

MySQL jest otwarto-źródłowym (open-source) systemem zarządzania bazą danych, powszechnie instalowanym jako część popularnego zestawu LAMP (Linux, Apache, MySQL, PHP / Python / Perl). Implementuje on model relacyjny, a do zarządzania swoimi danymi wykorzystuje Structured Query Language (znanego lepiej jako SQL).

Spis treści

1.	Instalacja MySQL na Ubuntu 20.04	2
2.	Konfiguracja MySQL	2
3.	Dostosowywanie uwierzytelnienia i uprawnień użytkownika (opcjonalnie)	3
4.	Z wiersza poleceń MySQL sprawdź wersję mysql	5
5.	Jak utworzyć bazę danych MySQL przez konsolę Linux ?	6
	Tworzenie bazy danych MySQL	6
	Wyświetlanie wszystkich baz danych MySQL	7
6.	Jak usunąć bazę danych MySQL używając konsoli Linux ?	7
	Usunięcie bazy danych w MySQL	8
	Wyświetlanie wszystkich baz danych MySQL	8
7.	Jak zresetować hasło root do MySQL	9
	Identyfikacja wersji bazy danych	9
	Zatrzymywanie serwera bazy danych	9
	Ponowne uruchomienie serwera bazy danych bez sprawdzania uprawnień	10
	Zmiana hasła administratora	11
	Uruchom ponownie serwer bazy danych w standardowy sposób	11

1.Instalacja MySQL na Ubuntu 20.04

Aby zainstalować MySQL, zaktualizuj najpierw indeks pakietu serwera, jeśli nie robiłeś tego ostatnio:

```
sudo apt update
```

Następnie zainstaluj pakiet *mysql-server*:

```
sudo apt install mysql-server
```

2.Konfiguracja MySQL

Uruchom (z sudo) dołączony skrypt bezpieczeństwa MySQL:

```
sudo mysql_secure_installation
```

Spowoduje to przejście przez serię monitów, w których możesz wprowadzić zmiany w opcjach bezpieczeństwa instalacji MySQL. W pierwszym monicie pojawi się pytanie, czy chcesz skonfigurować wtyczkę Validate Password, której można użyć do przetestowania siły hasła MySQL.

Jeśli zdecydujesz się skonfigurować wtyczkę Validate Password, skrypt poprosi cię o wybranie poziomu sprawdzania hasła (przy czym najsłabszym jest 0, a najsilniejszym 2):

```
Securing the MySQL server deployment.
```

```
Connecting to MySQL using a blank password.
```

```
VALIDATE PASSWORD COMPONENT can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD component?
```

```
Press y|Y for Yes, any other key for No: Y
```

```
There are three levels of password validation policy:
```

```
LOW      Length >= 8
MEDIUM  Length >= 8, numeric, mixed case, and special characters
STRONG Length >= 8, numeric, mixed case, special characters and dictionary
file
```

```
Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG:
```

```
2
```

Następnym pytaniem będzie wprowadzenie i potwierdzenie hasła dla użytkownika root MySQL:

```
Please set the password for root here.
```

```
New password:
```

```
Re-enter new password:
```

Skrypt zapyta wtedy, czy chcesz kontynuować z wprowadzonym hasłem, czy chcesz wprowadzić nowe. Jeśli hasło jest zadowalające, wpisz *Y*, aby kontynuować działanie skryptu:

```
Estimated strength of the password: 100
Do you wish to continue with the password provided?(Press y|Y for Yes, any
other key for No) : Y
```

Od tego momentu możesz naciskać *Y*, a następnie ENTER, aby zaakceptować wartości domyślne dla wszystkich kolejnych pytań.

