

AWS POC 1 - Load Government Debt Total % of GDP

Overview



The Lambda function will trigger when a new file is uploaded to the S3 bucket. The Lambda functions logs to the Cloudwatch log the file details and whether it is part of the World Bank *GC.DOD.TOTL.GD.ZS* dataset based on the file location.

S3 Bucket Setup

Bucket overview	
AWS Region Asia Pacific (Sydney) ap-southeast-2	Amazon Resource Name (ARN) arn:aws:s3::poc-staging

Setup a Standard S3 Bucket with default options.

Lambda Setup

S3 Bucket Role

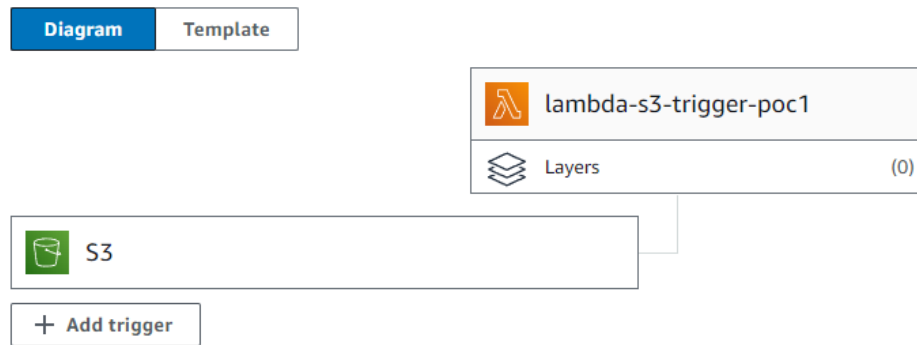
Create a new role with permissions to access the S3 bucket.

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "logs:PutLogEvents",
8         "logs:CreateLogGroup",
9         "logs:CreateLogStream"
10      ],
11      "Resource": "arn:aws:logs:*:*:*"
12    },
13    {
14      "Effect": "Allow",
15      "Action": [
16        "s3:GetObject"
17      ],
18      "Resource": "arn:aws:s3:::*/*"
19    }
20  ]
21 }
```

Role saved as: *s3-trigger-aws-poc1*

Lambda Function

Create a new Lambda Function and associate the S3 Bucket and the S3 Bucket Role with the new function.



Set the function runtime to be Java 21 and upload the *fat jar* to AWS.

Runtime settings Info			Edit	Edit runtime management configuration
Runtime Java 21 Runtime management configuration	Handler Info com.mattswart.aws.LambdaFuncHandler::handleRequest	Architecture Info x86_64		

Test Event

Create a Test Event in AWS and save the test event.

```
1 {
2   "Records": [
3     {
4       "eventVersion": "2.0",
5       "eventSource": "aws:s3",
6       "awsRegion": "ap-southeast-2",
7       "eventTime": "1970-01-01T00:00:00.000Z",
8       "eventName": "ObjectCreated:Put",
9       "userIdentity": {
10        "principalId": "EXAMPLE"
11      },
12      "requestParameters": {
13        "sourceIPAddress": "127.0.0.1"
14      },
15      "responseElements": {
16        "x-amz-request-id": "EXAMPLE123456789",
17        "x-amz-id-2": "EXAMPLE123/5678abcdefghijklmbdaisawesome/mnopqrstuvwxyzABCDEFGH"
18      },
19      "s3": {
20        "s3SchemaVersion": "1.0",
21        "configurationId": "testConfigRule",
22        "bucket": {
23          "name": "poc-staging",
24          "ownerIdentity": {
25            "principalId": "EXAMPLE"
26          },
27          "arn": "arn:aws:s3:::poc-staging"
28        },
29        "object": {
30          "key": "world-bank-open-data/GC.DOD.TOTL.GD.ZS/API_GC.DOD.TOTL.GD.ZS_DS2_en_csv_v2_509.csv",
```

