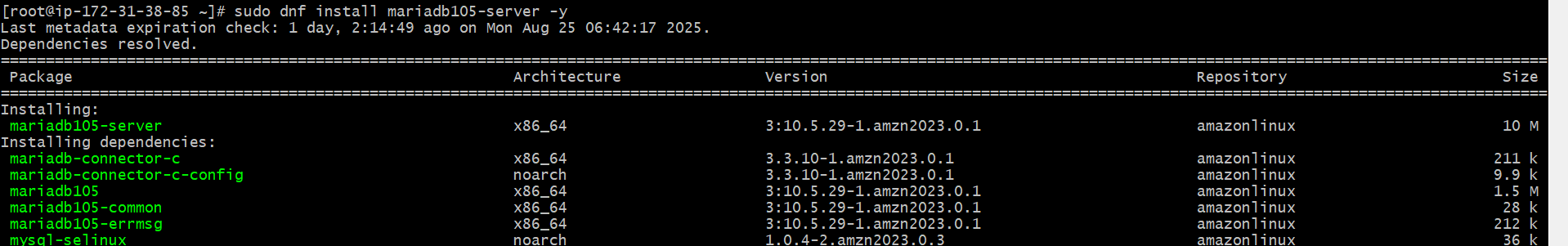
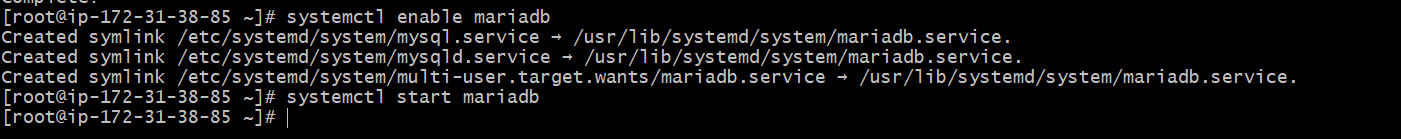
1. **Create mariadb db on ec2.**

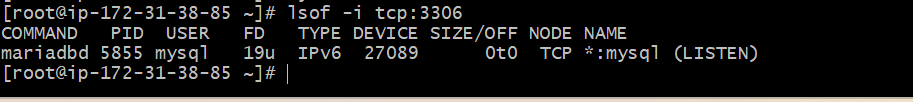
STEP:1  
Launch instance  
Go to AWS Console → EC2 → Launch instance