3. Dostosowywanie uwierzytelnienia i uprawnień użytkownika (opcjonalnie)

Aby użyć hasła do połączenia się z MySQL jako root, musisz zmienić jego metodę uwierzytelniania z domyślnej wtyczki uwierzytelniającej *auth_socket* na inną, taką jak *caching_sha2_password* lub *mysql_native_password*. Aby to zrobić, otwórz okno MySQL w swoim terminalu:

```
sudo mysql
```

Następnie uruchom instrukcję *ALTER USER*, aby zmienić używaną wtyczkę uwierzytelniającą i ustawić nowe *hasło*. Zmień hasło na wybrane przez siebie silne hasło i pamiętaj, że to polecenie zmieni hasło root ustawione w kroku 2:

```
ALTER USER 'root'@'localhost' IDENTIFIED WITH caching_sha2_password BY
'hasło';
```

Uwaga: *caching_sha2_password* jest preferowaną wtyczką uwierzytelniającą MySQL, ponieważ zapewnia bezpieczniejsze szyfrowanie haseł niż starsza (ale wciąż powszechnie używane) wtyczka *mysql_native_password*.

Jednak wiele aplikacji PHP – na przykład phpMyAdmin – nie działa niezawodnie z *caching_sha2_password*. Jeśli planujesz używać tej bazy danych z aplikacją PHP, dla użytkownika root możesz ustawić uwierzytelnianie za pomocą *mysql_native_password*:

```
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY
'password';
```

Następnie uruchom instrukcję *FLUSH PRIVILEGES*, która nakazuje serwerowi ponowne załadowanie tabel uprawnień i wprowadzenie nowych zmian w życie:

```
FLUSH PRIVILEGES;
```

Sprawdź ponownie metody uwierzytelniania stosowane przez każdego z użytkowników, aby potwierdzić, że *root* nie uwierzytelnia się już przy użyciu wtyczki *auth_socket*:

```
SELECT user, authentication_string, plugin, host FROM mysql.user;
```

```

+-----+-----+
+-----+-----+
| user          | authentication_string |
| plugin        | host                  |
+-----+-----+
+-----+-----+
| debian-sys-maint | $A$005$1S|M#3K
#XslZ.dXUq.lkEqTjMvhgOIX7B/jgh5DeLA3JB9nh0KwENTwQ4 | caching_sha2_password
| localhost |
| mysql.infoschema | $A$005$KOMBINACJAHASLAZSZYFREMKTORENIEWYKORZYSTUJESIE
| caching_sha2_password | localhost |
| mysql.session    | $A$005$KOMBINACJAHASLAZSZYFREMKTORENIEWYKORZYSTUJESIE
| caching_sha2_password | localhost |
| mysql.sys        | $A$005$KOMBINACJAHASLAZSZYFREMKTORENIEWYKORZYSTUJESIE
| caching_sha2_password | localhost |
| root             | *7564DAVC9151D997782ADD0839F92C1571D6D78F
| caching_sha2_password | localhost |
+-----+-----+
+-----+-----+
5 rows in set (0.00 sec)

```

Po potwierdzeniu tego na swoim serwerze, możesz wyjść z powłoki MySQL:

```
exit
```

Alternatywnie, możesz połączyć się z MySQL, używając do tego dedykowanego użytkownika zamiast użytkownika **root**. Aby utworzyć takiego użytkownika, ponownie otwórz powłokę MySQL:

```
sudo mysql
```

Uwaga: Jeśli włączono uwierzytelnianie za pomocą hasła dla użytkownika root – jak opisano w poprzednich akapitach – należy zamiast tego uruchomić następujące polecenie:

```
mysql -u root -p
```

Następnie stwórz nowego użytkownika i nadaj mu silne hasło:

```
CREATE USER 'adam'@'localhost' IDENTIFIED BY 'haslo';
```

Następnie nadaj nowemu użytkownikowi odpowiednie uprawnienia. Możesz na przykład nadać użytkownikowi uprawnienia do wszystkich tabel w bazie danych, a także uprawnienia do dodawania, zmieniania i usuwania danych za pomocą tego polecenia:

```
GRANT ALL PRIVILEGES ON *.* TO 'adam'@'localhost' WITH GRANT OPTION;
```

