INSTALACJA I ZARZĄDZANIE BAZĄ DANYCH MYSQL

źródło P.DuBois „MySQL”

Instalacja sererwa mysql odbywa się za pomocą polecenia

sudo apt-get install mysql-server mysql-common mysql-client

Później można wykonywać:
#/etc/init.d/mysql start – uruchamia serwer mysql
stop – zatrzymuje serwer mysql
restart – restartuje serwer mysql
Po zainstalowaniu serwera **mysql** otrzymujemy

a) dwie bazy danych:
   - bazę mysql, w której znajdują się tabele uprawnień
   - bazę testową.

b) dwa rodzaje kont:
   - root – konto superużytkownika przeznaczone do celów administracyjnych. To nie jest to samo konto co root systemowy!!! Początkowo to konto jest bez hasła – należy je ustawić!
   - konta bez nazwy użytkownika, tzw. „anonimowe”. Są one niebezpieczne, należy więc je usunąć.

Wszystkie konta znane serwerowi MySQL są wymienione w tabeli **user** bazy danych **mysql**.

Logowanie do bazy:

za pierwszym razem:
#mysql – u root

ustawienie dla konta root hasła:
mysql>SET PASSWORD FOR 'root'@'localhost'=PASSWORD('haslo');

kolejne logowania:
#mysql -u root -p

Po zalogowaniu:
show databases; - pokazuje bazy
use baza; – wybiera bazę
show tables; – pokazuje tabele w bazie
describe tabela; – pokazuje strukturę wybranej tabeli
slect * from tabela; - wyświetla zawartość tabeli.
create database baza; - tworzy bazę
drop database baza; – usuwa bazę
create user konto'@'host identified by 'haslo'; – tworzy konto bez dostępu do bazy danych.
set password for konto'@'host'=password('haslo'); - zmienia hasło.

Po zalogowaniu na zwykłe konto, widać tylko te bazy, do których ma się przypisane uprawnienia (łącznie z bazą testową).

Struktura i zawartość tabel uprawnień w MySQL

Dostęp do bazy danych MySQL kontrolowany jest przez informacje zapisane w tabelach uprawnień. Tabele te zlokalizowane są w bazie mysql. Są to:

a) user
b) tables_priv
c) columns_priv
d) host
e) db

a) Tabela **user** zawiera konta użytkowników, którzy mogą łączyć się z serwerem, ich hasła oraz definiuje ich uprawnienia globalne (superużytkownika), jeśli je mają. Należy pamiętać, że wszystkie uprawnienia definiowane w tabeli user są globalne i z racji tego dotyczą wszystkich baz danych. Jeśli jednej pozycji z tabeli user zostaną przypisane uprawnienia do operacji DELETE, konto z nią stowarzyszone będzie mogło usuwać rekordy z każdej tabeli. Dlatego też należy badzo ostrożnie nadawać tego rodzaju uprawnienia. Tabela user zawiera również kolumny dla opcji ssl, które dotyczą ustanowienia połączeń bezpiecznych przy użyciu protokołu ssl.
b) Tabela **tables_priv** definiuje uprawnienia działające na poziomie tabel. Uprawnienia tutaj zdefiniowane odnoszą się do wszystkich kolumn w danej tabeli.
c) Tabela **columns_priv** definiuje uprawnienia działające na poziomie kolumn. Uprawnienia tutaj definiowane odnosić się będą tylko do określonej kolumny wybranej tabeli.
d) Tabela **db** podaje bazy danych i to, które konta mają uprawnienia sięgania do nich. Uprawnienia definiowane będą się odnosić do wszystkich tabel w określonej bazie danych.
e) Tabela **host** w połączeniu z tabelą db pozwala kontrolować uprawnienia dostępu w zależności od hosta użytkownika, dając większą precyzję definiowania uprawnień niż byłoby to możliwe za pomocą samej tabeli db. Zawartości tej tabeli nie można zmieniać instrukcjami GRANT i REVOKE, dlatego może się zdażyć, że administrator nie będzie nigdy z niej korzystał.

