**AWS Pop-up Loft – Database Week**

Upgrading and Consolidating MySQL

Hands-on Lab

1. Create an Amazon EC2 t2.small instance with Amazon Linux 2
   * See documentation at

<http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EC2_GetStarted.html>

* + Create and use a security group that allows inbound TCP access using SSH (port 22) and MYSQL (port 3306) for all AWS sources:

172.0.0.0/8

10.0.0.0/8

You might get an automated warning that your EC2 instance is “open to the world”, because we’re not limiting the source range for SSH. This is expected. In a production system, you’ll want to provide a limited IP range for allowed SSH access. For this lab, disregard the warning.

1. Access the linux console. See documentation at

http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AccessingInstances.html

Use ssh for Linux or Mac; use PuTTY for Windows  
Example:

ssh –i ~/Downloads/key.pem ec2-user@ec2-01-02-03-99.us-west-2.compute.amazonaws.com

[ec2-user@ip-192-168-0-1 ~]$

* + Install MySql development client

$ sudo yum install mysql

$ sudo yum install mysql-devel

1. Create an Amazon RDS MySQL db.t2.small instance
   * See documentation at

<http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/CHAP_GettingStarted.CreatingConnecting.MySQL.html>

* + Name the instance sql-lab. Be sure to choose MySQL (not Aurora and not MariaDB), the db.t2.small instance size, a Dev/Test use case, and MySQL version 5.5.57
  + Choose a master username and a password (and don’t forget them!)
  + Do not create a database (we will do that later)
  + Once the instance is created, find your mysql endpoint name
    1. On the AWS Console, choose Services, then RDS
    2. On the RDS dashboard, choose DB Instances



* + 1. Select your DB Instance
    2. Note your endpoint name. You will need it later!

When using the endpoint name, you usually should not use the port extension (:3306), just the name.

* + Verify you can access the mysql> console from your EC2 instance  
    $ mysql –h <mysql node name> -u <user name> -p

Example:   
$ mysql -h sql-lab.cxpjiluqq0c9.us-west-2.rds.amazonaws.com -u awsuser -p

Enter password:

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 9999

Server version: 5.6.27-log MySQL Community Server (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>

To exit from the mysql> prompt, use CTRL-D

1. Download and Prepare Landsat scenes
   * See documentation at

<https://aws.amazon.com/public-data-sets/landsat/>

* + From your EC2 instance, download the Landsat scenes  
    $ wget http://landsat-pds.s3.amazonaws.com/scene\_list.gz
  + Unzip the scene list  
    $ gunzip scene\_list.gz

1. Load to MySQL
   * Log into the mysql> console
   * Create a landsat database  
     mysql> CREATE DATABASE landsat;  
     mysql> USE landsat;
   * Create the scene\_list table  
     mysql> CREATE TABLE scene\_list (entityId VARCHAR(64), acquisitionDate DATETIME,cloudCover DECIMAL(5,2),processingLevel VARCHAR(8),path INT,row INT,min\_lat DECIMAL(8,5),min\_lon DECIMAL(8,5),max\_lat DECIMAL(8,5),max\_lon DECIMAL(8,5),download\_url VARCHAR(128));
   * Load the landsat data  
     mysql> LOAD DATA LOCAL INFILE 'scene\_list' INTO TABLE scene\_list FIELDS TERMINATED BY ',';
2. Run SQL query

* Log in to your MySQL node and run a query

mysql> SELECT COUNT(\*) FROM scene\_list;

1. Create an Amazon Aurora MySQL db.r3.large instance
   * See documentation at

<https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/CHAP_GettingStarted.CreatingConnecting.Aurora.html>

* + Name the instance sql-aurora. Be sure to choose Aurora, MySQL 5.7-compatible, and the db.r3.largeinstance size. Do not create a replica in a different Zone.
  + Choose a master username and a password (and don’t forget them!)
  + Do not create a database (we will do that later)
  + Once the instance is created, find your mysql endpoint name
    1. On the AWS Console, choose Services, then RDS
    2. On the RDS dashboard, choose DB Instances



* + 1. Select your Aurora DB Instance
    2. Note your endpoint name. You will need it later!

When using the endpoint name, you usually should not use the port extension (:3306), just the name.

* + Verify you can access the mysql> console from your EC2 instance  
    $ mysql –h <mysql node name> -u <user name> -p

Example:   
$ mysql -h sql-aurora.cxpuqq0c9.us-west-2.rds.amazonaws.com -u awsuser -p

Enter password:

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 9999

Server version: 5.6.27-log MySQL Community Server (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>

To exit from the mysql> prompt, use CTRL-D

1. Create a dms.t2.medium DMS replication instance
   * See documentation at

<https://us-west-2.console.aws.amazon.com/dms/home?region=us-west-2#create-first-migration>

Name the replication instance dms-lab. Be sure to choose the dms.t2.medium instance size. Do not choose Multi-AZ.

* + Enter source and target database information. Use SSL mode “none”. Be sure to test each connection

1. Create and start your migration Task  
   Name the task migrate.
   * Enable validation and logging
   * Include all tables in the “landsat” schema
   * Be sure to START the task once it is created

After the migration is complete, log in to your Aurora instance and run

mysql> SELECT COUNT(\*) FROM scene\_list;

Did all of the data migrate?