

**РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ
НАРОДОВ**

**Факультет физико-математических и естественных
наук**

Кафедра теории вероятностей и кибербезопасности

Отчет лабораторной работы 6

Дисциплина: Администрирование сетевых подсистем

Студент: Астахова Марина

Группа: НПИбд-02-23

Тема: Установка и настройка системы управления базами данных MariaDB

6.1. Цель работы.

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

6.2. Выполнение работы

1. Установите необходимые для работы MariaDB пакеты.

Установка MariaDB.

```
[astakhovam@localhost.user.net ~]$ sudo -i
[sudo] password for astakhovam:
[root@localhost.user.net ~]# dnf -y install mariadb mariadb-server
Last metadata expiration check: 2:11:46 ago on Mon 17 Nov 2025 09:44:16 PM UTC.
Dependencies resolved.
=====
 Package           Architecture   Version        Repository      Size
=====
Installing:
 mariadb          x86_64        3:10.5.29-2.el9_6    appstream     1.6 M
 mariadb-server   x86_64        3:10.5.29-2.el9_6    appstream     9.7 M
Installing dependencies:
 mariadb-common   x86_64        3:10.5.29-2.el9_6    appstream     27 k
 mariadb-errmsg   x86_64        3:10.5.29-2.el9_6    appstream     210 k
 mysql-selinux    noarch       1.0.14-1.el9_6      appstream     36 k
 perl-DBD-MariaDB x86_64        1.21-16.el9_0      appstream     151 k
 perl-Sys-Hostname x86_64        1.23-481.1.el9_6    appstream     15 k
Installing weak dependencies:
 mariadb-backup   x86_64        3:10.5.29-2.el9_6    appstream     6.5 M
 mariadb-gssapi-server x86_64        3:10.5.29-2.el9_6    appstream     14 k
 mariadb-server-utils x86_64        3:10.5.29-2.el9_6    appstream     210 k
Transaction Summary
=====
Install 10 Packages

Total download size: 18 M
Installed size: 109 M
Downloading Packages:
(1/3/10): mariadb-10.5.29-2.el9_6.x86_64.rpm  0% [                                ] --- B/s | 0 B --:-- ETA

Complete!
[root@localhost.user.net ~]# cd /etc
[root@localhost.user.net etc]# cat my.cnf.d
cat: my.cnf.d: Is a directory
[root@localhost.user.net etc]# cat /etc/my.cnf
#
# This group is read 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@localhost.user.net etc]# ls /etc/my.cnf.d/
auth_gssapi.cnf  client.cnf  enable_encryption.preset  mariadb-server.cnf  mysql-clients.cnf  spider.cnf
[root@localhost.user.net etc]# cat /etc/my.cnf.d
cat: /etc/my.cnf.d: Is a directory
[root@localhost.user.net etc]# cat /etc/my.cnf.d/mariadb-server.cnf
```

```
root@localhost:/etc

socket=/var/lib/mysql/mysql.sock
log-error=/var/log/mariadb/mariadb.log
pid-file=/run/mariadb/mariadb.pid

#
# * Galera-related settings
#
[galera]
# Mandatory settings
#wsrep_on=ON
#wsrep_provider=
#wsrep_cluster_address=
#binlog_format=row
#default_storage_engine=InnoDB
#innodb_autoinc_lock_mode=2
#
# Allow server to accept connections on all interfaces.
#
#bind-address=0.0.0.0
#
# Optional setting
#wsrep_slave_threads=1
#innodb_flush_log_at_trx_commit=0

# this is only for embedded server
[embedded]

# This group is only read by MariaDB servers, not by MySQL.
# If you use the same .cnf file for MySQL and MariaDB,
# you can put MariaDB-only options here
[mariadb]

# This group is only read by MariaDB-10.5 servers.
# If you use the same .cnf file for MariaDB of different versions,
```

```
Remove anonymous users? [Y/n] Y
... Success!

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] Y
... Success!

By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] Y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] Y
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
[root@localhost.user.net etc]#
```

