

# **Отчет по лабораторной работе №6**

**Установка и настройка системы управления базами данных MariaDB**

Галацан Николай, НПИбд-01-22

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>5</b>
2.1	Установка MariaDB . . . . .	5
2.2	Конфигурация кодировки символов . . . . .	7
2.3	Создание базы данных . . . . .	8
2.4	Резервные копии . . . . .	11
2.5	Внесение изменений в настройки внутреннего окружения виртуальной машины . . . . .	12
<b>3</b>	<b>Выводы</b>	<b>14</b>
<b>4</b>	<b>Ответы на контрольные вопросы</b>	<b>15</b>

# Список иллюстраций

2.1	Просмотр конфигурационных файлов . . . . .	5
2.2	Прослушивание порта 3306 . . . . .	6
2.3	Вход в БД и просмотр списка команд . . . . .	6
2.4	Имеющиеся в системе БД . . . . .	7
2.5	Статус MariaDB . . . . .	7
2.6	Редактирование файла /etc/my.cnf.d/utf8.cnf . . . . .	8
2.7	Статус MariaDB после конфигурации кодировки символов . . . . .	8
2.8	Создание БД addressbook и таблицы city . . . . .	9
2.9	Вставка данных в таблицу . . . . .	9
2.10	Просмотр таблицы, создание пользователя, предоставление прав, обновление привилегий . . . . .	10
2.11	Общая информация о таблице . . . . .	10
2.12	Список БД, список таблиц БД addressbook . . . . .	11
2.13	Созданные резервные копии, восстановление резервных копий . .	11
2.14	Создание скрипта mysql.sh . . . . .	12

# 1 Цель работы

Приобретение практических навыков по установке и конфигурированию системы управления базами данных на примере программного обеспечения MariaDB.

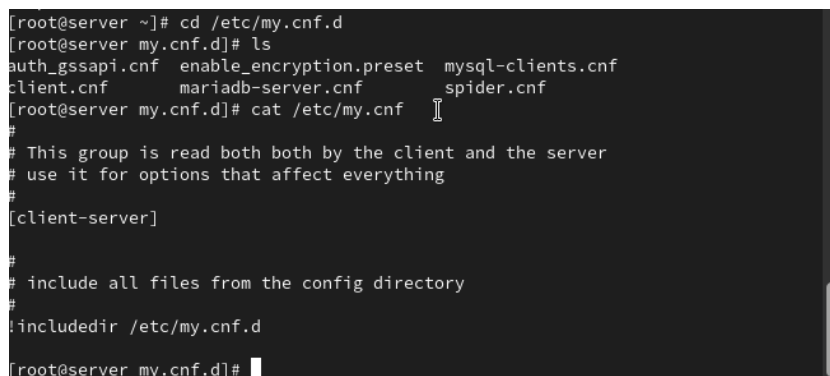
## 2 Выполнение лабораторной работы

### 2.1 Установка MariaDB

Запускаю VM через рабочий каталог. На VM server захожу под собственным пользователем и перехожу в режим суперпользователя. Устанавливаю необходимые пакеты:

```
dnf -y install mariadb mariadb-server
```

Просматриваю конфигурационные файлы mariadb в каталоге /etc/my.cnf.d и в файле /etc/my.cnf (рис. 2.1).



```
[root@server ~]# cd /etc/my.cnf.d
[root@server my.cnf.d]# ls
auth_gssapi.cnf  enable_encryption.preset  mysql-clients.cnf
client.cnf       mariadb-server.cnf        spider.cnf
[root@server my.cnf.d]# cat /etc/my.cnf
#
# This group is read both both by the client and the server
# use it for options that affect everything
#
[client-server]
#
# include all files from the config directory
#
includedir /etc/my.cnf.d
[root@server my.cnf.d]#
```

