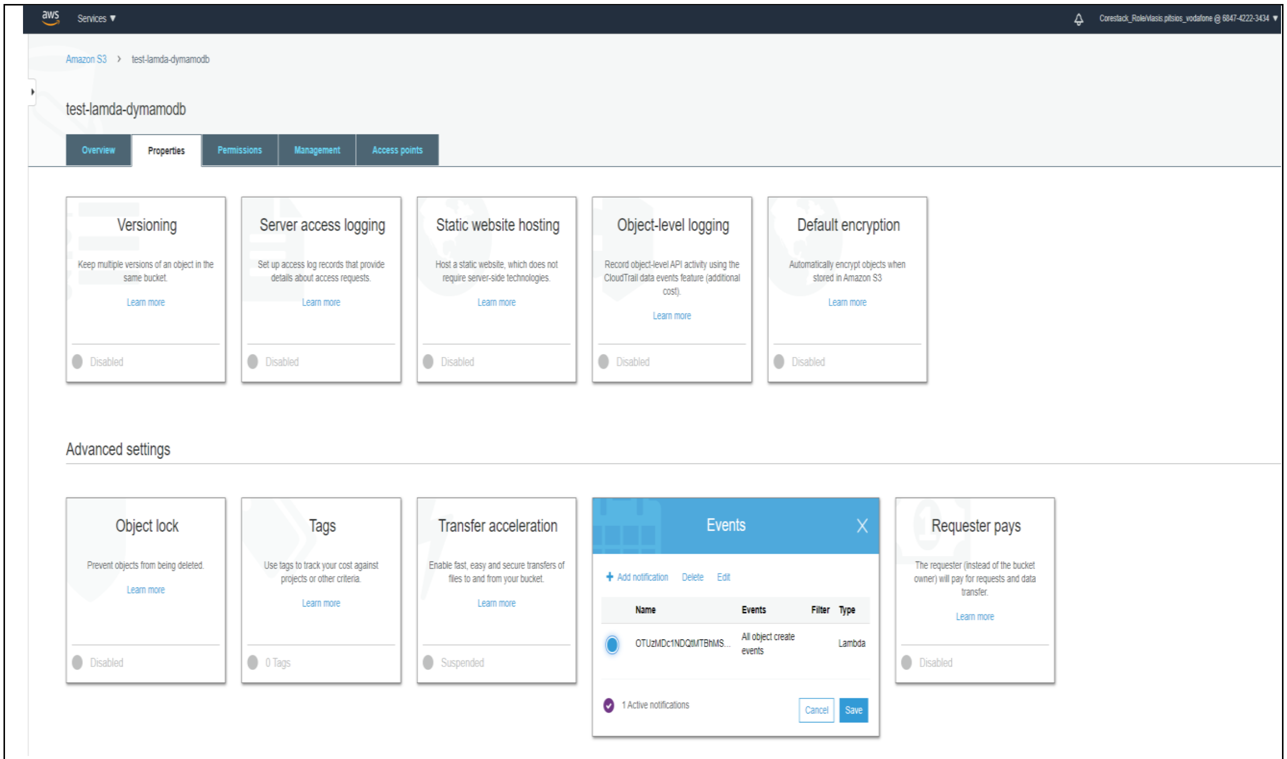
s3-record

Scan: [Table] s3-record: recordId

Scan | [Table] s3-record: recordId

An item consists of one or more attributes. Each attribute consists of a name, a data type, and a value. When you read or write an item, the only attributes that are required are those that make up the primary key. More info

Screenshot 4: Create table “s3-record” in DynamoDB, because lamda function needs to store on this table information about the uploaded file.



Amazon S3 › test-lamda-dymamodb

test-lamda-dymamodb

Versioning
Keep multiple versions of an object in the same bucket.
Learn more
Disabled

Server access logging
Set up access log records that provide details about access requests.
Learn more
Disabled

Static website hosting
Host a static website, which does not require server-side technologies.
Learn more
Disabled

Object-level logging
Record object-level API activity using the CloudTrail data events feature (additional cost).
Learn more
Disabled

Default encryption
Automatically encrypt objects when stored in Amazon S3
Learn more
Disabled

Advanced settings

Object lock
Prevent objects from being deleted.
Learn more
Disabled

Tags
Use tags to track your cost against projects or other criteria.
Learn more
0 Tags

