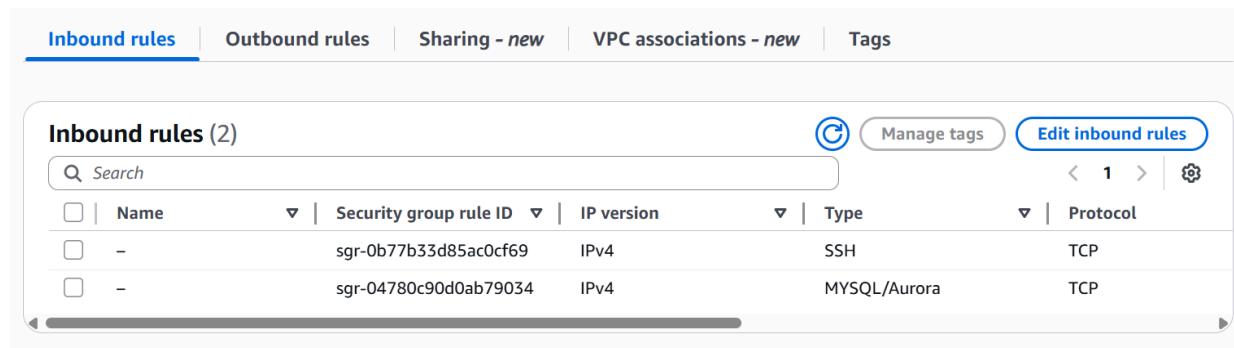


Database Project-2

Project	Description
MySQL / MariaDB Setup	 Install & configure MySQL or MariaDB, create database, manage users, and secure it (disable remote root access, strong passwords), automate regular backups using Shell Scripting & Crontab.

Step 1 . Launch an EC2 Instance

- **AMI:** Ubuntu Server 22.04 or above
- **Type:** t2.micro or above
- **Storage:** 8–10 GB
- **Security Group (Inbound Rules):**
 - SSH → 22
 - *Optional* (only if needed): MySQL/MariaDB → 3306



The screenshot shows the AWS Management Console interface for managing security group inbound rules. The top navigation bar includes tabs for Inbound rules, Outbound rules, Sharing - new, VPC associations - new, and Tags. The Inbound rules tab is selected, indicated by a blue underline. Below the tabs, there is a search bar labeled "Search". A table displays the current inbound rules, with two entries visible:

Name	Security group rule ID	IP version	Type	Protocol
-	sgr-0b77b33d85ac0cf69	IPv4	SSH	TCP
-	sgr-04780c90d0ab79034	IPv4	MySQL/Aurora	TCP

```
root@ip-172-31-46-45: ~
root@ip-172-31-46-45:~# cat /etc/os-release
PRETTY_NAME="Ubuntu 22.04.5 LTS"
NAME="Ubuntu"
VERSION_ID="22.04"
VERSION="22.04.5 LTS (Jammy Jellyfish)"
VERSION_CODENAME=jammy
ID=ubuntu
ID_LIKE=debian
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
UBUNTU_CODENAME=jammy
root@ip-172-31-46-45:~#
```

Step 2 . Connect to Your EC2 Instance

```
ssh -i my-key.pem ubuntu@EC2_PUBLIC_IP
```

```
Command Prompt
Microsoft Windows [Version 10.0.22631.6199]
(c) Microsoft Corporation. All rights reserved.

C:\Users\User>cd Downloads

C:\Users\User\Downloads>ssh -i my-key.pem ubuntu@13.201.31.241
```

Step 3 . Install MariaDB Server

```
apt update && sudo apt upgrade -y
```

```
sudo apt install mariadb-server -y
```

```
Check version: mariadb --version
```