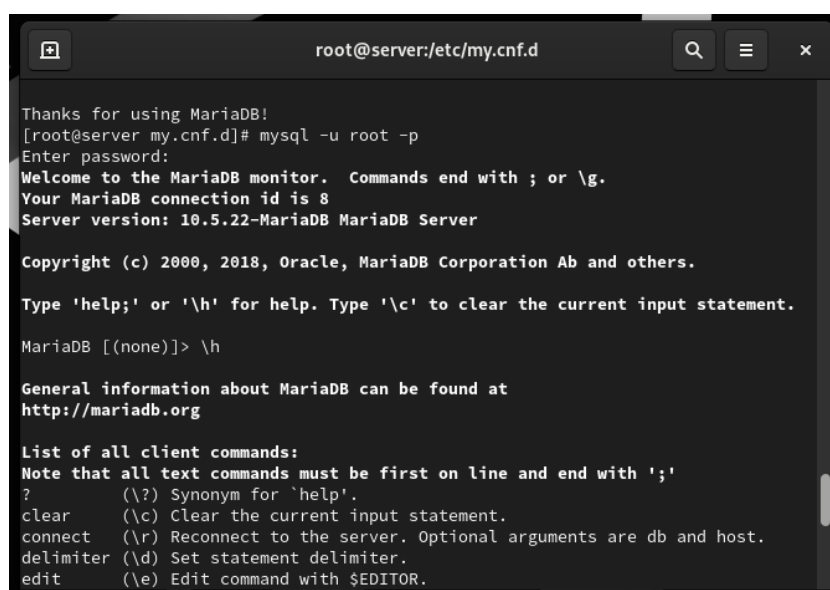
Рис. 2.1: Просмотр конфигурационных файлов

Запускаю и включаю ПО mariadb. Убеждаюсь, что прослушивается порт (рис. 2.2)

```
[root@server my.cnf.d]# ss -tulpen | grep 3306
tcp    LISTEN 0      80          *:3306      *:~         users:(("mariadb",pid=9966,fd=19)) uid:27 ino:44553 sk:14 cgroup:/system.slice/mariadb.service
v6only:0 <->
[root@server my.cnf.d]#
```

Рис. 2.2: Прослушивание порта 3306

Запускаю скрипт конфигурации безопасности `mysqld(mysql_secure_installation)`. Устанавливаю пароль для пользователя `root` базы данных, отключаю удалённый корневой доступ и удаляю тестовую базу данных и любых анонимных пользователей. Вхожу в базу данных с правами администратора и просматриваю список команд (рис. 2.3)



```
root@server:/etc/my.cnf.d
Thanks for using MariaDB!
[root@server my.cnf.d]# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 8
Server version: 10.5.22-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> \h

General information about MariaDB can be found at
http://mariadb.org

List of all client commands:
Note that all text commands must be first on line and end with ';'
?      (\?) Synonym for 'help'.
clear  (\c) Clear the current input statement.
connect (\r) Reconnect to the server. Optional arguments are db and host.
delimiter (\d) Set statement delimiter.
edit   (\e) Edit command with $EDITOR.
```

Рис. 2.3: Вход в БД и просмотр списка команд

Просматриваю имеющиеся базы данных, введя запрос, после чего выхожу из оболочки. В системе есть БД `information_schema`, `mysql`, `performance_schema` (рис. 2.4)

```
MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
+-----+
3 rows in set (0.001 sec)

MariaDB [(none)]> exit;
Bye
[root@server my.cnf.d]#
```

Рис. 2.4: Имеющиеся в системе БД

## 2.2 Конфигурация кодировки символов

Войдя в БД с правами администратора, просматриваю статус (рис. 2.5).

```
root@server:~
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> status
-----
mysql Ver 15.1 Distrib 10.5.22-MariaDB, for Linux (x86_64) using EditLine wrap
per

Connection id:          9
Current database:
Current user:            root@localhost
SSL:                    Not in use
Current pager:          stdout
Using outfile:           ''
Using delimiter:        ;
Server:                 MariaDB
Server version:         10.5.22-MariaDB MariaDB Server
Protocol version:       10
Connection:             Localhost via UNIX socket
Server characterset:    latin1
Db characterset:        latin1
Client characterset:    utf8
Conn. characterset:     utf8
UNIX socket:            /var/lib/mysql/mysql.sock
Uptime:                 8 min 30 sec
```

Рис. 2.5: Статус MariaDB

В каталоге /etc/my.cnf.d создаю файл utf8.cnf и редактирую его (рис. 2.6).

