5.3 Управління правами користувачів / Manage user privileges

СУБД MySQL є багатокористувацьким середовищем, тому для доступу до таблиць БД можуть бути створені різні облікові записи з різним рівнем привілеїв

MySQL is a multi-user environment, so different accounts with different levels of privileges can be created to access the database tables

Обліковому запису користувача можна надати привілеї на перегляд таблиці, додавання нових записів і оновлення вже існуючих

The user's account can be granted privileges to view the table, add new entries, and update existing ones

Адміністратору БД можна надати більш широкі повноваження (можливість створення таблиць, редагування та видалення вже існуючих)

The DBA can be given greater authority (the ability to create tables, edit and delete existing ones)

5.3 Управління правами користувачів / Manage user privileges

Для гостя досить лише перегляду таблиць

For a guest, just viewing the tables is enough

Розглянемо наступні питання / Let’s consider the following questions:

- створення, редагування і видалення облікових записів користувачів
- create, edit and delete user accounts

- призначення і скасування привілеїв
- assignment and cancellation of privileges

5.3 Управління правами користувачів / Manage user privileges

Обліковий запис є складовою і приймає форму 'username' @ 'host', де username – ім'я користувача, а host – найменування хоста, з якого користувач може звертатися до сервера

The account has composite structure and takes the form of 'username' @ 'host', where *username* is the name of the user, and *host* is the name of the host from which the user can access the server

Наприклад, записи 'root' @ '127.0.0.1' і 'wet' @ '62 .78.56.34 ' означають, що користувач з ім'ям root може звертатися з хоста, на якому розташований сервер, а wet – тільки з хоста з IP-адресою 62.78 .56.34

For example, the entries 'root' @ '127.0.0.1' and 'wet' @ '62.78.56.34' mean that the user with the name *root* can access from the host where the server is located, and *wet* – only from the host with IP address 62.78 .56.34

5.3 Управління правами користувачів / Manage user privileges

IP-адреса 127.0.0.1 завжди відноситься до локального хосту

IP address 127.0.0.1 always refers to the local host

Якщо сервер і клієнт встановлені на одному хості, то сервер слухає з'єднання за цією адресою, а клієнт відправляє на нього SQL-запити

If the server and client are installed on the same host, the server listens for connections to this address, and the client sends SQL queries to it

IP-адреса 127.0.0.1 має псевдонім localhost, тому облікові записи виду 'root' @ '127.0.0.1' можна записувати у вигляді 'root' @ 'localhost‘

The IP address 127.0.0.1 has an alias of *localhost*, so accounts like 'root' @ '127.0.0.1' can be written as 'root' @ 'localhost'

5.3 Управління правами користувачів / Manage user privileges

Число адрес, з яких необхідно забезпечити доступ користувачеві, може бути значним

The number of addresses from which user access should be provided can be significant

Для завдання діапазону в імені хоста використовується спеціальний символ "%"

The special character "%" is used to set the range in the host name

Так, обліковий запис 'wet' @ '%' дозволяє користувачеві wet звертатися до сервера MySQL з будь-яких комп'ютерів мережі

So, the 'wet' @ '%' account allows the *wet* user to access the MySQL server from any network computers

5.3 Управління правами користувачів / Manage user privileges

Усі облікові записи зберігаються в таблиці user системної бази даних з ім'ям mysql

All accounts are stored in the *user* table of the system database named *mysql*

```
mysql> SELECT Host,User,Password FROM mysql.user;
+-----------------------+------+----------+
| Host                  | User | Password |
+-----------------------+------+----------+
| localhost             | root |          |
| production.mysql.com  | root |          |
| 127.0.0.1             | root |          |
| localhost             |      |          |
| production.mysql.com  |      |          |
+-----------------------+------+----------+
5 rows in set (0.27 sec)
```

5.3 Управління правами користувачів / Manage user privileges

CREATE USER 'username' @ 'host'
[**IDENTIFIED BY** [**PASSWORD**] 'password'];

Оператор створює новий обліковий запис з необов'язковим паролем
The operator creates a new account with an optional password

Якщо пароль не вказано, в його якості виступає порожній рядок
If the password is not specified, an empty string is used as the password

