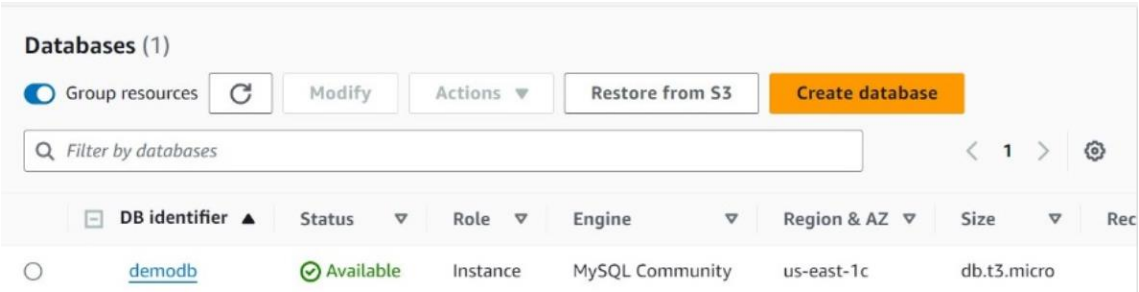
TASK 2

Use Sqoop command to ingest the data from RDS into the HBase Table.

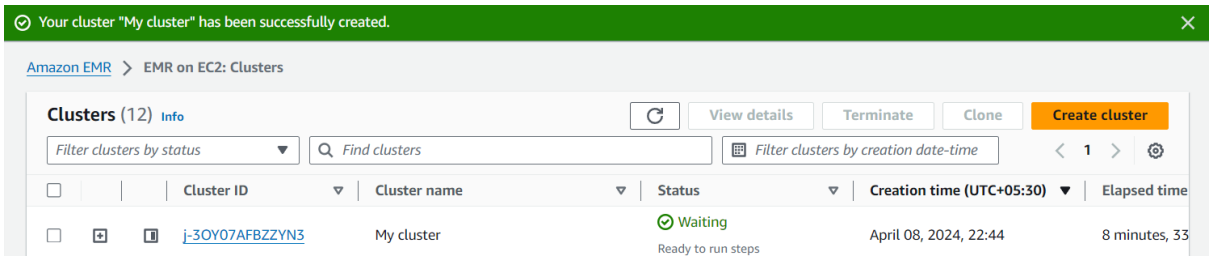
Step 1: Creating a RDS instance.

- Search for RDS on Amazon Web Server.
- Click on create database under database menu.
- Further choose “Standard create” from the database creation method.
- Next click on MySQL from Engine Options to run a MySQL engine.
- Click on Free tier under the Templates option to use the engine for free.
- Now, give a name to your database under DB cluster identifier.
- Next, give a public access to your database by clicking on yes under the Public Access option.
- Ensure to make the database password authentication under database authentication for security.
- At last, click on Create Database to create the RDS Database.
- The database will take 5-10 minutes to get active.



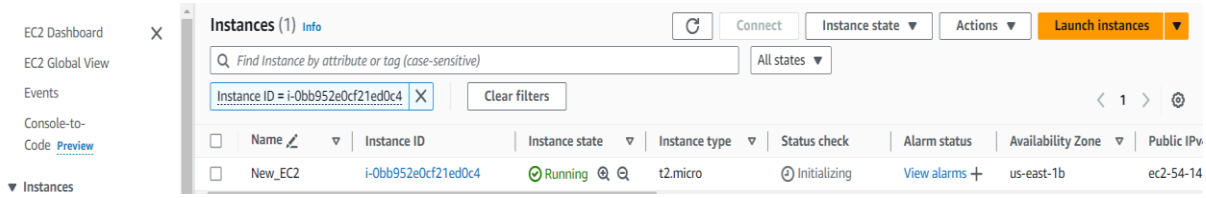
Step 2: Creating an EMR cluster.

- Search for EMR on Amazon Web Server.
- Click on “create cluster” to create a EMR cluster
- Now, give name to your cluster under the name column.
- Next, tick the required applications under the application bundle for example Hadoop, HBase, Sqoop etc.
- Further choose an EC2 instance type for Primary node.
- Next enter the Size required (30 GiB) under EBS root volume.
- Now, select the EC2 key pair for SSH to the cluster by creating a new key pair.
- Now at last, select the service role and instance profile as default role
- Finally, click on Create Cluster to create an EMR Cluster
- The cluster will take 5-10 minutes to get active



