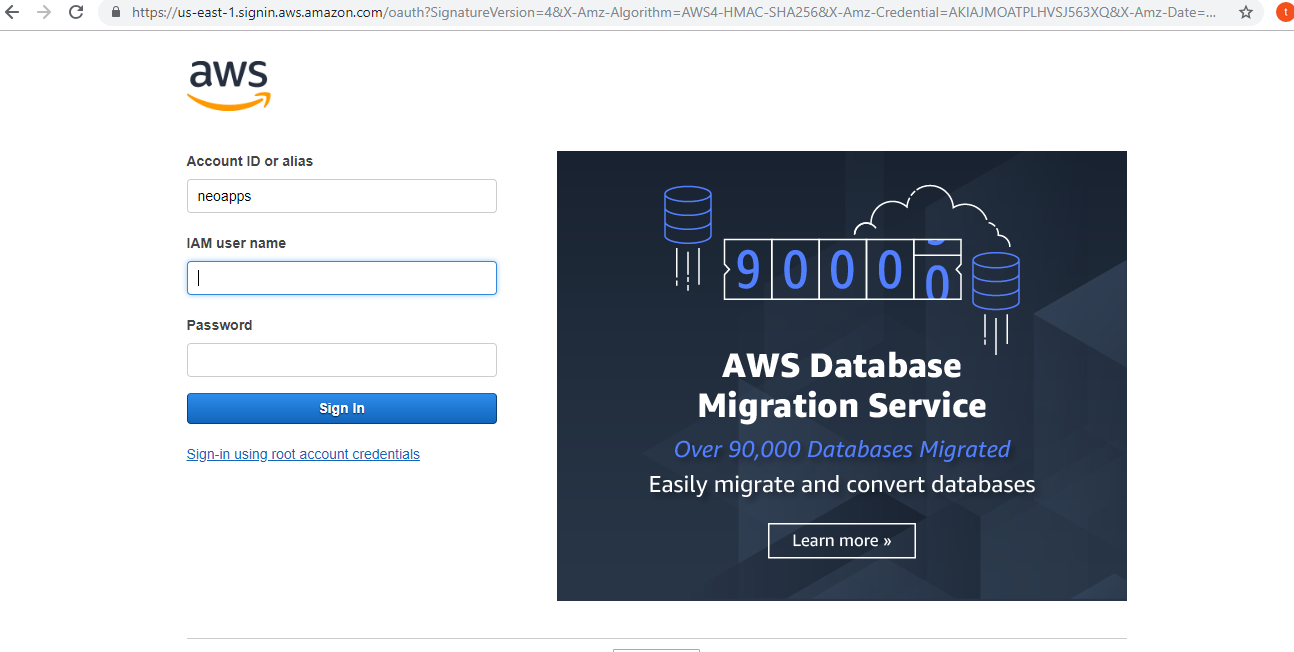
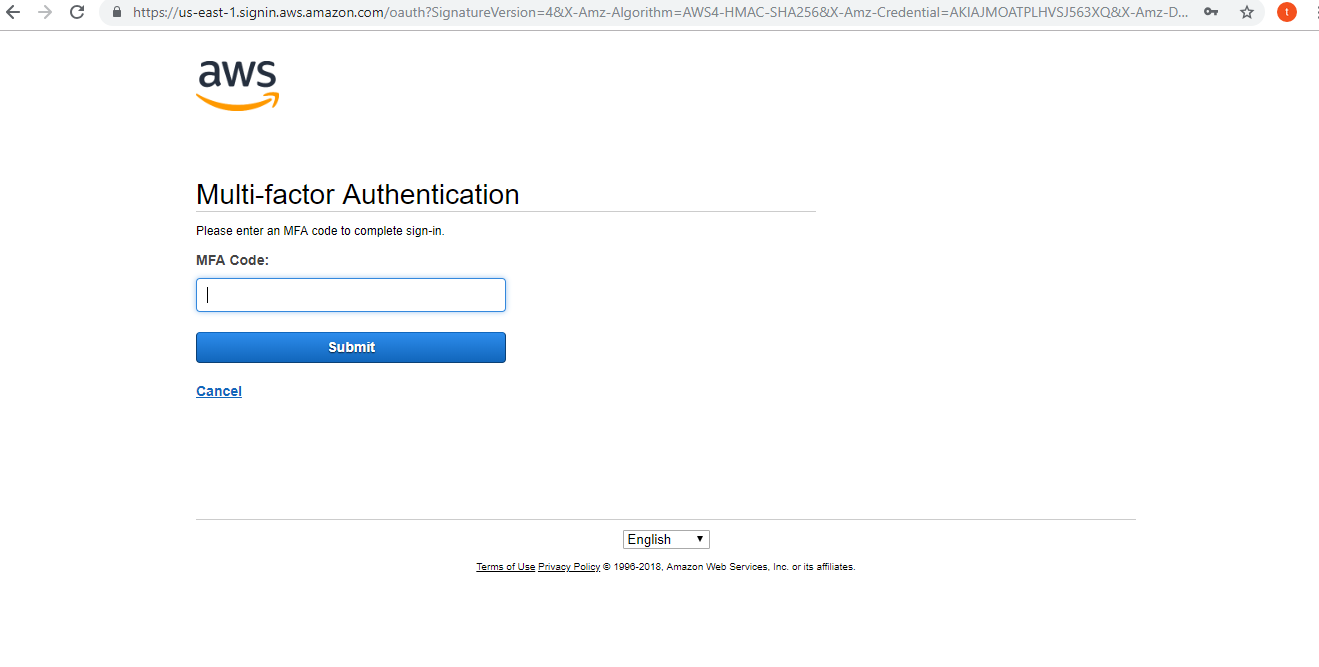
Description: Flow guide for Toshiba project