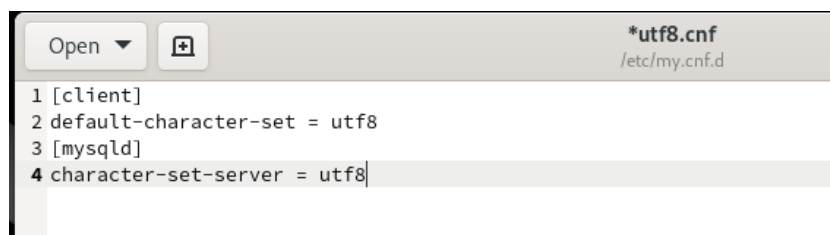


Рис. 2.6: Редактирование файла /etc/my.cnf.d/utf8.cnf

Перезапустив MariaDB, вхожу в БД и просматриваю статус. Наблюдаю изменения в Server characterset и Db characterset. Теперь поддерживаются не только латинские символы, но и кириллица (рис. 2.7).

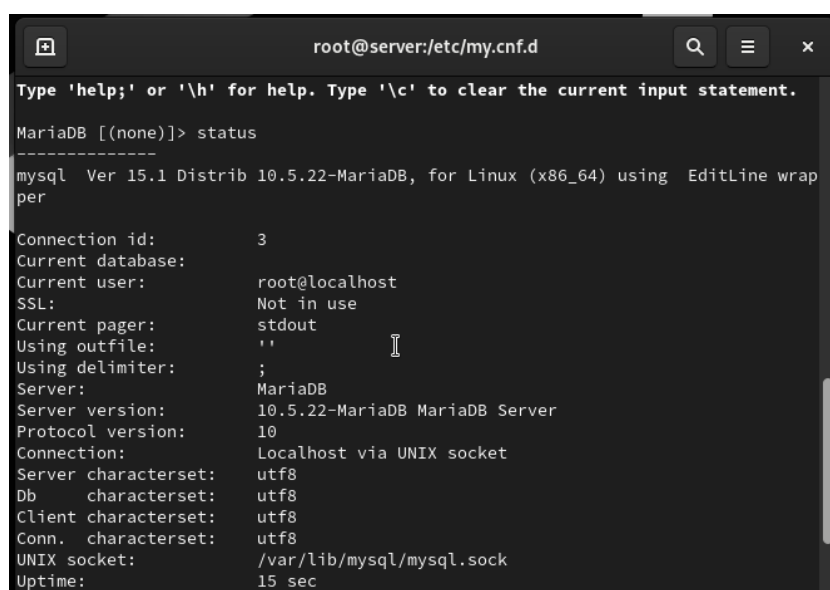
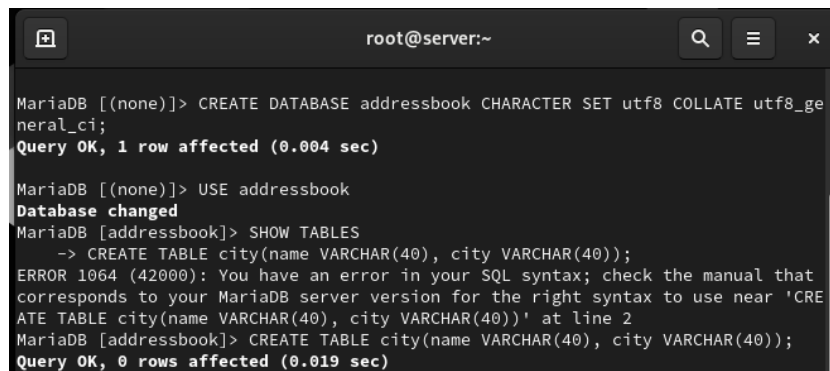


Рис. 2.7: Статус MariaDB после конфигурации кодировки символов

## 2.3 Создание базы данных

Вхожу в БД с правами администратора. Создаю БД addressbook и перехожу к ней. Создаю таблицу city с полями name и city (рис. 2.8).