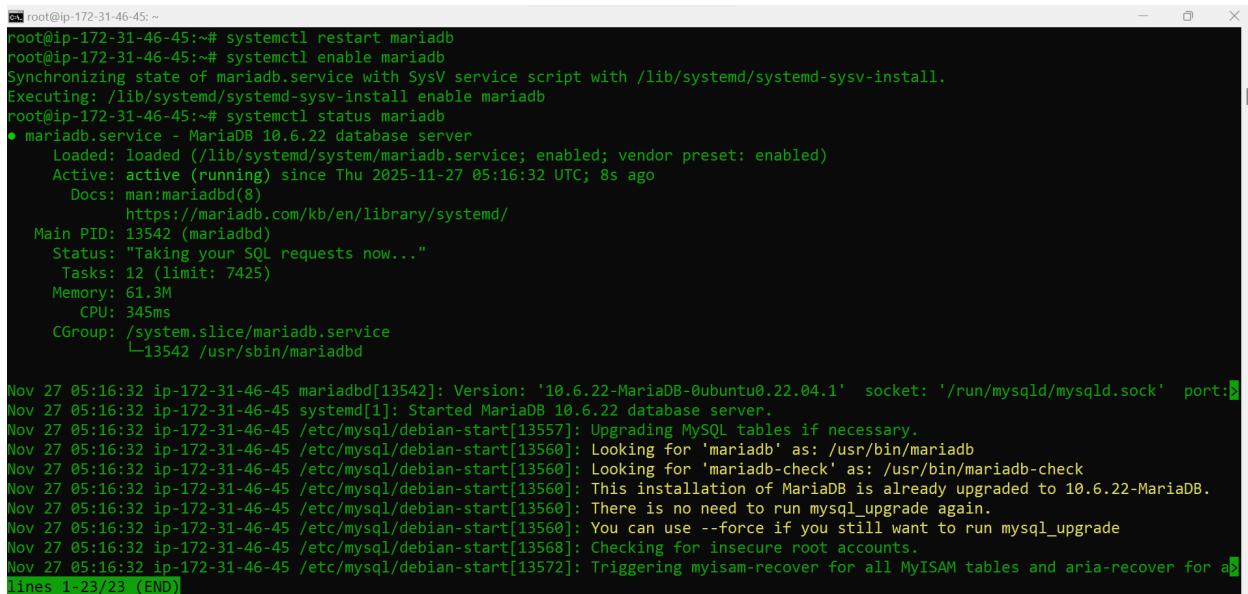
```
root@ip-172-31-46-45: ~
root@ip-172-31-46-45:~# mariadb --version
mariadb Ver 15.1 Distrib 10.6.22-MariaDB, for debian-linux-gnu (x86_64) using EditLine wrapper
root@ip-172-31-46-45:~#
```

Step 4 . Start & Enable MariaDB

```
systemctl start mariadb
```

```
systemctl enable mariadb
```

```
systemctl status mariadb
```



```
root@ip-172-31-46-45:~# systemctl restart mariadb
root@ip-172-31-46-45:~# systemctl enable mariadb
Synchronizing state of mariadb.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable mariadb
root@ip-172-31-46-45:~# systemctl status mariadb
● mariadb.service - MariaDB 10.6.22 database server
   Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2025-11-27 05:16:32 UTC; 8s ago
     Docs: man:mariadb(8)
           https://mariadb.com/kb/en/library/systemd/
 Main PID: 13542 (mariadb)
   Status: "Taking your SQL requests now..."
    Tasks: 12 (limit: 7425)
   Memory: 61.3M
      CPU: 345ms
     CGroup: /system.slice/mariadb.service
             └─13542 /usr/sbin/mariadb

Nov 27 05:16:32 ip-172-31-46-45 mariadb[13542]: Version: '10.6.22-MariaDB-0ubuntu0.22.04.1' socket: '/run/mysqld/mysqld.sock' port: 3306
Nov 27 05:16:32 ip-172-31-46-45 systemd[1]: Started MariaDB 10.6.22 database server.
Nov 27 05:16:32 ip-172-31-46-45 /etc/mysql/debian-start[13557]: Upgrading MySQL tables if necessary.
Nov 27 05:16:32 ip-172-31-46-45 /etc/mysql/debian-start[13560]: Looking for 'mariadb' as: /usr/bin/mariadb
Nov 27 05:16:32 ip-172-31-46-45 /etc/mysql/debian-start[13560]: Looking for 'mariadb-check' as: /usr/bin/mariadb-check
Nov 27 05:16:32 ip-172-31-46-45 /etc/mysql/debian-start[13560]: This installation of MariaDB is already upgraded to 10.6.22-MariaDB.
Nov 27 05:16:32 ip-172-31-46-45 /etc/mysql/debian-start[13560]: There is no need to run mysql_upgrade again.
Nov 27 05:16:32 ip-172-31-46-45 /etc/mysql/debian-start[13560]: You can use --force if you still want to run mysql_upgrade.
Nov 27 05:16:32 ip-172-31-46-45 /etc/mysql/debian-start[13568]: Checking for insecure root accounts.
Nov 27 05:16:32 ip-172-31-46-45 /etc/mysql/debian-start[13572]: Triggering myisam-recover for all MyISAM tables and aria-recover for ap
lines 1-23/23 (END)
```

