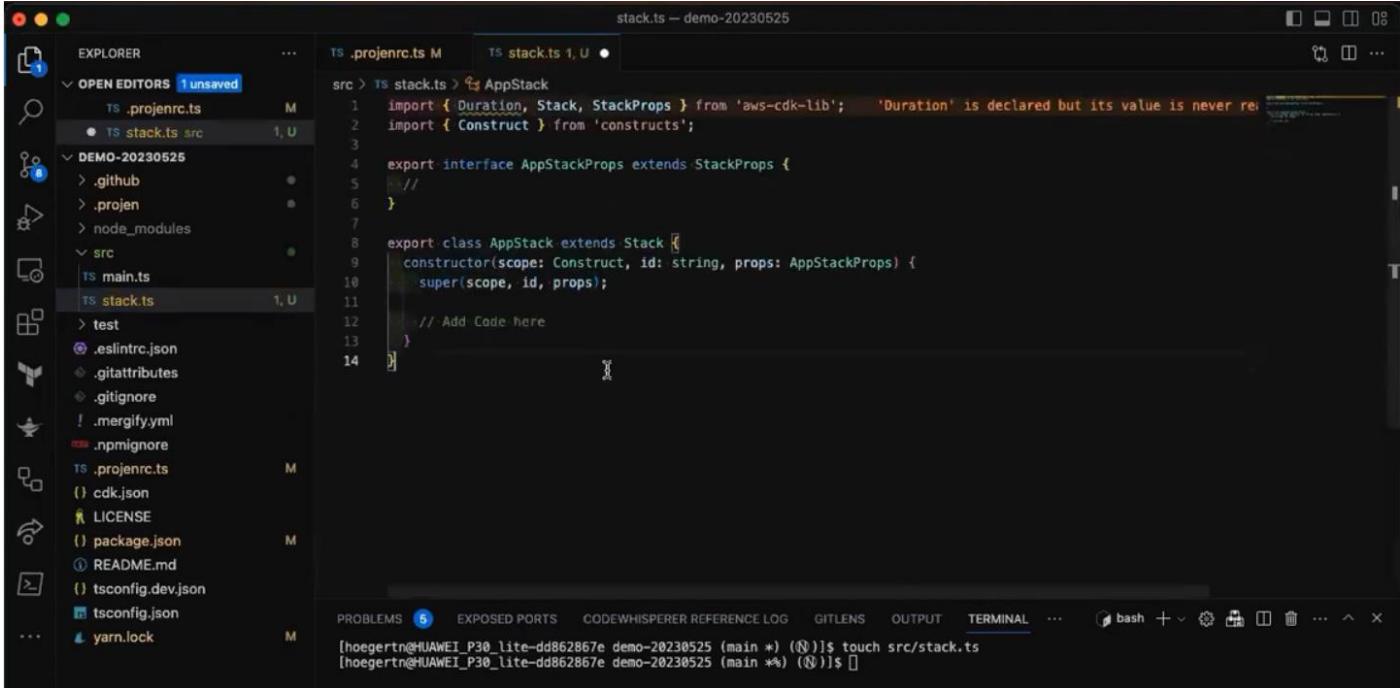
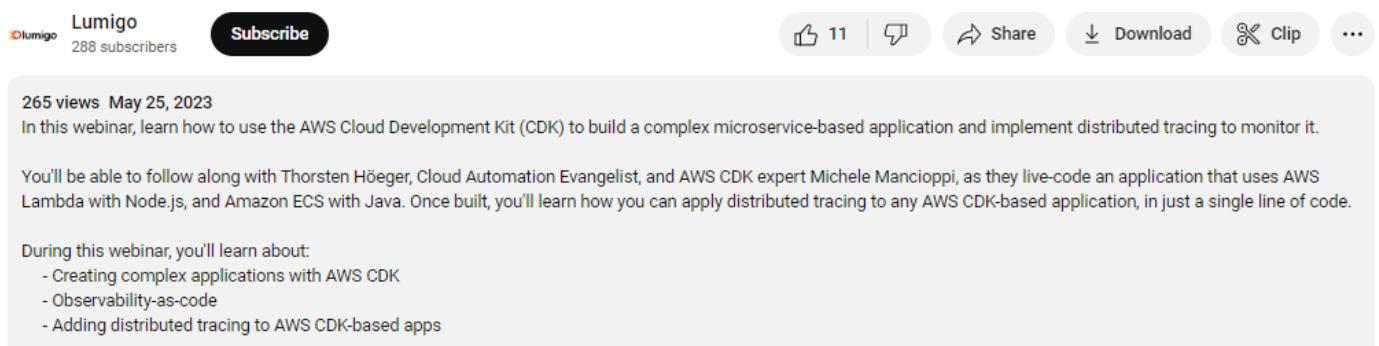
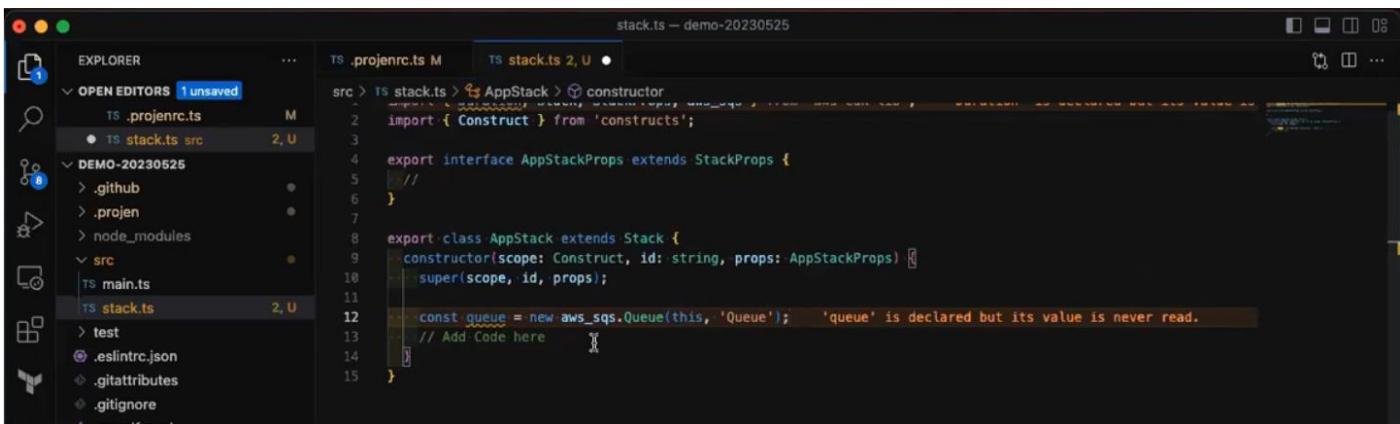


Crash Course on Building and Monitoring AWS CDK Apps



We have our App shell. Next, we want to have an **SQS** queue that will listen to events from the **EventBridge** and a **Lambda** function that will take the events out of the queue.



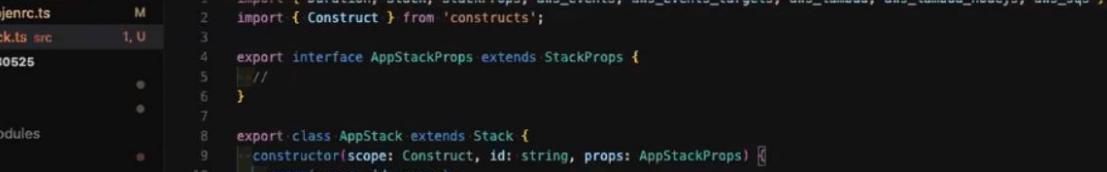
The screenshot shows a code editor interface with the following details:

- File Explorer:** On the left, it lists project files including `.projenrc.ts`, `stack.ts`, `main.ts`, and various configuration files like `.github`, `.projen`, and `.gitignore`.
- Editor Area:** The main area displays `stack.ts` with the following code:

```
src > TS stack.ts > AppStack > constructor
1 import { Construct } from 'constructs';
2
3 export interface AppStackProps extends StackProps {
4   ...
5 }
6
7 export class AppStack extends Stack {
8   constructor(scope: Construct, id: string, props: AppStackProps) {
9     super(scope, id, props);
10
11     const queue = new aws_sqs.Queue(this, 'Queue');
12
13     new aws_events.Rule(this, 'EBToQueueRule', {
14       eventPattern: {
15         source: ['lumigo-test'], "lumigo": Unknown word.
16       },
17       targets: [new aws_events_targets.SqsQueue(queue)],
18     });
19   }
20
21   // Add Code here
22
23 }
24 }
```

The code uses AWS CDK constructs and AWS Lambda functions, specifically interacting with AWS SQS and AWS Events.

Any message sent to the **lumigo-test** topic in **EventBridge** will be sent to our **SQS** queue called **Queue**. Next, we want to listen to the SQS queue using a **Lambda** function as below.



The screenshot shows the VS Code interface with the following details:

- File Explorer:** On the left, it lists files and folders. The `stack.ts` file is currently selected.
- Code Editor:** The main area displays the `stack.ts` file content. A specific line of code, which defines a new AWS Lambda function with a runtime of Node.js 18.x, is highlighted with a blue selection bar. This line contains the word "lambda".
- Status Bar:** At the top right, it shows the file name as `stack.ts — demo-20230526`.