Розумно зберігати пароль у вигляді хеш-коду, отриманого в результаті незворотного шифрування
It is reasonable to store the password in the form of a hash code obtained from irreversible encryption

5.3 Управління правами користувачів / Manage user privileges

Щоб скористатися цим механізмом шифрування, необхідно помістити між ключовим словом IDENTIFIED BY і паролем ключове слово PASSWORD

To use this encryption mechanism, you should place the keyword PASSWORD between the IDENTIFIED BY keyword and the password

DROP USER 'username' @ 'host';

Даний оператор дозволяє видалити обліковий запис

This operator allows you to delete an account

Зміна імені користувача в обліковому записі здійснюється за допомогою оператора

Use the operator to change the username of the account

RENAME USER old_name **TO** new_name;

5.3 Управління правами користувачів / Manage user privileges

Розглянуті вище оператори дозволяють створювати, видаляти і редагувати облікові записи, але вони не дозволяють змінювати привілеї користувача – повідомляти MySQL, який користувач має право тільки на читання інформації, який на читання і редагування, а кому надані права змінювати структуру БД і створювати облікові записи

The above operators allow you to create, delete and edit accounts, but they do not allow changing user privileges – tell MySQL which user has the right to read information only, which one to read and edit, and who has the right to change the database structure and create accounts

5.3 Управління правами користувачів / Manage user privileges

Для вирішення цих завдань призначені оператори **GRANT** (призначає привілеї) і **REVOKE** (видаляє привілеї)

The **GRANT** (assigns privileges) and **REVOKE** (deletes privileges) statements are intended for solving these tasks

Якщо облікового запису, який показаний в операторі **GRANT**, не існує, то він автоматично створюється

If the account shown in the **GRANT** statement does not exist, it is automatically created

Видалення всіх привілеїв за допомогою оператора **REVOKE** не приводить до автоматичного знищення облікового запису

Removing all privileges using the **REVOKE** statement does not automatically destroy the account

5.3 Управління правами користувачів / Manage user privileges

У найпростішому випадку оператор **GRANT** виглядає наступним чином

In the simplest case, the GRANT statement looks like this

```
mysql> GRANT ALL ON *.* TO 'wet'@'localhost' IDENTIFIED BY 'pass';
Query OK, 0 rows affected (0.17 sec)
```

Даний запит створює користувача з ім'ям wet і паролем pass, який може звертатися до сервера з локального хоста (localhost) і має всі права (ALL) для всіх баз даних (*. *)

This query creates a user with the name *wet* and a password *pass*, which can access the server from the local host (*localhost*) and has all rights (*ALL*) for all databases (*. *)

Якщо такий користувач існує, то його привілеї будуть змінені на ALL

If such a user exists, his privileges will be changed to ALL

5.3 Управління правами користувачів / Manage user privileges

Ключове слово ON в операторі GRANT задає рівень привілеїв, які можуть бути задані на одному з чотирьох рівнів

The ON keyword in the GRANT statement sets the level of privileges that can be set at one of four levels

Для таблиць можна встановити тільки такі типи привілеїв:

SELECT, **INSERT**, **UPDATE**, **DELETE**, **CREATE**, **DROP**, **GRANT OPTION**, **INDEX** и **ALTER**

Only the following types of privileges can be set for tables: **SELECT**, **INSERT**, **UPDATE**, **DELETE**, **CREATE**, **DROP**, **GRANT OPTION**, **INDEX** and **ALTER**

Це слід враховувати при використанні конструкції GRANT ALL, яка призначає привілеї на поточному рівні

This should be considered when using the GRANT ALL construct, which assigns privileges at the current level

5.3 Управління правами користувачів / Manage user privileges

Так, запит рівня бази даних GRANT ALL ON db. * не надає ніяких глобальних привілеїв

So the database level query GRANT ALL ON db. * does not provide any global privileges

Для скасування привілеїв використовується оператор **REVOKE**

To cancel privileges, use the **REVOKE** operator

```
mysql> REVOKE DELETE, UPDATE ON *.* FROM 'wet'@'localhost';
Query OK, 0 rows affected (0.02 sec)
```