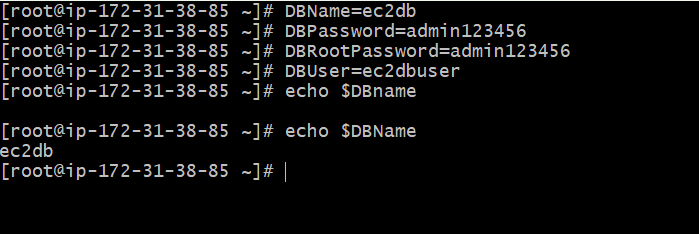
Install maria db

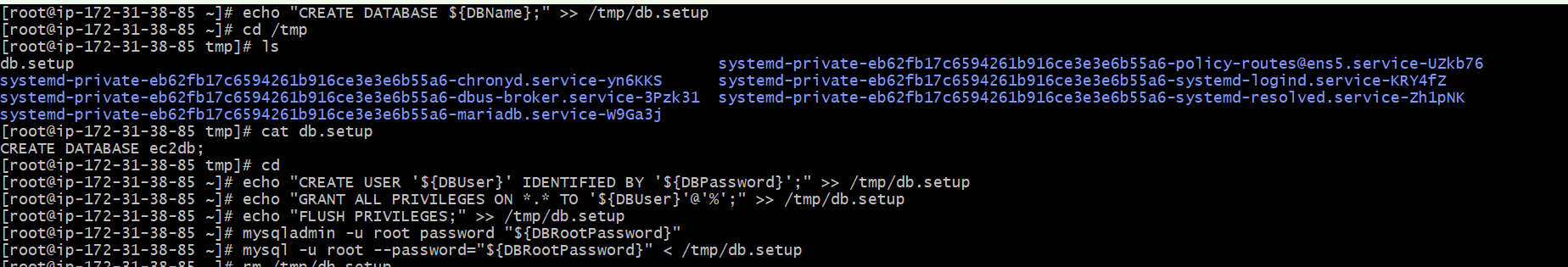
Enable and start

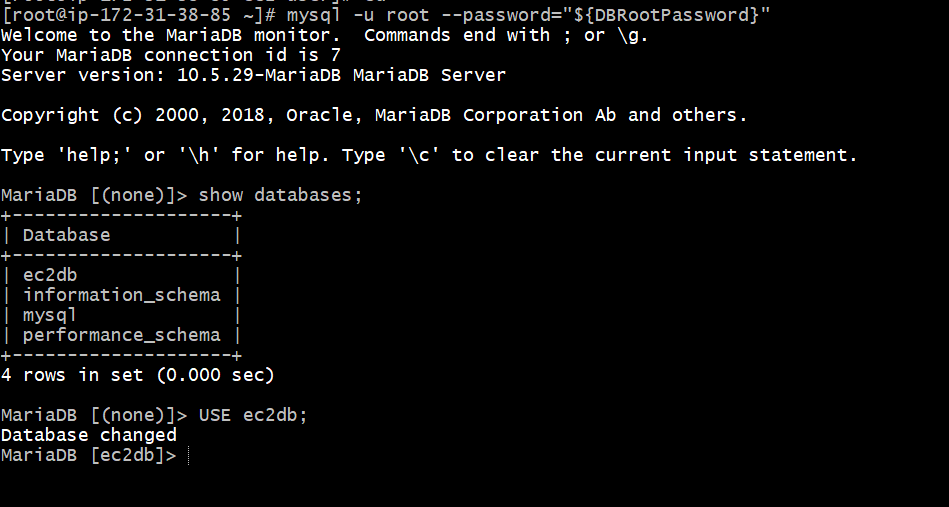






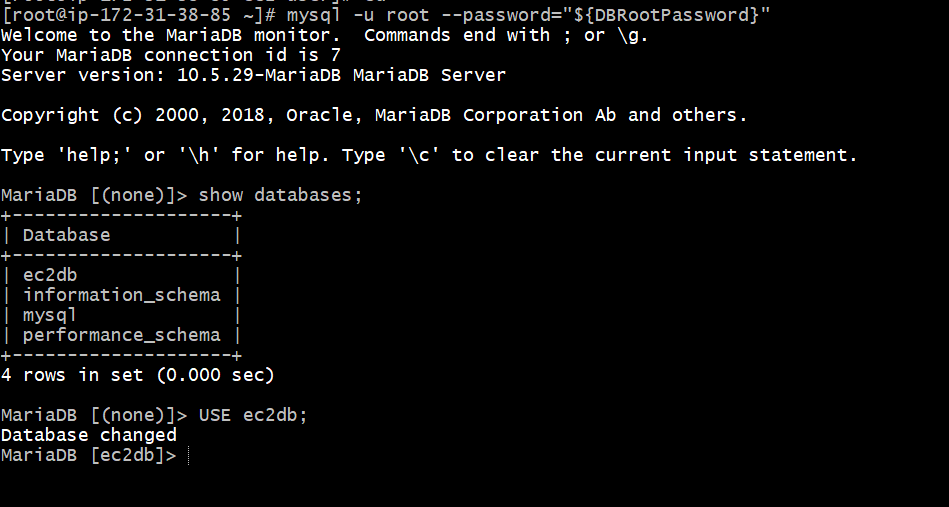


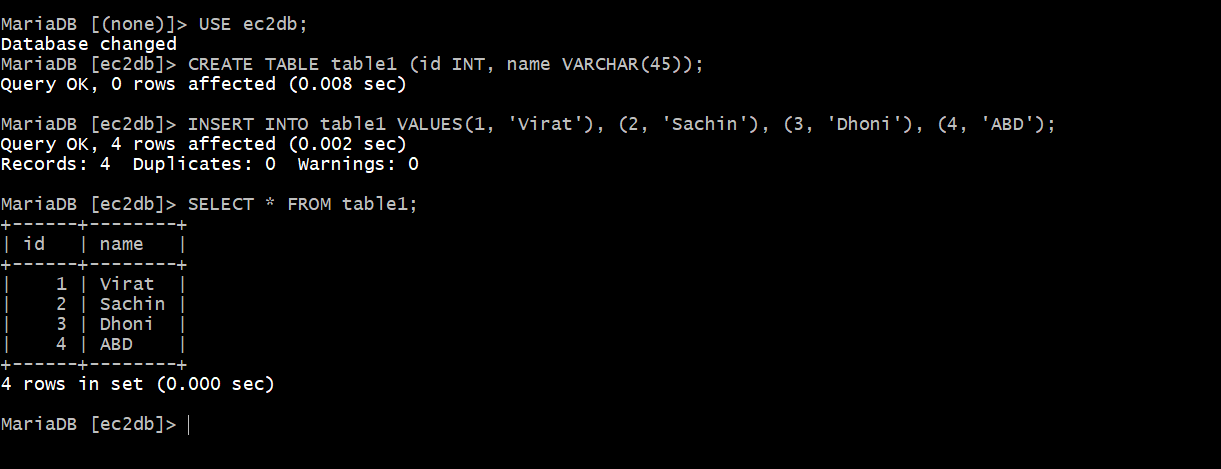




**2) Insert some dummy data**

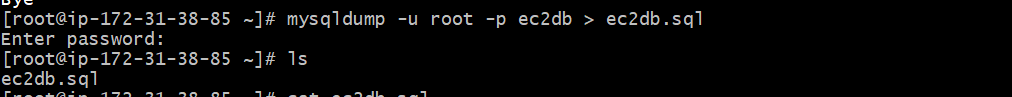
Inserting dummy data using below commands



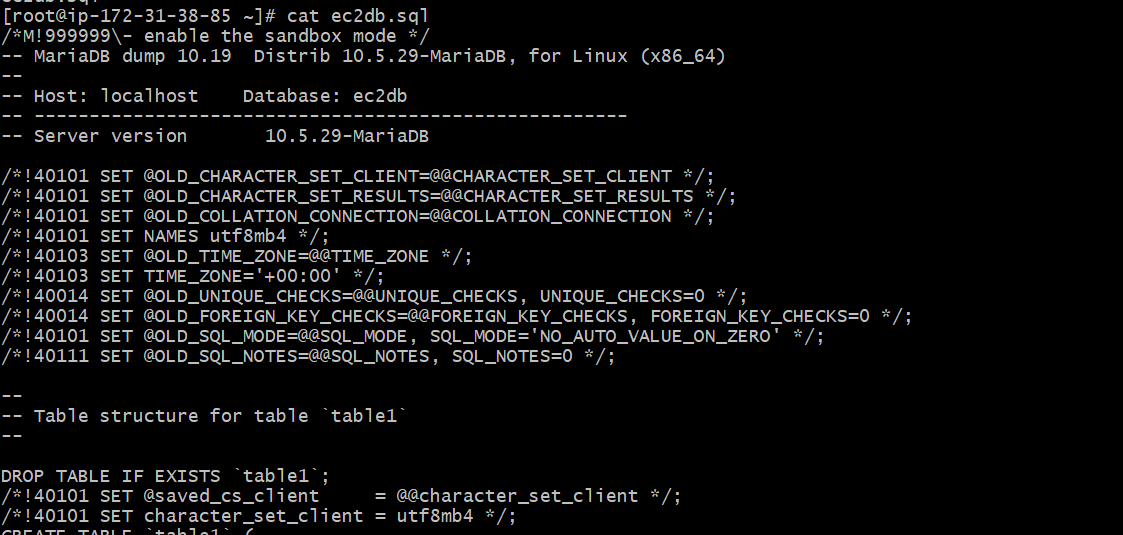


**3) Take the backup of dummy data on ec2**

Step-1  
Run the backup command:  
mysqldump -u root -p database\_name > file\_name.sql  
mysqldump -u root -p ec2db > ec2db.sql  
Enter password  
Checking file exist or not. :- ls