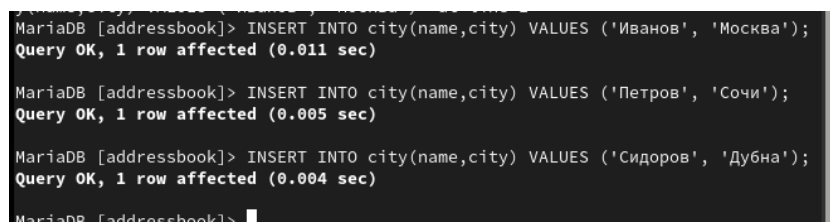




```
root@server:~  
MariaDB [(none)]> CREATE DATABASE addressbook CHARACTER SET utf8 COLLATE utf8_general_ci;  
Query OK, 1 row affected (0.004 sec)  
  
MariaDB [(none)]> USE addressbook  
Database changed  
MariaDB [addressbook]> SHOW TABLES  
-> CREATE TABLE city(name VARCHAR(40), city VARCHAR(40));  
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near 'CREATE TABLE city(name VARCHAR(40), city VARCHAR(40))' at line 2  
MariaDB [addressbook]> CREATE TABLE city(name VARCHAR(40), city VARCHAR(40));  
Query OK, 0 rows affected (0.019 sec)
```

Рис. 2.8: Создание БД addressbook и таблицы city

Заполняю таблицу (рис. 2.9).



```
MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Иванов', 'Москва');  
Query OK, 1 row affected (0.011 sec)  
  
MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Петров', 'Сочи');  
Query OK, 1 row affected (0.005 sec)  
  
MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Сидоров', 'Дубна');  
Query OK, 1 row affected (0.004 sec)  
  
MariaDB [addressbook]>
```

Рис. 2.9: Вставка данных в таблицу

Ввожу запрос `SELECT * FROM city;`. Выводятся все данные из только что созданной таблицы. Далее создаю пользователя, предоставляю права доступа, обновляю привилегии (рис. 2.10).

```
root@server:~  
  
Database changed  
MariaDB [addressbook]> SELECT * FROM city;  
+-----+-----+  
| name      | city      |  
+-----+-----+  
| Иванов    | Москва    |  
| Петров    | Сочи      |  
| Сидоров   | Дубна     |  
+-----+-----+  
3 rows in set (0.012 sec)  
  
MariaDB [addressbook]> CREATE USER ngalacan@'%' IDENTIFIED BY '123456';  
Query OK, 0 rows affected (0.017 sec)  
  
MariaDB [addressbook]> GRANT SELECT, INSERT, UPDATE,DELETE ON addressbook.* TO n  
galacan@'%';  
Query OK, 0 rows affected (0.005 sec)  
  
MariaDB [addressbook]> FLUSH PRIVILEGES  
-> ;  
Query OK, 0 rows affected (0.004 sec)  
  
MariaDB [addressbook]> FLUSH PRIVILEGES;
```

Рис. 2.10: Просмотр таблицы, создание пользователя, предоставление прав, обновление привилегий

Просматриваю общую информацию о таблице (рис. 2.11).

```
MariaDB [addressbook]> DESCRIBE city;  
+-----+-----+-----+-----+-----+-----+  
| Field | Type      | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| name  | varchar(40) | YES  |     | NULL    |       |  
| city  | varchar(40) | YES  |     | NULL    |       |  
+-----+-----+-----+-----+-----+-----+  
2 rows in set (0.005 sec)  
  
MariaDB [addressbook]> quit  
Bye  
[root@server ~]#
```

