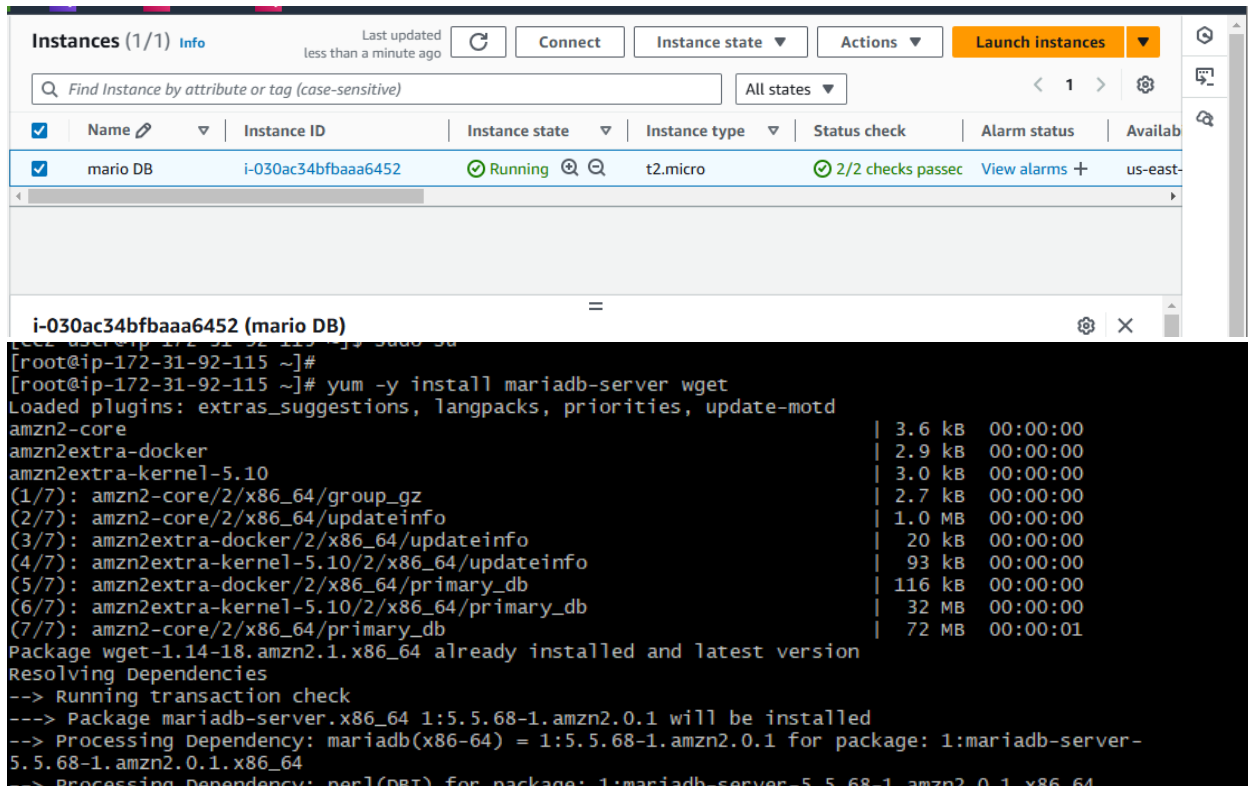


RDS

1) Create mariadb db on ec2.



