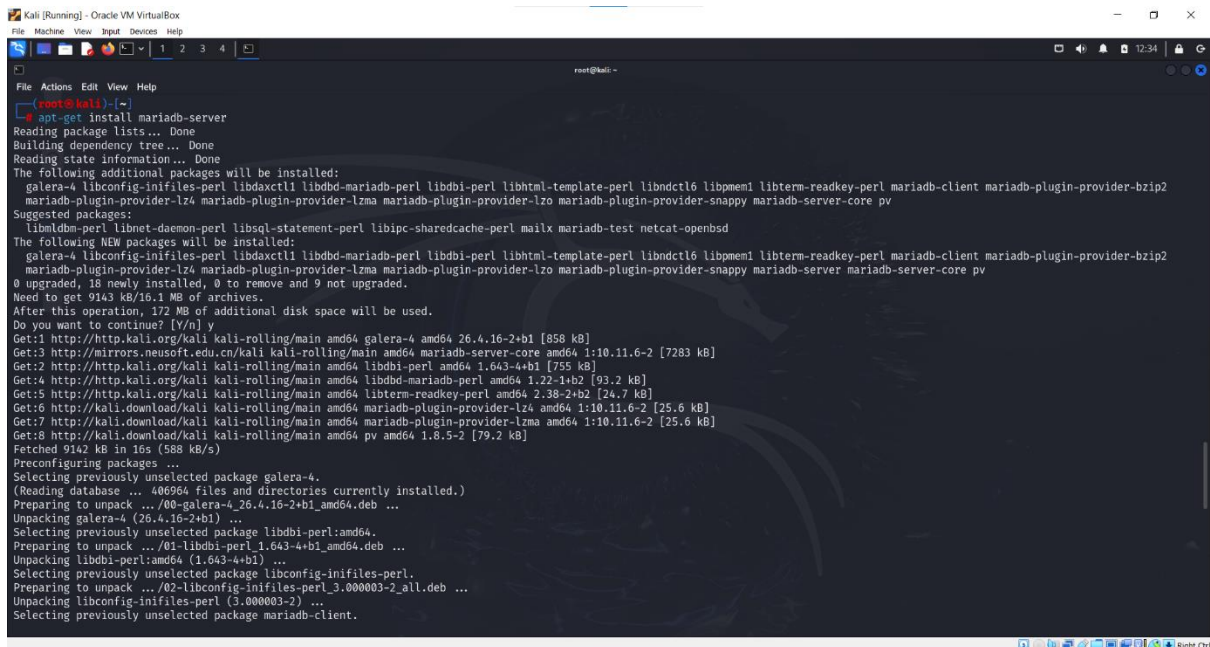


Onpremise to Cloud Migration

Step 1 : Create a database in your local system.

In your local system, download MariaDB package using the following command.