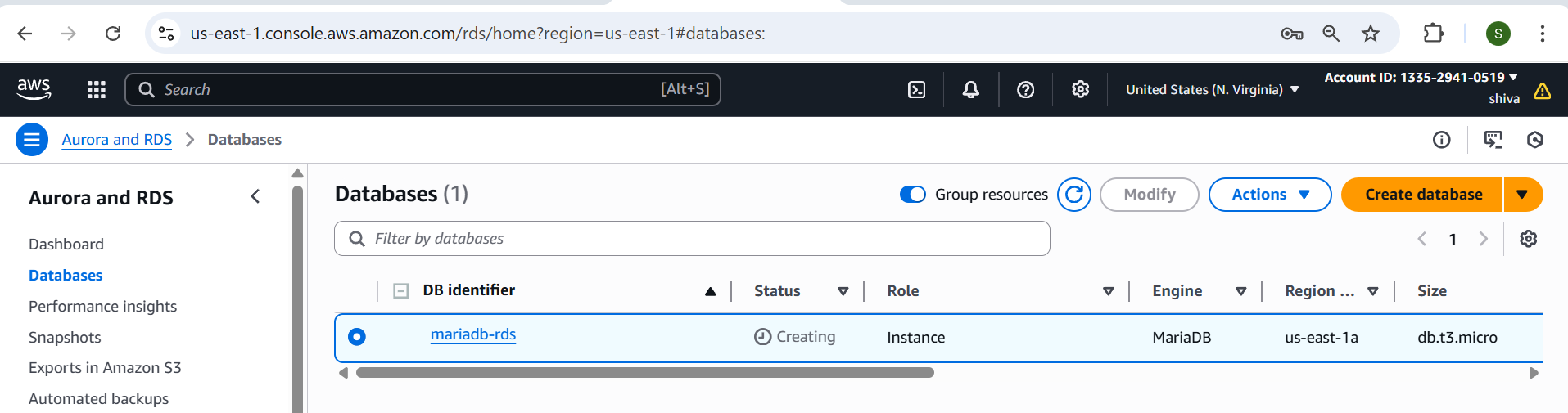
****

Check the backup file  
Cat ec2db.sq

****

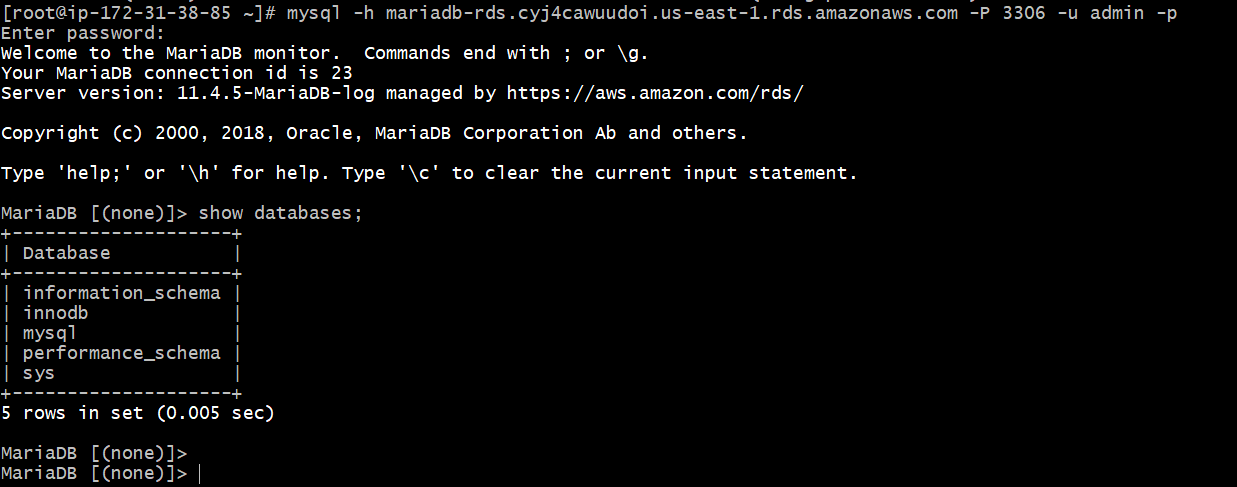
**4) launch MariaDB RDS instance.**

Go to AWS Console → RDS → Databases → Create database.  
Choose:  
Engine type: MariaDB  
Templates: Free tier (if eligible)  
Settings:  
DB instance identifier: mariadb-rds  
Master username: admin  
Master password: (set something strong, e.g.   
Admin123456!)  
Instance size:  
Choose db.t3.micro (free tier eligible)  
Connectivity:  
Choose your VPC (ideally the same one where your EC2   
lives).  
Enable Public Access = Yes (for testing).  
Select a VPC Security Group that allows inbound port 3306   
from your EC2 instance’s security group or IP.  
Click Create Database

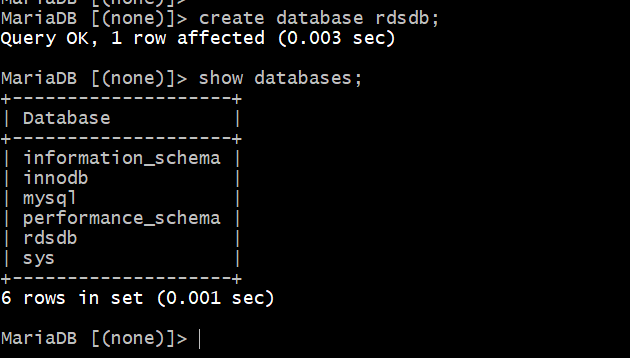
****

**5) Migrate database from ec2 to RDS.**

Step -1  
Connect to RDS and inspect current databases  
mysql -h <replace-rds-end-point-here> -P 3306 -u admin –p

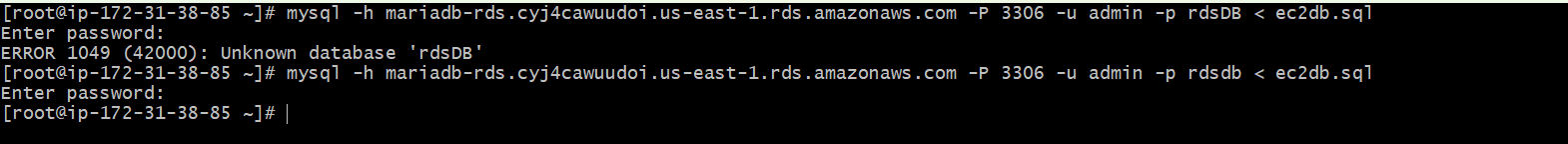


Step -2  
 Create the target DB on RDS  
CREATE DATABASE rdsdb;  
SHOW DATABASES;