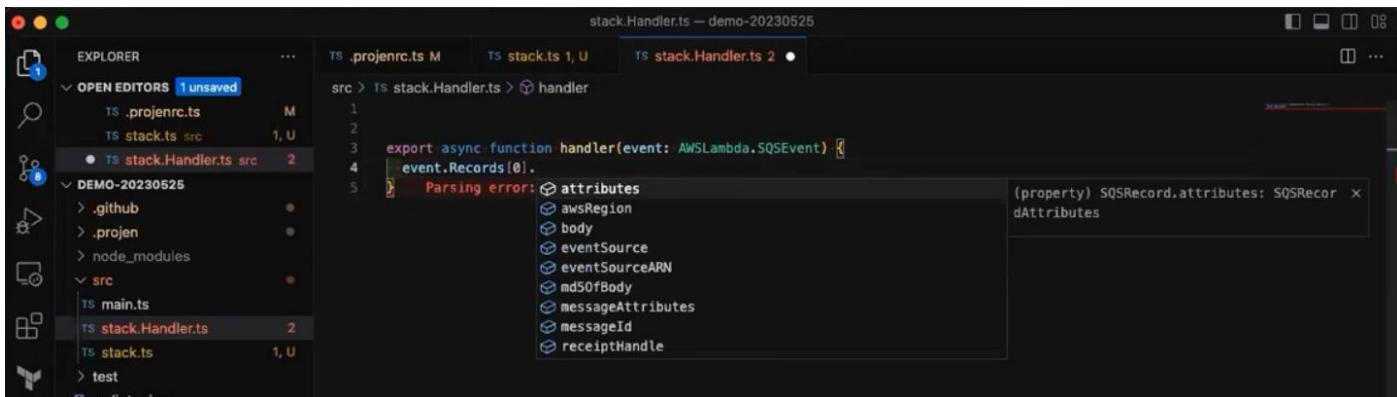
Next, we need to write the Lambda handler code to execute as below

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows files like .projencrc.ts, stack.ts, and stack.Handler.ts.
- Editor:** The stack.Handler.ts file is open, showing the following code:

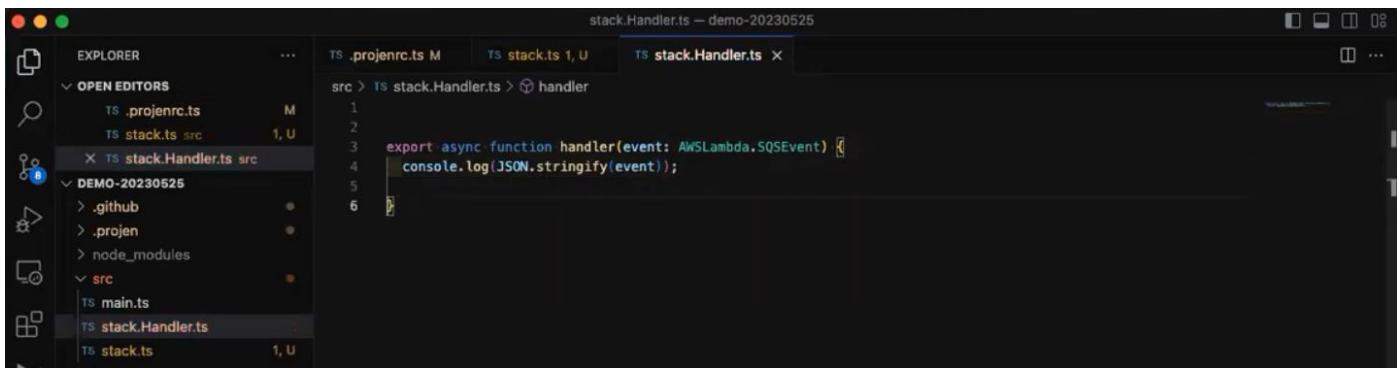
```
1
2
3 export async function handler[event: ()] { Parsing error: Type expected.
4
5 }
```

A red bar highlights the error at line 3: "export async function handler[event: ()] {".
- Terminal:** The terminal shows the command: `[hoegertn@HUAWEI_P30_lite-dd862867e demo-20230525 (main *)]$ touch src/stack.ts`



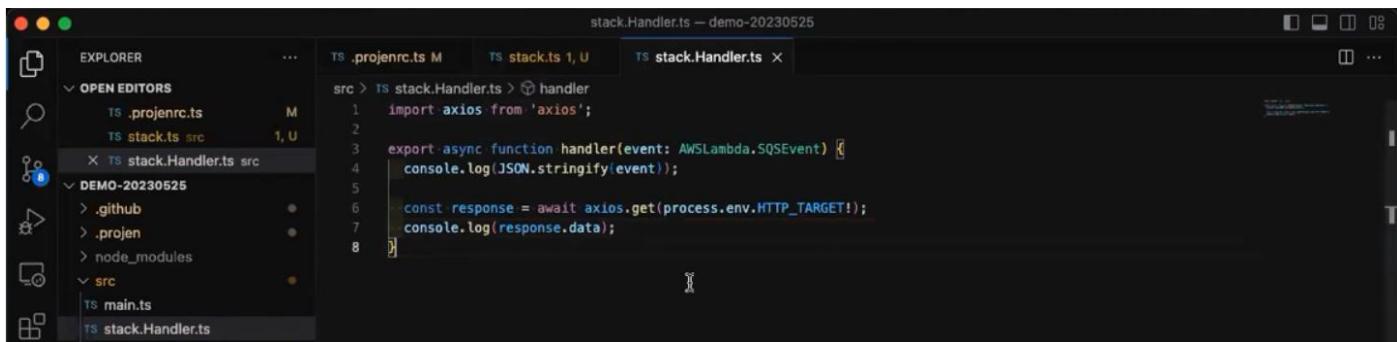
```
stack.Handler.ts — demo-20230525
src > TS stack.Handler.ts > handler
1
2
3 export async function handler(event: AWSLambda.SQSEvent) {
4   event.Records[0].
5     Parsing error: attributes
6     awsRegion
7     body
8     eventSource
9     eventSourceARN
10    md5OfBody
11    messageAttributes
12    messageId
13    receiptHandle
```

(property) SQSRecord.attributes: SQSRecord attributes

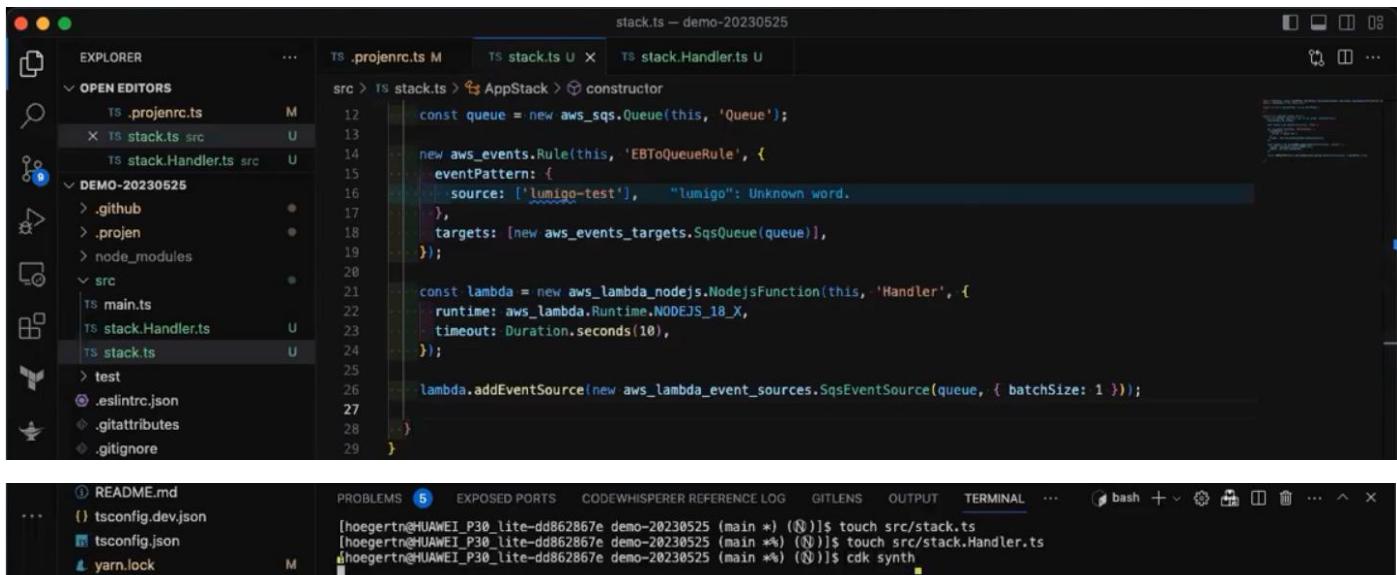


```
stack.Handler.ts — demo-20230525
src > TS stack.Handler.ts > handler
1
2
3 export async function handler(event: AWSLambda.SQSEvent) {
4   console.log(JSON.stringify(event));
5 }
```

This completes the Lambda function. Next, we want the Lambda to call an **HTTP endpoint** as below.



```
stack.Handler.ts — demo-20230525
src > TS stack.Handler.ts > handler
1 import axios from 'axios';
2
3 export async function handler(event: AWSLambda.SQSEvent) {
4   console.log(JSON.stringify(event));
5
6   const response = await axios.get(process.env.HTTP_TARGET!);
7   console.log(response.data);
8 }
```



```
stack.ts — demo-20230525
src > TS stack.ts > AppStack > constructor
12   const queue = new aws_sqs.Queue(this, 'Queue');
13
14   new aws_events.Rule(this, 'EBToQueueRule', {
15     eventPattern: {
16       source: ['lumigo-test'], "lumigo": Unknown word.
17     },
18     targets: [new aws_events_targets.SqsQueue(queue)],
19   });
20
21   const lambda = new aws_lambda_nodejs.NodejsFunction(this, 'Handler', {
22     runtime: aws_lambda.Runtime.NODEJS_18_X,
23     timeout: Duration.seconds(10),
24   });
25
26   lambda.addEventSource(new aws_lambda_event_sources.SqsEventSource(queue, { batchSize: 1 }));
27
28 }
```

When connecting a Lambda to an SQS queue, we need to create an event-source mapping, make sure that the permissions for the Lambda are correct for polling the queue as above. Then run **cdk synth** command.