```

31     "size": 1024,
32     "eTag": "0123456789abcdef0123456789abcdef",
33     "sequencer": "0A1B2C3D4E5F678901"
34 }
35 }
36 }
37 ]
38 }

```

Run the test event and view the Cloudwatch Log

Executing function: succeeded (logs)

▼ Details

The area below shows the last 4 KB of the execution log.

"Ok"

Summary

Code SHA-256 d022GAERH6bsajykgWo/UmA6k8pgx70w1WTAawLRqO=	Execution time 58 seconds ago (February 18, 2024 at 12:47 PM GMT+11)
Request ID 466a405d-49e6-47e8-a646-2f953ed0934f	Function version \$LATEST
Init duration 426.56 ms	Duration 7521.62 ms
Billed duration 7522 ms	Resources configured 512 MB
Max memory used 163 MB	

Log output

The section below shows the logging calls in your code. [Click here](#) to view the corresponding CloudWatch log group.

```

START RequestId: 466a405d-49e6-47e8-a646-2f953ed0934f Version: $LATEST
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
Successfully retrieved poc-staging/world-bank-open-data/GC.DOD.TOTL.GD.ZS/API_GC.DOD.TOTL.GD.ZS_DS2_en_csv_v2_509.csv of type text/csvThe uploaded file is a part of the World Bank indicator GC.DOD.TOTL.GD.ZS - poc-staging/world-bank-open-data/GC.DOD.TOTL.GD.ZS/API_GC.DOD.TOTL.GD.ZS_DS2_en_csv_v2_509.csv of type text/csvEND RequestId: 466a405d-49e6-47e8-a646-2f953ed0934f
REPORT RequestId: 466a405d-49e6-47e8-a646-2f953ed0934f Duration: 7521.62 ms Billed Duration: 7522 ms Memory Size: 512 MB Max Memory Used: 163 MB Init Duration: 426.56 ms
    
```

The CloudWatch log should have the output logged from the Java Lambda function.

```

1 INIT_START Runtime Version: java:21.v10 Runtime Version ARN: arn:aws:lambda:ap-southeast-2::runtime:2b04f677de1d4
2 START RequestId: 466a405d-49e6-47e8-a646-2f953ed0934f Version: $LATEST
3 SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
4 SLF4J: Defaulting to no-operation (NOP) logger implementation
5 SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
6 Successfully retrieved poc-staging/world-bank-open-data/GC.DOD.TOTL.GD.ZS/API_GC.DOD.TOTL.GD.ZS_DS2_en_csv_v2_509
7 The uploaded file is a part of the World Bank indicator GC.DOD.TOTL.GD.ZS - poc-staging/world-bank-open-data/GC.D
8 END RequestId: 466a405d-49e6-47e8-a646-2f953ed0934f
9 REPORT RequestId: 466a405d-49e6-47e8-a646-2f953ed0934f Duration: 7521.62 ms Billed Duration: 7522 ms Memor
    
```

Lambda Function (Java) Deep Dive

Lambda Function Code (Java)

```

1 package com.mattswart.aws;
2
3 import software.amazon.awssdk.services.s3.model.HeadObjectRequest;
4 import software.amazon.awssdk.services.s3.model.HeadObjectResponse;
5 import software.amazon.awssdk.services.s3.S3Client;
6
7 import com.amazonaws.services.lambda.runtime.Context;
8 import com.amazonaws.services.lambda.runtime.LambdaLogger;
    