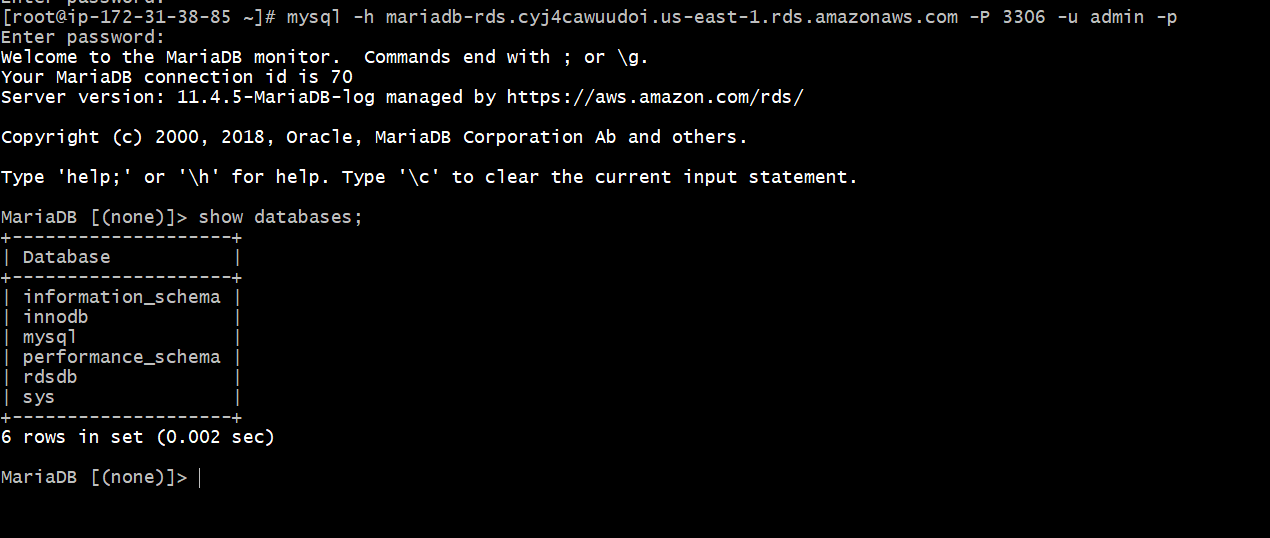
****

Step -3

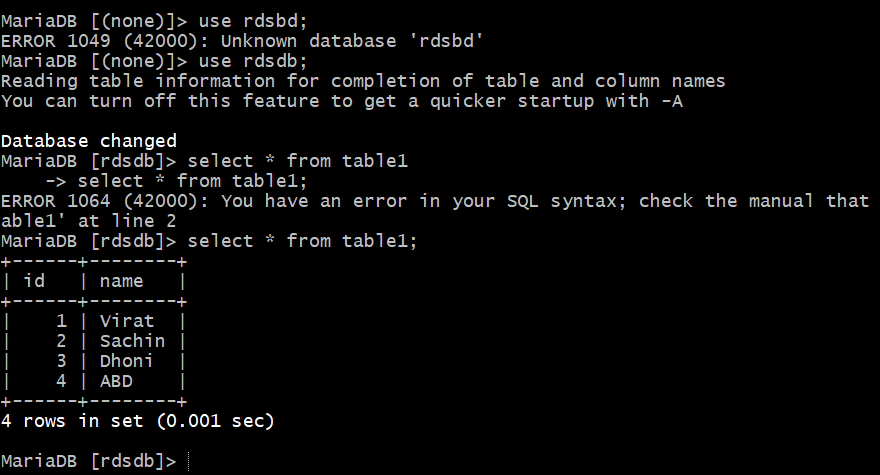
Migrate the DB dump that you have taken  
mysql -h <replace-rds-end-point-here> -P 3306 -u   
<user\_name> -p database\_name < ec2db.sql



Step -4  
Reconnect to RDS and verify data  
mysql -h <replace-rds-end-point-here> -P 3306 -u rdsuser –p



Step -5  
USE rdsdb;  
SHOW TABLES;  
SELECT \* FROM table1;