apt-get install mariadb-server



```
Kali [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
root@kali: ~
File Actions Edit View Help
root@kali: ~
# apt-get install mariadb-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  galera-4 libconfig-inifiles-perl libdaxctl1 libdbd-mariadb-perl libdbi-perl libhtml-template-perl libndctl6 libpmem1 libterm-readkey-perl mariadb-client mariadb-plugin-provider-bzip2
  mariadb-plugin-provider-lz4 mariadb-plugin-provider-lzma mariadb-plugin-provider-lzo mariadb-plugin-provider-snappy mariadb-server-core pv
Suggested packages:
  libmdbm-perl libnet-daemon-perl libsql-statement-perl libipc-sharedcache-perl mailx mariadb-test netcat-openbsd
The following NEW packages will be installed:
  galera-4 libconfig-inifiles-perl libdaxctl1 libdbd-mariadb-perl libdbi-perl libhtml-template-perl libndctl6 libpmem1 libterm-readkey-perl mariadb-client mariadb-plugin-provider-bzip2
  mariadb-plugin-provider-lz4 mariadb-plugin-provider-lzma mariadb-plugin-provider-lzo mariadb-plugin-provider-snappy mariadb-server mariadb-server-core pv
0 upgraded, 18 newly installed, 0 to remove and 9 not upgraded.
Need to get 9143 kB/16.1 MB of archives.
After this operation, 172 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://http.kali.org/kali kali-rolling/main amd64 galera-4 amd64 26.4.16-2+b1 [858 kB]
Get:2 http://mirrors.neusoft.edu.cn/kali kali-rolling/main amd64 mariadb-server-core amd64 1:10.11.6-2 [7283 kB]
Get:3 http://http.kali.org/kali kali-rolling/main amd64 libdbi-perl amd64 1.643-4+b1 [755 kB]
Get:4 http://http.kali.org/kali kali-rolling/main amd64 libdbd-mariadb-perl amd64 1.22-1+b2 [93.2 kB]
Get:5 http://http.kali.org/kali kali-rolling/main amd64 libterm-readkey-perl amd64 2.38-2+b2 [24.7 kB]
Get:6 http://kali.download/kali kali-rolling/main amd64 mariadb-plugin-provider-lz4 amd64 1:10.11.6-2 [25.6 kB]
Get:7 http://http.kali.org/kali kali-rolling/main amd64 mariadb-plugin-provider-lzma amd64 1:10.11.6-2 [25.6 kB]
Get:8 http://kali.download/kali kali-rolling/main amd64 pv amd64 1.8.5-2 [79.2 kB]
Fetched 9142 kB in 16s (588 kB/s)
Preconfiguring packages ...
Selecting previously unselected package galera-4.
(Reading database ... 406964 files and directories currently installed.)
Preparing to unpack .../00-galera-4_26.4.16-2+b1_amd64.deb ...
Unpacking galera-4 (26.4.16-2+b1) ...
Selecting previously unselected package libdbi-perl:amd64.
Preparing to unpack .../01-libdbi-perl_1.643-4+b1_amd64.deb ...
Unpacking libdbi-perl:amd64 (1.643-4+b1) ...
Selecting previously unselected package libconfig-inifiles-perl.
Preparing to unpack .../02-libconfig-inifiles-perl_3.000003-2_all.deb ...
Unpacking libconfig-inifiles-perl (3.000003-2) ...
Selecting previously unselected package mariadb-client.
Preparing to unpack .../03-mariadb-client_10.11.6-2_amd64.deb ...
Unpacking mariadb-client (10.11.6-2) ...
Selecting previously unselected package mariadb-server-core.
Preparing to unpack .../04-mariadb-server-core_10.11.6-2_amd64.deb ...
Unpacking mariadb-server-core (10.11.6-2) ...
Selecting previously unselected package mariadb-plugin-provider-lz4.
Preparing to unpack .../05-mariadb-plugin-provider-lz4_10.11.6-2_amd64.deb ...
Unpacking mariadb-plugin-provider-lz4 (10.11.6-2) ...
Selecting previously unselected package mariadb-plugin-provider-lzma.
Preparing to unpack .../06-mariadb-plugin-provider-lzma_10.11.6-2_amd64.deb ...
Unpacking mariadb-plugin-provider-lzma (10.11.6-2) ...
Selecting previously unselected package mariadb-plugin-provider-lzo.
Preparing to unpack .../07-mariadb-plugin-provider-lzo_10.11.6-2_amd64.deb ...
Unpacking mariadb-plugin-provider-lzo (10.11.6-2) ...
Selecting previously unselected package mariadb-plugin-provider-snappy.
Preparing to unpack .../08-mariadb-plugin-provider-snappy_10.11.6-2_amd64.deb ...
Unpacking mariadb-plugin-provider-snappy (10.11.6-2) ...
Selecting previously unselected package mariadb-server.
Preparing to unpack .../09-mariadb-server_10.11.6-2_amd64.deb ...
Unpacking mariadb-server (10.11.6-2) ...
Setting up galera-4 (26.4.16-2+b1) ...
Setting up libconfig-inifiles-perl (3.000003-2) ...
Setting up libdbi-perl:amd64 (1.643-4+b1) ...
Setting up libdbd-mariadb-perl (1.22-1+b2) ...
Setting up libterm-readkey-perl (2.38-2+b2) ...
Setting up mariadb-client (10.11.6-2) ...
Setting up mariadb-plugin-provider-lz4 (10.11.6-2) ...
Setting up mariadb-plugin-provider-lzma (10.11.6-2) ...
Setting up mariadb-plugin-provider-lzo (10.11.6-2) ...
Setting up mariadb-plugin-provider-snappy (10.11.6-2) ...
Setting up mariadb-server-core (10.11.6-2) ...
Setting up mariadb-server (10.11.6-2) ...
Processing triggers for libc-bin (2.35-0ubuntu3) ...
```

We want to create a database which we will migrate to the cloud. To create a database, login to the MariaDB server as root user using following command.

mysql -h localhost -u root

After logging into mariadb server, create a database by firing the query given below.