После установки пакетов убедились, что mariadb прослушивает порт

```
[root@localhost.user.net etc]# systemctl start mariadb
[root@localhost.user.net etc]# systemctl enable mariadb
Created symlink /etc/systemd/system/mysql.service → /usr/lib/systemd/system/mariadb.service.
Created symlink /etc/systemd/system/mysqld.service → /usr/lib/systemd/system/mariadb.service.
Created symlink /etc/systemd/system/multi-user.target.wants/mariadb.service → /usr/lib/systemd/system/mariadb.service.
[root@localhost.user.net etc]# ss -tulpen | grep mysql
[root@localhost.user.net etc]#
```

Просмотрели список команд MySQL, введя \h.

```
Thanks for using MariaDB!
[root@localhost.user.net etc]# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 13
Server version: 10.5.29-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'.
charset   (\C) Switch to another charset. Might be needed for processing binlog with multi-byte charsets.
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.
ego      (\G) Send command to MariaDB server, display result vertically.
exit     (\q) Exit mysql. Same as quit.
go       (\g) Send command to MariaDB server.
help     (\h) Display this help.
nopager  (\n) Disable pager, print to stdout.
noteee   (\t) Don't write into outfile.
nowarning (\w) Don't show warnings after every statement.
pager    (\P) Set PAGER [to_pager]. Print the query results via PAGER.
print    (\p) Print current command.
prompt   (\R) Change your mysql prompt.
```

Из приглашения интерактивной оболочки MariaDB для отображения доступных в настоящее время баз данных ввели MySQL-запрос

```
For server side help, type 'help contents'

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

MariaDB [(none)]>
```

```
MariaDB [(none)]> exit;
Bye
```

2. Конфигурация кодировки символов

В каталоге /etc/my.cnf.d создали файл utf8.cnf, открыли его на редактирование и указали в нём следующую конфигурацию:

```
root@localhost:/etc my.cnf.d
```

```
root@localhost:/etc x root@localhost
```

```
GNU nano 5.6.1 utf8.cnf
```

```
[client]
default-character-set = utf8
[mysqld]
character-set-server = utf8
```

3. В базе данных MariaDB создайте тестовую базу *addressbook*, содержащую таблицу *city* с полями *name* и *city*, т.е., например, для некоторого сотрудника указан город, в котором он работает. Создание базы данных.

Здесь мы завели addressbook и добавили через запросы фамилии Иванов , Петров и Сидоров с городами Москва, Сочи и Дубна.

```
[root@localhost user.net my.cnf.d]# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 4
Server version: 10.5.29-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)]> CREATE DATABASE addressbook CHARACTER SET utf8 COLLATE utf8_general_ci;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> USE addressbook;
Database changed
MariaDB [addressbook]> SHOW TABLES;
Empty set (0.000 sec)

MariaDB [addressbook]> CREATE TABLE city(name VARCHAR(40), city VARCHAR(40));
Query OK, 0 rows affected (0.006 sec)

MariaDB [addressbook]> CREATE TABLE city(name VARCHAR(40), city VARCHAR(40));
ERROR 1050 (42S01): Table 'city' already exists
MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Иванов','Москва');
Query OK, 1 row affected (0.003 sec)
```

```
MariaDB [(none)]> USE addressbook;
Database changed
MariaDB [addressbook]> SHOW TABLES;
Empty set (0.000 sec)

MariaDB [addressbook]> CREATE TABLE city(name VARCHAR(40), city VARCHAR(40));
Query OK, 0 rows affected (0.006 sec)

MariaDB [addressbook]> CREATE TABLE city(name VARCHAR(40), city VARCHAR(40));
ERROR 1050 (42S01): Table 'city' already exists
MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Иванов','Москва');
Query OK, 1 row affected (0.003 sec)

MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('о Петров','Сочи');
Query OK, 1 row affected (0.006 sec)

MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Сидоров','Дубна');
Query OK, 1 row affected (0.007 sec)

MariaDB [addressbook]> SELECT * FROM city;
+-----+-----+
| name | city |
+-----+-----+
| Иванов | Москва |
| о Петров | Сочи |
| Сидоров | Дубна |
+-----+-----+
3 rows in set (0.001 sec)
```

```
Query OK, 1 row affected (0.007 sec)

MariaDB [addressbook]> SELECT * FROM city;
+-----+-----+
| name | city |
+-----+-----+
| Иванов | Москва |
| о Петров | Сочи |
| Сидоров | Дубна |
+-----+-----+
3 rows in set (0.001 sec)

MariaDB [addressbook]> CREATE USER astakhovam@'%' IDENTIFIED BY 'password';
Query OK, 0 rows affected (0.007 sec)

MariaDB [addressbook]> GRANT SELECT,INSERT,UPDATE,DELETE ON addressbook.* TO astakhovam@'%';
Query OK, 0 rows affected (0.003 sec)

MariaDB [addressbook]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.001 sec)

MariaDB [addressbook]> DESCRIBE city;
+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| name | varchar(40) | YES | | NULL | |
| city | varchar(40) | YES | | NULL | |
+-----+-----+-----+-----+-----+
2 rows in set (0.001 sec)

MariaDB [addressbook]> quit
Bye
```