Transfer acceleration
Enable fast, easy and secure transfers of files to and from your bucket.
Learn more
Suspended

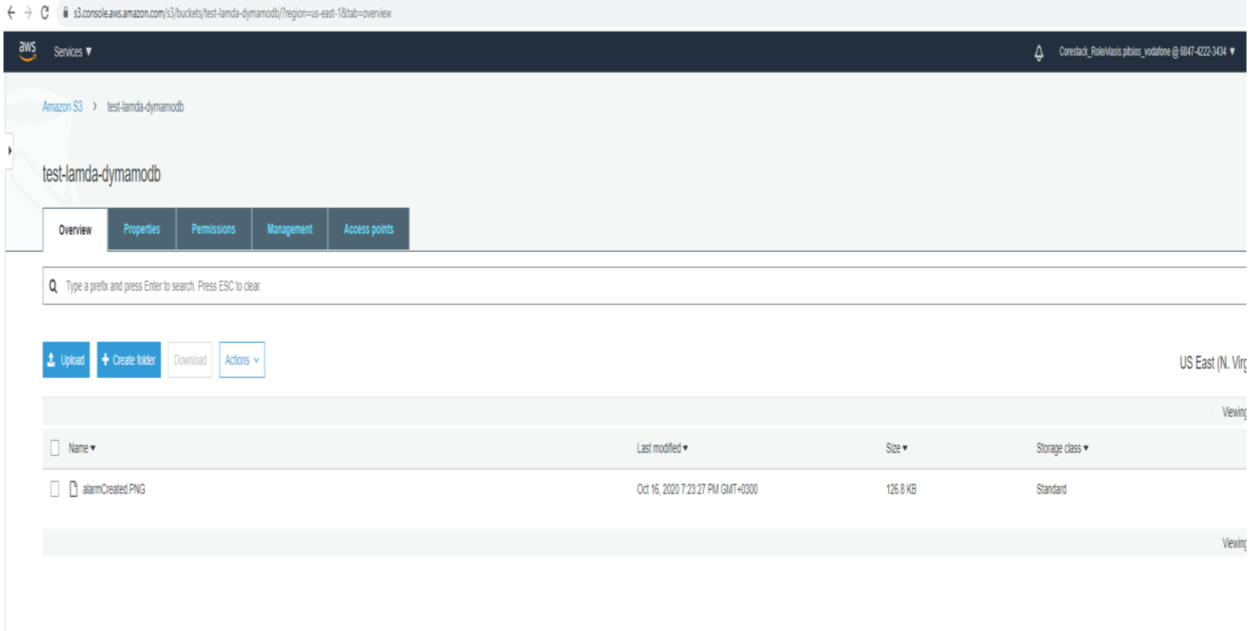
Events

Name	Events	Filter	Type
OTUzMDc1NDQtMTBhMS...	All object create events		Lambda

1 Active notifications

Requester pays
The requester (instead of the bucket owner) will pay for requests and data transfer.
Learn more
Disabled

Screenshot 5: S3 should create an event every time a new file is uploaded and trigger the lamda function.