kolumny zakresu dostępu tabel uprawnień

kolumny zakresu dostępu				
tabela user	**tabela db**	**tabela tables_priv**	**tabela columns_priv**	**tabela host**
Host	Host	Host	Host	Host
User	User	User	User	
Password	Db	Db	Db	Db
		Table_name	Table_name	
			Column_name	

kolumny uprawnień tabel uprawnień

kolumny uprawnień administracyjnych		
tabela user	**tabela db**	**tabela host**
Create_tmp_table_priv	Create_tmp_table_priv	Create_tmp_table_priv
Execute_priv		
File_priv		
Grant_priv	Grant_priv	Grant_priv

kolumny uprawnień do baz danych i tabel

kolumny uprawnień do baz danch i tabel		
tabela user	**tabela db**	**tabela host**
Lock_tables_priv	Lock_tables_priv	Lock_tables_priv
Process_priv		
Reload_priv		
Repl_client_priv		
Repl_slave_priv		
Show_db_priv		
Shutdown_priv		
Super_priv		
Alter_priv	Alter_priv	Alter_priv
Create_priv	Create_priv	Create_priv
Delete_priv	Delete_priv	Delete_priv
Drop_priv	Drop_priv	Drop_priv
Index_priv	Index_priv	Index_priv
Insert_priv	Insert_priv	Insert_priv
References_priv	References_priv	References_priv
select_priv	select_priv	select_priv
Update_priv	Update_priv	Update_priv
tabela tables_priv	**tabela columns_priv**	
Table_priv	Column_priv	

kolumny SSL i zarządzania zasobami (tylko tabela user) tabel uprawnień

kolumny ssl	**kolumny zarządzania zasobami**
ssl_type	max_connections
ssl_cipher	max_questions
x509_issuer	max_updates
x509_subject	

Uprawnienia administracyjne odnoszące się do tabel i baz danych:

- ALTER – pozwala użytkownikowi korzystać z instrukcji ALTER TABLE. Jest to wlaściwie tylko wstępne uprawnienie, konieczne jest dodanie kolejnych, aby ustalić, co konkretnie użytkownik może zrobić z tabelą.
- CREATE – umożliwia tworzenie baz danych i tabel. Nie daje uprawnień do tworzenia indeksów tabeli oprócz tych zadeklarowanych wstępnie w instrukcji CREATE TABLE.
- DELETE – pozwala usuwać rekordy z tabel.
- DROP – pozwala usuwać bazy danych i tabele. Nie daje możliwości usuwania indeksów.
- INDEX – umożliwia tworzenie i usuwanie indeksów z tabeli.
- INSERT – pozwala wstawiać do tabel nowe rekordy.
- REFERENES – to uprawnienie jest obecnie nieużwane. Być może będzie wykorzystywane do wskazywania osób, które będą miały możliwość definiowania ograniczeń doyczacych klucza obcego.
- SELECT – pozwala pobierać dane z tabel, korzystając z instrukcji SELET. To uprawnienie nie jest konieczne do wykonywania tych instrukcji Select, które nie dotyczą tabel, takich jak SELECT NOW() lub SELECT 4/2.
- UPDATE – pozwala modyfikować rekordy znajdujące się w tabelach.

Pozostałe uprawnienia

Specyfikator uprawnienia	**Dozwolone operacje**
CREATE TEMPORARY TABLES	Tworzenie tymczasowych tabel
EXECUTE	Wykonywanie przechowywanych procedur (zarezerwowane do użytku w przyszłości)
FILE	Odczytywanie i zapisywanie plików na serwerze
GRANT OPTION	Nadawanie uprawnień danego konta innym kontom
LOCK TABLES	Blokowanie tabel przy użyciu instrukcji LOK TABLES
PROCESS	Przeglądanie informacji na temat wątków działających w serwerze
RELOAD	Ponowne ładowanie tabel uprawnień lub czyszczenie i ponowne ładowanie dzienników, bufora tabeli i pamięci podręcznej hosta
REPLICATION CLIENT	Pytanie o lokalizacje serwera nadrzędnego i podległego
REPLICATION SLAVE	Pytanie o podległego serwera replikacji
SHOW DATABASES	Wydawanie instrukcji SHOW DATABASES

SHUTDOWN	Wyłączanie serwera
SUPER	Usuwanie wątków i wykonywanie innych operacji nadzorczych
ALL	Wszystkie operacje (oprócz GRANT); synonim ALL PRIVILEGES
USAGE	Specjalny specyfikator „bez uprawnień”

ZARZĄDZANIE KONTAMI UŻYTKOWNIKÓW

Polecenia **GRANT** i **REVOKE** pozwalają manipulować zawartością 4 tabel: user,db,tables_priv, columns_priv.