```

```

9  import com.amazonaws.services.lambda.runtime.RequestHandler;
10 import com.amazonaws.services.lambda.runtime.events.S3Event;
11 import com.amazonaws.services.lambda.runtime.events.models.s3.S3EventNotification.S3EventNotificationRecord;
12
13 public class LambdaFuncHandler implements RequestHandler<S3Event, String> {
14
15     @Override
16     public String handleRequest(S3Event s3event, Context context) {
17         try {
18             S3EventNotificationRecord record = s3event.getRecords().get(0);
19             String srcBucket = record.getS3().getBucket().getName();
20             String srcKey = record.getS3().getObject().getUrlDecodedKey();
21
22             S3Client s3Client = S3Client.builder().build();
23             HeadObjectResponse headObject = getHeadObject(s3Client, srcBucket, srcKey);
24
25             LambdaLogger logger = context.getLogger();
26             logger.log("Successfully retrieved " + srcBucket + "/" + srcKey + " of type " + headObject.contentType);
27
28             // check that it is the world-bank-open-data/GC.DOD.TOTL.GD.ZS/ location
29             if (srcKey.contains("world-bank-open-data/GC.DOD.TOTL.GD.ZS/")){
30                 logger.log("The uploaded file is a part of the World Bank indicator GC.DOD.TOTL.GD.ZS - " + srcBucket);
31             }
32
33             return "Ok";
34         } catch (Exception e) {
35             throw new RuntimeException(e);
36         }
37     }
38
39     private HeadObjectResponse getHeadObject(S3Client s3Client, String bucket, String key) {
40         HeadObjectRequest headObjectRequest = HeadObjectRequest.builder()
41             .bucket(bucket)
42             .key(key)
43             .build();
44         return s3Client.headObject(headObjectRequest);
45     }
46 }

```

Java Build

The JAR can be created as a Maven project or manually. This POC makes use of a *fat jar* and not layers.

The list external libraries required to run the Lambda functions:

```

1  annotations-2.24.5.jar
2  apache-client-2.24.5.jar
3  arns-2.24.5.jar
4  auth-2.24.5.jar
5  aws-core-2.24.5.jar
6  aws-lambda-java-core-1.2.3.jar
7  aws-lambda-java-events-3.11.4.jar
8  aws-query-protocol-2.24.5.jar
9  aws-xml-protocol-2.24.5.jar
10 checksums-2.24.5.jar
11 checksums-spi-2.24.5.jar
12 commons-codec-1.15.jar
13 commons-logging-1.2.jar

```

```
14 crt-core-2.24.5.jar
15 endpoints-spi-2.24.5.jar
16 eventstream-1.0.1.jar
17 http-auth-2.24.5.jar
18 http-auth-aws-2.24.5.jar
19 http-auth-spi-2.24.5.jar
20 http-client-spi-2.24.5.jar
21 httpclient-4.5.13.jar
22 httpcore-4.4.13.jar
23 identity-spi-2.24.5.jar
24 joda-time-2.10.8.jar
25 json-utils-2.24.5.jar
26 metrics-spi-2.24.5.jar
27 netty-buffer-4.1.100.Final.jar
28 netty-codec-4.1.100.Final.jar
29 netty-codec-http-4.1.100.Final.jar
30 netty-codec-http2-4.1.100.Final.jar
31 netty-common-4.1.100.Final.jar
32 netty-handler-4.1.100.Final.jar
33 netty-nio-client-2.24.5.jar
34 netty-resolver-4.1.100.Final.jar
35 netty-transport-4.1.100.Final.jar
36 netty-transport-classes-epoll-4.1.100.Final.jar
37 netty-transport-native-unix-common-4.1.100.Final.jar
38 profiles-2.24.5.jar
39 protocol-core-2.24.5.jar
40 reactive-streams-1.0.4.jar
41 regions-2.24.5.jar
42 s3-2.24.5.jar
43 sdk-core-2.24.5.jar
44 slf4j-api-1.7.30.jar
45 third-party-jackson-core-2.24.5.jar
46 utils-2.24.5.jar
```

The required library pack can be downloaded at: <https://github.com/marthinusswart/aws-proof-of-concepts/tree/main/poc1/reqs>

Resources

Proof of Concept Github	https://github.com/marthinusswart/aws-proof-of-concepts/tree/main
World Bank Open Data - Central Government Debt Total	https://data.worldbank.org/indicator/GC.DOD.TOTL.GD.ZS?view=chart
Connect S3 to Lambda Example	https://docs.aws.amazon.com/lambda/latest/dg/with-s3-example.html