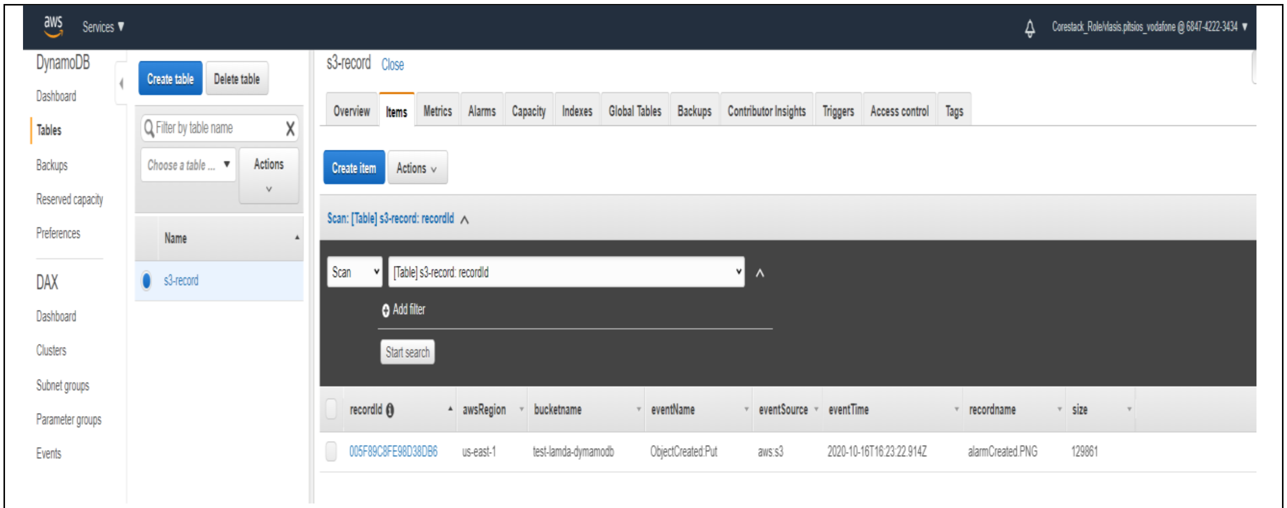
s3.console.aws.amazon.com/s3/buckets/test-lamda-dymamodb/?region=us-east-1&tab=overview

Amazon S3 › test-lamda-dymamodb

test-lamda-dymamodb

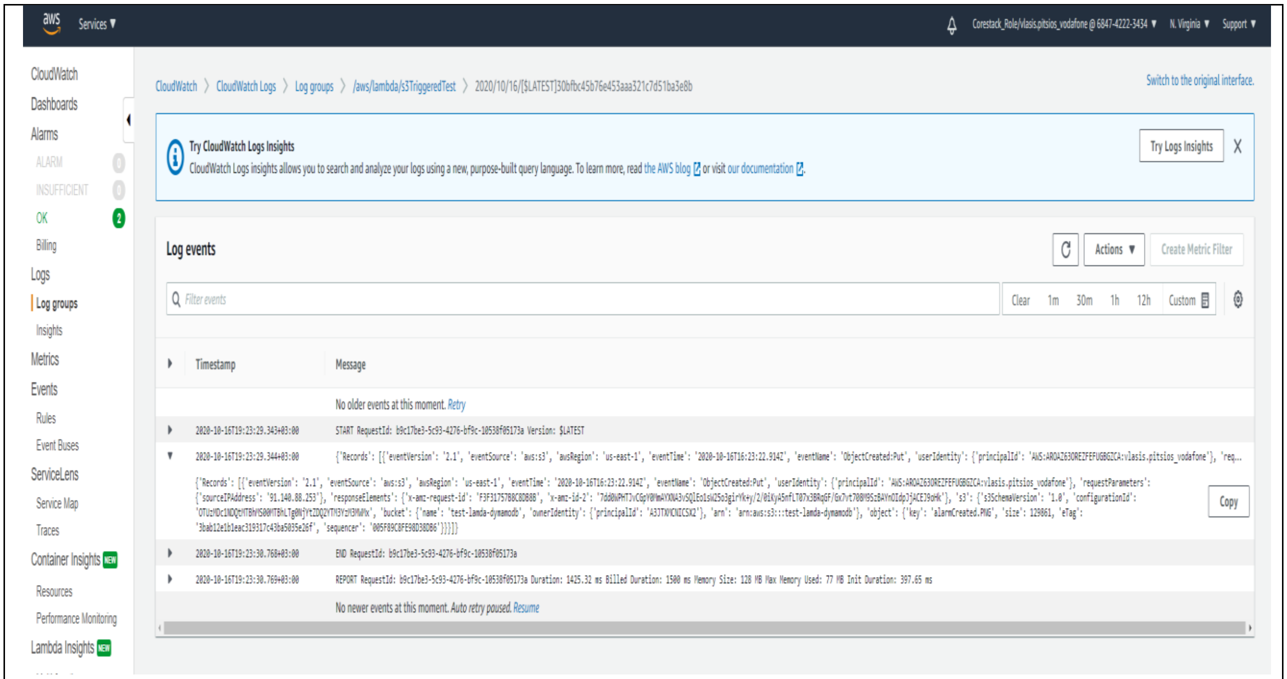
Name ▾	Last modified ▾	Size ▾	Storage class ▾
alarmCreated.PNG	Oct 16, 2020 7:23:27 PM GMT+0300	126.8 KB	Standard