CREATE DATABASE test;



```
root@kali:~# mysql -h localhost -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 32
Server version: 10.11.6-MariaDB-2 Debian n/a

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
4 rows in set (0.001 sec)

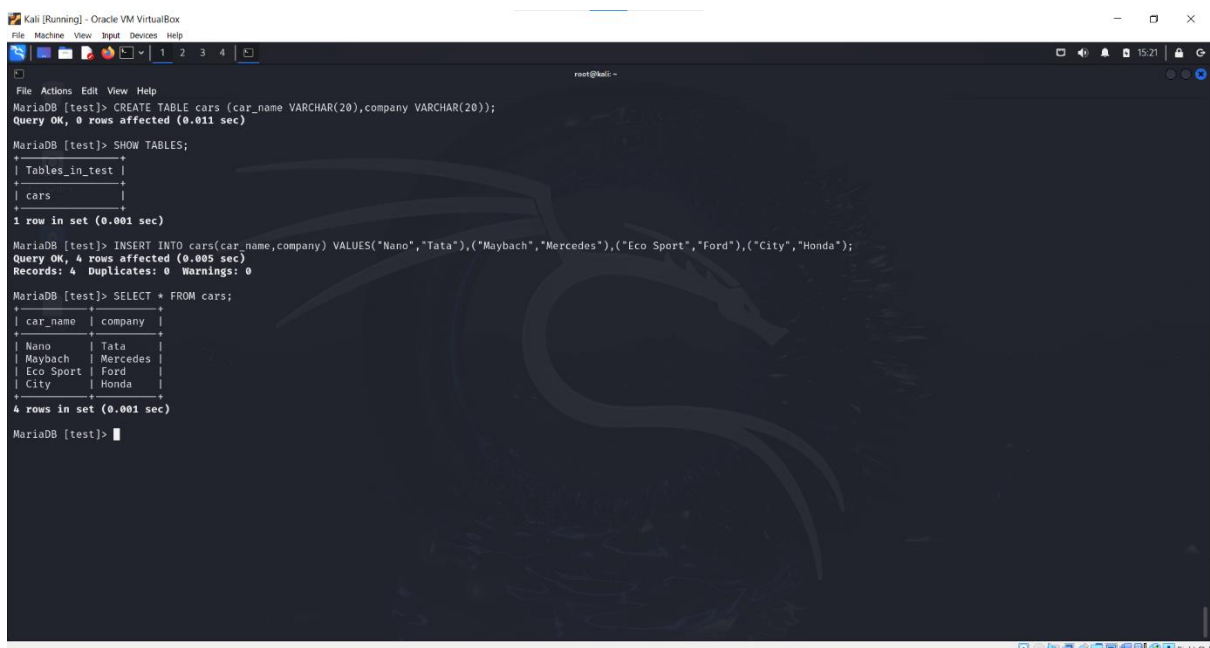
MariaDB [(none)]> CREATE DATABASE test
→ ;
Query OK, 1 row affected (0.001 sec)

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
| test |
+-----+
5 rows in set (0.001 sec)

MariaDB [(none)]>
```

Now populate your database by adding a table and data into it, I have used following queries to create a table and add data into that table.