The screenshot displays the AWS Management Console interface for EC2 instances. At the top, there's a header for 'Instances (1/1)' with an 'Info' link and a 'Last updated' timestamp. Below this is a search bar and a table of instances. The table has columns for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, and Availability zone. One instance, 'mario DB', is listed with ID 'i-030ac34bfbaaa6452', state 'Running', type 't2.micro', and status '2/2 checks passed'. Below the table, a terminal window for the instance 'i-030ac34bfbaaa6452 (mario DB)' shows the execution of the command 'yum -y install mariadb-server wget'. The terminal output shows the installation progress for various packages, including 'amzn2-core', 'amzn2extra-docker', and 'amzn2extra-kernel-5.10', and the successful installation of 'mariadb-server'.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability zone
mario DB	i-030ac34bfbaaa6452	Running	t2.micro	2/2 checks passed	View alarms	us-east-

```
[root@ip-172-31-92-115 ~]# yum -y install mariadb-server wget
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core | 3.6 kB | 00:00:00
amzn2extra-docker | 2.9 kB | 00:00:00
amzn2extra-kernel-5.10 | 3.0 kB | 00:00:00
(1/7): amzn2-core/2/x86_64/group_gz | 2.7 kB | 00:00:00
(2/7): amzn2-core/2/x86_64/updateinfo | 1.0 MB | 00:00:00
(3/7): amzn2extra-docker/2/x86_64/updateinfo | 20 kB | 00:00:00
(4/7): amzn2extra-kernel-5.10/2/x86_64/updateinfo | 93 kB | 00:00:00
(5/7): amzn2extra-docker/2/x86_64/primary_db | 116 kB | 00:00:00
(6/7): amzn2extra-kernel-5.10/2/x86_64/primary_db | 32 MB | 00:00:00
(7/7): amzn2-core/2/x86_64/primary_db | 72 MB | 00:00:01
Package wget-1.14-18.amzn2.1.x86_64 already installed and latest version
Resolving Dependencies
--> Running transaction check
---> Package mariadb-server.x86_64 1:5.5.68-1.amzn2.0.1 will be installed
--> Processing Dependency: mariadb(x86-64) = 1:5.5.68-1.amzn2.0.1 for package: 1:mariadb-server-5.5.68-1.amzn2.0.1.x86_64
--> Processing Dependency: perl(DBI) for package: 1:mariadb-server-5.5.68-1.amzn2.0.1.x86_64
```

Dependency installed:

```
mariadb.x86_64 1:5.5.68-1.amzn2.0.1
perl-Compress-Raw-Bzip2.x86_64 0:2.061-3.amzn2.0.2
perl-Compress-Raw-Zlib.x86_64 1:2.061-4.amzn2.0.2
perl-DBD-MySQL.x86_64 0:4.023-6.amzn2
perl-DBI.x86_64 0:1.627-4.amzn2.0.2
perl-Data-Dumper.x86_64 0:2.145-3.amzn2.0.2
perl-IO-Compress.noarch 0:2.061-2.amzn2
perl-Net-Daemon.noarch 0:0.48-5.amzn2
perl-PlRPC.noarch 0:0.2020-14.amzn2
```

Complete!

[root@ip-172-31-92-115 ~]# |

```
[root@ip-172-31-92-115 ~]# ls
[root@ip-172-31-92-115 ~]# systemctl status mariadb
Unit mariadb.service could not be found.
[root@ip-172-31-92-115 ~]# systemctl status mariadb
Unit mariadb.service could not be found.
[root@ip-172-31-92-115 ~]# systemctl status mariadb
● mariadb.service - MariaDB database server
   Loaded: loaded (/usr/lib/systemd/system/mariadb.service; enabled; vendor preset: disabled)
   Active: active (running) since Fri 2024-11-22 11:59:22 UTC; 5min ago
     Process: 3540 ExecStartPost=/usr/libexec/mariadb-wait-ready $MAINPID (code=exited, status=0/SUCCESS)
     Process: 3457 ExecStartPre=/usr/libexec/mariadb-prepare-db-dir %n (code=exited, status=0/SUCCESS)
    Main PID: 3539 (mysqld_safe)
      CGroup: /system.slice/mariadb.service
              └─3539 /bin/sh /usr/bin/mysqld_safe --basedir=/usr
                  └─3709 /usr/libexec/mysqld --basedir=/usr --datadir=/var/lib/mysql --plugin-dir=/usr/lib64/mysql/plugin --log-error=/var/log/mariadb/mariadb.log --pid-f
```

Nov 22 11:59:20 ip-172-31-92-115.ec2.internal mariadb-prepare-db-dir[3457]: MySQL manual for more instructions.

