Amazon OpenSearch Serverless

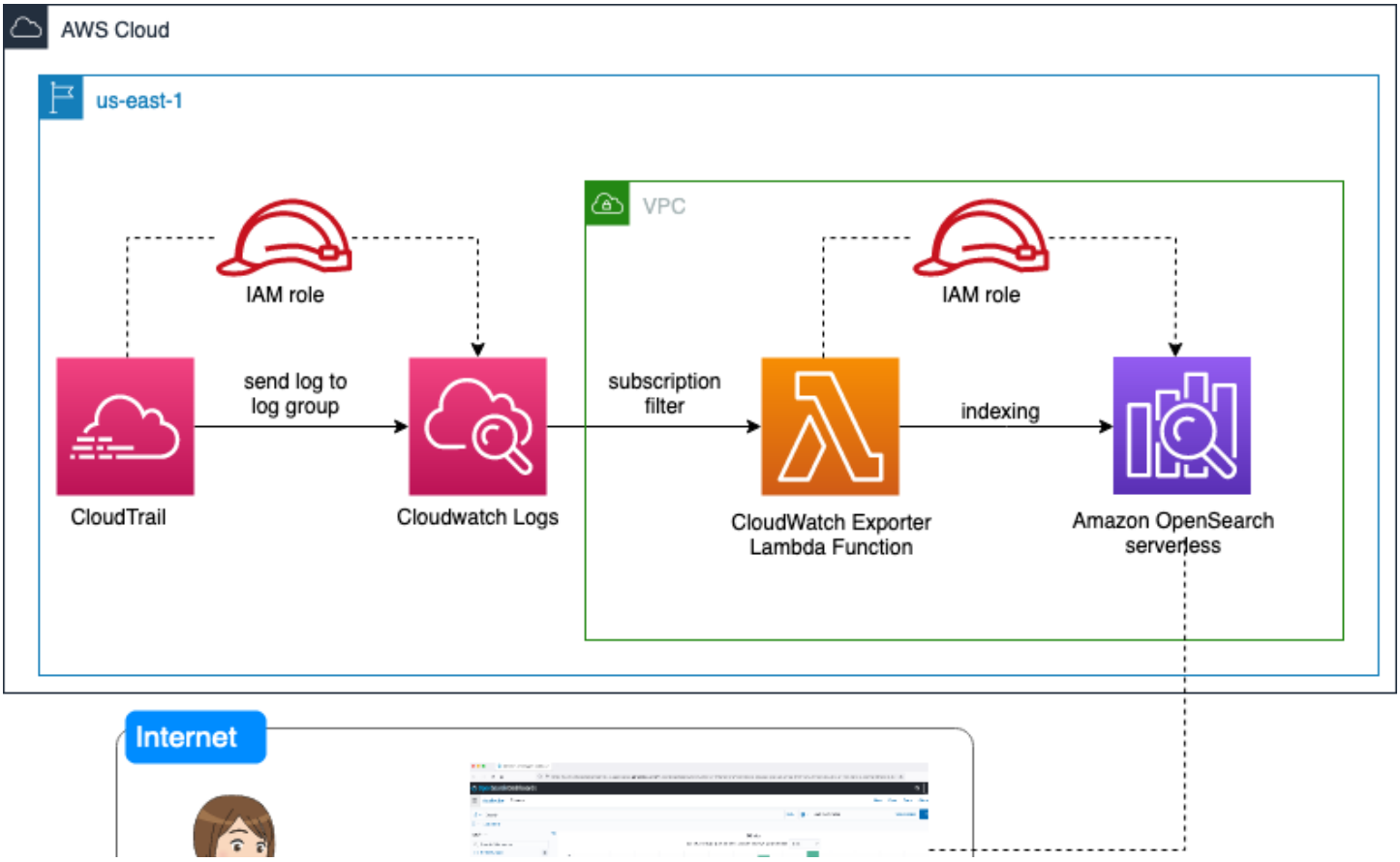
STABILITY STABLE

> **This is a stable example. It should successfully build out of the box**
>
> This example is built on Construct Libraries marked "Stable" and does not have any infrastructure prerequisites to build.

Overview

Do you want to stream AWS CloudTrail logs to Amazon OpenSearch Serverless and monitor it using OpenSearch Dashboard? Then this CDK is for you!

CDK example to create an Amazon OpenSearch Serverless, AWS Lambda, Amazon CloudWatch Logs and AWS CloudTrail logs using Python.



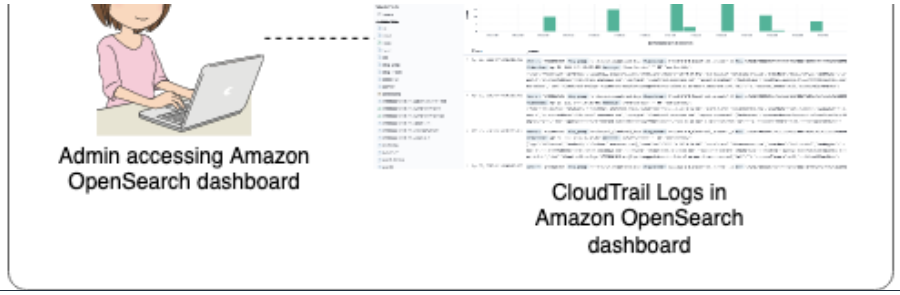


figure1.Architecture Diagram of CloudTrail log streaming to Amazon OpenSearch Serverless

This example demonstrates setting up a OpenSearch Serverless, CloudTrail and CloudWatch group to stream CloudTrail logs to OpenSearch Collection with Lambda and subscription filter using CDK.

Security

You can change the CloudTrail group name, Amazon OpenSearch collection name, CloudWatch retention. Below are default values. Here is the documentation of Amazon OpenSearch Serverless Security to read for you.

```
LOG_GROUP_NAME = "handler/svl_cloudtrail_logs"
COLLECTION_NAME = "ctcollection"
```

Currently we are using TLS with AWS encryption key, read more about Amazon OpenSearch Serverless encryption.

Once deployed, navigate to Amazon OpenSearch service console, select collections in Serverless section, select `ctcollection` and select dashboard URL, create your own Index pattern and explore the logs.

Build and Deploy

The `cdk.json` file tells the CDK Toolkit how to execute your app.

Python setup

This project is set up like a standard Python project. The initialization process also creates a virtualenv within this project, stored under the `.env` directory. To create the virtualenv it assumes that there is a `python3` (or `python` for Windows) executable in your path with access to the `venv` package. If for any reason the automatic creation of the virtualenv fails, you can create the virtualenv manually.

Run below command to create a virtualenv and to install local dependencies/requirements for lambda on MacOS and Linux.

```
$ bash bootstrap.sh
```

After the bootstrap process completes and the virtualenv is created, you can use the following step to activate your virtualenv.

```
$ source .env/bin/activate
```

If you are a Windows platform, you would activate the virtualenv like this:

```
% .env\Scripts\activate.bat
```

Once the virtualenv is activated, you can install the required dependencies.

```
$ pip install -r requirements.txt
```

At this point you can now synthesize the CloudFormation template for this code.

```
$ cdk synth
```

CDK Deploy

At this point you can deploy the stack to create OpenSearch Serverless domain, an AWS Lambda and cloud trail group, Index template and dashboards.

Using the default profile

```
$ cdk deploy
```

With specific profile

```
$ cdk deploy --profile test
```

CDK Destroy

To clean up AWS resources run below script.

⚠️Please delete below resource manually.

1. `CloudWatch Log groups`
2. `s3 buckets .`

```
$ cdk destroy
```