Screenshot 6: A file “alarmCreated.PNG” is uploaded on the S3 bucket.



recordId	awsRegion	bucketname	eventName	eventSource	eventTime	recordname	size
005F89C8FE98D38DB6	us-east-1	test-lamda-dymamodb	ObjectCreated:Put	aws:s3	2020-10-16T16:23:22.914Z	alarmCreated.PNG	129861

Screenshot 7: Information about the file “alarmCreated.PNG” are inserted into the DynamoDB table “s3-record”.


aws Services
Corestack_Role/vlasis.pitsios_vodafone @ 6847-4222-3434
N. Virginia
Support
CloudWatch
Dashboards
Alarms
ALARM
INSUFFICIENT
OK
Billing
Logs
Log groups
Insights
Metrics
Events
Rules
Event Buses
ServiceLens
Service Map
Traces
Container Insights NEW
Resources
Performance Monitoring
Lambda Insights NEW
CloudWatch > CloudWatch Logs > Log groups > /aws/lambda/s3TriggeredTest > 2020/10/16/[$LATEST]30bfbc45b76e453aa321c7d51ba3e8b
Switch to the original interface.
Try CloudWatch Logs Insights
CloudWatch Logs Insights allows you to search and analyze your logs using a new, purpose-built query language. To learn more, read the AWS blog or visit our documentation.
Try Logs Insights
Log events
Actions
Create Metric Filter
Filter events
Clear
1m
30m
1h
12h
Custom
Copy

```
Timestamp                     Message
                              No older events at this moment. Retry
2020-10-16T19:23:29.343+03:00 START RequestId: b9c17be3-5c93-4276-bf9c-10538f05173a Version: $LATEST
2020-10-16T19:23:29.344+03:00 {'Records': [{'eventVersion': '2.1', 'eventSource': 'aws:s3', 'awsRegion': 'us-east-1', 'eventTime': '2020-10-16T16:23:22.914Z', 'eventName': 'ObjectCreated:Put', 'userIdentity': {'principalId': 'AWS:AROAZ63OREZFEFUGBGZCA:vlasis.pitsios_vodafone'}, 'req...
{'Records': [{'eventVersion': '2.1', 'eventSource': 'aws:s3', 'awsRegion': 'us-east-1', 'eventTime': '2020-10-16T16:23:22.914Z', 'eventName': 'ObjectCreated:Put', 'userIdentity': {'principalId': 'AWS:AROAZ63OREZFEFUGBGZCA:vlasis.pitsios_vodafone'}, 'requestParameters':
{'sourceIPAddress': '91.140.88.253'}, 'responseElements': {'x-amz-request-id': 'F3F31757B8C8D88B', 'x-amz-id-2': '7dd0WPHTJvCGpY0MmAYXNA3vSQlEo1sW25o3girYk+y/2/0iKyA5nfLT07x3BRqGF/Gx7vt708M9SzBAYnOIdpJjACEJ9oHk'}, 's3': {'s3SchemaVersion': '1.0', 'configurationId':
'OTUzMDc1NDQtMTBhYS00MTBhLTg0NjYtZDQ2YTM3YzM3MWMx', 'bucket': {'name': 'test-lamda-dymamodb', 'ownerIdentity': {'principalId': 'A3JTXMCNICSX2'}, 'arn': 'arn:aws:s3:::test-lamda-dymamodb'}, 'object': {'key': 'alarmCreated.PNG', 'size': 129861, 'eTag':
'3bab12e1b1eac319317c43ba5035e26f', 'sequencer': '005F89C8FE98D38DB6'}}}]}
2020-10-16T19:23:30.768+03:00 END RequestId: b9c17be3-5c93-4276-bf9c-10538f05173a
2020-10-16T19:23:30.769+03:00 REPORT RequestId: b9c17be3-5c93-4276-bf9c-10538f05173a Duration: 1425.32 ms Billed Duration: 1500 ms Memory Size: 128 MB Max Memory Used: 77 MB Init Duration: 397.65 ms
                              No newer events at this moment. Auto retry paused. Resume
```

Screenshot 8: Logs from AWS Cloudwatch. Lamda is triggered when the file “alarmCreated.PNG” was uploaded. The event is created and logged.