Składnia GRANT

GRANT ***uprawnienia(kolumny)*** **ON** ***do_czego*** **TO** ***konto*** **IDENTIFIED BY „*****haslo*****”**
REQUIRE ***wymogi_dotyczące_szyfrowania***
WITH ***opcje_nadawania_uprawnień_lub_zarządzanie_zasobami*****;**

uprawnienia – przypisane dla danego użytkownika
kolumny – definiuje kolumny, których uprawnienia będą dotyczyć. Wykorzystywany do uprawnień związanych z kolumnami, gdy ma dotyczyć wielu kolumn oddzielamy je przecinkami.
do_czego – czego będą dotyczyć uprawnienia.
konto – definiuje konto, którego uprawnienia dotyczą. Składa się z nazwy konta i nazwy hosta:

'*nazwa_użytkownika*'@'*nazwa_hosta*'

nazwa_użytkownika – kto się łączy, ta nazwa nie musi być taka jak nazwa użytkownika w systemie nazwa_hosta – skąd się łączy.
haslo – podajemy w dokładnej postaci, bez szyfrowania, bo operacje kodowania wykonuje automatycznie polecenie GRANT. Jak IDENTIFIED BY nie został podany, to użytkownikowi nie jest przypisane hasło.
REQUIRE- opcjonalne, dostępne od v. 4.0.0, służy do definiowania kont, które muszą się łączyć przez zabezpieczone łącza przy użyciu protokołu SSL. Nie ma sensu korzystać w przypadku kont, które nie łączą się z serwerem przez sieć zewnętrzną.
WITH – opcjonalne, do nadawania uprawnień GRANT OPTION, które zezwala kontu nadawać uprawnienia innym użytkownikom. Od v. 4.0.2służy też do zarządzania zasobami, np. do nakładania ograniczeń dotyczących liczby, połączeń lub zapytań, jakie konto może dokonać na godzinę.

Nie ma znaczenia, czy nazwy użytkowników, hasła oaz nazwy baz danych i tabel zapisane w tabelach uprawnień są pisane dużymi czy małymi literami.

Oglądanie przyznanych przywilejów poprzez bezpośrednie zaglądanie do tabel systemowych jest dość niewygodne i na dodatek łatwo o pomyłkę. Dużo wygodniejsze jest posługiwanie się poleceniem **SHOW GRANTS**, które w przejrzysty sposób pokazuje aktualnie przyznane przywileje dla wybranego użytkownika, przykładowo:
mysql> SHOW GRANTS FOR ’konto’@’host’;

Przykład 1
udostępnianie wszystkich tabel w bazie danych dla kont o określonym hoście:

GRANT ALL ON baza.* TO 'konto'@'host' IDENTIFIED BY 'haslo';

host – localhsot, student.zsl.gda.pl, % (oznacza dowolny host, najmniej bezpieczne)

od v. 3.23 można też w host podawać adres IP, np.
'konto'@'192.168.128.3'
'konto'@'192.168.128.%'
'konto'@'192.168.128.0/255.255.128.0'

'konto' i 'konto'@'%' sącałkowicie równoważne.
gdyby ktoś napisał 'konto@localhost', to MySQL uznałby całość jako nazwę użytkownika i dodał sobie znak %.

Specyfikator uprawnienia	**Poziom, do którego stosowane są uprawnienia**
ON *.*	Uprawnienia globalne; wszystkie bazy danych, wszystkie tabele
ON *	Uprawnienia globalne; jeśli nie wybrano domyślnej bazy danych; w przeciwnym razie uprawnienia na poziomie bazy danych dla bieżącej bazy danych.
ON nazwa_bazy.*	Uprawnienia na poziomie bazy danych; wszystkie tabele w nazwanej bazie danych.
ON nazwa_bazy.nazwa_nazwa_tabeli	Uprawnienia poziomu tabeli; wszystkie kolumny w nazwanej tabeli.
ON nazwa_tabeli	Uprawnienia poziomu tabeli; wszystkie kolumny w nazwanej tabeli w domyślnej bazie danych.

Przykład 2
nadawanie uprawnień do kontrolowania dostępu do bazy danych
na końcu dodaje się opcję WITH GRANT OPTION, np.

GRANT ALL ON baza.* TO 'konto'@'host' IDENTIFIED BY 'haslo'
WITH GRANT OPTION;

lub

GRANT GRANT OPTION ON baza.* TO 'konto'@'host';

Przykład 3
ograniczenie zużycia zasobów przez użytkownika
np.
...WITH MAX_ONNETIONS_PER_HOUR 10 MAX_QUERIES_PER_HOUR 200 MAX_UPDATES_PER_HOUR 50;
oznacza to, że użytkownik może łączyć się tylko 10 razy na godzinę, wydać w tym czasie 200 zapytań (z czego najwżej 50 może być aktualizacjami).

