

Serverless Architectures with Amazon DynamoDB and Amazon Kinesis Streams with AWS Lambda

-Sonal Varshney

1. Create an Amazon Kinesis Stream

Resources

Resource Groups

EC2

S3

IAM

sonalvarshney

N. Virginia

Support

Create Kinesis stream

Kinesis stream name*

Acceptable characters are uppercase and lowercase letters, numbers, underscores, hyphens, and periods.

Shards

A shard is a unit of throughput capacity. Each shard ingests up to 1MB/sec and 1000 records/sec, and emits up to 2MB/sec. To accommodate for higher or lower throughput, the number of shards can be modified after the Kinesis stream is created using the API. [Learn more](#)

Producers → Kinesis stream (Shard, Shard) → Consumers

► [Estimate the number of shards you'll need](#)

Number of shards*

Activate Windows
Go to Settings to activate Windows.

2. Create Lambda Function

aws Services Resource Groups EC2 S3 IAM

Lambda > Functions > Create function > Using blueprint kinesis-process-record-python

Basic information

Function name

Execution role
Choose a role that defines the permissions of your function. To create a custom role, go to the [IAM console](#).

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.

[View the lambda_basic_execution role](#) on the IAM console.

Python 2.7

```
1 from __future__ import print_function
2
3 import base64
4 import json
5
6 print('Loading function')
7
8
9 def lambda_handler(event, context):
10     #print("Received event: " + json.dumps(event, indent=2))
11     for record in event['Records']:
12         # Kinesis data is base64 encoded so decode here
13         payload = base64.b64decode(record['kinesis']['data'])
14         print("Decoded payload: " + payload)
15     return 'Successfully processed {} records.'.format(len(event['Records']))
16
```

3. Test the function

Execution result: succeeded (logs)

Details

The section below shows the result returned by your function execution.

"Successfully processed 1 records."

Summary

Code SHA-256

//ny7mgRryaQZTHVlvBPP7HYDrYsBftf7dwGyWT+L/I=

Duration

0.25 ms

Resources configured

128 MB

Request ID

c1d2d804-d7f6-432a-ba01-5a637d67717a

Billed duration

100 ms

Max memory used

42 MB

Log output

The section below shows the logging calls in your code. These correspond to a single row within the CloudWatch log group corresponding to this Lambda function. [Click here](#) to view the CloudWatch log group.

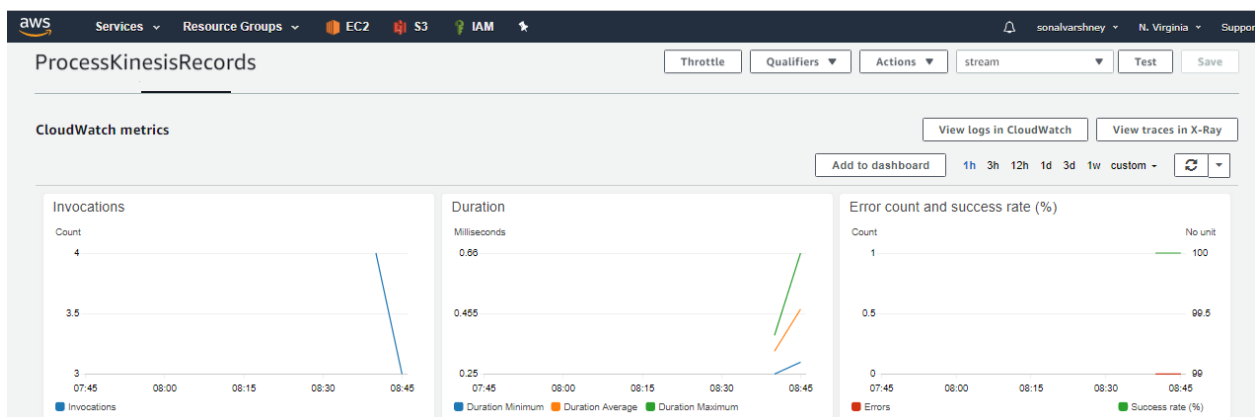
START RequestId: c1d2d804-d7f6-432a-ba01-5a637d67717a Version: \$LATEST

Decoded payload: Hello, this is a test 123.

END RequestId: c1d2d804-d7f6-432a-ba01-5a637d67717a

REPORT RequestId: c1d2d804-d7f6-432a-ba01-5a637d67717a Duration: 0.25 ms Billed Duration: 100 ms Memory Size: 128 MB Max Memory Used: 42 MB

4. Monitoring



5. Create DynamoDB tables (GameScoreRecords, GameScoresByUser)

The screenshot shows the AWS IAM console interface for creating a new DynamoDB table. On the left, a sidebar contains a 'Create table' button and a search bar labeled 'Filter by table name'. Below the search bar, a list of tables is shown, with 'GameScoresByUser' selected. The main panel displays the 'GameScoresByUser' table creation progress, with a status bar indicating 'Table is being created'. Below this, there are sections for 'Recent alerts' (showing no CloudWatch alarms) and 'Stream details' (showing stream enabled status as 'No').

6. Create Lambda Function

The screenshot shows the AWS Lambda console interface for the 'AggregateScoresByUser' function. The top navigation bar includes the AWS logo, 'Services', 'Resource Groups', and various service icons (EC2, S3, IAM, etc.). The main panel displays the function configuration, including the function name 'AggregateScoresByUser', its ARN, and a 'Designer' section. The 'Designer' section shows a visual representation of the function's configuration, including a 'DynamoDB' trigger (labeled '(2)') and a 'Layers' section (labeled '(0)'). The 'Designer' section also includes a 'Save' button and a 'Test' button.

7. Configure test event

Configure test event



A function can have up to 10 test events. The events are persisted so you can switch to another computer or web browser and test your function with the same events.

- ☒ Create new test event
- ☐ Edit saved test events

Event template

Hello World

Event name

score

```
1 {  
2   "Records": [  
3     {  
4       "eventID": "1",  
5       "eventVersion": "1.0",  
6       "dynamodb": {  
7         "Keys": {  
8           "RecordID": {  
9             "S": "2"  
10          }  
11        },  
12        "NewImage": {  
13          "RecordID": {  
14            "S": "2"
```

8. Verify in DynamoDB

Create tableDelete table

Filter by table name X

Choose a table ... Actions

Name
<input type="radio"/> GameScoreRecords
<input checked="" type="radio"/> GameScoresByUser

GameScoresByUserClose

OverviewItemsMetricsAlarmsCapacityIndexesC

Create itemActions

Scan: [Table] GameScoresByUser: Username ^

Scan [Table] GameScoresByUser: Username

+ Add filter

Start search

Username	Score
<input type="checkbox"/> Jane Doe	100

9. Trigger Update

Create item

Tree

Item {3}

RecordID Number : 11

Username String : Sonal

Score Number : 150

Cancel

Save

GameScoreRecords

GameScoresByUser

Scan

[Table] GameScoresByUser: Username

+ Add filter

Start search

	Username	Score
<input type="checkbox"/>	Jane Doe	100
<input type="checkbox"/>	Sonal	150