**6) Install MySQL DB on ec2**

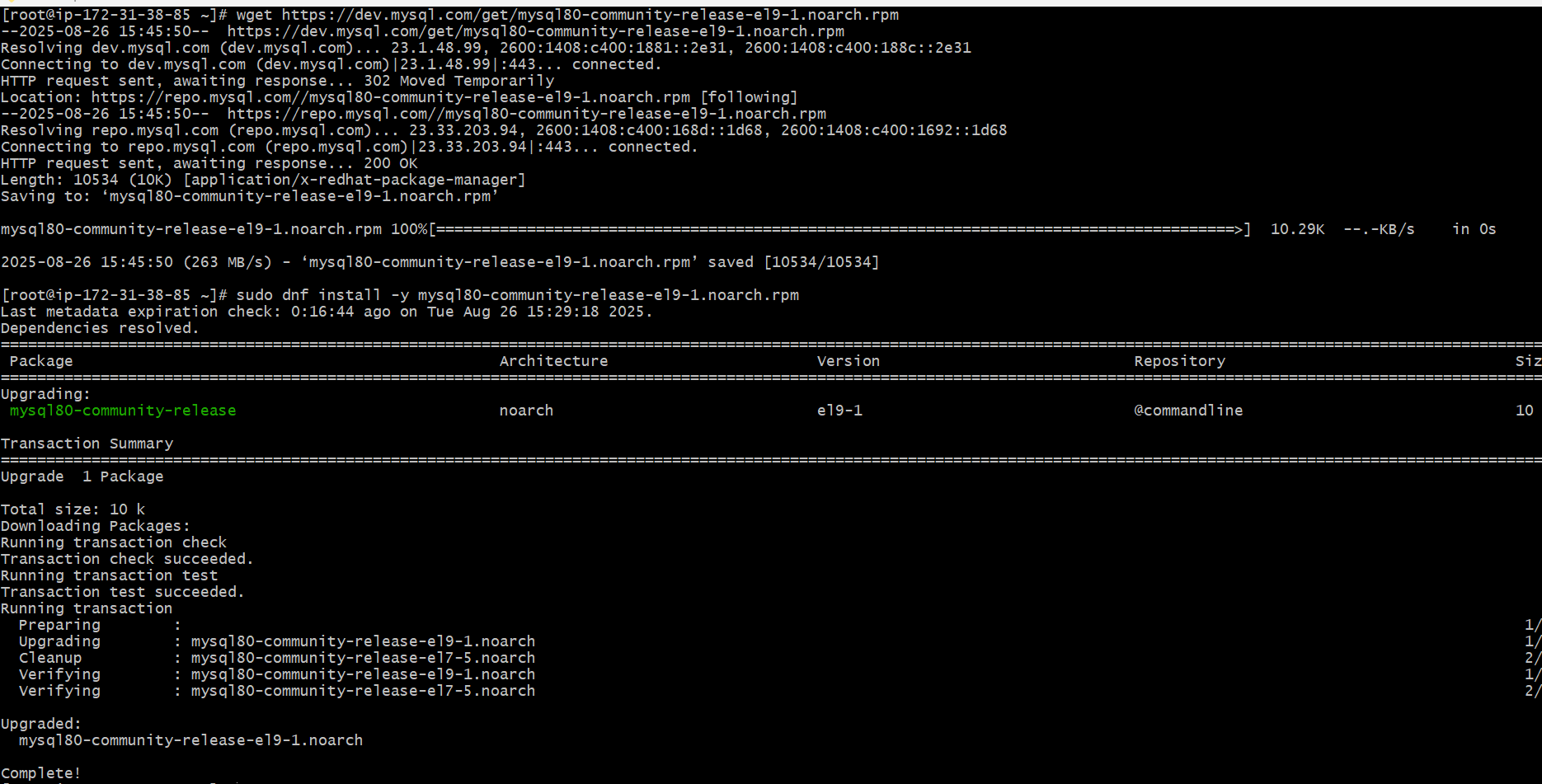
Import the correct new MySQL GPG key

sudo rpm --import https://repo.mysql.com/RPM-GPG-KEY-mysql-2023

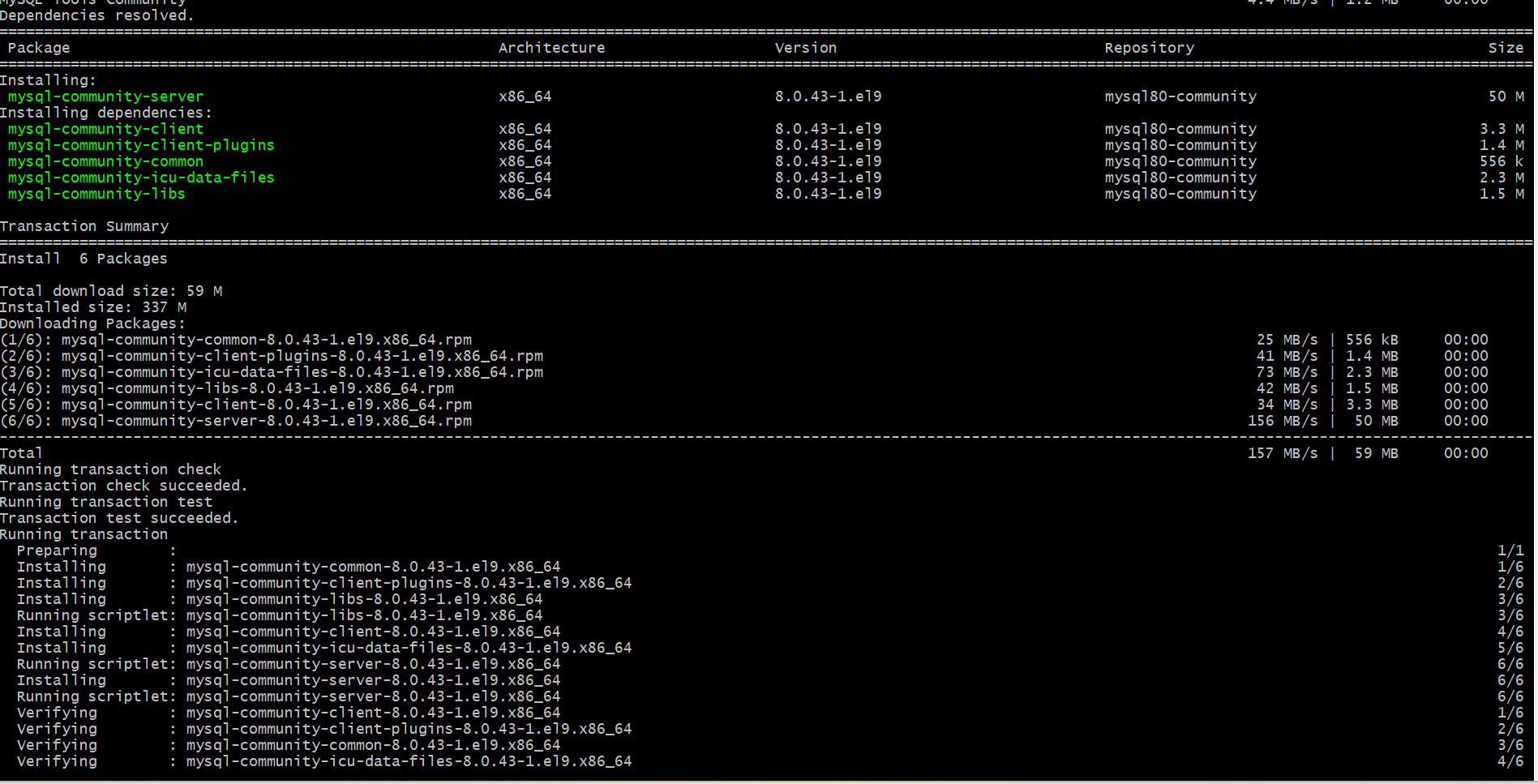
install the official repo