Author: Thulasiram.

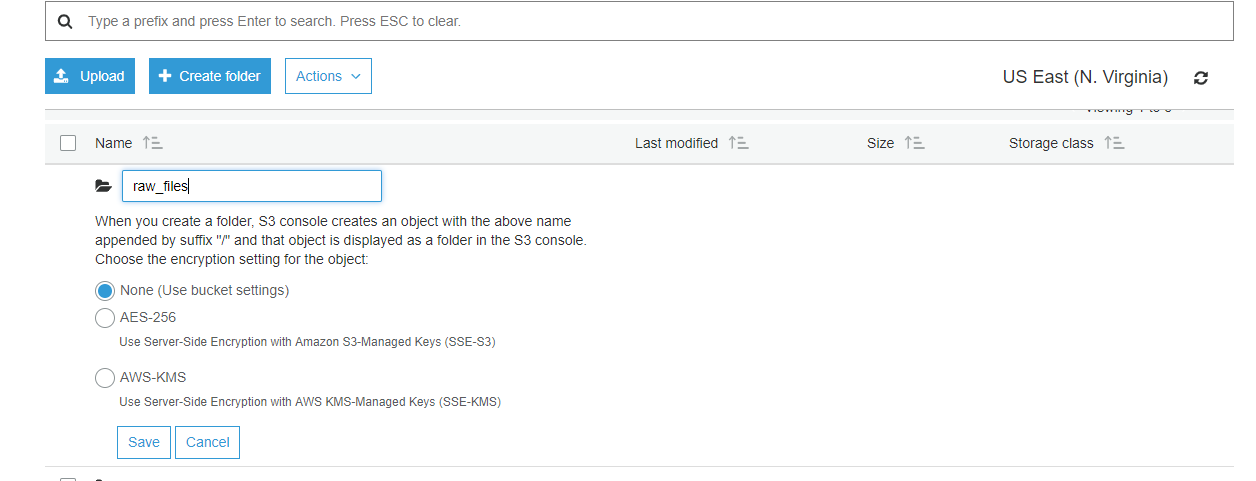
1. Login to the Neo apps console giving the IAM User name, Password and MFA code