Добавили пароль для базы данных и посмотрели общую информацию о таблицах

```
MariaDB [addressbook]> quit
Bye
[root@localhost.user.net my.cnf.d]# mysqlshow -u root -p
Enter password:
+-----+
| Databases |
+-----+
| addressbook |
| information_schema |
| mysql |
| performance_schema |
+-----+
[root@localhost.user.net my.cnf.d]# mysqlshow -u root -p addressbook
Enter password:
Database: addressbook
+-----+
| Tables |
+-----+
| city |
+-----+
[root@localhost.user.net my.cnf.d]#
```

4. Создайте резервную копию базы данных *addressbook* и восстановите из неё данные. Резервные копии.

```
root@localhost:/etc      x      root@localhost:/etc/my.cnf.d      x      root@localhost:~      x
[astakhovamd@localhost.user.net ~]$ sudo -i
[sudo] password for astakhovamd:
[root@localhost.user.net ~]# mkdir -p /var/backup
[root@localhost.user.net ~]# mysqldump -u root -p addressbook > /var/backup/addressbook.sql
Enter password:
[root@localhost.user.net ~]# mysqldump -u root -p addressbook | gzip > /var/backup/addressbook.sql.gz
Enter password:
[root@localhost.user.net ~]# mysqldump -u root -p addressbook | gzip > $(date +/var/backup/addressbook.%Y%m%d.%H%M%S.sql.gz)
Enter password:
[root@localhost.user.net ~]# mysql -u root -p addressbook < /var/backup/addressbook.sql
Enter password:
[root@localhost.user.net ~]# zcat /var/backup/addressbook.sql.gz | mysql -u root -p addressbook
Enter password:
[root@localhost.user.net ~]#
```

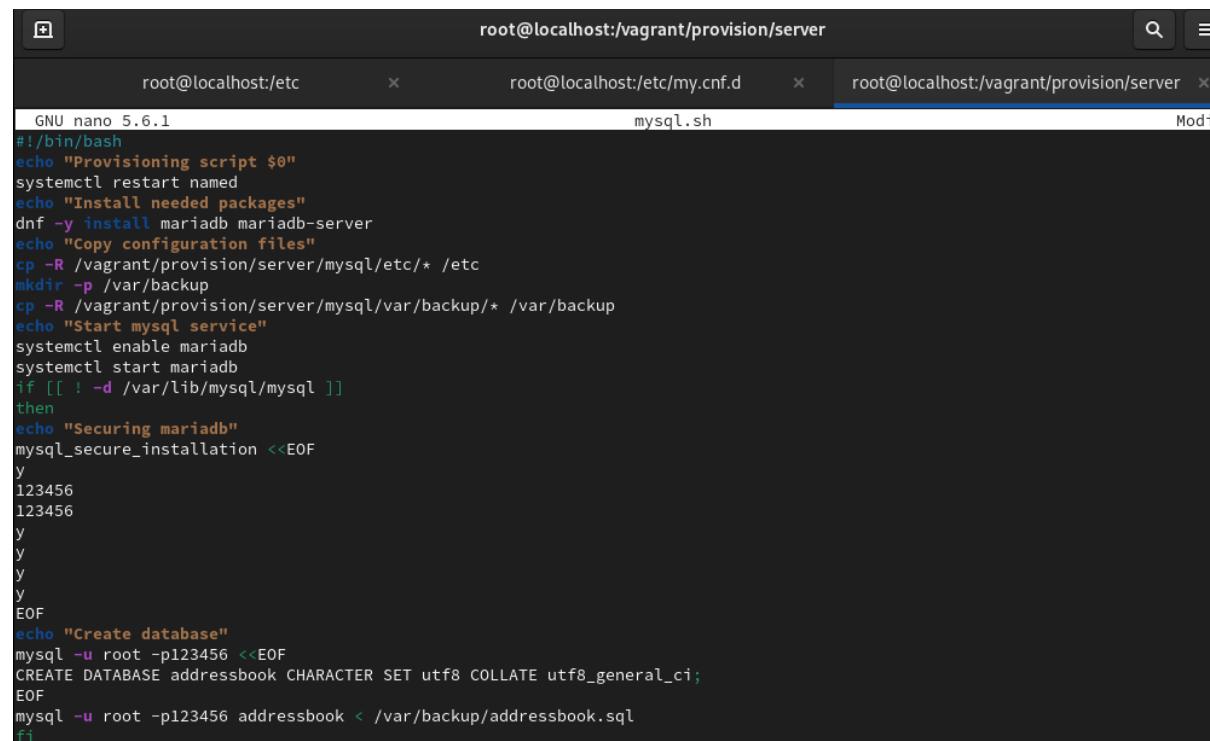
5. Напишите скрипт для Vagrant, фиксирующий действия по установке и настройке базы данных MariaDB во внутреннем окружении виртуальной машины server. Соответствующим образом следует внести изменения в Vagrantfile.