CREATE TABLE cars(car_name VARCHAR(20),company VARCHAR(20));



```
MariaDB [test]> CREATE TABLE cars(car_name VARCHAR(20),company VARCHAR(20));
Query OK, 0 rows affected (0.011 sec)

MariaDB [test]> SHOW TABLES;
+-----+
| Tables_in_test |
+-----+
| cars |
+-----+
1 row in set (0.001 sec)

MariaDB [test]> INSERT INTO cars(car_name,company) VALUES('Nano','Tata'),('Maybach','Mercedes'),('Eco Sport','Ford'),('City','Honda');
Query OK, 4 rows affected (0.005 sec)
Records: 4  Duplicates: 0  Warnings: 0

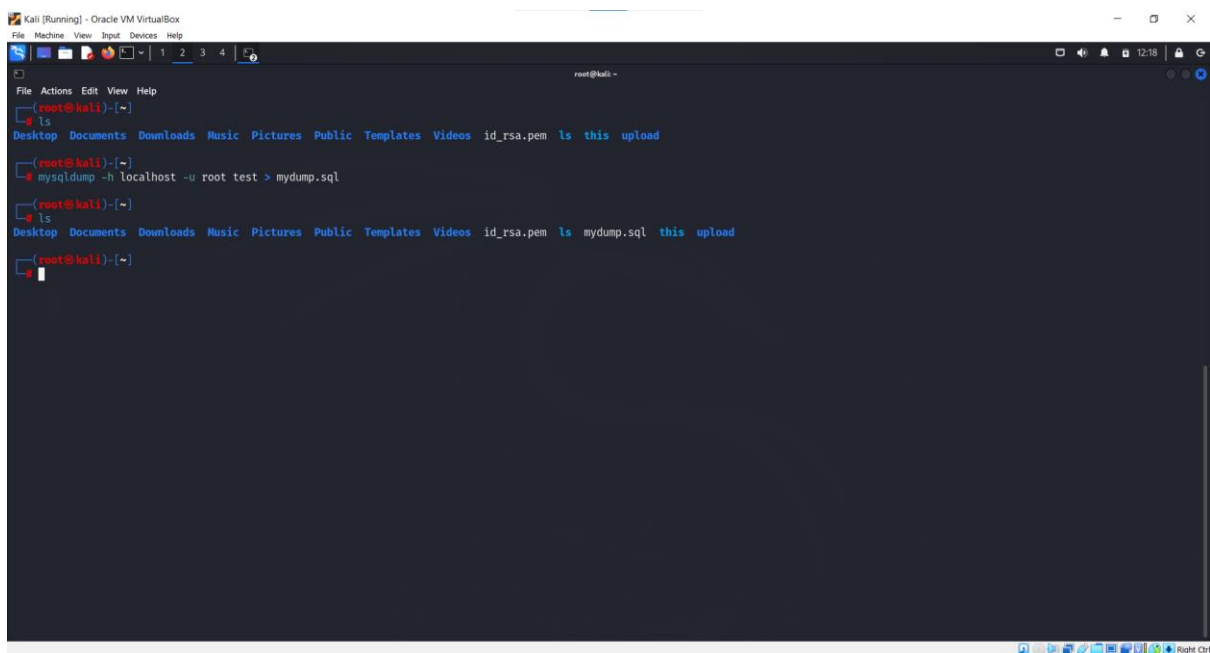
MariaDB [test]> SELECT * FROM cars;
+-----+-----+
| car_name | company |
+-----+-----+
| Nano | Tata |
| Maybach | Mercedes |
| Eco Sport | Ford |
| City | Honda |
+-----+-----+
4 rows in set (0.001 sec)

MariaDB [test]>
```

Step 2 : Create backup file for database

Inorder to migrate our database to the cloud environment we need to first create its backup file. This can be done using 'mysqldump' command as shown below.

