AI-03006 MariaDB Information

Owner	Ⓡ Rigel Arcayan
Tags	Project
Created time	@September 16, 2025 8:05 PM

What is MariaDB? What is the history of the database creation and development? Create a comprehensive guide in the installation, deployment, management. List all the commands used in the Ubuntu server environment. Create a guide on how to perform basic database creation, user creation, passwords, logins, reset account, drop db etc. create a guide on how to install and configure phasing , it's prerequisites and needed dependencies.

MariaDB is an open-source relational database management system (RDBMS) that is a community-developed, commercially supported fork of MySQL. It was created to ensure that the RDBMS would remain free and open-source after the acquisition of MySQL by Oracle Corporation. MariaDB is known for its high performance, scalability, and robust features.

History of MariaDB

The story of MariaDB begins in 2009. Michael "Monty" Widenius, one of the original creators of MySQL, forked the database in response to Oracle's acquisition of Sun Microsystems, which had previously acquired MySQL. The move was a direct response to widespread concerns within the open-source community that Oracle would change MySQL's licensing and open-source nature. Widenius named the new project MariaDB after his younger daughter, Maria (he had previously named MySQL after his older daughter, My). The development was initially non-commercial and open-source. In 2010, the first stable version was released, and the MariaDB Corporation AB was established to build a business around the database. In 2012, MariaDB's versioning diverged from MySQL's, and the MariaDB Foundation was announced to oversee its development and ensure its open-source nature.

A Comprehensive Guide to MariaDB on Ubuntu

1. Installation and Deployment This guide assumes you are working in an Ubuntu server environment.

- **Update System Packages:** Always start by updating your package lists to ensure you're getting the latest version available.Bash

  ```
  sudo apt update
  ```

- **Install MariaDB Server:** This command installs the MariaDB server package and its necessary dependencies. The `y` flag automatically answers "yes" to any prompts.Bash

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

- **Verify Installation:** Check the status of the MariaDB service to ensure it's running correctly.Bash

  ```
  sudo systemctl status mariadb
  ```

- **Secure the Installation:** Run the security script to improve the default security of your MariaDB installation. This script will guide you through setting a root password, removing anonymous users, disallowing remote root logins, and removing the test database.Bash

  ```
  sudo mysql_secure_installation
  ```

---

2. Basic Database Management Commands

Once you have installed and secured MariaDB, you can log in and start managing databases. By default, the `root` user authenticates using the `unix_socket` plugin, so you don't need a password initially.

- **Log in to MariaDB:**Bash

  ```
  sudo mariadb
  ```

  This command will open the MariaDB command prompt. To exit, type `exit;` and press Enter.

- **Create a New Database:**SQL

  ```
  CREATE DATABASE mynewdatabase;
  ```

  Remember to end all SQL commands with a semicolon (;).

- **Show All Databases:**SQL

  ```
  SHOW DATABASES;
  ```

- **Drop (Delete) a Database:**SQL

  ```
  DROP DATABASE mynewdatabase;
  ```

  Warning: This command is irreversible and will permanently delete the database and all its data.

---

3. User Management and Security

It is best practice to avoid using the `root` user for everyday operations. Instead, create specific users with limited privileges.

- **Create a New User:**SQL

  ```
  CREATE USER 'newuser'@'localhost' IDENTIFIED BY 'strong_password_here';
  ```

  - `newuser`: The desired username.
  - `localhost`: The host from which the user can connect. Replace `localhost` with `%` to allow connections from any host (this is less secure and should be used with caution).
  - `strong_password_here`: The password for the new user.

- **Grant Privileges to a User:**SQL

  ```
  GRANT ALL PRIVILEGES ON mynewdatabase.* TO 'newuser'@'localhost';
  ```

  This command grants all privileges on the `mynewdatabase` to `newuser`.

  - `mynewdatabase.*`: Specifies that the privileges apply to all tables within the `mynewdatabase`.

    - `ALL PRIVILEGES`: Grants all available permissions. You can specify more granular permissions like `SELECT`, `INSERT`, `UPDATE`, etc.
- **Flush Privileges:** After creating a new user or changing their privileges, always run this command to apply the changes immediately.SQL

  ```
  FLUSH PRIVILEGES;
  ```
- **Log in as a New User:**Bash

  ```
  mariadb -u newuser -p
  ```

  You will be prompted to enter the password you set for `newuser`.
- **Reset a User's Password:**SQL

  ```
  ALTER USER 'newuser'@'localhost' IDENTIFIED BY 'even_stronger_password';
  ```

  This changes the password for the existing user.

Phasing in MariaDB

The user prompt mentions "phasing." This term is not a standard feature, tool, or concept in MariaDB. It is possible this is a typo or refers to a third-party tool or a highly specialized internal process that is not widely documented or publicly available. Therefore, a guide on how to install and configure "phasing" for MariaDB cannot be created with the information available. If you're referring to a different tool, please clarify.