The screenshot shows the VS Code interface with the following details:

- EXPLORER** view: Shows the project structure including `.projenrc.ts`, `stack.ts`, `demo-20230525-dev.template.json`, and `stack.Handler.ts`.
- OPEN EDITORS** view: Shows the `demo-20230525-dev.template.json` file open.
- demo-20230525-dev.template.json** content (partial):

```

1  {
2    "Resources": {
3      "CDKMetadata": {
4        "Type": "AWS::CDK::Metadata",
5        "Properties": {
6          "Analytics": "v2:deflate64:H4sIAAAAAAA/zPSzDQM1BMLC/WTU7J1s3JTNKrDi5JTM7WcU7LC0tzi8tSk6t1cnLT0nVyyrWLzH"
7        }
8      },
9      "Metadata": {
10        "aws:cdk:path": "demo-20230525-dev/CDKMetadata/Default"
11      }
12    },
13    "Parameters": {
14      "BootstrapVersion": {
15        "Type": "AWS::SSM::Parameter::Value<String>",
16        "Default": "/cdk-bootstrap/hnb659fds/version",
17        "Description": "Version of the CDK Bootstrap resources in this environment, automatically retrieved from SSM"
18      }
19    },
20    "Rules": {
21      "CheckBootstrapVersion": {
22        "Assertions": [
23          {
24            "Assert": {
25              "Fn::Not": [
26                ...
27              ]
28            }
29          }
30        ]
31      }
32    }
33  }

```
- PROBLEMS** view: Shows an error message: `Fn::Not: - Fn::Contains: - - "1" - - "2" - - "3" - - "4" - - "5" - Ref: BootstrapVersion AssertDescription: CDK bootstrap stack version 6 required. Please run 'cdk bootstrap' with a recent version of the CDK CLI.`
- TERMINAL** view: Shows the command `lhoegertn@HUAWEI_P30_lite-dd862867e demo-20230525 (main *) (S$) $` and the timestamp `2023-05-25 17:22:36`.

This is the `cdk.out` file for the CF code. Next, let us now use the App Stack in the `main.ts` file.

The screenshot shows the VS Code interface with the following details:

- EXPLORER** view: Shows the project structure including `.projenrc.ts`, `stack.ts`, `main.ts`, and `stack.Handler.ts`.
- OPEN EDITORS** view: Shows the `main.ts` file open.
- main.ts** content:

```

src > TS main.ts ...
You, 17 minutes ago | 1 author (You)
1 import { App, Stack, StackProps } from 'aws-cdk-lib';
2 import { Construct } from 'constructs';
3
You, 17 minutes ago | 1 author (You)
4 export class MyStack extends Stack {
5   constructor(scope: Construct, id: string, props: StackProps = {}) {
6     super(scope, id, props);
7     // define resources here...
8   }
9
10  // for development, use account/region from cdk cli
11  const devEnv = {
12    account: process.env.CDK_DEFAULT_ACCOUNT,
13    region: process.env.CDK_DEFAULT_REGION,
14  };
15
16  const app = new App();
17
18  new MyStack(app, 'demo-20230525-dev', { env: devEnv });
19
20  // new MyStack(app, 'demo-20230525-prod', { env: prodEnv });
21
22  app.synth();
23

```

```
src > TS main.ts > ...
  You, 16 seconds ago | author (You)
1  import { App } from 'aws-cdk-lib';
2  import { AppStack } from './stack';
3
4  const app = new App();
5
6  new AppStack(app, 'demo-20230525-dev', {
7    env: {
8      account: '538118019757',
9      region: 'eu-central-1',
10     },
11   });
12   You, 15 seconds ago + Uncommitted changes
13   app.synth();
```

```
cdk.out > () demo-20230525-dev.template.json > ...
4   "Type": "AWS::SQS::Queue",
5   "UpdateReplacePolicy": "Delete",
6   "DeletionPolicy": "Delete",
7   "Metadata": {
8     "aws:cdk:path": "demo-20230525-dev/Queue/Resource"
9   },
10  "QueuePolicy25439813": {
11    "Type": "AWS::SQS::QueuePolicy",
12    "Properties": {
13      "PolicyDocument": {
14        "Statement": [
15          {
16            "Action": [
17              "sns:SendMessage",
18              "sns:GetQueueAttributes",
19              "sns:GetQueueUrl"
20            ],
21            "Condition": {
22              "ArnEquals": {
23                "aws:SourceArn": {
24                  "Fn::GetAtt": [
25                    "EBTQueueRule380B1882",
26                    "Arn"
27                  ]
28                }
29              }
30            }
31          ]
32        }
33      }
34    }
35  }
```

```
cdk.out > asset.72191ad743352f83e9d4cc/def10cb0385cae05a3ea58c48942f9661593ee280 > JS index.js
1  "use strict";
2  var __create = Object.create;
3  var __defProp = Object.defineProperty;
4  var __getOwnPropDesc = Object.getOwnPropertyDescriptor;
5  var __getOwnPropNames = Object.getOwnPropertyNames;
6  var __getProtoOf = Object.getPrototypeOf;
7  var __hasOwnProperty = Object.prototype.hasOwnProperty;
8  var __commonJS = (cb, mod) => function __require() {
9    return mod || (0, cb.__getOwnPropNames(cb)[0])((mod = { exports: {} }).exports, mod), mod.exports;
10  };
11 var __export = (target, all3) => {
12   for (var name in all3)
13     __defProp(target, name, { get: all3[name], enumerable: true });
14 };
15 var __copyProps = (to, from, except, desc) => {
16   if (from && typeof from === "object" || typeof from === "function") {
17     for (let key of __getOwnPropNames(from))
18       if (!__hasOwnProperty.call(to, key) && key !== except)
19         __defProp(to, key, { get: () => from[key], enumerable: !(desc = __getOwnPropDesc(from, key)) || desc });
20   }
21 }
```

This is our JS code and everything else needed

The screenshot shows a VS Code interface with the following details:

- File Explorer:** Shows files like `.projencrc.ts`, `stack.ts`, `main.ts`, `index.js`, and `stack.Handler.ts`.
- Code Editor:** The `index.js` file is open, displaying code related to AWS CDK. It includes imports from `axios` and `process.env`, and annotations for CommonJS exports.
- Terminal:** The terminal shows the command `[hoeger@Huawei_P30_lite-dd862867e demo-20230525 (main)*]$` and the timestamp `2023-05-25 17:25:32`.
- Bottom Status Bar:** Shows the file path `index.js — demo-20230525`.

▶ OUTLINE    ⏪ PREV    ⏹ NEXT    🔍 SEARCH    🌐

Timeline: demo-20230525-dev [hoegerth@HUAWEI\_P30\_lite-dd862867e demo-20230525 (main \*)] \$ cdka deploy

2023-05-25 17:25:54

The screenshot shows the AWS Toolkit for VS Code interface. On the left, there's a sidebar with icons for test, outline, timeline, and AWS artifacts. The main area has tabs for PROBLEMS, EXPOSED PORTS, CODEWHISPERER, REFERENCE LOG, GITLENS, OUTPUT, TERMINAL, and BASH. The TERMINAL tab is active. Below it, a modal window titled "IAM Policy Changes" is open, showing a table with one row. The table has two columns: "Resource" and "Managed Policy ARN". The resource is `\${Handler/ServiceRole}` and the ARN is `arn:\${AWS::Partition}:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole`. A note at the bottom says "(NOTE: There may be security-related changes not in this list. See https://github.com/aws/aws-cdk/issues/1299)". At the bottom right, there's a message asking "Do you wish to deploy these changes (y/n)?". The status bar at the bottom right shows the date and time: 2023-05-25 17:26:02.

The screenshot shows the AWS Toolkit for VS Code interface. On the left, there's a sidebar with icons for users, outline, and timeline. The main area has tabs for PROBLEMS, EXPOSED PORTS, CODEWHISPERER REFERENCE LOG, GITLENS, OUTPUT, TERMINAL, and more. The TERMINAL tab is active, displaying the command-line output of the AWS CLI. It shows the user is deploying changes to a 'dev' environment and creating a CloudFormation changeset. The changeset status is shown as [1/9]. Below the terminal, the CloudFormation stack status table is displayed.

Timestamp	Action	Type	Resource
17:26:08	CREATE_IN_PROGRESS	AWS::CloudFormation::Stack	demo-20230525-dev
17:26:11	CREATE_IN_PROGRESS	AWS::IAM::Role	Handler/ServiceRole
17:26:12	CREATE_IN_PROGRESS	AWS::SQS::Queue	Queue

The screenshot shows the AWS Toolkit for VS Code interface. The left sidebar displays files like `.eslintrc.json`, `.gitattributes`, `.gitignore`, `.mergify.yml`, and `.npmignore`. The main area has tabs for PROBLEMS (with 8 errors), EXPOSED PORTS, CODEWHISPERER REFERENCE LOG, GITLENS, OUTPUT, TERMINAL, and more. A note at the top says: '(NOTE: There may be security-related changes not in this list. See <https://github.com/aws/aws-cdk/issues/1299>)'. Below it, a message asks 'Do you wish to deploy these changes (y/n)? y'. The command palette shows 'demo-20230525-dev: deploying... [1/1]' and 'demo-20230525-dev: creating CloudFormation changeset...'. The bottom pane shows a table of CloudFormation resources:

17:26:08	CREATE_IN_PROGRESS	AWS::CloudFormation::Stack	demo-20230525-dev	
17:27:24	CREATE_IN_PROGRESS	AWS::Events::Rule	EBSITEQueueRule	
17:27:25	CREATE_IN_PROGRESS	AWS::IAM::Policy	Handler/ServiceRole/DefaultPolicy	

On the right, there's a timestamp '2023-05-25 17:27:40'.

