

Practical 14

Configuring MariaDB

MariaDB Base Configuration

1) Check for MariaDB and MariaDB server on the server machine using yum repository.

```
[root@localhost ~]# yum list mariadb mariadb-server
Loaded plugins: langpacks, product-id, search-disabled-repos, subscription-manager
This system is not registered with an entitlement server. You can use subscription-manager to register.
Installed Packages
mariadb.x86_64                                1:5.5.56-2.el7                                @anaconda/7.4
mariadb-server.x86_64                        1:5.5.56-2.el7                                @anaconda/7.4
[root@localhost ~]# systemctl restart mariadb
[root@localhost ~]# systemctl status mariadb
● mariadb.service - MariaDB database server
   Loaded: loaded (/usr/lib/systemd/system/mariadb.service; disabled; vendor preset: disabled)
   Active: active (running) since Mon 2024-09-30 14:14:42 IST; 1min 7s ago
     Process: 4430 ExecStartPost=/usr/libexec/mariadb-wait-ready $MAINPID (code=exited, status=0/SUCCESS)
    Process: 4340 ExecStartPre=/usr/libexec/mariadb-prepare-db-dir %n (code=exited, status=0/SUCCESS)
   Main PID: 4429 (mysqld_safe)
      CGroup: /system.slice/mariadb.service
              └─4429 /bin/sh /usr/bin/mysqld_safe --basedir=/usr
                └─4591 /usr/libexec/mysqld --basedir=/usr --datadir=/var/lib/mysql -...
```

2) Then use **mysql_secure_installation** command.

```
[root@localhost ~]# mysql_secure_installation
```

```
NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!
```

```
In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.
```

```
Enter current password for root (enter for none):
OK, successfully used password, moving on...
```

```
Setting the root password ensures that nobody can log into the MariaDB
root user without the proper authorisation.
```

```
Set root password? [Y/n] y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!
```

```
By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them. This is intended only for testing, and to make the installation
go a bit smoother. You should remove them before moving into a
production environment.
```

```
Remove anonymous users? [Y/n] y
... Success!
```

```
Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.
```

Normally, root should only be allowed to connect from 'localhost'. This ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] y
... Success!

By default, MariaDB comes with a database named 'test' that anyone can access. This is also intended only for testing, and should be removed before moving into a production environment.

Remove test database and access to it? [Y/n] n
... skipping.

Reloading the privilege tables will ensure that all changes made so far will take effect immediately.

Reload privilege tables now? [Y/n] y
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB installation should now be secure.

Thanks for using MariaDB!
[root@localhost ~]# █

3) Type **mysql -u root -p** to login into the database as root and enter password.

```
[root@localhost ~]# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 8
Server version: 5.5.56-MariaDB MariaDB Server

Copyright (c) 2000, 2017, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> █
```

4) You have now logged into MariaDB. You can now run SQL commands.

```
MariaDB [(none)]> CREATE DATABASE model;
Query OK, 1 row affected (0.00 sec)

MariaDB [(none)]> USE model;
Database changed
MariaDB [model]> CREATE TABLE students (id INT(5) DEFAULT NULL, name VARCHAR(20) DEFAULT NULL, course VARCHAR(20) DEFAULT NULL, day DATE DEFAULT NULL, status BOOL DEFAULT NULL);
Query OK, 0 rows affected (0.01 sec)
```

```
MariaDB [model]> SHOW COLUMNS IN students;
```

Field	Type	Null	Key	Default	Extra
id	int(5)	YES		NULL	
name	varchar(20)	YES		NULL	
course	varchar(20)	YES		NULL	
day	date	YES		NULL	
status	tinyint(1)	YES		NULL	

```
5 rows in set (0.00 sec)

MariaDB [model]> INSERT INTO students VALUES('1234','jerry','DSA',NOW(),'1');
Query OK, 1 row affected, 1 warning (0.00 sec)

MariaDB [model]> SELECT * FROM students
-> ;
```

id	name	course	day	status
1234	jerry	DSA	2024-09-30	1

```
1 row in set (0.00 sec)
```

```
MariaDB [model]> SELECT * FROM students;
```

id	name	course	day	status
1234	jerry	DSA	2024-09-30	1

```
1 row in set (0.00 sec)
```

```
MariaDB [model]> SHOW DATABASES;
```

Database
information_schema
model
mysql
performance_schema
test

```
5 rows in set (0.00 sec)
```

5) To quit MariaDB, type exit.

```
MariaDB [model]> exit
Bye
```

6) Use following command to take backup of SQL databases.

```
[root@localhost ~]# mysqldump -u root -p model > /root/model > /root/model.sql
Enter password:
[root@localhost ~]# ls
anaconda-ks.cfg  initial-setup-ks.cfg  model  model.sql  tmprequest

[root@localhost ~]# mysqldump -u root -p --all-databases > /root/model2.sql
Enter password:
[root@localhost ~]# ls
anaconda-ks.cfg  initial-setup-ks.cfg  model  model2.sql  model.sql  tmprequest
```