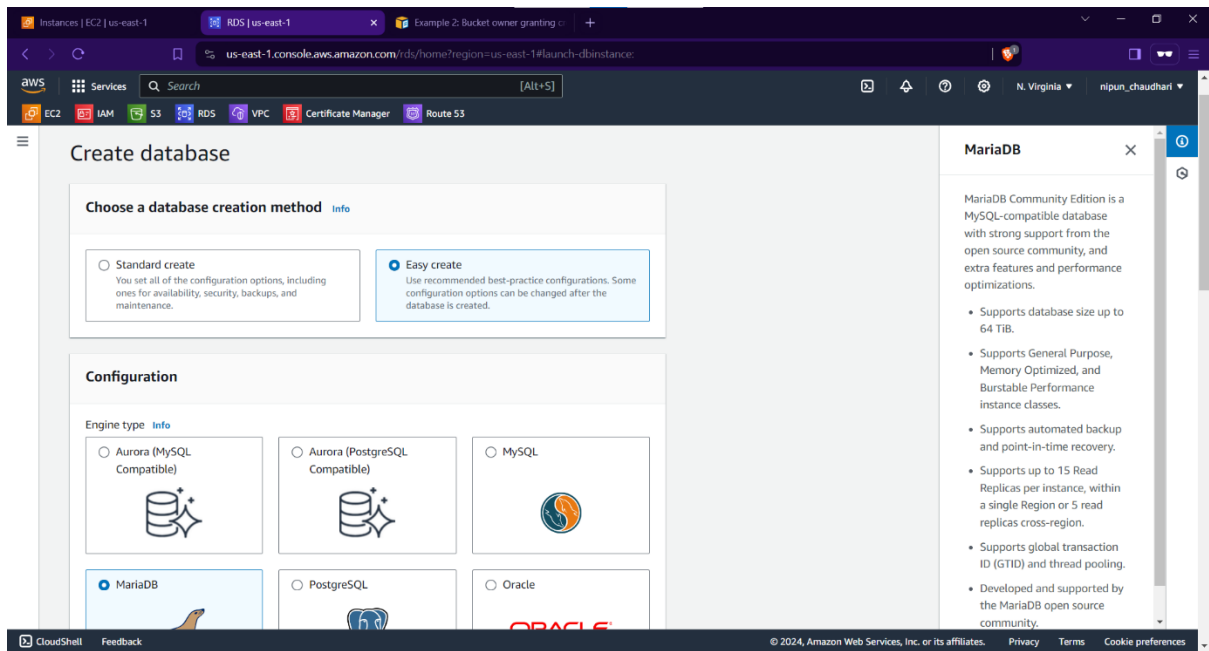
mysqldump -h localhost -u root test > mydump.sql



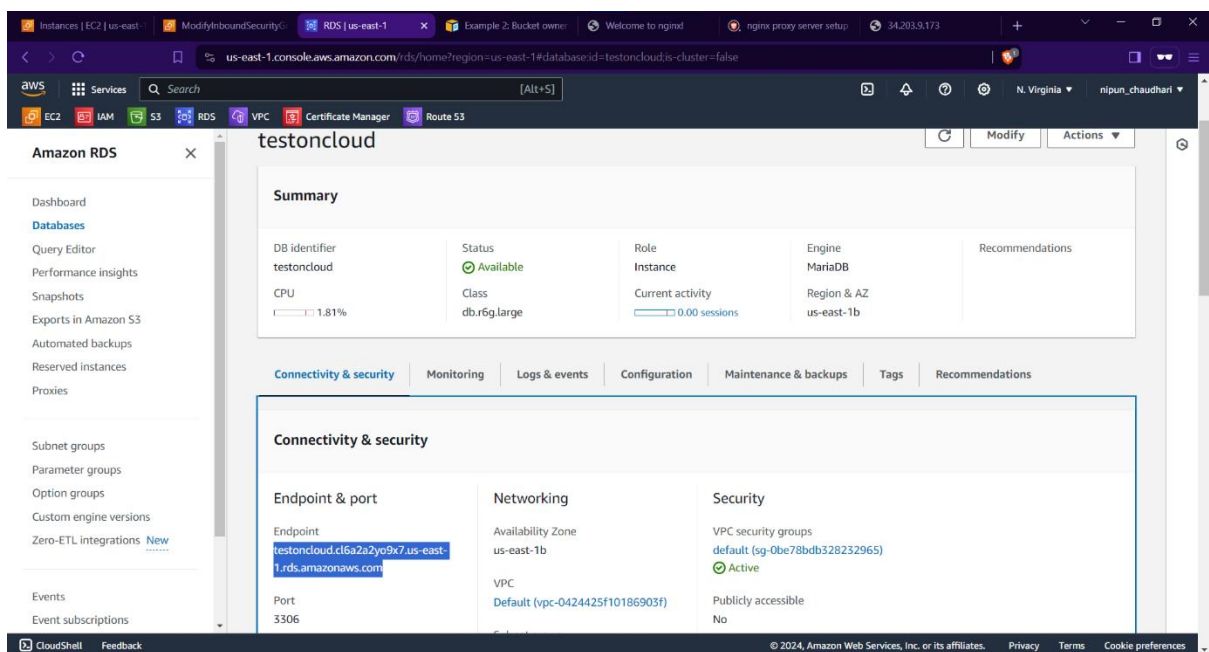
```
Kali [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@kali: ~
ls
Desktop Documents Downloads Music Pictures Public Templates Videos id_rsa.pem ls this upload
root@kali: ~
mysqldump -h localhost -u root test > mydump.sql
root@kali: ~
ls
Desktop Documents Downloads Music Pictures Public Templates Videos id_rsa.pem ls mydump.sql this upload
root@kali: ~
```