wget https://dev.mysql.com/get/mysql80-community-release-el9-1.noarch.rpm

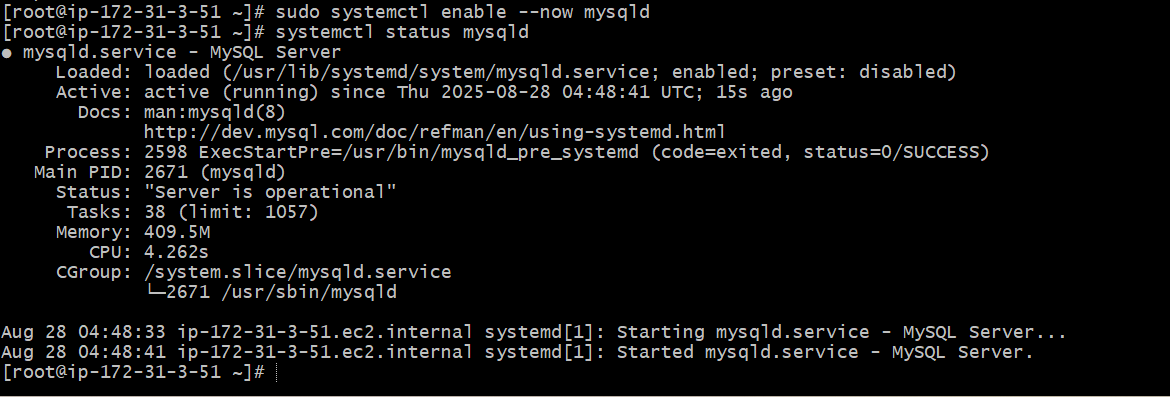
sudo dnf install -y mysql80-community-release-el9-1.noarch.rpm



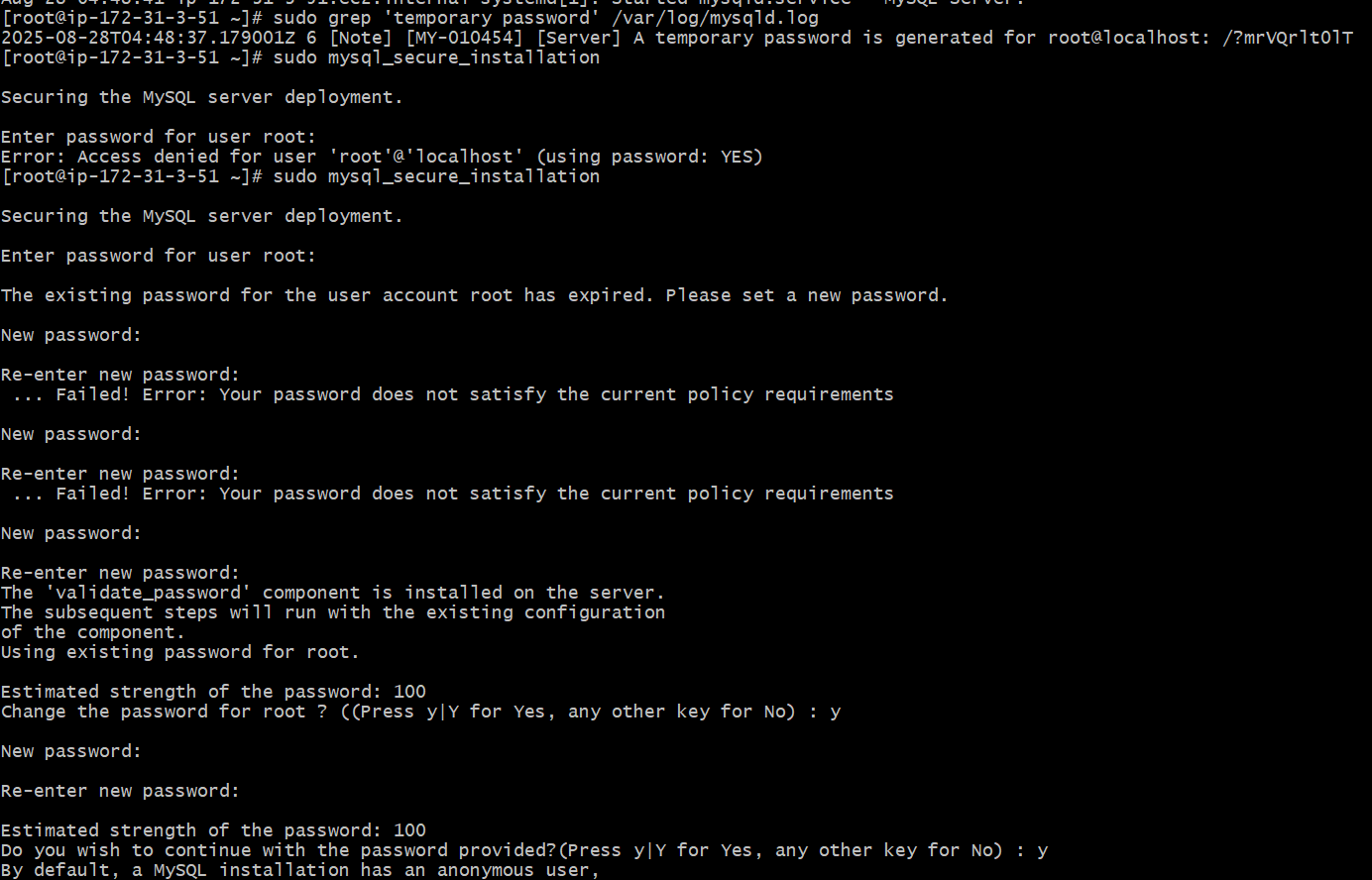
Install MySQL server  
sudo dnf install -y mysql-community-server