Następnie możesz wyjść z powłoki MySQL:

```
Exit
```

4.Z wiersza poleceń MySQL sprawdź wersje mysql

Uzyskać informację o wersji MySQL można za pomocą następującego polecenia:

```
mysql
```

Uzyskasz wynik podobny do tego:

```
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 2
Server version: 5.5.41-MariaDB MariaDB Server

Copyright (c) 2000, 2014, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]>
```

o podłączeniu do wierszu poleceń MySQL, można również uruchomić polecenie:

```
SHOW VARIABLES LIKE "%version%";
```

Wynik będzie podobny do tego:

```
+-----+-----+
| Variable_name | Value |
+-----+-----+
| innodb_version | 5.5.40-MariaDB-36.1 |
| protocol_version | 10 |
| slave_type_conversions | |
| version | 5.5.41-MariaDB |
| version_comment | MariaDB Server |
| version_compile_machine | x86_64 |
| version_compile_os | Linux |
+-----+-----+
7 rows in set (0.10 sec)
```

Gratulacje, teraz już umiesz sprawdzać wersję MySQL!

5. Jak utworzyć bazę danych MySQL przez konsolę Linux ?

Krótki artykuł z podstawowymi poleceniami do tworzenia nowej bazy danych MySQL poprzez terminal/konsolę na serwerze z systemem operacyjnym Linux.

MySQL z konsoli Linux: Podstawy interakcji z bazą danych

Przedstartowe sprawdzenie

- Ten artykuł dotyczy tworzenia bazy danych MySQL w systemie Linux używając konsoli.

Tworzenie bazy danych MySQL

Zacznijmy od połączenia z serwerem MySQL w konsoli Linux używając następującego polecenia:

```
mysql -u root -p
```

W tym przypadku wskazałem użytkownika **root** używając parametr `-u`. Parametr `-p` wykorzystałem w taki sposób aby wpisać hasło w kolejnej linii konsoli. Wprowadź swoje hasło, aby zalogować się do systemu.

Teraz musisz być zalogowany w MySQL:

```
mysql>
```

Utworzyć bazę danych o nazwie **tutorial_database** można używając polecenia:

```
CREATE DATABASE tutorial_database;
```

Jeżeli baza danych o podanej nazwie już istnieje, to nowa baza danych nie zostanie utworzona, a natomiast otrzymasz błąd:

```
ERROR 1007 (HY000): Can't create database 'tutorial_database'; database exists
```

Aby ten błąd był niewidoczny użyj następującego polecenia:

```
CREATE DATABASE IF NOT EXISTS tutorial_database;
```

Wyświetlanie wszystkich baz danych MySQL

Aby wyświetlić listę baz danych, należy użyć polecenia:

```
SHOW DATABASES;
```

Wynik powinien być podobny do tego:

```
mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| mysql      |
| test       |
| tutorial_database |
+-----+
4 rows in set (0.00 sec)
```

Gratulacje, teraz już umiesz tworzyć bazy danych MySQL z konsoli Linux!

6. Jak usunąć bazę danych MySQL używając konsoli Linux ?

Krótką instrukcją z poleceniem usuwania bazy danych w MySQL z poziomu konsoli lub terminalu na serwerach z systemem operacyjnym Linux.

Przedstartowe sprawdzenie

- Ten artykuł dotyczy usuwania bazy danych MySQL w systemie Linux używając konsoli.

Najpierw zacznijmy od połączenia z serwerem MySQL w konsoli Linux używając następującego polecenia:

```
mysql -u root -p
```

W tym przypadku wskazałem użytkownika **root** używając parametr `-u`. Parametr `-p` wykorzystałem w taki sposób aby wpisać hasło w kolejnej linii konsoli. Wprowadź swoje hasło, aby zalogować się do systemu.