Wewnątrz klauzuli kolejność opcji zarządzania zasobami nie ma znaczenia. Domyślną wartością każdej opcji jest zero, co oznacza „bez ograniczenia”.

Przykład 4
odbieranie uprawnień i usuwanie użytkowników

REVOKE *uprawnienia(kolumny)* ON *do_czego* FROM *konto*;

parametr ***konto*** odpowiada kontu z instrukcji GRANT.
Uprawnienia nie muszą odpowiadać dokładnie tym z GRANT, można instrukcją GRANT nadać kilka uprawnień, a następnie instrukcją REVOKE odebrać niektóre z nich, np.
GRANT ALL ...
a potem
REVOKE DELETE, UPDATE ON baza.* FROM 'konto'@'host';

Przykład 5

usunięcie uprawnień GRANT OPTION
REVOKE GRANT OPTION ON...

REVOKE usuwa tylko uprawnienia a nie całe konta!!!. Aby usunąć konto, należy usunąć jego rekord z tabeli user specjalną instrukcją DELETE.

#mysql -u root -p
mysql>USE mysql;
mysql>DELETE FROM user WHERE user='użytkownik' and host='host';
mysql>FLUSH PRIVILEGES;

FLUSH PRIVILEGES – przeładowanie uprawnień, jest to niezbędne, ponieważ serwer ładuje automatycznie tabele na nowo po zastosowaniu instrukcji GRANT lub REVOKE, ale nie odświeża ich w przypadku bezpośredniej modyfikacji (warto sprawdzić przed zastosowaniem instrukcji FLUSH, czy rekordy związane z tym kontem nie występują w winnych tabelach uprawnień i je usunąć).

MySQL – instalacja sererwa mysql oraz zarządzanie użytkownikami bazy danych

Zadanie 1
Zainstaluj serwer mysql.
Zadanie 2
Zaloguj się do serera mysql, ustaw dla konta root haslo, a następnie wykonaj poniższe ćwiczenia.
UWAGA!!! - do wykonanych ćwiczeń wykonaj sprawozdanie. W sprawozdaniu mają się znajdować polecenia, które użyłeś do wykonania danego ćwiczenia oraz wygląd tabel z bazy mysql. Ponadto zanim cokolwiek usuniesz, zmodyfikujesz itp. zawołaj prowadzącego zajęcia.

Ćwiczenie 1
Utworzyć 4 konta użytkowników o nazwach **test1**, **test2**, **test3**, **root2**.
Użytkownicy **test1** oraz **test2** nie powinni mieć nadanych żadnych uprawnień natomiast użytkownik **test3** powinien mieć nadane pełne prawa do bazy **baza3** (baza o tej nazwie nie musi w tym momencie fizycznie istnieć). Ostatnie konto (**root2**) ma mieć uprawnienia administratora (takie same jakie posiada użytkownik root).
Ćwiczenie 2
Utworzyć dwie nowe bazy danych o nazwach **baza1** oraz **baza2**.
Ćwiczenie 3
Nadać pełne prawa do bazy **baza1** dla użytkownika test1 oraz pełne prawa do bazy **baza2** dla

użytkownika **test2**.

Ćwiczenie 4

Zmienić hasła użytkownikom **test1** oraz **test2** na dowolne inne.

Ćwiczenie 5

Odebrać użytkownikowi **test3** wszystkie uprawnienia do bazy **baza3**.

Ćwiczenie 6

Utworzyć 4 tabele (struktura tabel może być zupełnie dowolna). Dwie w bazie **baza1** oraz dwie w bazie **baza2**. Użytkownikowi **test1** nadać prawa SELECT, INSERT, UPDATE, DELETE do pierwszej tabeli użytkownika **test2** oraz prawo SELECT do jednej wybranej kolumny w drugiej tabeli użytkownika **test2**. Użytkownikowi **test2** nadać pełne prawa do wszystkich obiektów w bazie **baza1**.

Ćwiczenie 7

Za pomocą polecenia SHOW GRANTS wyświetlić uprawnienia wszystkich użytkowników.

Ćwiczenie 8

Zapoznać się z aktualną zawartością tabel przywilejów (user, db, tables priv, columns priv).