url: <https://neoapps.signin.aws.amazon.com/console>

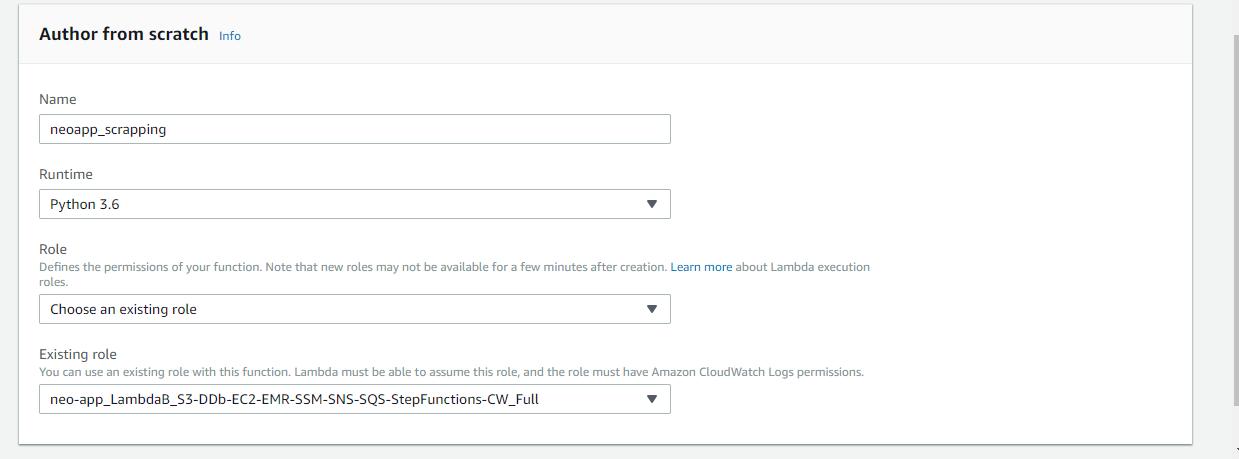




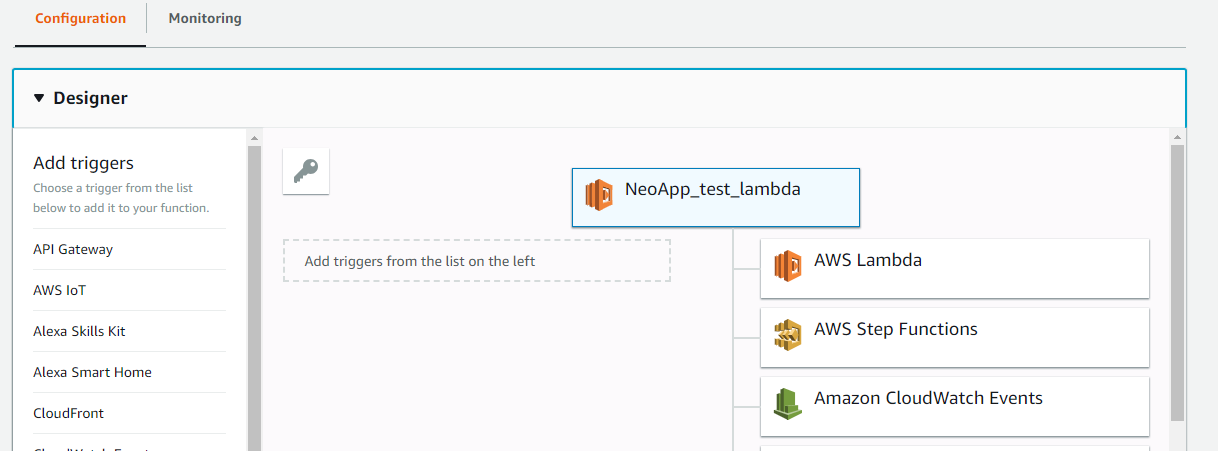
1. After logging into AWS choose the S3 service and select the Bucket which was assigned for the project. Once in the bucket, you can click on ‘create folder’ to create a folder or upload file directly.