Teraz musisz być zalogowany w MySQL:

```
mysql>
```

Usunięcie bazy danych w MySQL

Bazę danych można usunąć używając jednego polecenia. **Uważaj! Usuwanie bazy danych nie może być anulowane!**

Polecenie wygląda następująco:

```
DROP DATABASE tutorial_database;
```

Jeżeli baza danych o nazwie **tutorial_database** nie istnieje, to pojawi się ten błąd:

```
ERROR 1008 (HY000): Can't drop database 'tutorial_database'; database doesn't exist
```

Aby ten błąd był niewidoczny użyj następującego polecenia:

```
DROP DATABASE IF EXISTS tutorial_database;
```

Wyświetlanie wszystkich baz danych MySQL

Aby wyświetlić listę baz danych, należy użyć polecenia:

```
SHOW DATABASES;
```

Wynik powinien być podobny do tego:

```
mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| mysql      |
| test       |
+-----+
4 rows in set (0.00 sec)
```

Gratulacje, teraz już umiesz usuwać bazę danych MySQL używając konsoli Linux!

7. Jak zresetować hasło root do MySQL

Zapominanie haseł zdarza się najlepszym z nas. Jeśli zapomnisz lub stracisz hasło administratora do bazy danych MySQL lub MariaDB, nadal możesz uzyskać dostęp i zresetować hasło, jeśli posiadasz dostęp do serwera i konta użytkownika z włączonym sudo.

W tym artykule omówimy sposób resetowania hasła administratora dla starszych i nowszych wersji MySQL i MariaDB.

Będziemy potrzebowali

Aby odzyskać hasło administratora MySQL/MariaDB, będziesz potrzebować:

- dostępu do serwera Linux z MySQL lub MariaDB z użytkownikiem sudo.

Identyfikacja wersji bazy danych

Większość współczesnych dystrybucji Linuksa jest dostarczana z MySQL lub MariaDB, popularnym zamiennikiem, który jest w pełni kompatybilny z MySQL. W zależności od użytej bazy danych i jej wersji musisz użyć różnych poleceń, aby odzyskać hasło administratora.

Możesz sprawdzić swoją wersję za pomocą następującego polecenia:

```
mysql --version
```

W przypadku MySQL, otrzymasz następujący rezultat:

```
Dane wyjściowe MySQL  
mysql Ver 14.14 Distrib 5.7.16, for Linux (x86_64) using EditLine wrapper
```

Albo następujący rezultat dla MariaDB:

```
Dane wyjściowe MariaDB  
mysql Ver 15.1 Distrib 5.5.52-MariaDB, for Linux (x86_64) using readline  
5.1
```

Zanotuj, której bazy danych i której wersji używasz, ponieważ będziesz korzystać z tych informacji później. Następnie musisz zatrzymać bazę danych, aby uzyskać do niej dostęp ręcznie.

Zatrzymywanie serwera bazy danych

Aby zmienić hasło administratora, musisz wcześniej zatrzymać serwer bazy danych.

Dla bazy MySQL możesz to zrobić poniższym poleceniem:

```
sudo systemctl stop mysql
```

Dla bazy MariaDB – takim:

```
sudo systemctl stop mariadb
```

Po zatrzymaniu serwera bazy danych uzyskasz do niego dostęp ręcznie, aby zresetować hasło administratora.

Ponowne uruchomienie serwera bazy danych bez sprawdzania uprawnień

Jeśli uruchomisz MySQL i MariaDB bez wczytania informacji o uprawnieniach użytkownika, pozwoli ci to uzyskać dostęp do wiersza poleceń bazy danych z uprawnieniami administratora bez podawania hasła. Pozwoli ci to uzyskać dostęp do bazy danych, nie znając go.

Aby to zrobić, musisz powstrzymać bazę danych przed wczytaniem tabel uprawnień, które przechowują informacje o uprawnieniach użytkownika. Ponieważ jest to trochę ryzykowne z punktu widzenia bezpieczeństwa, należy również pominąć sieć, aby uniemożliwić innym klientom nawiązywanie połączeń.