Step 3 : Create a database on cloud.

Login to your amazon AWS account, go to RDS service control panel and create a database instance with MariaDB.



Upon the creation of your database instance, click on your database and copy database endpoint.



Step 4 : Get your backup file on EC2 instance.

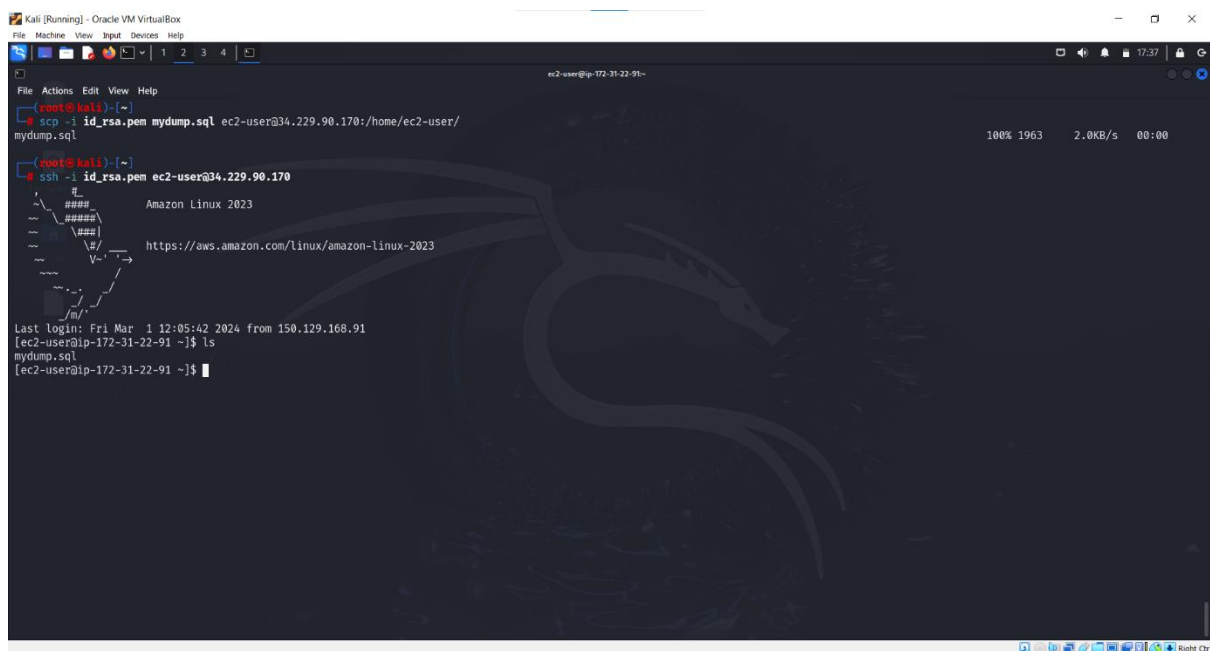
We cannot access our RDS instance from our local machine. In order to access it we need an EC2 instance. Go to EC2 service and launch an instance with MYSQL/Aurora inbound traffic allowed on it.

