

Безопасность баз данных и администрирование

Создание новых пользователей MySQL с различными уровнями доступа:

```
user@ubuntu-server:~$ sudo mysql -u root -p
[sudo] password for user:
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 15
Server version: 8.0.39-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

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

mysql>
```

Администратор базы данных

Создание пользователя с полными правами на определенную базу данных:

```
mysql> CREATE USER 'admin_db'@'%' IDENTIFIED BY '5555';
Query OK, 0 rows affected (0.05 sec)

mysql> GRANT ALL PRIVILEGES ON test_db.* TO 'admin_db'@'%' WITH GRANT OPTION;
Query OK, 0 rows affected (0.01 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.01 sec)

mysql> _
```

Пользователь с правом только на чтение из определенной базы данных

```
mysql> CREATE USER 'read_user'@'%' IDENTIFIED BY '7777';
Query OK, 0 rows affected (0.04 sec)

mysql> GRANT SELECT ON test_db.* TO 'read_user'@'%;
Query OK, 0 rows affected (0.01 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.01 sec)

mysql> _
```

Пользователь с правами на добавление и удаление данных в определённой таблице

Создание пользователя, который имеет права на добавление (INSERT) и удаление (DELETE) данных в определённой таблице (users):

```
mysql> CREATE USER 'table_user'@'%' IDENTIFIED BY '8888';
Query OK, 0 rows affected (0.04 sec)

mysql> GRANT SELECT, INSERT, DELETE ON test_db.users TO 'table_user'@'%;
Query OK, 0 rows affected (0.01 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.01 sec)

mysql> _
```

Проверка созданных пользователей и их прав

Показать всех пользователей:

```
mysql> SELECT User, HOST FROM mysql.user;
+-----+-----+
| User          | HOST          |
+-----+-----+
| admin_db      | %             |
| read_user     | %             |
| table_user    | %             |
| testuser      | %             |
| debian-sys-maint | localhost    |
| mysql.infoschema | localhost    |
| mysql.session | localhost    |
| mysql.sys     | localhost    |
| root          | localhost    |
+-----+-----+
9 rows in set (0.00 sec)

mysql> _
```

Проверить привилегии конкретного пользователя:

```
mysql> SHOW GRANTS FOR 'admin_db'@'%;
+-----+-----+
| Grants for admin_db@% |
+-----+-----+
| GRANT USAGE ON *.* TO `admin_db`@`%` |
| GRANT ALL PRIVILEGES ON `test_db`.* TO `admin_db`@`%` WITH GRANT OPTION |
+-----+-----+
2 rows in set (0.01 sec)

mysql> _
```

Настройка ролей и назначение прав

Создание роли для администратора базы данных

```
mysql> CREATE ROLE 'admin_role';
Query OK, 0 rows affected (0.01 sec)
```

Создание роли только для чтения

```
mysql> CREATE ROLE 'read_role';
Query OK, 0 rows affected (0.01 sec)
```

Создание роли для изменения данных

```
mysql> CREATE ROLE 'modifier_role';
Query OK, 0 rows affected (0.01 sec)
```

Назначение прав для ролей

Назначение прав для роли admin-role

```
mysql> GRANT ALL PRIVILEGES ON test_db.* TO 'admin_role';
Query OK, 0 rows affected (0.05 sec)
```

Назначение прав для роли read_role

```
mysql> GRANT SELECT ON test_db.* TO 'read_role';
Query OK, 0 rows affected (0.01 sec)
```

Назначение прав для роли modifier_role

```
mysql> GRANT INSERT, DELETE ON test_db.* TO 'modifier_role';
Query OK, 0 rows affected (0.02 sec)
```

Назначение ролей пользователям

Теперь можно назначить созданные роли конкретным пользователям, например:

```
mysql> CREATE USER 'vasya'@'%' IDENTIFIED BY 'vasya';
Query OK, 0 rows affected (0.04 sec)

mysql> GRANT 'admin_db' TO 'vasya'@'%';
Query OK, 0 rows affected (0.01 sec)

mysql> _
```

Настройка доступа к порту MySQL только с определённого IP

Узнаем свой адрес:

```
C:\Users\user>ipconfig

Настройка протокола IP для Windows

Адаптер Ethernet Ethernet:

    DNS-суффикс подключения . . . . . : 
    Локальный IPv6-адрес канала . . . : fe80::9a9:8b57:93b:7b4a%11
    IPv4-адрес. . . . . : 192.168.1.183
    Маска подсети . . . . . : 255.255.255.0
    Основной шлюз. . . . . : fe80::1%11
                             192.168.1.1
```

Настроим подключение только с этого IP:

```
user@ubuntu-server:~$ sudo ufw allow from 192.168.1.183 to any port 3306
Rule added
user@ubuntu-server:~$ sudo ufw deny 3306
Rule added
Rule added (v6)
user@ubuntu-server:~$
```

Проверим правила, которые настроены в UFW:

```
user@ubuntu-server:~$ sudo ufw status
Status: active

To Action From
--
3306/tcp ALLOW Anywhere
22/tcp ALLOW Anywhere
3306 ALLOW 192.168.1.183
3306 DENY Anywhere
3306/tcp (v6) ALLOW Anywhere (v6)
22/tcp (v6) ALLOW Anywhere (v6)
3306 (v6) DENY Anywhere (v6)

user@ubuntu-server:~$
```

Это означает, что доступ к порту 3306 разрешён только с IP-адреса 192.168.1.183, и для всех остальных он закрыт.

Убедимся, что MySQL работает и слушает подключения:

```
user@ubuntu-server:~$ sudo netstat -tulnp | grep mysql
tcp        0      0 127.0.0.1:33060      0.0.0.0:*             LISTEN      32078/mysqld
tcp        0      0 0.0.0.0:3306         0.0.0.0:*             LISTEN      32078/mysqld
user@ubuntu-server:~$
```

Проверим, что Mysql Workbanch работает:

MySQL Workbench

UbuntuMysql x MySQL Model* x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

test_db

Tables

cities

users

Columns

- id_user
- name
- surname
- city_id

Indexes

Foreign Keys

Administration Schemas

Information

Table: users

Columns:

- id_user int AI PK
- name varchar(64)
- surname varchar(45)
- city_id int

Query 1 cities - Table users - Table cities users x

```
1 • USE test_db;
2 • SELECT * FROM users;
3 • SELECT COUNT(*) from users;
4 • INSERT INTO users (name, surname, city_id) VALUES ('Ivan',
```

Limit to 1000 rows

Result Grid

	id_user	name	surname	city_id
▶	1	Будимир	Феликсович	12574
	2	Иванна	Робертовна	21011
	3	Евфросиния	Федоровна	24987
	4	Тарас	Гавриилович	19797
	5	Юлия	Валентиновна	31467

Настройка завершена.