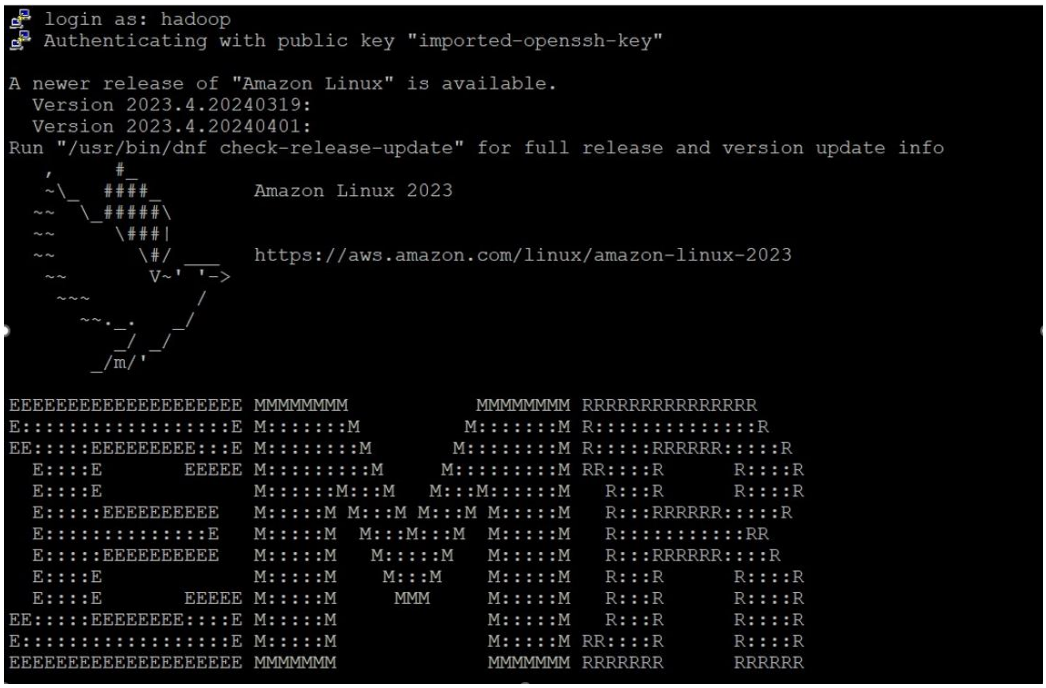
Step 3: Creating an EC2 instance

- Search for EC2 on Amazon Web Server
- Click on Launch instances to create a EC2 instance
- Now, give a name to your EC2 instance under the Name and Tags option
- Now, Select a desired Amazon Machine Image for example, Amazon Linux, macOS etc.
- Now select the instance type which is required for example t2.micro with 1 vCPU and 1GiB memory
- Now, select a key pair name from the drop-down menu or create a new key pair
- At last, click on Launch Instance to create a EC2 Instance
- The creation of the instance may take around 5-10 minutes to get active



Step 4. Connecting EMR instance with the local computer using Putty.

- Firstly, copy the Primary node public DNS from the EMR cluster and paste it as a host name or IP Address in Putty
- Now, click on credentials under the Auth menu from SSH and browse for private key file for authentication.
- After inserting the private key, click open and the connection between EMR instance with the local computer will be executed.



Step 5: Execute below command

sudo yum update

sudo yum install python3 python3-pip

python3 --version

pip3 --version

pip3 install thriftpy2

Step 6: Execute the below command on Hadoop

sqoop import --connect jdbc:mysql://demodb.crog2ckcif6b.us-east-1.rds.amazonaws.com/taxi_records --username admin --password admin123 --table trip_log --hbase-table trip_log_hbase --column-family cf1 --hbase-create-table --hbase-row-key tpep_pickup_datetime,tpep_dropoff_datetime --hbase-bulkload --split-by payment_type

Step 7: Execute below python script in root

vi task3.py

----------Python script----------

```
import happybase

# create connection
connection = happybase.Connection('localhost', port=9090, autoconnect=False)

# open connection to perform operations
def open_connection():
    connection.open()

# close opened connection
def close_connection():
    connection.close()

# get the pointer to a table
def get_table(name):
    open_connection()
```

```
    table = connection.table(name)
    close_connection()
    return table
def batch_insert_data(filename, tablename):
    print("starting batch insert of "+filename)
    file = open(filename, 'r')
    table = get_table(tablename)
    open_connection()
    i = 0
    with table.batch(batch_size=50000) as b:
        for line in file:
            if i!=0:
                temp = line.strip().split(",")
                b.put(temp[1]+temp[2] ,{'cf1:VendorID': str(temp[0]), 'cf1:tpep_pickup_datetime':
str(temp[1]), 'cf1:tpep_dropoff_datetime': str(temp[2]), 'cf1:passenger_count': str(temp[3]),
'cf1:trip_distance': str(temp[4]), 'cf1:RatecodeID': str(temp[5]), 'cf1:store_and_fwd_flag':
str(temp[6]), 'cf1:PULocationID': str(temp[7]), 'cf1:DOLocationID': str(temp[8]), 'cf1:payment_type':
str(temp[9]),'cf1:fare_amount': str(temp[10]), 'cf1:extra': str(temp[11]), 'cf1:mta_tax': str(temp[12]),
'cf1:tip_amount': str(temp[13]), 'cf1:tolls_amount': str(temp[14]), 'cf1:improvement_surcharge':
str(temp[15]), 'cf1:total_amount': str(temp[16]), 'cf1:congestion_surcharge': str(temp[17]),
'cf1:airport_fee': str(temp[18]) })


            i+=1


    file.close()
    print("batch insert done")
    close_connection()


batch_insert_data('yellow_tripdata_2017-01.csv', 'trip_log')
batch_insert_data('yellow_tripdata_2017-02.csv', 'trip_log')
```

----------Python script----------

```
python task3.py
```

Step 8: After this, Execute below commands

sudo -i

cd hbase

hbase shell

list

describe 'trip_log_hbase'

count 'trip_log_hbase'

Resulting in,

Ingesting the data from RDS into the HBase Table.