PROBLEMS 5 EXPOSED PORTS CODEWHISPERER REFERENCE LOG GITLENS OUTPUT TERMINAL ... bash + ⚙️ 📁 🗑️ ⌂ ⌃ ⌄

demo-20230525-dev

Deployment time: 168.09s

Stack ARN:  
arn:aws:cloudformation:eu-central-1:538118019757:stack/demo-20230525-dev/74e7a4e0-fb10-11ed-8bf6-06b905975718

Total time: 168.52s

[hoegerlin@HUAWEI-P30 ~]\$ ls

2023-05-25 17:30:01

eu-central-1.console.aws.amazon.com/console/home?region=eu-central-1#

Services Search [Option+S] Frankfurt AdminAccess/hoegertn

CloudFormation Secrets Manager DynamoDB Step Functions

## Console Home

Recently visited

- CloudFormation
- CloudWatch
- Lambda
- Amazon EventBridge
- Elastic Container Service
- EC2
- Secrets Manager
- Simple Queue Service
- API Gateway
- Step Functions
- DynamoDB
- EC2 Image Builder
- Systems Manager

Reset to default layout + Add widgets

eu-central-1.console.aws.amazon.com/cloudformation/home?region=eu-central-1#/stacks?filteringT...

Services Search [Option+S] Frankfurt AdminAccess/hoegertn

CloudFormation Secrets Manager DynamoDB Step Functions

## CloudFormation

### Stacks

- Stacks
- StackSets
- Exports

### Designer

### Registry

- Public extensions
- Activated extensions
- Publisher

### Spotlight

CloudFormation > Stacks

### Stacks (24)

Stack name	Status	Created time	Description
demo-20230525-dev	CREATE_IN_PROGRESS	2023-05-25 17:26:02 UTC+0200	-
LumigoIntegrationV9	CREATE_COMPLETE	2023-05-24 00:28:31 UTC+0200	This template which only L...
aws-cdk-demo-taimos-github	UPDATE_COMPLETE	2023-04-30 20:36:09 UTC+0200	-
aws-cdk-demo-taimos-hoegertn	UPDATE_COMPLETE	2023-04-30 20:20:32 UTC+0200	-
cdk-serverless-v2-demo-dev	CREATE_COMPLETE	2023-01-30 15:13:29 UTC+0100	-
bucketav-baader	UPDATE_COMPLETE	2022-11-22 00:26:08 UTC+0100	bucketAV - / for Amazon scanners run

eu-central-1.console.aws.amazon.com/cloudformation/home?region=eu-central-1#/stacks/demo-20230525-dev

Services Search [Option+S] Frankfurt AdminAccess/hoegertn

CloudFormation Secrets Manager DynamoDB Step Functions

## CloudFormation > Stacks > demo-20230525-dev

### Stacks (24)

Stack name	Status	Created time	Description
demo-20230525-dev	CREATE_IN_PROGRESS	2023-05-25 17:26:02 UTC+0200	-
LumigoIntegrationV9	CREATE_COMPLETE	2023-05-24 00:28:31 UTC+0200	-
aws-cdk-demo-taimos-github	UPDATE_COMPLETE	2023-04-30 20:36:09 UTC+0200	-

### demo-20230525-dev

Stack info	Events	Resources	Outputs	Parameters	Template	Change sets
<h4>Overview</h4> <p>Stack ID: arn:aws:cloudformation:eu-central-1:538118019757:stack/demo-20230525-dev/74e7a4e0-fb10-11ed-8bf6-06b905975718</p> <p>Status: CREATE_IN_PROGRESS</p> <p>Root stack: -</p> <p>Created time: 2023-05-25 17:26:02 UTC+0200</p> <p>Deleted time: -</p>						

AWS CloudFormation Stack demo-20230525-dev

**Stacks (24)**

- demo-20230525-dev (CREATE\_IN\_PROGRESS)
- LumigointegrationV9 (CREATE\_COMPLETE)
- aws-cdk-demo-talmos-github (CREATE\_COMPLETE)

**Events (18)**

Timestamp	Logical ID	Status	Status reason
2023-05-25 17:28:12 UTC+0200	Handler886CB40B	CREATE_IN_PROGRESS	Resource creation Initiated
2023-05-25 17:28:11 UTC+0200	Handler886CB40B	CREATE_IN_PROGRESS	-
2023-05-25 17:28:09 UTC+0200	HandlerServiceRoleDefinitionPolicyCBD0CC91	CREATE_COMPLETE	-
2023-05-25 17:27:25 UTC+0200	HandlerServiceRoleDefinitionPolicyCBD0CC91	CREATE_IN_PROGRESS	Resource creation Initiated

AWS CloudFormation Stack demo-20230525-dev

**Stacks (24)**

- demo-20230525-dev (CREATE\_COMPLETE)
- LumigointegrationV9 (CREATE\_COMPLETE)
- aws-cdk-demo-talmos-github (CREATE\_COMPLETE)

**Resources (8)**

Logical ID	Physical ID	Type	Status
Queue	-	-	CREATE_COMPLETE
EToQueueRule	-	-	CREATE_COMPLETE
Handler	-	-	CREATE_COMPLETE
ServiceRole	-	-	CREATE_COMPLETE
SqsEventSource:demo20230525dev...	-	-	CREATE_COMPLETE
Handler886CB40B	demo-20230525-dev-Handler886CB40B-1caEVfNIk3y	AWS::Lambda::Function	CREATE_COMPLETE
CDKMetadata	-	-	CREATE_COMPLETE

AWS CloudFormation Search results for 'eventbridge'

Try searching with longer queries for more relevant results

**Services**

- Amazon EventBridge

Serverless service for building event-driven applications.

Screenshot of the AWS Lambda console showing the Event buses page.

**Event buses** Info

Event buses receive events from a variety of sources and match them to rules in your account. Different types of event buses receive events from different sources, including AWS services in your account and other accounts, custom applications and services, and partner applications and services.

Enabling event discovery on an event bus will generate EventBridge Schemas for events on that bus. This may incur a cost (the first five million ingested events in each month is free).

Default event bus		
Name	Amazon Resource Name (ARN)	Schema discovery
default	arn:aws:events:eu-central-1:538118019757:event-bus/default	Not initiated

Custom event bus (1)		
Name	Amazon Resource Name (ARN)	Schema discovery
myEventBus	arn:aws:events:eu-central-1:538118019757:event-bus/myEventBus	Stopped

Screenshot of the AWS Lambda console showing the default event bus details.

**Important Message**  
If you have existing cross account event bus targets that do not have an IAM role configured, we recommend adding IAM roles to grant users access to resources in another account and set organization boundaries using Service Control Policies (SCPs) to determine who can send and receive events from accounts in your organization. You can attach IAM roles using EventBridge PutTarget calls. To learn more about permissions for cross account event bus targets, please refer to our documentation.

**default**

**Event bus details**

Event bus name	Schema discovery
default	Not initiated
Event bus ARN	Discoverer ID
arn:aws:events:eu-central-1:538118019757:event-bus/default	No discoverer
	Discoverer ARN
	No discoverer

Screenshot of the AWS Lambda console showing the Send events page.

**Event bus**  
Select the event bus to send the event to.

**Event source**  
The event source to use for the event.  
  
Max 256 characters.

**Detail type**  
The detail type to use for the event. This determines which fields are included in the event.  
  
Max 128 characters.

**Event detail**  
Enter the JSON for the detail type. You can copy and paste from another source, or drag a .json file from your computer and drop it in the box to add the JSON contents of the file.

Screenshot of the AWS CloudFormation console showing the creation of a new event type named "test". The "Event detail" section contains a JSON object with an empty "detail" field.

```
1 {}
```

Screenshot of the AWS EventBridge console showing a successful event send operation. An event entry for "Event entry 1" was sent on May 25, 2023, at 05:31 PM GMT+2 to the "default" event bus. The status is "Success".

```
{ "version": "0", "id": "166845cc-5f01-95ed-5467-5a2d67477582", "detail-type": "test", "source": "lumigo-test", "account": "538118019757", "time": "2023-05-25T15:31:14Z", "region": "eu-central-1", "resources": [], "detail": {} }
```

Screenshot of the AWS Lambda console showing the function "demo-20230525-dev-Handler886CB40B-icaEVfiNIk3v". The function belongs to an application and has an SQS trigger.

This function belongs to an application. Click here to manage it.

**Function overview**

**demo-20230525-dev-Handler886CB40B-icaEVfiNIk3v**

**Description**  
-

**Last modified**  
3 minutes ago

**Function ARN**  
arn:aws:lambda:eu-central-1:538118019757:function:demo-20230525-dev-Handler886CB40B-icaEVfiNIk3v

**Application**  
demo-20230525-dev

The screenshot shows the AWS Lambda service interface. The top navigation bar includes the AWS logo, a 'Services' dropdown, a search bar, and account information for 'Frankfurt' and 'AdminAccess/hoegert'. Below the search bar are links for CloudFormation, Secrets Manager, DynamoDB, and Step Functions. The main content area has tabs for 'Code', 'Test', 'Monitor' (which is selected), 'Configuration', 'Aliases', and 'Versions'. Under the 'Monitor' tab, there are buttons for 'Metrics', 'Logs', and 'Traces', with 'Metrics' currently selected. Below these are four links: 'View CloudWatch logs', 'View X-Ray traces', 'View Lambda Insights', and 'View CodeGuru profiles'. A section titled 'CloudWatch metrics' contains a description of how Lambda sends runtime metrics to CloudWatch, a 'Filter by' dropdown set to 'Function', and a 'Function' button. A note at the bottom explains how to view metrics for specific function versions or aliases.

Lambda sends runtime metrics for your functions to Amazon CloudWatch. The metrics shown are an aggregate view of all function runtime activity. To view metrics for the unqualified or \$LATEST resource, choose **Filter by**. To view metrics for a specific function version or alias, choose **Aliases** or **Versions**, select the alias or version, and then choose **Monitor**.