Оператор REVOKE скасовує привілеї, але не видаляє облікові записи

Operator REVOKE revokes privileges, but does not delete accounts

5.3 Управління правами користувачів / Manage user privileges

Privilege	Description
ALL [PRIVILEGES]	Комбінація всіх привілеїв, за винятком привілеї GRANT OPTION, яка задається окремо Combination of all privileges, except GRANT OPTION privilege, which is specified separately
ALTER	Дозволяє редагувати таблицю за допомогою оператора ALTER TABLE Allows you to edit a table using the ALTER TABLE statement
ALTER ROUTINE	Дозволяє редагувати або видаляти збережену процедуру Allows you to edit or delete a stored procedure
CREATE	Дозволяє створювати таблицю за допомогою оператора CREATE TABLE Allows you to create a table using the operator CREATE TABLE
CREATE ROUTINE	Дозволяє створювати збережену процедуру Allows you to create a stored procedure

5.3 Управління правами користувачів / Manage user privileges

Privilege	Description
CREATE TEMPORARY TABLES	Дозволяє створювати тимчасові таблиці Allows you to create temporary tables
CREATE USER	Дозволяє працювати з обліковими записами с допомогою CREATE USER, DROP USER, RENAME USER і REVOKE ALL PRIVILEGES Allows you to work with accounts using CREATE USER, DROP USER, RENAME USER and REVOKE ALL PRIVILEGES
CREATE VIEW	Дозволяє створювати уявлення за допомогою оператора CREATE VIEW Allows you to create a view using the CREATE VIEW statement
DELETE	Дозволяє видаляти записи за допомогою оператора DELETE Allows you to delete records using the operator DELETE
DROP	Дозволяє видаляти таблиці за допомогою оператора DROP TABLE Allows you to delete tables using the DROP TABLE statement

5.3 Управління правами користувачів / Manage user privileges

Privilege	Description
EXECUTE	Дозволяє виконувати збережені процедури Allows you to execute stored procedures
INDEX	Дозволяє працювати з індексами, зокрема, використовувати оператори CREATE INDEX і DROP INDEX Allows you to work with indexes, in particular, to use the operators CREATE INDEX and DROP INDEX
INSERT	Дозволяє додавати в таблицю нові записи оператором INSERT Allows you to add new entries to the table using the INSERT statement
LOCK TABLES	Дозволяє здійснювати блокування таблиць за допомогою операторів LOCK TABLES і UNLOCK TABLES Allows locking tables using the LOCK TABLES and UNLOCK TABLES statements

5.3 Управління правами користувачів / Manage user privileges

Privilege	Description
SELECT	Дозволяє здійснювати вибірки таблиць оператором SELECT Allows table selection with a SELECT statement
SHOW DATABASES	Дозволяє переглядати список всіх таблиць на сервері за допомогою оператора SHOW DATABASES Allows you to view a list of all tables on the server using the operator SHOW DATABASES
SHOW VIEW	Дозволяє використовувати оператор SHOW CREATE VIEW Allows the use of the SHOW CREATE VIEW statement
UPDATE	Дозволяє оновлювати вміст таблиць оператором UPDATE Allows updating table contents with UPDATE statement
USAGE	Синонім для статусу «відсутні привілеї» Synonym for “missing privileges” status
GRANT OPTION	Дозволяє управляти привілеями інших користувачів, без цього привілею неможливо виконати оператори GRANT I REVOKE Allows you to manage the privileges of other users, without this privilege it is impossible to execute GRANT and REVOKE statements

5.3 Управління правами користувачів / Manage user privileges

ON Keyword	Level
ON *.*	Глобальний рівень - користувач із правами на глобальному рівні може звертатися до всіх БД і таблиць, що входять до їх складу Global level – a user with authority at the global level can access all databases and tables included in them
ON db.*	Рівень бази даних - привілеї поширюються на таблиці бази даних db Database Level - Privileges apply to *db* database tables
ON db.tbl	Рівень таблиці - привілеї поширюються на таблицю tbl бази даних db Table Level - Privileges apply to the *tbl* table of the *db* database
ON db.tbl	Рівень стовпчика - привілеї стосуються окремих стовпців в таблиці tbl бази даних db. Список стовпців вказується в дужках через кому після ключових слів SELECT, INSERT, UPDATE Column Level — Privileges relate to individual columns in the *tbl* table of the *db* database. The list of columns is indicated in parentheses, separated by commas after the keywords SELECT, INSERT, UPDATE