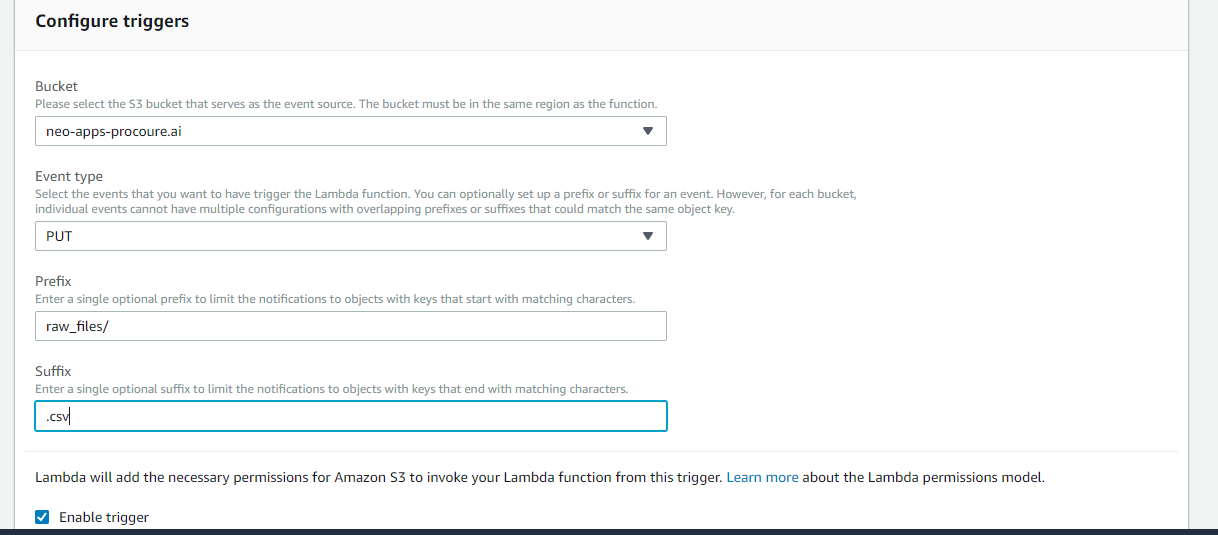
1. Now to create the lambda function, as per the below screen shot give a proper name to it, select runtime as python3.6 or the language in which you want to write the code, role should be chosen based on the requirements,



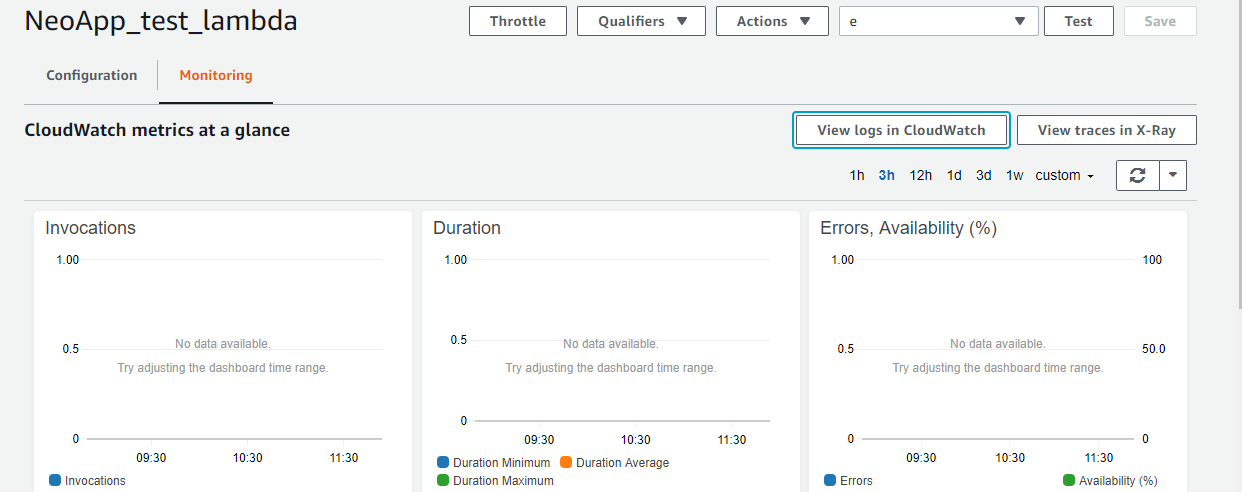
1. In the lambda function click on Designer and choose s3 from left hand side as a trigger.