The screenshot shows the AWS CloudWatch interface with the URL [eu-central-1.console.aws.amazon.com/cloudwatch/home?region=eu-central-1#logsV2:log-groups/o...](https://eu-central-1.console.aws.amazon.com/cloudwatch/home?region=eu-central-1#logsV2:log-groups/o...). The left sidebar is titled "CloudWatch" and includes sections for "Favorites and recents", "Dashboards", "Alarms" (with 1 alarm), "In alarm", "All alarms", "Logs" (selected, showing "Log groups" and "Logs Insights"), "Metrics" (with "All metrics", "Explorer", "Streams"), and "X-Ray traces" (with "Service map"). The main content area shows the path "CloudWatch > Log groups > /aws/lambda/demo-20230525-dev-Handler886CB40B-icaEVfiNIk3v". Below this, there are three buttons: "Actions", "View in Logs Insights", and "Search log group". The "Log group details" section displays the following information:

ARN	Metric filters	Data protection - new
arn:aws:logs:eu-central-1:538118019757:log-group:/aws/lambda/demo-20230525-dev-Handler886CB40B-icaEVfiNIk3v:"	0	-
	Subscription filters	Sensitive data found - new
	0	-
	Contributor Insights rules	KMS key ID
	-	-
Creation time		
Now		
Retention		
Never expire		
Stored bytes		
-		

The screenshot shows the AWS CloudWatch Log Groups interface. The left sidebar has sections for CloudWatch, Favorites and recents, Dashboards, Alarms (with 1 in alarm), Logs (Log groups, Logs Insights), Metrics (All metrics, Explorer, Streams), and X-Ray traces (Service map, Traces). The main area shows a breadcrumb path: CloudWatch > Log groups > /aws/lambda/demo-20230525-dev-Handler886CB40B-icaEVflNlk3v > 2023/05/25/[\${LATEST}5bdeb8317ef24a0b90b3f9dfac9d38b0]. Below this, there's a search bar, filter buttons (Actions, Create metric filter), and a timestamp range selector (Clear, 1m, 30m, 1h, 12h, Custom, Display dropdown, and a refresh icon). A message says "There are older events to load. Load more." followed by several log entries. Each entry includes a timestamp, message content, and a detailed view icon.

	Timestamp	Message
▶	2023-05-25T17:31:15.003+02:00	INIT_START Runtime Version: nodejs:18.v6 Runtime Version ARN: arn:aws:lambda:eu-c...
▶	2023-05-25T17:31:15.205+02:00	START RequestId: d490b302-40d4-5ffa-b8f0-efebc1565958 Version: \${LATEST}
▶	2023-05-25T17:31:15.207+02:00	2023-05-25T15:31:15.207Z d490b302-40d4-5ffa-b8f0-efebc1565958 INFO {"Records": [{"...
▶	2023-05-25T17:31:15.505+02:00	2023-05-25T15:31:15.505Z d490b302-40d4-5ffa-b8f0-efebc1565958 ERROR Invoke Error -
▶	2023-05-25T17:31:15.543+02:00	END RequestId: d490b302-40d4-5ffa-b8f0-efebc1565958
▶	2023-05-25T17:31:15.543+02:00	REPORT RequestId: d490b302-40d4-5ffa-b8f0-efebc1565958 Duration: 337.48 ms Billed...

aws Services Search [Option+S] Frankfurt AdminAccess/hoegertn

**CloudWatch**

- Favorites and recents
- Dashboards
- Alarms** 1 4 0
  - In alarm
  - All alarms
- Logs**
  - Log groups
  - Logs Insights
- Metrics**
  - All metrics
  - Explorer
  - Streams
- X-Ray traces**
  - Service map
  - Traces

▶ 2023-05-25T17:31:15.205+02:00 START RequestId: d490b302-40d4-5ffa-b8f0-eefebc1565958 Version: \$LATEST

▼ 2023-05-25T17:31:15.207+02:00 2023-05-25T15:31:15.207Z d490b302-40d4-5ffa-b8f0-eefebc1565958 INFO {"Records": [{"version": "1.0", "id": "166845cc-5f01-95ed-5467-5a2d67477582", "detail-type": "test", "source": "lumigo-test", "account": "538118019757", "time": "2023-05-25T15:31:14Z", "region": "eu-central-1", "resources": {}, "detail": {}}, {"body": "connect ECONNREFUSED 127.0.0.1:80", "name": "Error", "stack": "Error: connect ECONNREFUSED 127.0.0.1:80\n at AxiosError.from (/var/task/index.js:11177:14)\n at RedirectableRequest.handleRequestError (/var/task/index.js:12779:33)\n at RedirectableRequest.emit (node:events:513:28)\n at eventHandlers.<computed> (/var/task/index.js:10306:28)\n at ClientRequest.emit (node:events:513:28)\n at Socket.socketErrorListener (node:http\_client:502:9)\n at Socket.emit (node:internal/streams/destroy:151:8)\n at emitErrorCloseNT (node:internal/streams/destroy:116:3)\n at process.processTicksAndRejections (node:internal/process/task\_queues:82:21)"}, {"body": "lumigo-test", "name": "Unknown word", "stack": "EventPattern: {\n source: ['lumigo-test'],\n lumigo: Unknown word.\n},\n targets: [new aws\_events\_targets.SqsQueue(queue)],\n}\n\nconst lambda = new aws\_lambda\_nodejs.NodejsFunction(this, 'Handler', {\n runtime: aws\_lambda.Runtime.NODEJS\_18\_X,\n timeout: Duration.seconds(10),\n});\nlambda.addEventsSource(new aws\_lambda\_event\_sources.SqsEventSource(queue, { batchSize: 1 }));"}], "md5OfBody": "ca166253af18cc143bccce534151ff5", "eventSource": "aws:sqs", "eventSourceARN": "arn:aws:sqs:eu-central-1:538118019757:demo-20230525-dev-Queue4A7E3555-ltF8HUUdCIKX", "awsRegion": "eu-central-1"}]}

Copy

▶ 2023-05-25T17:31:15.505+02:00 2023-05-25T15:31:15.505Z d490b302-40d4-5ffa-b8f0-eefebc1565958 ERROR Invoke Error

Back to top

eu-central-1.console.aws.amazon.com/cloudwatch/home?region=eu-central-1#logsV2:log-groups/o... Frankfurt AdminAccess/hoegertn

**CloudWatch**

- Favorites and recents
- Dashboards
- Alarms** 1 4 0
  - In alarm
  - All alarms
- Logs**
  - Log groups
  - Logs Insights
- Metrics**
  - All metrics
  - Explorer
  - Streams
- X-Ray traces**
  - Service map

messageAttributes": {}, "md5OfBody": "ca166253af18cc143bccce534151ff5", "eventSource": "aws:sqs", "eventSourceARN": "arn:aws:sqs:eu-central-1:538118019757:demo-20230525-dev-Queue4A7E3555-ltF8HUUdCIKX", "awsRegion": "eu-central-1"}]

▶ 2023-05-25T17:31:15.505+02:00 2023-05-25T15:31:15.505Z d490b302-40d4-5ffa-b8f0-eefebc1565958 ERROR Invoke Error

message": "connect ECONNREFUSED 127.0.0.1:80", "name": "Error", "stack": "Error: connect ECONNREFUSED 127.0.0.1:80\n at AxiosError.from (/var/task/index.js:11177:14)\n at RedirectableRequest.handleRequestError (/var/task/index.js:12779:33)\n at RedirectableRequest.emit (node:events:513:28)\n at eventHandlers.<computed> (/var/task/index.js:10306:28)\n at ClientRequest.emit (node:events:513:28)\n at Socket.socketErrorListener (node:http\_client:502:9)\n at Socket.emit (node:internal/streams/destroy:151:8)\n at emitErrorCloseNT (node:internal/streams/destroy:116:3)\n at process.processTicksAndRejections (node:internal/process/task\_queues:82:21)"}, "config": { "transitional": { "silentJSONParsing": true, "forcedJSONParsing": true, "clarifyTimeoutError": false }, "adapter": [ "xhr", "http" ] }

Copy

It worked as expected.

```

stack.ts - demo-20230525
src > TS stack.ts M TS stack.Handler.ts U
export class AppStack extends Stack {
  constructor(scope: Construct, id: string, props: AppStackProps) {
    super(scope, id, props);
    const queue = new aws_sqs.Queue(this, 'Queue');
    new aws_events.Rule(this, 'EBToQueueRule', {
      eventPattern: {
        source: ['lumigo-test'], lumigo: Unknown word.
      },
      targets: [new aws_events_targets.SqsQueue(queue)],
    });
    const lambda = new aws_lambda_nodejs.NodejsFunction(this, 'Handler', {
      runtime: aws_lambda.Runtime.NODEJS_18_X,
      timeout: Duration.seconds(10),
    });
    lambda.addEventsSource(new aws_lambda_event_sources.SqsEventSource(queue, { batchSize: 1 }));
  }
}

```

```

stack.Handler.ts — demo-20230525
src > TS stack.Handler.ts > handler
1 import axios from 'axios';
2
3 export async function handler(event: AWSLambda.SQSEvent) {
4     console.log(JSON.stringify(event));
5
6     const response = await axios.get(process.env.HTTP_TARGET!);
7     console.log(response.data);
8 }

```

```

ecs.ts — demo-20230525
src > TS ecs.ts > ...
1 import * as cdk from 'aws-cdk-lib';
2 import { Construct } from 'constructs';
3
4 export interface EcsAppProps {
5
6 }
7
8 export class EcsApp extends Construct {
9     constructor(scope: Construct, id: string, _props: EcsAppProps) {
10        super(scope, id);
11
12    }
13
14 }

```

Next, let us create an ECS service that will be deployed in a VPC

```

ecs.ts — demo-20230525
src > TS ecs.ts > EcsApp > constructor > vpc > subnetConfiguration
1 import * as cdk from 'aws-cdk-lib';
2 import { Construct } from 'constructs';
3
4 export interface EcsAppProps {
5
6 }
7
8 export class EcsApp extends Construct {
9     constructor(scope: Construct, id: string, _props: EcsAppProps) {
10        super(scope, id);
11
12        const vpc = new cdk.aws_ec2.Vpc(this, 'Vpc', {
13            maxAzs: 3,
14            natGateways: 0,
15            subnetConfiguration: [
16                {
17                    name: 'public',
18                    subnetType: cdk.aws_ec2.SubnetType.PUBLIC,
19                    mapPublicIpOnLaunch: true,
20                },
21            ],
22        });
23
24        this.service = new aws_ecs_patterns.ApplicationLoadBalancedFargateService(this, 'Resource', {
25            vpc,
26            assignPublicIp: true,
27            circuitBreaker: {
28                rollback: true,
29            },
30            taskImageOptions: {
31                containerPort: 8080,
32                image: cdk.aws_ecs.ContainerImage.fromAsset(path.join(__dirname, '..', 'ecs-app'), {
33                    platform: cdk.aws_ecr_assets.Platform.LINUX_AMD64,
34                }),
35            },
36            healthCheckGracePeriod: cdk.Duration.minutes(5),
37        });
38
39    }
40
41 }