Рис. 2.11: Общая информация о таблице

Просматриваю список баз данных (к имеющимся ранее БД добавилась addressbook), а также список таблиц БД addressbook (рис. 2.12)

```
[root@server ~]# mysqlshow -u root -p
Enter password:
+-----+
| Databases |
+-----+
| addressbook |
| information_schema |
| mysql |
| performance_schema |
+-----+
[root@server ~]# mysqlshow -u root -p addressbook
Enter password:
Database: addressbook
+-----+
| Tables |
+-----+
| city |
+-----+
[root@server ~]#
```

Рис. 2.12: Список БД, список таблиц БД addressbook

## 2.4 Резервные копии

Создаю каталог /var/backup для резервных копий. Создаю резервную копию БД addressbook, сжатую резервную копию, сжатую резервную копию с указанием даты создания копии:

```
mysqldump -u root -p addressbook > /var/backup/addressbook.sql
```

```
mysqldump -u root -p addressbook | gzip > /var/backup/addressbook.sql.gz
```

```
mysqldump -u root -p addressbook | gzip > $(date
    ↪ +/var/backup/addressbook.%Y%m%d.%H%M%S.sql.gz)
```

Восстанавливаю БД из резервной копии, а также из сжатой резервной копии (рис. 2.13).

```
[root@server backup]# ls
addressbook.20240929.204328.sql.gz  addressbook.sql  addressbook.sql.gz
[root@server backup]# mysql -u root -p addressbook < /var/backup/addressbook.sql
Enter password:
[root@server backup]# zcat /var/backup/addressbook.sql.gz | mysql -u root -p addressbook
Enter password:
[root@server backup]#
```

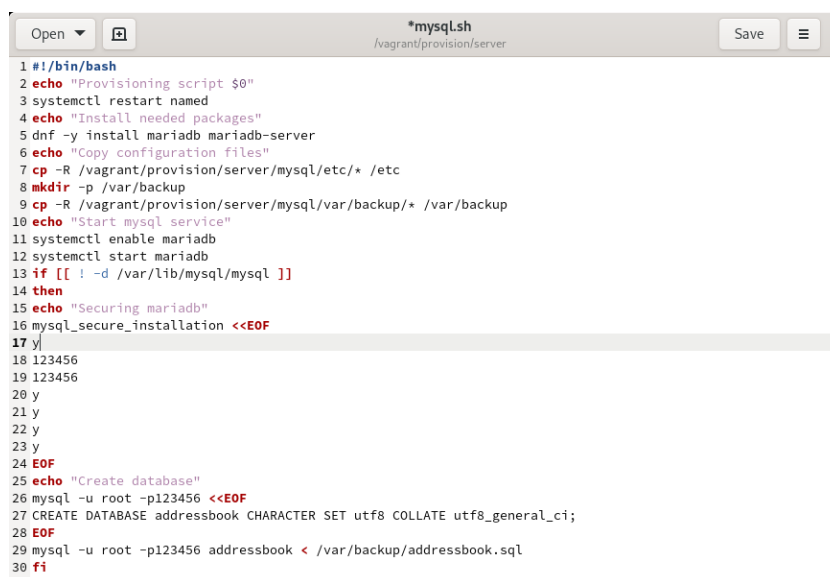
Рис. 2.13: Созданные резервные копии, восстановление резервных копий

## 2.5 Внесение изменений в настройки внутреннего окружения виртуальной машины

На VM server перехожу в каталог для внесения изменений в настройки внутреннего окружения `/vagrant/provision/server/` и копирую в соответствующие каталоги конфигурационные файлы:

```
cd /vagrant/provision/server
mkdir -p /vagrant/provision/server/mysql/etc/my.cnf.d
mkdir -p /vagrant/provision/server/mysql/var/backup
cp -R /etc/my.cnf.d/utf8.cnf
    ↪ /vagrant/provision/server/mysql/etc/my.cnf.d/
cp -R /var/backup/* /vagrant/provision/server/mysql/var/backup/
```