Step 5 . Secure the MariaDB Installation

```
Mysql_secure_installation
```

Step 6 . Login to MariaDB & Create Database & User (for portfolio)

```
mysql -u root -p
```

Inside MariaDB shell:

Create database

```
Show databases;
```

```
create database shabbir_db1;
```

```
root@ip-172-31-46-45: ~
MariaDB [(none)]> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| mysql          |
| performance_schema |
| sys            |
+-----+
4 rows in set (0.000 sec)

MariaDB [(none)]> create database shabbir_db1;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| mysql          |
| performance_schema |
| shabbir_db1    |
| sys            |
+-----+
5 rows in set (0.000 sec)

MariaDB [(none)]>
```

Create a user

```
create user shabbir@localhost identified by 'password@123';
```

```
root@ip-172-31-46-45: ~
MariaDB [(none)]> create user shabbir@localhost identified by 'password@123';
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]>
```

Grant permissions for Shabbir to shabbir_db1

```
GRANT ALL PRIVILEGES ON shabbir_db1.* TO 'shabbir'@'localhost';
```

```
[root@ip-172-31-46-45: ~] MariaDB [(none)]> GRANT ALL PRIVILEGES ON shabbir_db1.* TO 'shabbir'@'localhost';
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]>
```

```
EXIT;
```

Re-login with Shabbir user & Show database

```
[root@ip-172-31-46-45: ~] MariaDB [(none)]> GRANT ALL PRIVILEGES ON shabbir_db1.* TO 'shabbir'@'localhost';
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> exit ;
Bye
root@ip-172-31-46-45:~# mariadb -u shabbir -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 38
Server version: 10.6.22-MariaDB-0ubuntu0.22.04.1 Ubuntu 22.04

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 |
| shabbir_db1    |
+-----+
2 rows in set (0.000 sec)

MariaDB [(none)]>
```

Step 7 . Create New DB,use/go-inside new DB ,Create table,show tables,SELECT * FROM employees;

```
root@ip-172-31-46-45: ~
MariaDB [(none)]> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| mysql          |
| performance_schema |
| shabbir_db1    |
| sys            |
+-----+
5 rows in set (0.000 sec)

MariaDB [(none)]> create database shabbir_db2;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| mysql          |
| performance_schema |
| shabbir_db1    |
| shabbir_db2    |
| sys            |
+-----+
6 rows in set (0.000 sec)

MariaDB [(none)]>
```

```

root@ip-172-31-46-45: ~
MariaDB [(none)]> use shabbir_db2
Database changed
MariaDB [shabbir_db2]> show tables;
Empty set (0.000 sec)

MariaDB [shabbir_db2]> CREATE TABLE employees (
    ->     id INT AUTO_INCREMENT PRIMARY KEY,
    ->     name VARCHAR(50) NOT NULL,
    ->     email VARCHAR(100) UNIQUE,
    ->     department VARCHAR(50),
    ->     salary DECIMAL(10,2),
    ->     created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
    -> );
Query OK, 0 rows affected (0.012 sec)

MariaDB [shabbir_db2]> show tables;
+-----+
| Tables_in_shabbir_db2 |
+-----+
| employees             |
+-----+
1 row in set (0.000 sec)

MariaDB [shabbir_db2]> INSERT INTO employees(name, email, department, salary)
-> VALUES
-> ('Shabbir Ahmad','shabbir@example.com','IT',50000),
-> ('Rahul Kumar','rahul@gmail.com','DevOps',65000);
Query OK, 2 rows affected (0.002 sec)
Records: 2  Duplicates: 0  Warnings: 0

MariaDB [shabbir_db2]>