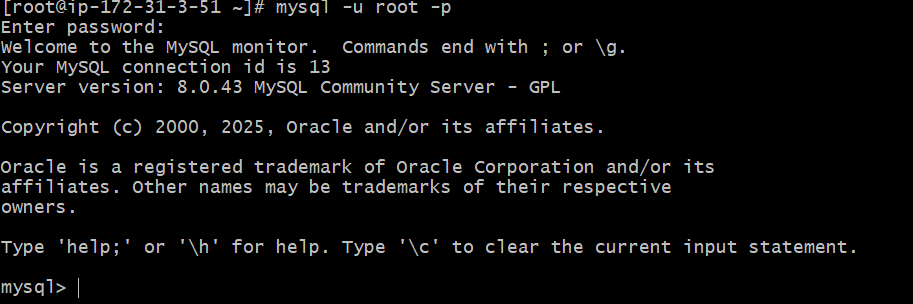
Start & enable MySQL  
sudo systemctl enable --now mysqld  
systemctl status mysqld



Get root temporary password  
sudo grep 'temporary password' /var/log/mysqld.log  
Secure installation  
sudo mysql\_secure\_installation

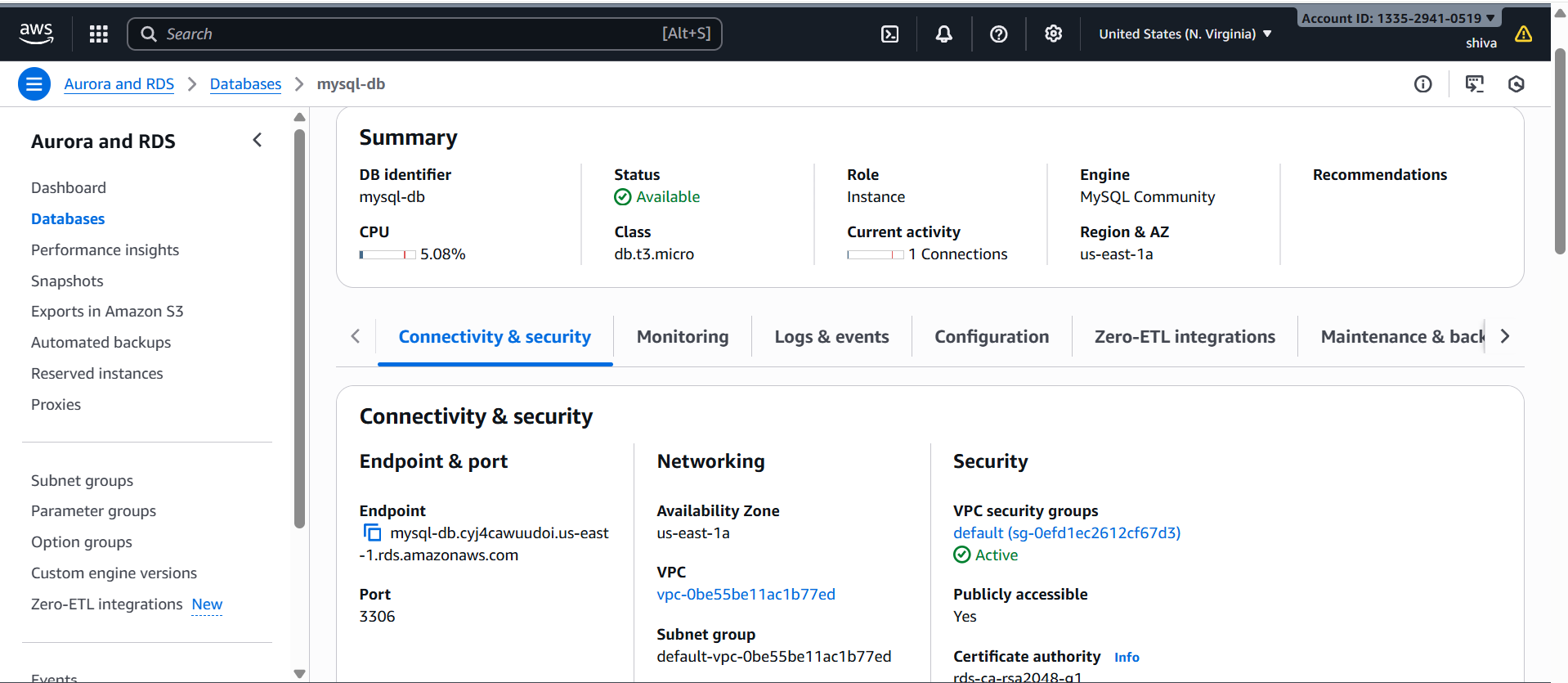


Login to mysql --mysql -u root -p



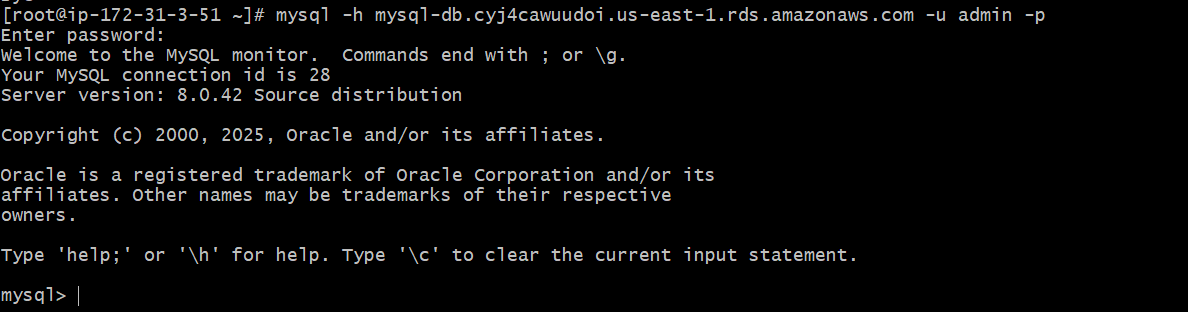
**7) Launch MySQL RDS image**

Sign in to AWS Console  
Go to → Services → RDS → Databases → Create database.  
Choose Database Creation Method  
•Standard create (gives you full control)  
•Or Easy create (simpler, but fewer options)  
Select Standard create for more flexibility.  
 Engine Options  
•Engine type → Select MySQL  
•Version → Choose a stable version (e.g., MySQL 8.0.x LTS)  
Templates  
Choose based on use case:  
Relational Data Base Task  
•Production (Multi-AZ, backups enabled, more costly)  
•Dev/Test (Single-AZ, cheaper)  
Settings  
•DB instance identifier → e.g., mydb-mysql  
•Master username → e.g., admin  
•Master password → choose a strong password (or auto-generate and  
download)  
 Instance Configuration  
•Instance type → e.g., db.t3.micro (free-tier eligible) or db.t3.medium for   
small  
workloads  
•Storage → General Purpose (gp3), start with 20GB (auto-scaling optional)  
Connectivity  
•VPC → Choose your VPC (default or custom)  
•Subnet group → Usually default unless you made custom subnets  
•Public access →  
-Yes if you want to connect from outside AWS (not recommended for  
production)  
-No if only accessed inside VPC (best practice)  
VPC security group → Create/attach one allowing TCP 3306 from your IP   
or app servers  
 Additional Config  
•Initial database name → e.g., appdb  
•Automatic backups → Enable (set retention, e.g., 7 days)  
•Encryption → Enable if required  
•Monitoring → Enable Enhanced monitoring if needed  
Create Database  
Click Create database → AWS will provision your RDS instance (may take 5–  
10  
minutes).



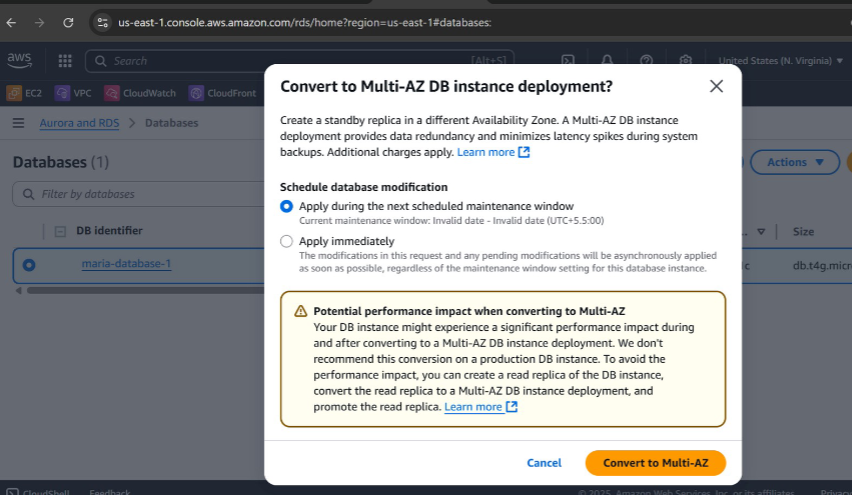
Connect to RDS  
Once status = Available:  
Copy the Endpoint

Connect from EC2 or laptop:  
mysql -h mysql-db.cyj4cawuudoi.us-east-1.rds.amazonaws.com -u admin –p



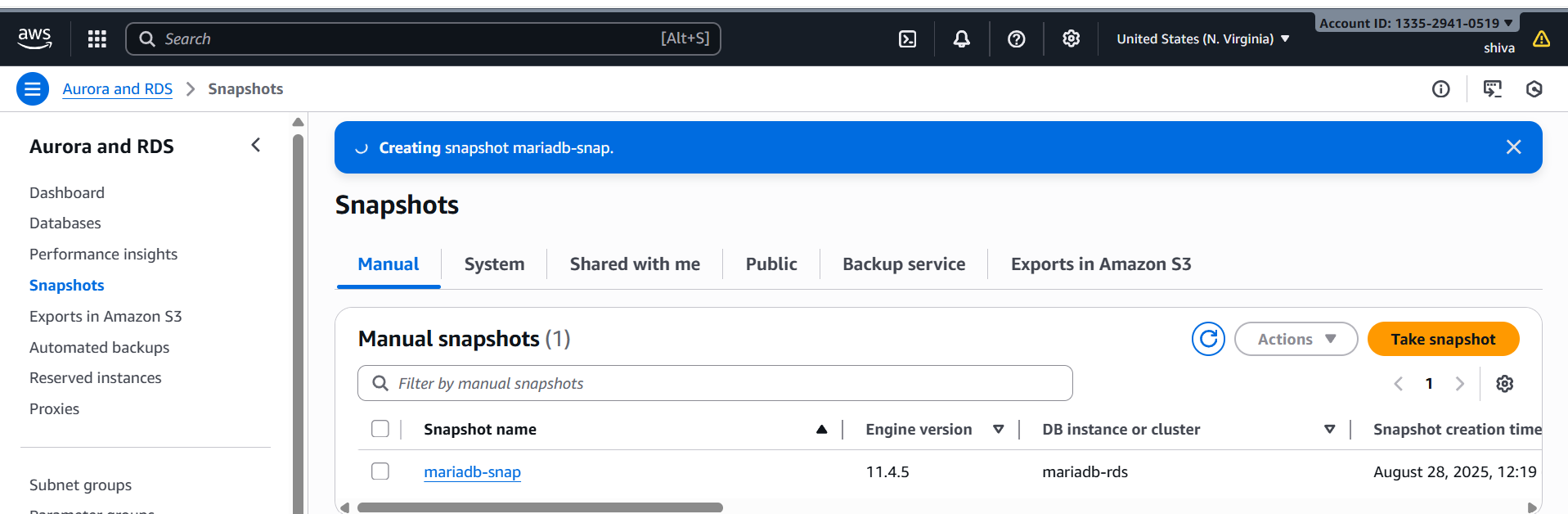
**8) Configure multi AZ**

In AWS Console → RDS → Select your DB → Modify or (we can go from   
action → convert to Multi-AZ)  
Under Availability & durability, enable Multi-AZ deployment  
Apply immediately