Uruchom bazę danych bez ładowania tabel uprawnień i włączania sieci:

```
sudo mysqld_safe --skip-grant-tables --skip-networking &
```

Znak ampersand na końcu tego polecenia spowoduje, że proces ten będzie działał w tle, dzięki czemu będziesz mógł nadal korzystać z terminala.

Teraz możesz połączyć się z bazą danych jako użytkownik root, co nie powinno spowodować pytania o hasło.

```
mysql -u root
```

Zamiast tego natychmiast zobaczysz shell bazy danych.

Shell MySQL:

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

```
mysql>
```

Shell MariaDB:

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

```
MariaDB [(none)]>
```

Teraz, gdy masz dostęp do konta administratora, możesz zmienić jego hasło.

Zmiana hasła administratora

Jednym prostym sposobem zmiany hasła roota dla nowszych wersji MySQL jest użycie polecenia *ALTER USER*. Jednak to polecenie teraz nie zadziała, ponieważ tabele uprawnień nie są wczytane.

Powiedzmy serwerowi bazy danych, aby ponownie załadował tabele uprawnień, wydając polecenie *FLUSH PRIVILEGES*.

```
FLUSH PRIVILEGES;
```

Teraz możemy zmienić hasło administratora.

W przypadku MySQL 5.7.6 i nowszych, a także MariaDB 10.1.20 i nowszych, użyj następującego polecenia:

```
ALTER USER 'root'@'localhost' IDENTIFIED BY 'nowe_haslo';
```

W przypadku MySQL 5.7.5 i starszych, a także MariaDB 10.1.20 i starszych, użyj:

```
SET PASSWORD FOR 'root'@'localhost' = PASSWORD('nowe_haslo');
```

Pamiętaj, aby zastąpić *nowe_haslo* nowym wybranym hasłem.

Uwaga: jeśli polecenie *ALTER USER* nie działa, zwykle oznacza to większy problem. Możesz jednak spróbować *UPDATE ... SET*, aby zamiast tego zresetować hasło administratora.

```
UPDATE mysql.user SET authentication_string = PASSWORD('nowe_haslo') WHERE User = 'root' AND Host = 'localhost';
```

Pamiętaj, aby po tym ponownie załadować tabele uprawnień.

W obu przypadkach powinieneś zobaczyć potwierdzenie, że polecenie zostało wykonane pomyślnie.

```
Query OK, 0 rows affected (0.00 sec)
```

Hasło zostało zmienione, dzięki czemu można teraz zatrzymać ręczną instancję serwera bazy danych i zrestartować go tak, jak wcześniej.

Uruchom ponownie serwer bazy danych w standardowy sposób

Najpierw zatrzymaj instancję serwera bazy danych, którą ręcznie uruchomiłeś w kroku 3. To polecenie wyszukuje PID lub identyfikator procesu MySQL lub MariaDB i wysyła SIGTERM, aby nakazał mu płynne zakończenie działania po wykonaniu operacji czyszczenia.

W przypadku MySQL, użyj polecenia:

```
sudo kill `cat /var/run/mysqld/mysqld.pid`
```

W przypadku MariaDB, skorzystaj z:

```
sudo kill `/var/run/mariadb/mariadb.pid`
```

Następnie uruchom ponownie usługę za pomocą *systemctl*.

MySQL:

```
sudo systemctl start mysql
```

MariaDB:

```
sudo systemctl start mariadb
```

Teraz możesz potwierdzić, że nowe hasło zostało poprawnie zastosowane, uruchamiając:

```
mysql -u root -p
```

Polecenie powinno teraz wyświetlać monit o podanie nowego hasła. Wpisz go, a powinieneś uzyskać dostęp do terminala bazy danych zgodnie z oczekiwaniami.

Podsumowanie

Masz teraz dostęp administratora do przywróconego serwera MySQL lub MariaDB. Upewnij się, że nowe hasło administratora jest silne i bezpieczne i przechowuj je w bezpiecznym miejscu.

8.Klient Mysql HeidiSQL (WINE)