**MariaDB**

MariaDB는 MySQL의 발전된 형태의 대체제로써, <https://downloads.mariadb.org/>에서 다운로드 받을 수 있으며, GPL v2 라이선스로 유지되고 있고, MariaDB 커뮤니티와 MariaDB 재단이 주축이 되어 개발되고 있다.

**보안**

보안은 현재의 세계에서 매우 중요하며 MariaDB 개발자들의 주의를 기울이고 있다. 이 프로젝트는 MySQL 프로젝트 기반으로 자체적인 보안 패치를 유지하고 있다. 각 MariaDB 릴리즈에 대해 개발자는 MySQL 보안 패치를 merge 할 수 있으며 필요한 경우 개선할 수도 있다. 중요한 보안 이슈가 발견되면, 개발자들은 즉시 이를 해결하는 새로운 MariaDB 릴리즈를 개발, 배포한다.

MySQL에서 발견된 많은 보안 이슈는 MariaDB에서도 발견되어 왔으며 MariaDB 팀에 보고되어 왔다. MariaDB 팀은 모든 보안 이슈들이 즉각 보고되고 충분히 자세하게 해결될 수 있도록 http://cve.mitre.org/ 와 긴밀히 협력하고 있다. 자세한 보안 이슈들은 일반적으로 이슈가 해결된MariaDB 와 MySQL 버전이 배포된 이후에 릴리즈된다.

**호환성**

MariaDB는 현재까지 최신의 MaySQL과 같은 브랜치로부터 릴리즈되며, 대개의 경우 MySQL과 마찬가지로 동작한다. MySQL의 모든 명령어, 인터페이스, 라이브러리와 API가 MariaDB에도 존재한다. 또한 MariaDB로 데이터베이스를 변환할 필요도 없다. 즉, MariaDB는 사실상 MySQL의 완벽한 대체제라고 말할 수 있다.

**현재 버전과 릴리즈 스케줄**

현재 MariaDB의 안정된 릴리즈는 MariaDB 5.5 이다.

개발 버전 릴리즈는 MariaDB 10.0.

이전의 안정된 릴리즈는 MariaDB 5.3.

현재와 개발 버전(이전 버전들)은 <https://downloads.mariadb.org/>에서 다운로드 받을 수 있다.

**MariaDB설치**

**~$ sudo apt-get install mariadb-server**

|  |
| --- |
| 패키지 목록을 읽는 중입니다... 완료  의존성 트리를 만드는 중입니다  상태 정보를 읽는 중입니다... 완료  다음의 추가 패키지가 설치될 것입니다:  alera-3 gawk libaio1 libconfig-inifiles-perl libdbd-mysql-perl libdbi-perl libhtml-template-perl  libjemalloc1 libmysqlclient20 libreadline5 libsigsegv2 libterm-readkey-perl mariadb-client-10.1  mariadb-client-core-10.1 mariadb-common mariadb-server-10.1 mariadb-server-core-10.1 mysql-common socat  제안하는 패키지:  gawk-doc libmldbm-perl libnet-daemon-perl libsql-statement-perl libipc-sharedcache-perl mailx tinyca  다음 새 패키지를 설치할 것입니다:  galera-3 gawk libaio1 libconfig-inifiles-perl libdbd-mysql-perl libdbi-perl libhmtl-template-perl  libjemalloc1 libmysqlclient20 libreadline5 libsigsegv2 libterm-readkwy-perl mariadb-client-10.1  mariadb-client-core-10.1 mariadb-common mariadb-server mariadb-server-10.1  mariadb-server-core-10.1 mysql-common socat  0개 업그레이드, 20개 새로 설치, 0개 제거 및 0개 업그레이드 안함.  24.1 M바이트 아카이브를 받아야 합니다.  이 작업 후 184 M바이트의 디스크 공간을 더 사용하게 됩니다.  계속 하시겠습니까? [Y/n] |

- 설치 날짜 기준(2019.03.30)으로 MariaDB의 10.1.38버전와 의존성 패키지 설치

**MariaDB의 권한 테이블 설정**

**~$ sudo mysql\_secure\_installation**

|  |
| --- |
| NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB SERVER 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... |

**Enter current password for root (enter for none)** → MariaDb의 root계정은 쉘인증이 기본적으로 성정되므로 root계정으로 실행됐다면 비밀번호 없이(Enter) 아니면 비밀번호 입력

|  |
| --- |
| Setting the root password ensures that nobody can log into the MariaDB  root user without the proper authorisation.  Set root password? [Y/n]  New password:  Re-enter new password:  Password updated successfully!  Reloading privilege tables.  ... Success! |

**Set root password? [Y/n]** → 따로 패스워드를 성정하고 싶으면 Y, root그대로 사용할려면 n

- New password: → 패스워드 입력

- Re-enter new password: → 패스워드 재입력

|  |
| --- |
| 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]  ... Success! |

**Remove anonymous users? [Y/n]** → 익명 사용자를 삭제할지여부(Y-삭제, n-삭제 안함)

|  |
| --- |
| Nomally, 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]  ... Success! |

**Disallow root login remotely? [Y/n]** → 원격 접속으로 루트 로그인 허용 여부(Y-거부, n-허용)

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

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

→ 기본적으로 테스트 데이터베이스를 제공해주기 때문에 이와같이 물어봄

→ 테스트 데이터베이스 삭제 여부(Y-삭제, n-삭제 안함)

|  |
| --- |
| Reload privilege tables now? [Y/n]  ... 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! |

**Reload privilege tables now? [Y/n]** → 지금까지 작성한 권한 테이블을 적용 여부(Y-적용, n-적용 안함)

**MariaDB의 Character sets 설정**

**~$ sudo gedit /etc/mysql/mariadb.conf.d/50-server.cnf**

|  |
| --- |
| # \* Character sets  #  # MySQL/MariaDB default is Latin1, but in Debian we rather default to the full  # utf8 4-byte character set. See also client.cnf  #  character-set-setver = utf8mb4  # collation-server = utf8mb4\_general\_ci  collation-server = utf8mb4\_unicode\_ci |

\* Character sets 설정 - 위와 같이 수정 → 저장 → 종료

utf8m64\_general\_ci(default)을 사용할 경우 정렬 속도가 utf8m64\_unicode\_ci에 비해 약간 빠르 긴 하지만 거의 차이는 없음, 한글/일본어와 같이 비 라틴계 언어들의 정렬이 조금 어색한 경우가 있기 때문에 utf8m64\_unicode\_ci 사용

\* MySQL에 대한 Character sets설정 → /etc/mysql/mariadb.conf.d/50-mysql-clients.cnf

\* Client에 대한 Character sets설정 → /etc/mysql/mariadb.conf.d/50-client.cnf

**MariaDB의 Version 확인**

**~$ mariadb –version**