Внесение изменений в настройки внутреннего окружения виртуальной машины.

В данном каталоге создали файл mysql.sh и прописали код

```
Enter password:  
[root@localhost.user.net ~]# cd /vagrant/provision/server  
[root@localhost.user.net server]# mkdir -p /vagrant/provision/server/mysql/etc/my.cnf.d  
[root@localhost.user.net server]# mkdir -p /vagrant/provision/server/mysql/var/backup  
[root@localhost.user.net server]# cp -R /etc/my.cnf.d/utf8.cnf /vagrant/provision/server/mysql/etc/my.cnf.d/  
[root@localhost.user.net server]# cp -R /var/backup/* /vagrant/provision/server/mysql/var/backup/  
[root@localhost.user.net server]#
```

```
[root@localhost.user.net server]# sudo nano mysql.sh  
[root@localhost.user.net server]# cd /vagrant/provision/server  
[root@localhost.user.net server]# touch mysql.sh  
[root@localhost.user.net server]# chmod +x mysql.sh  
[root@localhost.user.net server]# sudo nano mysql.sh  
[root@localhost.user.net server]#
```



```
root@localhost:/vagrant/provision/server  
root@localhost:/etc  
root@localhost:/vagrant/provision/server  
GNU nano 5.6.1  
#!/bin/bash  
echo "Provisioning script $0"  
systemctl restart named  
echo "Install needed packages"  
dnf -y install mariadb mariadb-server  
echo "Copy configuration files"  
cp -R /vagrant/provision/server/mysql/etc/* /etc  
mkdir -p /var/backup  
cp -R /vagrant/provision/server/mysql/var/backup/* /var/backup  
echo "Start mysql service"  
systemctl enable mariadb  
systemctl start mariadb  
if [[ ! -d /var/lib/mysql/mysql ]]  
then  
echo "Securing mariadb"  
mysql_secure_installation <<EOF  
y  
123456  
123456  
y  
y  
y  
EOF  
echo "Create database"  
mysql -u root -p123456 <<EOF  
CREATE DATABASE addressbook CHARACTER SET utf8 COLLATE utf8_general_ci;  
EOF  
mysql -u root -p123456 addressbook < /var/backup/addressbook.sql  
fi
```

Добавляем изменения в vagrantfile

```
      type: "shell",
      preserve_order: true,
      path: "provision/server/dhcp.sh"
    end

  server.vm.provision "server http",
    type: "shell",
    preserve_order: true,
    path: "provision/server/http.sh"
  end

  server.vm.provision "server mysql",
    type: "shell",
    preserve_order: true,
    path: "provision/server/mysql.sh"
end|  
  
## Client configuration
```

6.3. Итог работы

Были приобретены практические навыки по установке и конфигурированию систем управления базами данных на примере программного обеспечения MariaDB.

6.4 Контрольные вопросы

Какая команда отвечает за настройки безопасности в MariaDB?

