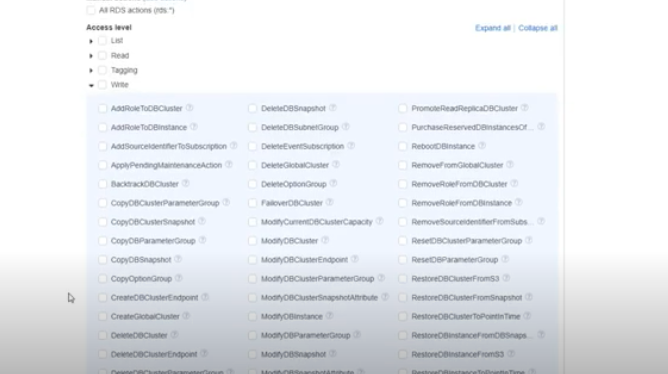
Use lambda to copy Automated RDS backups to another region

Create a policy that allows us to copy RDS snapshots to the new region

1. Open IAM console and select policies
2. Click on create policy
3. Select RDS service
4. Under the “write permission”, select “copyDBsnapshots”



1. Under resources select “All Resources”
2. Provide a name for your policy as “Copy-RDS-Backups” and click create policy
3. Click “Create Role”
4. Choose “Lambda” as the service since it will be assumed by Lambda
5. Select “AWSLAMBDABasicExecutionRole” (This role allows us to write to cloudwatch)
6. Select “Copy-RDS-Backups” as well (New policy that was created)
7. Provide a role name “Lambda-Copy-RDS-Backups”
8. Create Lambda function and name it “Lambda-Copy-RDS-Backups-Function”
9. Select Python 3.7 and select the Role that was created
10. Copy the lambda code below

# Written By GeekTopia

#

# Copy RDS Automated snapshots to a new region upon creation

# --Free to use under all conditions

# --Script is provied as is. No Warranty, Express or Implied

import json

import boto3

destinationRegion = "us-east-1"

def lambda\_handler(event, context):

sourceRegion = event['region']

rds = boto3.client('rds',region\_name=destinationRegion)

#Build the Snapshot ARN - Always use the ARN when copying snapshots across region.

sourceSnapshotARN = event['detail']['SourceArn']

sourceSnapshotARN= sourceSnapshotARN.replace(":db:",":snapshot:")

#build a new snapshot name

sourceSnapshotIdentifer = event['detail']['SourceIdentifier']

targetSnapshotIdentifer ="{0}-ManualCopy".format(sourceSnapshotIdentifer)

targetSnapshotIdentifer = targetSnapshotIdentifer.replace(":","-")

#Execute copy

try:

copy = rds.copy\_db\_snapshot(SourceDBSnapshotIdentifier=sourceSnapshotARN,TargetDBSnapshotIdentifier=targetSnapshotIdentifer,SourceRegion=sourceRegion)

print("Started Copy of Snapshot {0} in {2} to {1} in {3} ".format(sourceSnapshotIdentifer,targetSnapshotIdentifer,sourceRegion,destinationRegion))

except ClientError as ex:

if ex.response['Error']['Code'] == 'DBSnapshotAlreadyExists':

print("Snapshot {0} already exist".format(targetSnapshotIdentifer))

else:

print("ERROR: {0}".format(ex.response['Error']['Code']))

return {

'statusCode': 200,

'body': json.dumps('Opearation Complete')

}

1. Open cloud watch
2. Click “Rules” on the left
3. Click “Create Rule”
4. Click the edit button on the edit button on the right hand corner
5. Paste the following filter

{

"source": [

"aws.rds"

],

"detail-type": ["RDS DB Snapshot Event"],

"detail": {

"Message":["Automated snapshot created"]

}

}

1. On the right click on add target
2. Select the lambda function and click next
3. Provide a name for the rule “InvokeonRDSSnapshot”