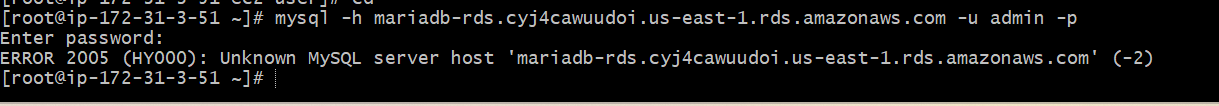
****

**9) Take Backup of DB and restore the DB**

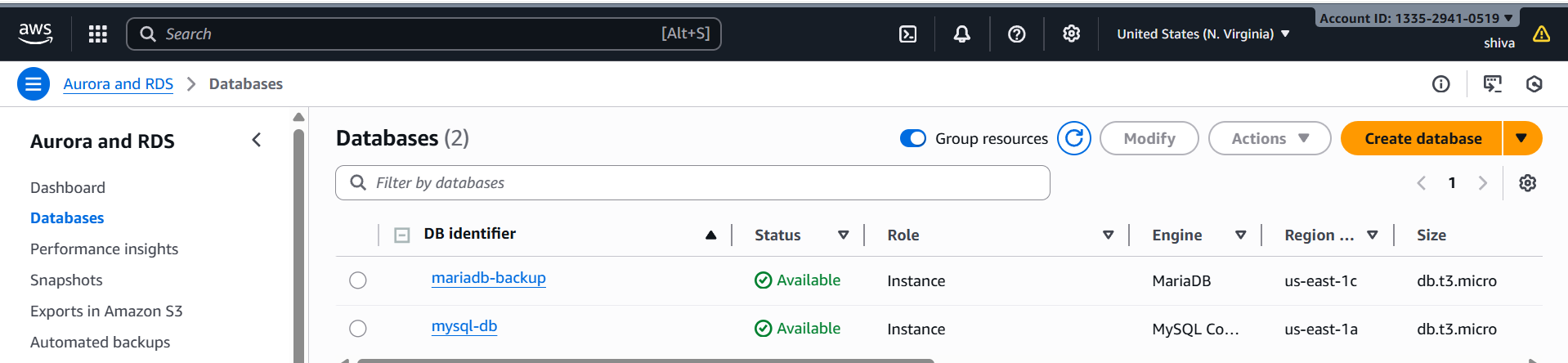
Step 1: Create a Manual Snapshot  
1. Go to AWS Console → RDS.  
2. In the left panel, click Databases.  
3. Select your RDS Maria instance.  
4. On the top right, click Actions → Take snapshot.  
5. Enter a Snapshot name.  
6.Click Take snapshot

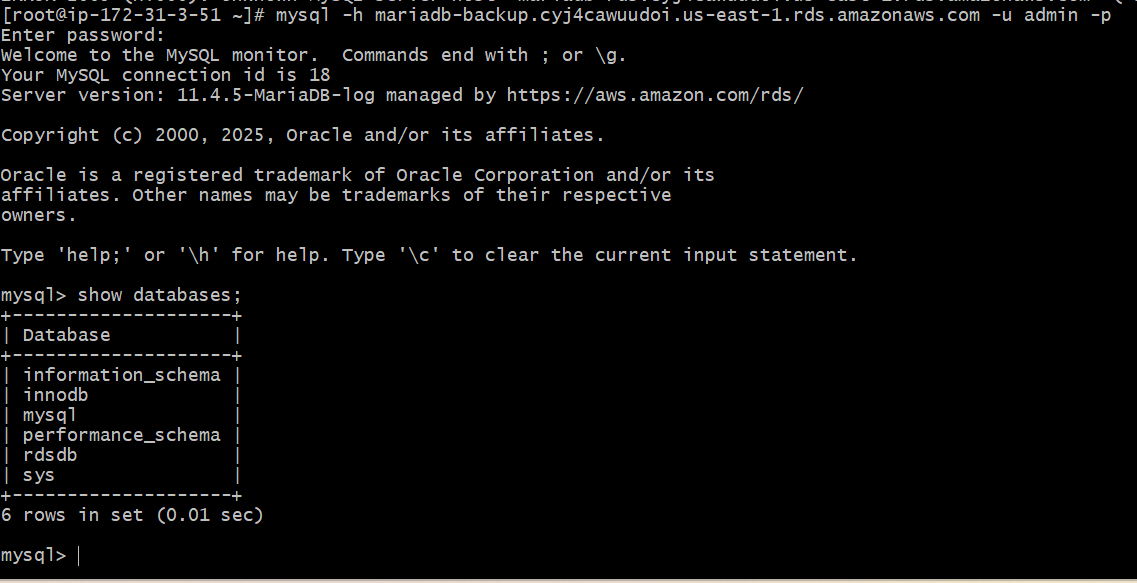
****

Delete the Maria –db



•Step 2: Restore Database from Snapshot  
•Go to RDS → Snapshots.  
•Select the snapshot you created.  
•Click Actions → Restore snapshot.

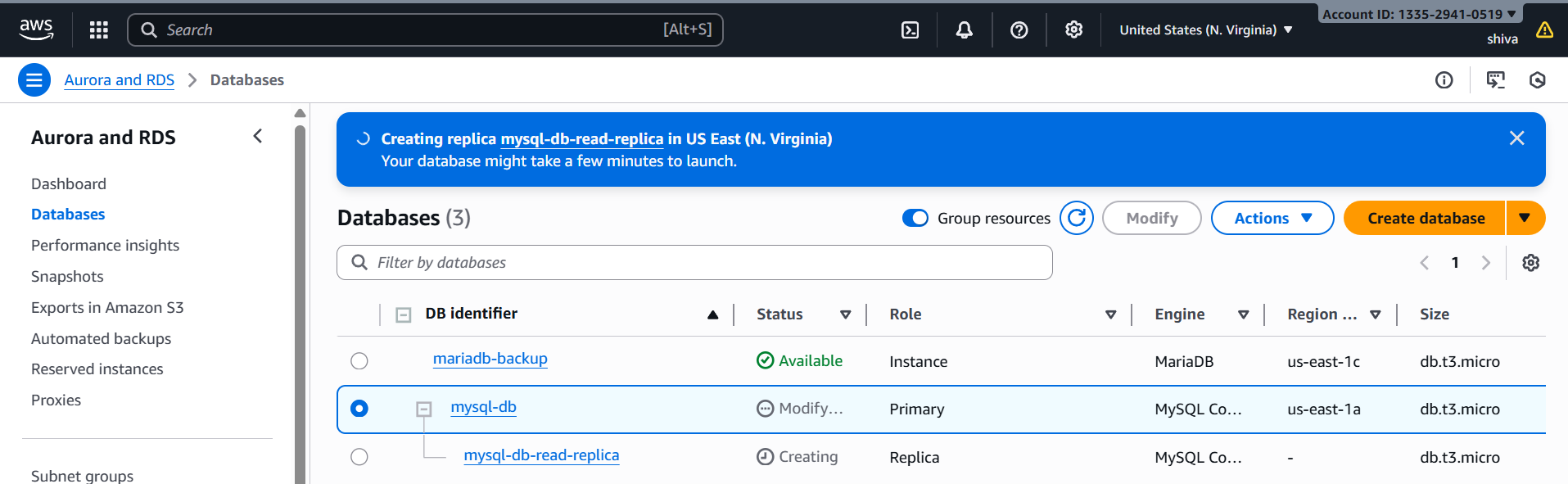
****

****

**10) Create Read Replica**

On the DB instance page,  
click on the database  
Actions → Create read replica.  
Configure the read replica:  
DB instance identifier: e.g., mysql-db-read-replica.  
DB instance class: Choose based on your workload.  
VPC, Subnet, Security Group: Must match or allow access from your   
app.  
Storage type: Same as primary (recommended).  
Enable replication features: leave defaults unless specific use case.

Click Create read replica

****