Next, open the terminal on your local machine and copy the backup file of your database into your EC2 instance using following command.

```
scp -i <key> <filename> ec2-user@<ip_address>:<destination>
```

SSH into your EC2 instance using following command.

```
ssh -i <key> ec2-user@<ip_address>
```

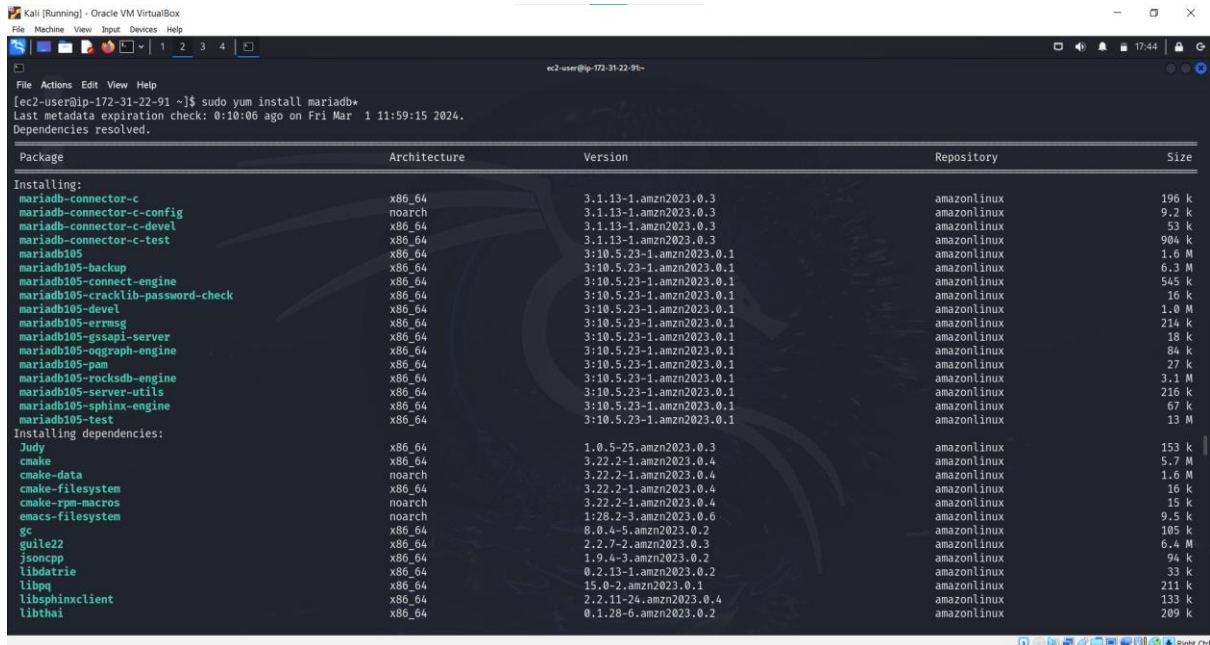


Step 5 : Migrate your database on Cloud.

We have already copied our database dump file from our local machine to our EC2 instance. Next step is to install MariaDB on our EC2 instance.