```

```

ecs.ts — demo-20230525
src > TS ecs.ts > EcsApp > constructor > taskImageOptions
1
2         name: 'public',
3         subnetType: cdk.aws_ec2.SubnetType.PUBLIC,
4         mapPublicIpOnLaunch: true,
5     };
6
7     this.service = new cdk.aws_ecs_patterns.ApplicationLoadBalancedFargateService(this, 'Resource', {
8         vpc,
9         assignPublicIp: true,
10        circuitBreaker: {
11            rollback: true,
12        },
13        taskImageOptions: {
14            containerPort: 8080,
15            image: cdk.aws_ecs.ContainerImage.fromAsset(path.join(__dirname, '..', 'ecs-app'), {
16                platform: cdk.aws_ecr_assets.Platform.LINUX_AMD64,
17            }),
18        },
19        healthCheckGracePeriod: cdk.Duration.minutes(5),
20    });
21
22 }

```

We are using a L3 construct in the **ecs\_patterns** (that combines multiple L2 Constructs) as above

```

stack.ts — demo-20230525

src > TS stack.ts > AppStack > constructor
  7
  8  export class AppStack extends Stack {
  9    constructor(scope: Construct, id: string, props: AppStackProps) {
 10      super(scope, id, props);
 11
 12      const queue = new aws_sqs.Queue(this, 'Queue');
 13
 14      new aws_events.Rule(this, 'EBToQueueRule', {
 15        eventPattern: {
 16          source: ['lumigo-test'], "lumigo": Unknown word.
 17        },
 18        targets: [new aws_events_targets.SqsQueue(queue)],
 19      });
 20
 21      const table = new aws_dynamodb.Table(this, 'Table', {
 22        billingMode: aws_dynamodb.BillingMode.PAY_PER_REQUEST,
 23        partitionKey: {
 24          name: 'PK',
 25          type: aws_dynamodb.AttributeType.STRING,
 26        },
 27      });
 28
 29      const lambda = new aws_lambda_nodejs.NodejsFunction(this, 'Handler', {
 30        runtime: aws_lambda.Runtime.NODEJS_18_X,
 31        timeout: Duration.seconds(10),
 32        environment: {
 33          TABLE: table.tableName,
 34        },
 35      });
 36
 37      lambda.addEventSource(new aws_lambda_event_sources.SqsEventSource(queue, { batchSize: 1 }));
 38
 39    }
 40
 41  };
 42
 43}

```

We then add a DynamoDB table as below

```

stack.ts — demo-20230525

src > TS stack.ts > AppStack > constructor > lambda > environment
  18
  19    targets: [new aws_events_targets.SqsQueue(queue)],
 20
 21    const table = new aws_dynamodb.Table(this, 'Table', {
 22      billingMode: aws_dynamodb.BillingMode.PAY_PER_REQUEST,
 23      partitionKey: {
 24        name: 'PK',
 25        type: aws_dynamodb.AttributeType.STRING,
 26      },
 27    });
 28
 29    const lambda = new aws_lambda_nodejs.NodejsFunction(this, 'Handler', {
 30      runtime: aws_lambda.Runtime.NODEJS_18_X,
 31      timeout: Duration.seconds(10),
 32      environment: [
 33        TABLE: table.tableName,
 34      ],
 35    });
 36
 37    lambda.addEventSource(new aws_lambda_event_sources.SqsEventSource(queue, { batchSize: 1 }));
 38
 39  }
 40
 41  };
 42
 43}

```

```

stack.ts — demo-20230525

src > TS stack.ts > AppStack > constructor > lambda > environment > ecsApp
  25
  26    name: 'PK',
  27    type: aws_dynamodb.AttributeType.STRING,
  28  );
 29
 30  const lambda = new aws_lambda_nodejs.NodejsFunction(this, 'Handler', {
 31    runtime: aws_lambda.Runtime.NODEJS_18_X,
 32    timeout: Duration.seconds(10),
 33    environment: [
 34      TABLE: table.tableName,
 35    ],
 36  });
 37
 38  lambda.addEventSource(new aws_lambda_event_sources.SqsEventSource(queue, { batchSize: 1 }));
 39
 40  const ecsApp = new EcsApp(this, 'EcsApp', {});
 41  lambda.addEnvironment('HTTP_TARGET', 'http://' + ecsApp.service.loadBalancer.loadBalancerDnsName);
 42
 43}

```

We then instantiated our ECS Construct in our app and gave it to the Lambda as a target.

The screenshot shows the VS Code interface with the file `ecs.ts` open. The code is written in TypeScript and defines an AWS CDK stack. It creates a VPC with one public subnet and an Application Load Balanced Fargate service. The service has a container port of 8080 and uses a Docker image from an asset named 'ecs-app'. The health check grace period is set to 5 minutes.

```
const vpc = new cdk.aws_ec2.Vpc(this, 'Vpc', {
    maxAzs: 3,
    natGateways: 0,
    subnetConfiguration: [
        {
            name: 'public',
            subnetType: cdk.aws_ec2.SubnetType.PUBLIC,
            mapPublicIpOnLaunch: true,
        },
    ],
});

this.service = new cdk.aws_ecs_patterns.ApplicationLoadBalancedFargateService(this, 'Resource', {
    vpc,
    assignPublicIp: true,
    circuitBreaker: {
        rollback: true,
    },
    taskImageOptions: {
        containerPort: 8080,
        image: cdk.aws_ecs.ContainerImage.fromAsset(path.join(__dirname, '..', 'ecs-app')),
        platform: cdk.aws_ecr_assets.Platform.LINUX_AMD64,
    },
},
    healthCheckGracePeriod: cdk.Duration.minutes(5),
});
```

This screenshot shows the same `ecs.ts` file in VS Code. The code is identical to the one in the previous screenshot, defining the same VPC and Fargate service configuration.

```
const vpc = new cdk.aws_ec2.Vpc(this, 'Vpc', {
    maxAzs: 3,
    natGateways: 0,
    subnetConfiguration: [
        {
            name: 'public',
            subnetType: cdk.aws_ec2.SubnetType.PUBLIC,
            mapPublicIpOnLaunch: true,
        },
    ],
});

this.service = new cdk.aws_ecs_patterns.ApplicationLoadBalancedFargateService(this, 'Resource', {
    vpc,
    assignPublicIp: true,
    circuitBreaker: {
        rollback: true,
    },
    taskImageOptions: {
        containerPort: 8080,
        image: cdk.aws_ecs.ContainerImage.fromAsset(path.join(__dirname, '..', 'ecs-app')),
        platform: cdk.aws_ecr_assets.Platform.LINUX_AMD64,
    },
},
    healthCheckGracePeriod: cdk.Duration.minutes(5),
});
```

The screenshot shows the VS Code interface with the file `MyEndpoint.java` open. The code is written in Java and defines a REST controller class named `MyEndpoint`. The class has a single endpoint method that logs a message when called.

```
/*
 * Copyright (c) 2023, Taimos GmbH http://www.taimos.de
 */
package de.taimos.lumigodemo;

import java.io.IOException;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;
import okhttp3.OkHttpClient;
import okhttp3.Request;
import okhttp3.Response;

/**
 * The following Java code defines a MyEndpoint REST controller class with a root endpoint that makes an HTTP
 */
@RestController
public class MyEndpoint {

    @GetMapping("/")
    public String getRoot() throws IOException {
        // Logging that this method has been called
        System.out.println("CALLED!");
    }
}
```

The ECS app is a springboot application that has a Dockerfile for building the image for deployment.

MyEndpoint.java — demo-20230525

```

EXPLORER ... TS .projenrc.ts M TS stack.ts U TS main.ts M TS stack.Handler.ts U TS ecs.ts U J MyEndpoint.java U ...
ecs-app > src > main > java > de > taimos > lumigodemo > J MyEndpoint.java
18 public class MyEndpoint {
19
20     @GetMapping("/")
21     public String getRoot() throws IOException {
22
23         // Logging that this method has been called
24         System.out.println("CALLED!");
25
26         // Creating a new instance of OkHttpClient
27         OkHttpClient client = new OkHttpClient();
28
29         // Calling the website
30         Request request = new Request.Builder()
31             .url("http://taimos.de")
32             .build();
33
34         try (Response response = client.newCall(request).execute()) {
35             // If the response is successful, get the response body as a string and return it
36             String res = response.body().string();
37             // System.out.println(res);
38             // Logging that we've received the result from the request
39             System.out.println("GOT RESULT");
40             return res;
41         } catch (Exception e) {
42             // Log any exceptions that occurred while making the request and throw them
43             e.printStackTrace();
44         }
45     }
46 }

```

MyEndpoint.java — demo-20230525

```

EXPLORER ... TS .projenrc.ts M TS stack.ts U TS main.ts M TS stack.Handler.ts U TS ecs.ts U J MyEndpoint.java U ...
ecs-app > src > main > java > de > taimos > lumigodemo > J MyEndpoint.java
24 // Logging that this method has been called
25 System.out.println("CALLED!");
26
27 // Creating a new instance of OkHttpClient
28 OkHttpClient client = new OkHttpClient();
29
30 // Calling the website
31 Request request = new Request.Builder()
32     .url("http://taimos.de")
33     .build();
34
35 try (Response response = client.newCall(request).execute()) {
36     // If the response is successful, get the response body as a string and return it
37     String res = response.body().string();
38     // System.out.println(res);
39     // Logging that we've received the result from the request
40     System.out.println("GOT RESULT");
41     return res;
42 } catch (Exception e) {
43     // Log any exceptions that occurred while making the request and throw them
44     e.printStackTrace();
45     throw e;
46 }