Создаю скрипт `/vagrant/provision/server/mysql.sh` (рис. 2.14).



```
1 #!/bin/bash
2 echo "Provisioning script $0"
3 systemctl restart named
4 echo "Install needed packages"
5 dnf -y install mariadb mariadb-server
6 echo "Copy configuration files"
7 cp -R /vagrant/provision/server/mysql/etc/* /etc
8 mkdir -p /var/backup
9 cp -R /vagrant/provision/server/mysql/var/backup/* /var/backup
10 echo "Start mysql service"
11 systemctl enable mariadb
12 systemctl start mariadb
13 if [[ ! -d /var/lib/mysql/mysql ]]
14 then
15     echo "Securing mariadb"
16     mysql_secure_installation <<EOF
17 y
18 123456
19 123456
20 y
21 y
22 y
23 y
24 EOF
25 echo "Create database"
26 mysql -u root -p123456 <<EOF
27 CREATE DATABASE addressbook CHARACTER SET utf8 COLLATE utf8_general_ci;
28 EOF
29 mysql -u root -p123456 addressbook < /var/backup/addressbook.sql
30 fi
```

Рис. 2.14: Создание скрипта `mysql.sh`

Для отработки созданного скрипта во время загрузки виртуальных машин в конфигурационном файле `Vagrantfile` добавляю запись в конфигурации сервера

```
server.vm.provision "server mysql",  
  type: "shell",  
  preserve_order: true,  
  path: "provision/server/mysql.sh"
```

## **3 Выводы**

В результате выполнения работы были приобретены практические навыки по установке и конфигурированию системы управления базами данных на примере программного обеспечения MariaDB.

## 4 Ответы на контрольные вопросы

1. Какая команда отвечает за настройки безопасности в MariaDB?
  - Настройки безопасности в MariaDB обычно управляются с помощью команды `mysql_secure_installation`. Эта команда выполняет несколько шагов, включая установку пароля для пользователя `root`, удаление анонимных учетных записей, отключение удаленного входа для пользователя `root` и удаление тестовых баз данных.
2. Как настроить MariaDB для доступа через сеть?
  - Для настройки MariaDB для доступа через сеть, вы можете отредактировать файл конфигурации MariaDB (обычно называемый `my.cnf`) и убедиться, что параметр `bind-address` установлен на IP-адрес, доступный в вашей сети. Также, убедитесь, что пользователь имеет права доступа извне, например, с использованием команды `GRANT`.
3. Какая команда позволяет получить обзор доступных баз данных после входа в среду оболочки MariaDB?
  - `SHOW DATABASES;`
4. Какая команда позволяет узнать, какие таблицы доступны в базе данных?
  - `SHOW TABLES;`
5. Какая команда позволяет узнать, какие поля доступны в таблице? -

- `DESCRIBE table_name;`
6. Какая команда позволяет узнать, какие записи доступны в таблице?
- `SELECT * FROM table_name;`
7. Как удалить запись из таблицы?
- `DELETE FROM table_name WHERE condition;`, где `condition` - условие, определяющее, какие записи следует удалить.
8. Где расположены файлы конфигурации MariaDB? Что можно настроить с их помощью?
- Файлы конфигурации MariaDB обычно располагаются в различных местах в зависимости от системы, но основной файл - `my.cnf`. Он может быть в `/etc/my.cnf`, `/etc/mysql/my.cnf` или `/usr/etc/my.cnf`. С помощью этих файлов можно настроить различные параметры, такие как порт, пути к файлам данных, параметры безопасности и другие.
9. Где располагаются файлы с базами данных MariaDB?
- Файлы с базами данных MariaDB располагаются в директории данных. Обычно это `/var/lib/mysql/` на Linux-системах.
10. Как сделать резервную копию базы данных и затем её восстановить?
- Для создания резервной копии базы данных используйте команду `mysqldump`. Например, `mysqldump -u username -p dbname > backup.sql`. Для восстановления базы данных из резервной копии используйте команду `mysql -u username -p dbname < backup.sql`.