Команда: `sudo mysql_secure_installation`

Пояснение: Эта интерактивная утилита помогает выполнить первоначальную настройку безопасности, такую как установка пароля для пользователя `root` базы данных, удаление анонимных пользователей, запрет удаленного входа для `root` и удаление тестовой базы данных.

Как настроить MariaDB для доступа через сеть?

Шаги:

1. Изменить файл конфигурации:
2. Откройте основной конфигурационный файл MariaDB. На Ubuntu 22.04 это обычно `/etc/mysql/mariadb.conf.d/50-server.cnf` или `/etc/mysql/my.cnf` (который может включать другие файлы).
3. Найдите строку `bind-address = 127.0.0.1` (или `bind-address = ::1` для IPv6).
4. Измените ее на `bind-address = 0.0.0.0` (для доступа со всех IP-адресов) или на конкретный IP-адрес сетевого интерфейса, через который вы хотите разрешить доступ.
5. Сохраните изменения и закройте файл.
6. Перезапустить службу MariaDB
7. Настроить брандмауэр (UFW)
8. Разрешите входящие соединения на порт MariaDB (по умолчанию 3306)
9. Создать пользователя и предоставить ему права: Подключитесь к MariaDB от имени `root` и создайте пользователя, который сможет подключаться удаленно

Какая команда позволяет получить обзор доступных баз данных после входа в среду оболочки MariaDB?

Команда: `SHOW DATABASES;`

Пояснение: Выполняется после подключения к MariaDB через клиент `mysql`.

Какая команда позволяет узнать, какие таблицы доступны в базе данных?

Команды:

1. Сначала выберите базу данных: USE database_name;
2. Затем покажите таблицы: SHOW TABLES;

Какая команда позволяет узнать, какие поля доступны в таблице?

Команды:

1. DESCRIBE table_name;
2. SHOW COLUMNS FROM table_name;

Как удалить запись из таблицы?

Команда:

DELETE FROM table_name WHERE condition;

Пояснение:

WHERE condition очень важен, чтобы удалить только нужные записи.

Где расположены файлы конфигурации MariaDB? Что можно настроить с их помощью?

Расположение:

На Ubuntu 22.04 основной конфигурационный файл находится по пути /etc/mysql/my.cnf.

Однако, чаще всего my.cnf включает другие файлы из директорий: /etc/mysql/conf.d/

/etc/mysql/mariadb.conf.d/ (наиболее часто используемая директория для настроек сервера MariaDB на Ubuntu, например, 50-server.cnf, 50-client.cnf и т.д.)

Что можно настроить:

1. *Сетевые параметры*: bind-address (IP-адрес для прослушивания), port (порт).
2. *Кодировки*: character_set_server, collation_server (для UTF-8).
3. *Производительность*: innodb_buffer_pool_size, key_buffer_size, max_connections, query_cache_size.
4. *Логирование*: log_error (файл логов ошибок), general_log (общий лог запросов), slow_query_log (лог медленных запросов).
5. *Пути к данным*: datadir (директория с базами данных).
6. *Безопасность*: skip-networking (отключить сетевой доступ), local-infile.

И многое другое, влияющее на поведение и ресурсы сервера.

Где располагаются файлы с базами данных MariaDB?

Расположение: По умолчанию файлы баз данных MariaDB располагаются в директории, указанной параметром datadir в конфигурационном файле.

На Ubuntu 22.04 это обычно /var/lib/mysql/

Как сделать резервную копию базы данных и затем её восстановить?

1. *Создание резервной копии (бэкапа):*
 1. Сжатая резервная копия с датой (рекомендуется)
 2. mysqldump: Утилита для создания дампа базы данных.
 3. * -u root -p: Подключение как пользователь root (запросит пароль).
 4. * addressbook: Имя базы данных для бэкапа.
 5. * | gzip: Сжимает вывод в формат Gzip.
 6. * > /var/backup/...: Перенаправляет вывод в файл с именем, включающим дату и время.
2. *Восстановление резервной копии:*
 1. Создать базу данных
 2. Восстановить данные из сжатого файла
 - i. zcat: Распаковывает .gz файл и выводит его содержимое на стандартный вывод.
 - ii. | mysql -u root -p addressbook: Перенаправляет распакованный SQL-скрипт в базу данных addressbook через клиент mysql.