1. After selecting the s3 as trigger. Configure the trigger by choosing the bucket name on which you are working, event type, prefix (which is nothing but folder in s3) and suffix to filter out file types that must trigger lambda. Example below.



1. In the Monitoring 🡪 View logs in cloudwatch we can see the logs corresponding to the lambda.

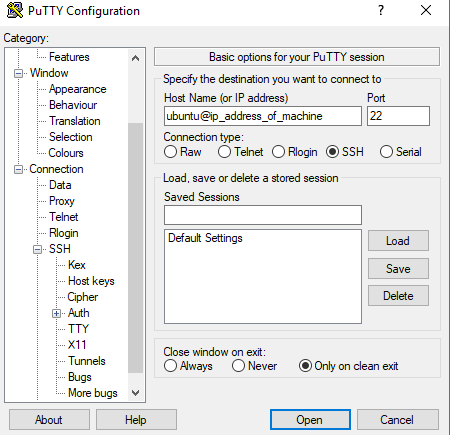


1. Install putty on windows machine.

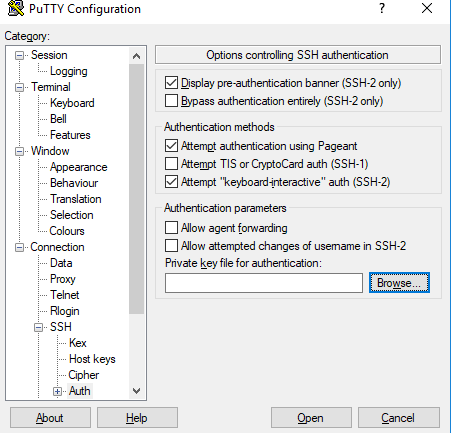
Links: <https://www.putty.org/>

1. After installing convert the pem file to ppk using ‘putty gen’. And connect to Ubuntu machine using Putty desktop app.

8.1) Give the ip address of ubuntu machine in session like below



8.2) In the SSH -> Auth-> Browse give the ppk file and click on open to connect to ubuntu machine.



1. Connect to Ubuntu machine from lambda to run a sample python code on server.

Example code:-

import time

import json

import paramiko

import boto3

import logging

logger = logging.getLogger()

logger.setLevel(logging.INFO)

def lambda\_handler(event, context):

""""

\* lambda function to start the instance and give the commands to execute program

""""

ec2\_client = boto3.client('ec2')

response = ec2\_client.start\_instances(

InstanceIds=[

'i-0bbe7527407062453'

])

logger.info(response)

# i-0bbe7527407062453 is the ec2 instance id

time.sleep(60) # In order to make sure that the instance is running before giving commands

s3\_client = boto3.client('s3')

s3\_client.download\_file('thulasi-ram-bucket','ec2andemr.pem', '/tmp/ec2.pem')

# to download ec2andemr.pem file from "thulasi-ram-bucket" bucket in to lambda temp folder

ssh = paramiko.SSHClient()

ssh.set\_missing\_host\_key\_policy(paramiko.AutoAddPolicy())

privkey = paramiko.RSAKey.from\_private\_key\_file('/tmp/ec2.pem')

ssh.connect('ec2-18-221-144-29.us-east-2.compute.amazonaws.com',username='ec2-user',pkey=privkey)

# ssh connect to the ec2 server

commands = ["wget -N https://s3-eu-west-1.amazonaws.com/thulasi-ram-dum/raw\_material\_final\_11.py",

"python36 raw\_material\_final\_11.py"]

# List of commands to run on ubuntu server

for command in commands:

ssh.exec\_command(command)

1. To stop the instance, give the following lines of code at the end of python program that you are running on the ec2 instance. In the above program eg: is raw\_material\_final\_11.py

Note: To stop the instance, ec2 instance should have the role attached which have policy ec2 full permission.

Code :

ec2\_client = boto3.client("ec2", region\_name = "eu-west-1")

resp = ec2\_client.stop\_instances(InstanceIds = ["i-0b774eaa8e8a77c56"])