```

Dockerfile — demo-20230525

```

EXPLORER ... TS .projenrc.ts M TS stack.ts U TS main.ts M TS stack.Handler.ts U TS ecs.ts U Dockerfile U ...
ecs-app > Dockerfile > FROM
1 FROM Maven:latest as build
2
3 # Copy the current directory into the build container
4 COPY .
5
6 # Build the project with Maven
7 RUN mvn clean package
8
9 FROM amazoncorretto:17
10
11 # Copy the jar file of the built application from the build container to the runtime container
12 COPY --from=build target/lumigo-demo-*.*.jar app.jar
13
14 # Set the command to run the Java application
15 CMD ["java","-jar","app.jar"]
16

```

Building and running this container will start up the Springboot application that can then be called via HTTP:80 by our Lambda function with the event data.

stack.ts — demo-20230525

```

src > ts stack.ts > AppStack > constructor
  25   name: 'PK',
  26   type: aws_dynamodb.AttributeType.STRING,
  27   ),
  28   );
  29
  30   const lambda = new aws_lambda_nodejs.NodejsFunction(this, 'Handler', {
  31     runtime: aws_lambda.Runtime.NODEJS_18_X,
  32     timeout: Duration.seconds(10),
  33     environment: {
  34       TABLE: table.tableName,
  35     },
  36   });
  37
  38   lambda.addEventSource(new aws_lambda_event_sources.SqsEventSource(queue, { batchSize: 1 }));
  39
  40   const ecsApp = new EcsApp(this, 'EcsApp', {});
  41   lambda.addEnvironment('HTTP_TARGET', `http://${ecsApp.service.loadBalancer.loadBalancerDnsName}`);
  42 }
  43

```

PROBLEMS 6 EXPOSED PORTS CODEWHISPERER REFERENCE LOG GITLENS OUTPUT TERMINAL ... bash + ⚡ 🔍

TS stack.Handler.ts U TS stack.ts U

demo-20230525-dev

Deployment time: 168.09s

Stack ARN:  
arn:aws:cloudformation:eu-central-1:1538118019757:stack/demo-20230525-dev/74e7a4e0-fb10-11ed-8bf6-96b905975718

Total time: 168.52s

[hoegernt@HUAWEI\_P30\_Lite-dd862867e demo-20230525 (main \*) (N)]\$ touch src/ecs.ts  
[hoegernt@HUAWEI\_P30\_Lite-dd862867e demo-20230525 (main \*) (N)]\$ cd deploy demo-20230525-dev

2023-05-25 17:43:20

PROBLEMS 6 EXPOSED PORTS CODEWHISPERER REFERENCE LOG GITLENS OUTPUT TERMINAL ... bash + ⚡ 🔍

TS main.ts M TS stack.Handler.ts U TS stack.ts U

cdk.out/bundling-temp-bfdbd45ef4504f78df4fb329d5d5e730be484f20ad918e00c5046d117c4ea1b/index.js 370.7kb

Done in 0.17s.

Synthesis time: 8.58s

demo-20230525-dev: start: Building 07aacfc20e80b77b7d17898f0a99add85f1db62a3ae93d52f4482d94d1179599:538118019757-eu-central-1  
demo-20230525-dev: success: Built 07aacfc20e80b77b7d17898f0a99add85f1db62a3ae93d52f4482d94d1179599:538118019757-eu-central-1  
demo-20230525-dev: start: Building 560af5f89da491698729e2ba125488e447b08e679cd3283821feef801636ace:538118019757-eu-central-1  
demo-20230525-dev: success: Published 07aacfc20e80b77b7d17898f0a99add85f1db62a3ae93d52f4482d94d1179599:538118019757-eu-central-1  
demo-20230525-dev: success: Published 07aacfc20e80b77b7d17898f0a99add85f1db62a3ae93d52f4482d94d1179599:538118019757-eu-central-1

2023-05-25 17:43:35

PROBLEMS 6 EXPOSED PORTS CODEWHISPERER REFERENCE LOG GITLENS OUTPUT TERMINAL ... bash + ⚡ 🔍

TS main.ts M TS stack.Handler.ts U TS stack.ts U

Downloading from central: https://repo.maven.apache.org/maven2/org/apache/commons/commons-lang3/3.12.0/commons-lang3-3.12.0.jar #9 32.51 Downloaded from central: https://repo.maven.apache.org/maven2/org/plexus/plexus-build-api/0.0.7/plexus-build-api-0.0.7.jar (8.5 KB at 110 kB/s)  
#9 32.52 Downloaded from central: https://repo.maven.apache.org/maven2/org/slf4j/slf4j-api/1.7.36/slf4j-api-1.7.36.jar (41 kB at 514 kB/s)  
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.5.1/plexus-utils-3.5.1.jar (269 kB at 2.2 MB/s)  
Downloaded from central: https://repo.maven.apache.org/maven2/commons-io/2.11.0/commons-io-2.11.0.jar (327 kB at 2.0 MB/s)  
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/commons/commons-lang3/3.12.0/commons-lang3-3.12.0.jar (587 kB at 3.2 MB/s)  
#9 33.03 [INFO] Copying 1 resource from src/main/resources to target/classes

2023-05-25 17:44:10

Dockerfile — demo-20230525

```

FROM maven:latest as build
COPY . .
RUN mvn clean package
FROM amazoncorretto:17
COPY --from=build target/lumigo-demo-*jar app.jar
CMD ["java", "-jar", "app.jar"]

```

ecs-app > Dockerfile > ...

FROM maven:latest as build

# Copy the current directory into the build container

COPY . .

# Build the project with Maven

RUN mvn clean package

FROM amazoncorretto:17

# Copy the jar file of the built application from the build container to the runtime container

COPY --from=build target/lumigo-demo-\*jar app.jar

# Set the command to run the Java application

CMD ["java", "-jar", "app.jar"]

This screenshot shows the AWS CloudFormation Changeset details in the terminal. It lists several resources being updated or created, such as Security Groups, Route Tables, and Listener/ECS Group associations.

Action	Resource Type	Resource ID
CREATE_IN_PROGRESS	AWS::CloudFormation::Stack	demo-20230525-dev
CREATE_IN_PROGRESS	AWS::ElasticLoadBalancingV2::TargetGroup	EcsApp/Resource/LB...cListener/ECSGroup
CREATE_IN_PROGRESS	AWS::EC2::RouteTable	EcsApp/Vpc/publicSubnet2/RouteTable
CREATE_IN_PROGRESS	AWS::EC2::RouteTable	EcsApp/Vpc/publicSubnet3/RouteTable
CREATE_IN_PROGRESS	AWS::EC2::RouteTable	EcsApp/Vpc/publicSubnet1/RouteTable
CREATE_IN_PROGRESS	AWS::EC2::VPCHostAttachment	EcsApp/Vpc/VPCHW

(NOTE: There may be security-related changes not in this list. See <https://github.com/aws/aws-cdk/issues/1299>)

Do you wish to deploy these changes (y/n)? y  
demo-20230525-dev: deploying... [1/1]  
demo-20230525-dev: creating CloudFormation changeset...

2023-05-25 17:45:16

This screenshot shows the AWS CloudFormation Changeset details in the terminal. It lists several resources being updated or created, such as Security Groups, Route Tables, and Listener/ECS Group associations.

Action	Resource Type	Resource ID
UPDATE_IN_PROGRESS	AWS::CloudFormation::Stack	demo-20230525-dev
CREATE_IN_PROGRESS	AWS::ElasticLoadBalancingV2::TargetGroup	EcsApp/Resource/LB...cListener/ECSGroup
CREATE_IN_PROGRESS	AWS::EC2::RouteTable	EcsApp/Vpc/publicSubnet2/RouteTable
CREATE_IN_PROGRESS	AWS::EC2::RouteTable	EcsApp/Vpc/publicSubnet3/RouteTable
CREATE_IN_PROGRESS	AWS::EC2::RouteTable	EcsApp/Vpc/publicSubnet1/RouteTable
CREATE_IN_PROGRESS	AWS::EC2::VPCHostAttachment	EcsApp/Vpc/VPCHW

This screenshot shows the AWS CloudFormation Changeset details in the terminal. It lists several resources being updated or created, such as Security Groups, Route Tables, and Listener/ECS Group associations.

Action	Resource Type	Resource ID
UPDATE_IN_PROGRESS	AWS::CloudFormation::Stack	demo-20230525-dev
CREATE_IN_PROGRESS	AWS::ElasticLoadBalancingV2::LoadBalancer	EcsApp/Resource/LB

This screenshot shows the AWS CloudFormation Changeset details in the terminal. It lists several resources being updated or created, such as Security Groups, Route Tables, and Listener/ECS Group associations.

Action	Resource Type	Resource ID
UPDATE_IN_PROGRESS	AWS::CloudFormation::Stack	demo-20230525-dev
CREATE_IN_PROGRESS	AWS::ElasticLoadBalancingV2::LoadBalancer	EcsApp/Resource/LB

ecs.ts — demo-20230525

The screenshot shows the VS Code interface with the following components:

- EXPLORER**: Shows the project structure with files like .projencrc.ts, TS stack.ts, TS ecs.ts, TS main.ts, and TS stack.Handler.ts.
- CODE EDITOR**: Displays the contents of `ecs.ts` which configures an Application Load Balanced Fargate Service.
- TERMINAL**: Shows logs from CloudFormation and AWS Lambda functions.
- PROBLEMS**: Shows 8 errors.
- OUTPUT**: Shows logs from CloudFormation and AWS ECS services.

eu-central-1.console.aws.amazon.com/events/home?region=eu-central-1#/eventbuses/sendevents?...

The screenshot shows the AWS Lambda console with the following details:

- Services**: CloudFormation, Secrets Manager, DynamoDB, Step Functions.
- Event entry 1**: Sent on May 25, 2023, 05:50 PM GMT+2.
- Status**: Success.
- Event**: A JSON object representing the event sent to the event bus.