Use following command to install MariaDB

sudo apt-get install mariadb*



```
Kali [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
ec2-user@ip-172-31-22-91:~$ sudo yum install mariadb*
Last metadata expiration check: 0:10:06 ago on Fri Mar 1 11:59:15 2024.
Dependencies resolved.

Package Architecture Version Repository Size
Installing:
mariadb-connector-c x86_64 3.1.13-1.amzn2023.0.3 amazonlinux 196 k
mariadb-connector-c-config noarch 3.1.13-1.amzn2023.0.3 amazonlinux 9.2 k
mariadb-connector-c-devel x86_64 3.1.13-1.amzn2023.0.3 amazonlinux 53 k
mariadb-connector-c-test x86_64 3.1.13-1.amzn2023.0.3 amazonlinux 904 k
mariadb105 x86_64 3:10.5.23-1.amzn2023.0.1 amazonlinux 1.6 M
mariadb105-backup x86_64 3:10.5.23-1.amzn2023.0.1 amazonlinux 6.3 M
mariadb105-connect-engine x86_64 3:10.5.23-1.amzn2023.0.1 amazonlinux 545 k
mariadb105-cracklib-password-check x86_64 3:10.5.23-1.amzn2023.0.1 amazonlinux 16 k
mariadb105-devel x86_64 3:10.5.23-1.amzn2023.0.1 amazonlinux 1.0 M
mariadb105-errmsg x86_64 3:10.5.23-1.amzn2023.0.1 amazonlinux 214 k
mariadb105-gssapi-server x86_64 3:10.5.23-1.amzn2023.0.1 amazonlinux 18 k
mariadb105-oggraph-engine x86_64 3:10.5.23-1.amzn2023.0.1 amazonlinux 84 k
mariadb105-pam x86_64 3:10.5.23-1.amzn2023.0.1 amazonlinux 27 k
mariadb105-rocksdb-engine x86_64 3:10.5.23-1.amzn2023.0.1 amazonlinux 3.1 M
mariadb105-server-utils x86_64 3:10.5.23-1.amzn2023.0.1 amazonlinux 216 k
mariadb105-sphinx-engine x86_64 3:10.5.23-1.amzn2023.0.1 amazonlinux 67 k
mariadb105-test x86_64 3:10.5.23-1.amzn2023.0.1 amazonlinux 13 M
Installing dependencies:
Judy x86_64 1.0.5-25.amzn2023.0.3 amazonlinux 153 k
cmake x86_64 3.22.2-1.amzn2023.0.4 amazonlinux 5.7 M
cmake-data noarch 3.22.2-1.amzn2023.0.4 amazonlinux 1.6 M
cmake-filesystem x86_64 3.22.2-1.amzn2023.0.4 amazonlinux 16 k
cmake-rpm-macros noarch 3.22.2-1.amzn2023.0.4 amazonlinux 15 k
cmake-filesystem x86_64 3.22.2-1.amzn2023.0.4 amazonlinux 9.5 k
gc x86_64 8.0.4-5.amzn2023.0.2 amazonlinux 105 k
guile22 x86_64 2.2.7-2.amzn2023.0.3 amazonlinux 6.4 M
jsoncpp x86_64 1.9.4-3.amzn2023.0.2 amazonlinux 94 k
libdatrie x86_64 0.2.13-1.amzn2023.0.2 amazonlinux 33 k
libpq x86_64 15.0-2.amzn2023.0.1 amazonlinux 211 k
libsphinxclient x86_64 2.2.11-24.amzn2023.0.4 amazonlinux 133 k
libthai x86_64 0.1.28-6.amzn2023.0.2 amazonlinux 209 k
```

Once the MariaDB is installed run following command to login into our RDS database instance.

mysql -h <database_endpoint> -u admin -p

After logging into database instance create a database named migrate into that instance by running following query.

CREATE DATABASE migrate;

Following screen will appear after you run **SHOW DATABASES;** query.



```
[ec2-user@ip-172-31-22-91 ~]$ mysql -h testoncloud.cl6a2a2yo9x7.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 87
Server version: 10.11.6-MariaDB-log managed by https://aws.amazon.com/rds/

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| innodb |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.001 sec)

MariaDB [(none)]> CREATE DATABASE migrate;
Query OK, 1 row affected (0.003 sec)

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| innodb |
| migrate |
| mysql |
| performance_schema |
| sys |
+-----+
```

Exit from database instance and run following command to create table and populate it using 'mydump.sql' file.

mysql -h <database_endpoint> -u admin -p migrate <mydump.sql

Connect to your RDS database instance and run following queries to check whether migration is successful or not.

SHOW DATABASES; ➔ show databases on RDS instance

USE migrate; ➔ change to 'migrate' database

SHOW TABLES; ➔ show tables in database

SELECT * FROM <table_name>; ➔ show contents of table

Following output will be shown on the screen if migration is successful.


```
Kali [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
[ec2-user@ip-172-31-22-91 ~]$ mysql -h testoncloud.cl6a2a2yo9x7.us-east-1.rds.amazonaws.com -u admin -p migrate < mydump.sql
Enter password:
[ec2-user@ip-172-31-22-91 ~]$ mysql -h testoncloud.cl6a2a2yo9x7.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 98
Server version: 10.11.6-MariaDB-log managed by https://aws.amazon.com/rds/

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> use migrate;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [migrate]> show tables;
+-----+
| Tables_in_migrate |
+-----+
| cars               |
+-----+
1 row in set (0.001 sec)

MariaDB [migrate]> select * from cars;
+-----+-----+
| car_name | company |
+-----+-----+
| Nano    | Tata    |
| Maybach | Mercedes|
| Eco Sport | Ford    |
| City    | Honda   |
+-----+-----+
4 rows in set (0.001 sec)

MariaDB [migrate]> 
```