```

`SELECT * FROM employees;`

```

MariaDB [shabbir_db2]> SELECT * FROM employees;
+----+-----+-----+-----+-----+-----+
| id | name      | email           | department | salary   | created_at      |
+----+-----+-----+-----+-----+-----+
|  1 | Shabbir Ahmad | shabbir@example.com | IT          | 50000.00 | 2025-11-27 06:03:31 |
|  2 | Rahul Kumar  | rahul@gmail.com   | DevOps      | 65000.00 | 2025-11-27 06:03:31 |
+----+-----+-----+-----+-----+-----+
2 rows in set (0.000 sec)

MariaDB [shabbir_db2]>

```

Step 8. Automate Daily Backups (Shell Script + Cron)

A- First take Manual Backup/Dump

```

root@ip-172-31-46-45: ~
root@ip-172-31-46-45:~# mariadb-dump -u root -p=root@123 shabbir_db2 > /mnt/manual_dump
root@ip-172-31-46-45:~# ls /mnt/
manual_dump
root@ip-172-31-46-45:~#

```

```
root@ip-172-31-46-45: ~
MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| shabbir_db1 |
| shabbir_db2 |
| sys |
+-----+
6 rows in set (0.000 sec)

MariaDB [(none)]> drop database shabbir_db2;
Query OK, 1 row affected (0.014 sec)

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| shabbir_db1 |
| sys |
+-----+
5 rows in set (0.000 sec)

MariaDB [(none)]>
```

Restore Database (If Required)

```
mariadb -u root -p=root@123 shabbir_db1 < /mnt/manual-dump
```

```
root@ip-172-31-46-45: ~
root@ip-172-31-46-45:~# mariadb -u root -p=root@123 shabbir_db1 < /mnt/manual_dump
```

B- Create backup script:

```
vim /root/auto-backup.sh
```

Add:

```
#!/bin/bash
```

```
mariadb-dump -u root -p=root@123 shabbir_db1 > /home/auto_backup
```

Make executable:

```
chmod 755 /root/auto_backup.sh
```

```
[root@ip-172-31-46-45: ~]# cat auto_backup.sh
#!/bin/bash
mariadb-dump -u root -p=root@123 shabbir_db1 > /home/auto_backup
[root@ip-172-31-46-45: ~]# ls /home/
auto_backup  ubuntu
[root@ip-172-31-46-45: ~]#
```

Step 9 . ⏳ Schedule Automatic Backup (Crontab)

Open cron:

```
crontab -e
```

Add :

```
*/2 * * * * /root/auto_backup.sh
```

Step 10 .🌐 Allow Remote Access (Not recommended for production/ Optional)

Edit MariaDB config:

```
vim /etc/mysql/mariadb.conf.d/50-server.cnf
```

Change :

```
bind-address = 0.0.0.0
```

```
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address            = 0.0.0.0
```

Restart:

```
sudo systemctl restart mariadb
```

Update EC2 Inbound Rule:

Port **3306**

Source: **My IP** (never 0.0.0.0/0)

```
# mariadb -h pub-IP -u root -p=root@123 Check-access from Remote
```



DONE – MariaDB Server on Ubuntu EC2 is Ready Thank you !!