Sales Services Search [Option+S] Frankfurt AdminAccess/hoegertn

CloudFormation Secrets Manager DynamoDB Step Functions

### CloudWatch

- Favorites and recents
- Dashboards
- Alarms** 1 4 0
  - In alarm
  - All alarms
- Logs**
  - Log groups**
  - Logs Insights
- Metrics**
  - All metrics
  - Explorer
  - Streams
- X-Ray traces**
  - Service map
  - Traces

ARN: arn:aws:logs:eu-central-1:538118019757:log-group:/aws/lambda/demo-20230525-dev-Handler886CB40B-icaEVfNIk3v:\*

Metric filters: 0  
Subscription filters: 0  
Contributor Insights rules: -

Creation time: 19 minutes ago  
Retention: Never expire  
Stored bytes: -

Data protection - new: -  
Sensitive data found - new: -  
KMS key ID: -

**Log streams** Metric filters Subscription filters Contributor Insights Tags Data protection - new

#### Log streams (2)

Filter log streams or try prefix search □ Exact match □ Show expired ⓘ Info < 1 > ⓘ

Log stream	Last event time
2023/05/25/[LATEST]612052c98c2c4a0b8b5238233cbc7548	2023-05-25 17:48:15 (UTC+02:00)

Sales Services Search [Option+S] Frankfurt AdminAccess/hoegertn

CloudFormation Secrets Manager DynamoDB Step Functions

### CloudWatch

- Favorites and recents
- Dashboards
- Alarms** 1 4 0
  - In alarm
  - All alarms
- Logs**
  - Log groups**
  - Logs Insights
- Metrics**
  - All metrics
  - Explorer
  - Streams

CloudWatch > Log groups > /aws/lambda/demo-20230525-dev-Handler886CB40B-icaEVfNIk3v > 2023/05/25/[LATEST]612052c98c2c4a0b8b5238233cbc7548

#### Log events

You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

Actions Create metric filter

Filter events Clear 1m 30m 1h 12h Custom Display

Timestamp	Message
No older events at this moment. <a href="#">Retry</a>	
2023-05-25T17:48:15.170+02:00	INIT_START Runtime Version: nodejs:18.v6 Runtime Version ARN: arn:aws:lambda:eu-c...
2023-05-25T17:48:15.375+02:00	START RequestId: 3dfdd84f-fb4e-510c-a40b-84a67e218717 Version: \$LATEST
2023-05-25T17:48:15.377Z	3dfdd84f-fb4e-510c-a40b-84a67e218717 INFO {"Records": [{"...
2023-05-25T15:48:15.748Z	3dfdd84f-fb4e-510c-a40b-84a67e218717 ERROR Invoke Error -
2023-05-25T17:48:15.806+02:00	END RequestId: 3dfdd84f-fb4e-510c-a40b-84a67e218717

Sales Services Search [Option+S] Frankfurt AdminAccess/hoegertn

CloudFormation Secrets Manager DynamoDB Step Functions

### CloudWatch

- Favorites and recents
- Dashboards
- Alarms** 1 4 0
  - In alarm
  - All alarms
- Logs**
  - Log groups**
  - Logs Insights
- Metrics**
  - All metrics
  - Explorer
  - Streams

```

2023-05-25T17:49:17.166+02:00 REPORT RequestId: 9f77a7db-b127-5095-8204-fb323491d5eb Duration: 2195.89 ms Bille...
2023-05-25T17:50:14.946+02:00 START RequestId: dbfb2fec-6b3b-5d2e-97f2-7305eecc0290 Version: $LATEST
2023-05-25T17:50:15.046+02:00 2023-05-25T15:50:15.046Z dbfb2fec-6b3b-5d2e-97f2-7305eecc0290 INFO {"Records": [{"...
2023-05-25T17:50:15.897Z 2023-05-25T15:50:15.897Z dbfb2fec-6b3b-5d2e-97f2-7305eecc0290 INFO <!doctype html...
<html xmlns:og="http://opengraphprotocol.org/schema/" xmlns:fb="http://www.facebook.com/2008/fbml" lang="de-DE" class="touch-styles">
<head>
<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
<meta name="viewport" content="width=device-width, initial-scale=1.0, shrink-to-fit=no">
<!-- This is Squarespace. --><!-- kazoo-dove-h5ra -->
<base href="">
<meta charset="utf-8" />
<title>Taimos GmbH</title>
<meta http-equiv="Accept-CH" content="Sec-CH-UA-Platform-Version, Sec-CH-UA-Model" /><link rel="shortcut icon" type="image/x-icon" href="https://images.squarespace-cdn.com/content/v1/5ec2819b26f8d37144868e96/1606869165208-XQ8EX8A048LN34R329ZD/favicon.ico?format=100w"/>
<link rel="canonical" href="https://taimos.de"/>
<meta property="og:site_name" content="Taimos GmbH"/>
<meta property="og:title" content="Taimos GmbH"/>
```

This is the site called from the ECS service

AWS Services Search [Option+S] Frankfurt AdminAccess/hoegertn

CloudFormation Secrets Manager DynamoDB Step Functions

### CloudWatch

- Favorites and recents
- Dashboards
- Alarms** 1 4 0
- In alarm
- All alarms
- Logs**
- Log groups**
- Logs Insights

CloudWatch > Log groups

**Log groups (39)** By default, we only load up to 10000 log groups.

<input type="checkbox"/>	Log group	Data pro...	Sensitiv...	Retention	Metric fi...
<input type="checkbox"/>	/aws/lambda/demo-20230525-dev-Handler886CB40...	-	-	Never expire	-
<input type="checkbox"/>	<a href="#">demo-20230525-dev-EcsAppTaskDefwebLogGroupC5...</a>	-	-	Never expire	-

demo-2023| Actions View in Logs Insights Create log group

AWS Services Search [Option+S] Frankfurt AdminAccess/hoegertn

CloudFormation Secrets Manager DynamoDB Step Functions

### CloudWatch

- Favorites and recents
- Dashboards
- Alarms** 1 4 0
- In alarm
- All alarms
- Logs**
- Log groups**
- Logs Insights
- Metrics**
- All metrics
- Explorer
- Streams
- X-Ray traces**
- Service map
- Traces

CloudWatch > Log groups > demo-20230525-dev-EcsAppTaskDefwebLogGroupC52CEF25-e3U6zSBPj0oG

### demo-20230525-dev-EcsAppTaskDefwebLogGroupC52CEF25-e3U6zSB...

Actions View in Logs Insights Search log group

**Log group details**

ARN	Metric filters	Data protection - new
arn:aws:logs:eu-central-1:538118019757:log-group:demo-20230525-dev-EcsAppTaskDefwebLogGroupC52CEF25-e3U6zSBPj0oG:*	0	-
	Subscription filters	Sensitive data found - new
	0	-
	Contributor Insights rules	-
		KMS key ID
Creation time		-
6 minutes ago		
Retention		
Never expire		
Stored bytes		
-		

AWS Services Search [Option+S] Frankfurt AdminAccess/hoegertn

CloudFormation Secrets Manager DynamoDB Step Functions

### CloudWatch

- Favorites and recents
- Dashboards
- Alarms** 1 4 0
- In alarm
- All alarms
- Logs**
- Log groups**
- Logs Insights
- Metrics**
- All metrics
- Explorer
- Streams
- X-Ray traces**
- Service map
- Traces

arn:aws:logs:eu-central-1:538118019757:log-group:demo-20230525-dev-EcsAppTaskDefwebLogGroupC52CEF25-e3U6zSBPj0oG:\*

Subscription filters	-
Contributor Insights rules	-
Creation time	-
6 minutes ago	-
Retention	-
Never expire	-
Stored bytes	-

**Log streams** Metric filters Subscription filters Contributor Insights Tags Data protection - new

**Log streams (1)**

<input type="checkbox"/>	Log stream	Last event time
<input type="checkbox"/>	<a href="#">Resource/web/bac824fc90e64e8a934ef47f7bafe74a</a>	2023-05-25 17:48:45 (UTC+02:00)

Filter log streams or try prefix search  Exact match  Show expired [info](#) < 1 > [@](#)

AWS Services Search [Option+S] Frankfurt AdminAccess/hoegert

CloudFormation Secrets Manager DynamoDB Step Functions

CloudWatch ×

Favorites and recents ▶

Dashboards

Alarms ▶ 1 ○ 4 ○ 0

In alarm

All alarms

Logs ▶

Log groups

Logs Insights

Metrics ▶

All metrics

Explorer

Streams

X-Ray traces ▶

Service map

Traces

2023-05-25T17:50:39.831+02:00 CALLED!

2023-05-25T17:50:40.571+02:00 GOT RESULT

2023-05-25T17:50:41.929+02:00 CALLED!

2023-05-25T17:50:42.829+02:00 GOT RESULT

2023-05-25T17:50:50.217+02:00 CALLED!

2023-05-25T17:50:51.148+02:00 GOT RESULT

2023-05-25T17:50:54.844+02:00 CALLED!

2023-05-25T17:50:55.735+02:00 GOT RESULT

2023-05-25T17:50:56.935+02:00 CALLED!

2023-05-25T17:50:57.850+02:00 GOT RESULT

2023-05-25T17:51:05.232+02:00 CALLED!

2023-05-25T17:51:05.996+02:00 GOT RESULT

2023-05-25T17:51:09.854+02:00 CALLED!

2023-05-25T17:51:10.581+02:00 GOT RESULT

2023-05-25T17:51:11.949+02:00 CALLED!

2023-05-25T17:51:12.650+02:00 GOT RESULT

No newer events at this moment. Auto retry paused. Resume Back to top ^

2023-05-25 17:51:16

This worked.

← → × buildpacks.io

Buildpacks.io Features Community Blog Registry Docs GitHub

Cloud Native Buildpacks transform your application source code into images that can run on any cloud.

Get Started

Cloud build example - build for packer@buildpacks/builder:full

Introduction to Buildpacks: Java Example

Detecting Building Exporting

Java Node.js Python

Why Cloud Native Buildpacks?