Nov 22 11:59:20 ip-172-31-92-115.ec2.internal mariadb-prepare-db-dir[3457]: Please report any problems at <http://mariadb.org/jira>

Nov 22 11:59:20 ip-172-31-92-115.ec2.internal mariadb-prepare-db-dir[3457]: The latest information about MariaDB is available at <http://mariadb.org/>.

Nov 22 11:59:20 ip-172-31-92-115.ec2.internal mariadb-prepare-db-dir[3457]: You can find additional information about the MySQL part at:

Nov 22 11:59:20 ip-172-31-92-115.ec2.internal mariadb-prepare-db-dir[3457]: <http://dev.mysql.com>

Nov 22 11:59:20 ip-172-31-92-115.ec2.internal mariadb-prepare-db-dir[3457]: Consider joining MariaDB's strong and vibrant community:

Nov 22 11:59:20 ip-172-31-92-115.ec2.internal mariadb-prepare-db-dir[3457]: <https://mariadb.org/get-involved/>

Nov 22 11:59:21 ip-172-31-92-115.ec2.internal mysqld_safe[3539]: 241122 11:59:21 mysqld_safe Logging to '/var/log/mariadb/mariadb.log'.

Nov 22 11:59:21 ip-172-31-92-115.ec2.internal mysqld_safe[3539]: 241122 11:59:21 mysqld_safe Starting mysqld daemon with databases from /var/lib/mysql

Nov 22 11:59:22 ip-172-31-92-115.ec2.internal systemd[1]: Started MariaDB database server.

[root@ip-172-31-92-115 ~]# |

2) Insert some dummy data

```

MariaDB [(none)]> USE ec2db;
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 [ec2db]> insert into table1 values(1, 'rakesh'), (2, 'naresh'), (3, 'ganesh')
, (4, 'sundar');
Query OK, 4 rows affected (0.01 sec)
Records: 4  Duplicates: 0  Warnings: 0

MariaDB [ec2db]> select * from table1:
->
->
->
->
->
-> select * from table1;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corre
sponds to your MariaDB server version for the right syntax to use near ':

select * from table1' at line 1
MariaDB [ec2db]> select * from table1;
+-----+-----+
| id  | name  |
+-----+-----+
| 1   | rakesh |
| 2   | naresh |
| 3   | ganesh |
| 4   | sundar |
+-----+-----+
4 rows in set (0.00 sec)

MariaDB [ec2db]> |

```

3) Take the backup of dummy data on ec2

```

[root@ip-172-31-92-115 backup]# ls
[root@ip-172-31-92-115 backup]# mysqldump -u root -p ec2db > backup_maria.sql
Enter password:
mysqldump: Got error: 1045: "Access denied for user 'root'@'localhost' (using password: YES)" when trying to connect
[root@ip-172-31-92-115 backup]# mysqldump -u root -p ec2db > backup_maria.sql
Enter password:
[root@ip-172-31-92-115 backup]# ls
backup_maria.sql
[root@ip-172-31-92-115 backup]# |

```

4) launch Mariadb RDS instance

Choose a database creation method

☐ Standard create

You set all of the configuration options, including ones for availability, security, backups, and maintenance.

☒ Easy create

Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

Configuration

Engine type [Info](#)

☐ Aurora (MySQL Compatible)



☐ Aurora (PostgreSQL Compatible)



☐ MySQL



☐ PostgreSQL



☒ MariaDB



☐ Oracle

ORACLE

Creating database my-database1

Your database might take a few minutes to launch. You can use settings from my-database1 to simplify configuration of suggested database add-ons while we finish creating your DB for you.

[View credential details](#)

Databases (1)

☒ Group resources



[Modify](#)

[Actions](#)

[Restore from S3](#)

[Create database](#)

< 1 >

<input type="checkbox"/>	DB identifier	Status	Role	Engine	Region ...	Size	Recon
<input type="radio"/>	my-database1	Creating	Instance	MariaDB	-	db.t4g.mi...	