



MariaDB 설치

📍 난이도	
🔧 기술	
🕒 생성 일시	@ 2023년 2월 1일 오후 1:36
👤 생성자	진영 장
🕒 최종 편집 일시	@ 2023년 2월 27일 오후 1:25

[Ubuntu] 우분투에 MySQL 설치하기

apt-get이라는 패키지 매니저를 이용해 설치를 진행한다. (우분투 리눅스를 설치했다면 이미 설치되어 있다.) \$ sudo apt-get update \$ sudo apt-get install mysql-server 설치 중간 중간 Password를 물어보는데 빈칸으로 두고 OK를 누르면 Ubuntu 서버 비밀번호와 자동으로 동기화 된다. 3-1. 외부 접속 기능 설정 (포트 3306 오픈) 처음에 이 부분을

📄 <https://velog.io/@seungsang00/Ubuntu-%EC%9A%B0%EB%B6%84%ED%88%AC%EC%97%90-MySQL-%EC%84%A4%EC%B9%98%ED%95%98%EA%B8%B0>

velog

mariaDB server 설치

```
#우분투 서버 업데이트
$ sudo apt-get update && sudo apt-get -y upgrade

#mariadb-server 설치
$ sudo apt-get install mariadb-server -y

#ssh 설치
$ sudo apt install openssh-server

#mariadb 접속
$ sudo mysql
#또는
$ sudo mariadb
```

mariaDB client 설치

```
#mariadb-client 설치
$ sudo apt install mariadb-client
```

mariaDB secure 설치

```
#mariadb-secure 설치
$ sudo 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] n
... skipping.

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] n
... skipping.

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] Y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!

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!

```

원격 Client 접속을 위한 MariaDB 설정 방법



아래 세가지 진행해야한다.

1. bind-address = 0.0.0.0
2. CREATE USER '<ID>'@'%' IDENTIFIED BY '<PWD>';
3. mysql> GRANT ALL PRIVILEGES ON *.* TO '<ID>'@'%';

외부서버에서 접속시 IaaS에서 Inbound 규칙에 포트를 열어줬는데 확인해야한다.

Configuration 파일에서 skip-networking 과 skip bind-address 수정

mariaDB(mysql) 원격 접속 허용하기 (How to allow 'Remote access' for mariaDB(mysql)) & docker로 mariadb 실행하기위한 docker-compose.yml & 구성
요즘은 maria DB 를 자주 사용하고 있습니다. 가볍고, 지원하는 툴도 가볍고, 무료이고.. 사용하기 편한 DB 인 것 같습니다. 하지만 모든 프로그램이 그렇듯, 최초에 설치하고 나면 설
것들이 조금 있습니다. mysql 기반 maria DB 역시 마찬가지인데, 가장 큰 문제가 되는 것이 원격 접속에 대한 것입니다.

📄 <https://4urdev.tistory.com/82>

우분투 20.04에서 MariaDB 설치 및 기본 동작 확인 명령어

관계형 데이터 베이스의 대표 주자는 Oracle입니다. 관계형 데이터베이스 업체는 오라클, IBM, 마이크로소프트 순이
고, Oracle의 DBMS의 가장 높으며, 오픈 소스 진영의 DBMS는 MySQL, PostgreSQL, SQLite을 많이 사용하고 있
습니다. 최근에 MySQL이 오라클 인수된 영향으로 기능 개선에 다소 미흡한 부분이 있고, 라이선스가 모호한 점이 있

📄 <https://kibua20.tistory.com/185>



```

#mysqld에 대한 default option 확인
#default 옵션을 보면 --bind-address가 127.0.0.1로 설정되어 있음을 확인할 수 있고, /etc/mysql/mariadb.conf.d/50-server.cnf 파일에서 설정되어 있다.
$ mysql --print-defaults

$ sudo vi /etc/mysql/my.cnf
...
[mysqld]
skip-networking=0
skip-bind-address
...

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

```

```
bind-address      = 0.0.0.0
...
port              = 13306
...

$ sudo systemctl restart mariadb.service
```

MySQL 기본 설정

외부 접속 기능 설정(포트 3306 오픈)

처음에 이 부분을 설정을 안해줬더니 MySQL Workbench를 사용하려고 했더니 오류가 발생했다.

```
#외부 접속 기능 설정(포트 3306 오픈)
$ sudo ufw allow mysql

#MySQL 실행
$ sudo systemctl start mysql

# Ubuntu 서버 재시작시 MySQL 자동 재시작
$ sudo systemctl enable mysql
```

MySQL 접속

```
$ sudo /usr/bin/mysql -u root -p
Enter password: 엔터
```

사용자 등록 및 권한 설정

사용자 정보 확인

```
mysql> SELECT User, Host, authentication_string FROM mysql.user;
```

계정 만들기

```
#localhost에 대해서 만들 때
mysql> CREATE USER 'testuser'@'localhost' IDENTIFIED BY 'mysql비번';

#모든 IP에 대해서 만들 때
mysql> CREATE USER 'paasta'@'%' IDENTIFIED BY 'vktmxktpsxj@22!';
mysql> FLUSH PRIVILEGES;
mysql> SELECT User, Host, authentication_string FROM mysql.user;
```

사용할 데이터베이스에 사용자 권한 부여

```
#mysql> GRANT ALL PRIVILEGES ON 데이터베이스이름.* FOR'testuser'@'localhost';
#localhost에 대해서 만들 때
mysql> GRANT ALL PRIVILEGES ON *.* FOR'testuser'@'localhost';

#모든 IP에 대해서 만들 때
mysql> GRANT ALL PRIVILEGES ON *.* TO 'paasta'@'%';
mysql> GRANT ALL PRIVILEGES ON *.* TO 'psta_release'@'%' IDENTIFIED BY 'paastaadmin' WITH GRANT OPTION;
mysql> FLUSH PRIVILEGES;
mysql> SELECT User, Host, authentication_string FROM mysql.user;

#권한부여 확인
mysql> SHOW GRANTS FOR 'paasta'@'%';
```

MySQL 비밀번호 변경

```
mysql> SET PASSWORD FOR 'root'@'localhost' = PASSWORD('바꿀비번');

또는
```

```
mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY '바꿀비번';

#예)
mysql> SET PASSWORD FOR 'paasta'@'%' = PASSWORD('vktmxktpsxbj@22!');
mysql> ALTER USER 'paasta'@'%' IDENTIFIED WITH mysql_native_password BY 'vktmxktpsxbj@22!';
```

ID 삭제하기

```
mysql> DELETE FROM mysql.user WHERE Host='%' AND User='아이디';
mysql> FLUSH PRIVILEGES;
```

DATABASE 생성

```
mysql> CREATE DATABASE paasta default CHARACTER SET UTF8;
mysql> CREATE DATABASE `pstadb-apply` default CHARACTER SET UTF8;
mysql> CREATE DATABASE `pstadb-board` default CHARACTER SET UTF8;
mysql> CREATE DATABASE `pstadb-common` default CHARACTER SET UTF8;
mysql> CREATE DATABASE `pstadb-release` default CHARACTER SET UTF8;
```