

Администрирование сетевых подсистем

Лабораторная работа №6

Чигладзе Майя Владиславовна

Содержание

1	Цель работы	6
2	Задание	7
3	Теоретическое введение	8
4	Выполнение лабораторной работы	10
5	Выводы	31
	Список литературы	32

Список иллюстраций

- Рисунок 1:1 - Входим под пользователем
- Рисунок 1:2 - Устанавливаем необходимые для работы пакеты
- Рисунок 1:3 - Конфигурационный файл
- Рисунок 1:4 - Запускаем программное обеспечение
- Рисунок 1:5 - Запускаем программное обеспечение
- Рисунок 1:6 - Запускаем скрипт безопасности
- Рисунок 1:9 - Для входа в базу данных с правами администратора вводим
- Рисунок 1:10 - Просмотрим список команд
- Рисунок 1:11 - Смотрим базы данных
- Рисунок 2:1 - Входим в базу данных
- Рисунок 2:2 - Для отображения вводим приглашения
- Рисунок 2:3 - В каталоге создаем файл
- Рисунок 2:4 - Редактируем файл
- Рисунок 2:5 - Перезапускаем базу
- Рисунок 2:6 - Смотрим статус
- Рисунок 3:1 - Смотрим статус
- Рисунок 3:2 - Переходим в базу данных
- Рисунок 3:3 - Отображаем таблицу
- Рисунок 3:4 - Создаем таблицу
- Рисунок 3:5 - Заполняем строки
- Рисунок 3:6 - Заполняем строки
- Рисунок 3:7 - Создаем пользователя
- Рисунок 3:8 - Предоставила права доступа, обновила привилегии
- Рисунок 3:9 - Вышла из окружения, посмотрела список баз данных
- Рисунок 3:10 - Просмотрела список таблиц
- Рисунок 4:1 - Переходим и создаем каталог
- Рисунок 4:2 - Создаем исполняемый файл
- Рисунок 4:3 - Пишем скрипт
- Рисунок 4:4 - Добавляем запись

Цель работы

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

1 Задание

Установите необходимые для работы MariaDB пакеты (см. раздел 6.4.1).

Настройте в качестве кодировки символов по умолчанию utf8 в базах данных.

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 (см. раздел 6.4.1).

Создайте резервную копию базы данных addressbook и восстановите из неё данные (см. раздел 6.4.1).

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

4 Выполнение лабораторной работы

1. Загрузите вашу операционную систему и перейдите в рабочий каталог с проектом:

```
cd /var/tmp/user_name/vagrant
```

2. Запустите виртуальную машину server:

```
make server-up
```

(или, если вы работаете под ОС Windows, то `vagrant up server`).

3. На виртуальной машине server войдите под вашим пользователем и откройте терминал. Перейдите в режим суперпользователя:

```
sudo -i
```



Рисунок 1:1 - Входим под пользователем

4. Установите необходимые для работы с базами данных пакеты:

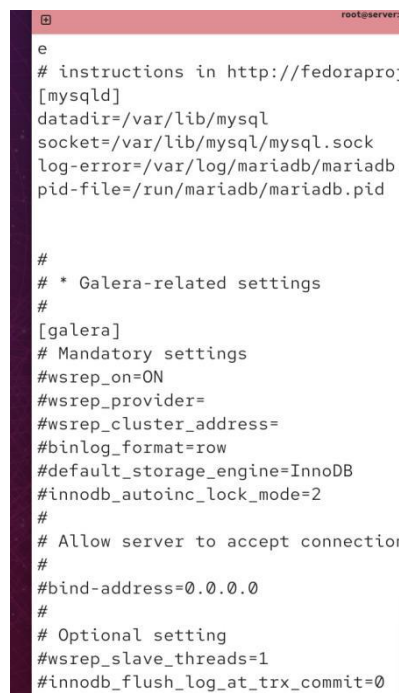
```
dnf -y install mariadb mariadb-server
```

```
[sudo] password for mvchigladze:
```

```
[root@server.mvchigladze.net ~]# dnf -y install mariadb mariadb-server
```

Рисунок 1:2 - Устанавливаем необходимые для работы пакеты

5. Просмотрите конфигурационные файлы mariadb в каталоге `/etc/my.cnf.d` и в файле `/etc/my.cnf`. В отчёте прокомментируйте построчно их содержание.



```
e
# instructions in http://fedoraproj.org/wiki/HowToStartMySQL
[mysqld]
datadir=/var/lib/mysql
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
#
#bind-address=0.0.0.0
#
# Optional setting
#wsrep_slave_threads=1
#innodb_flush_log_at_trx_commit=0
```

Рисунок 1:3 - Конфигурационный файл

Данный конфигурационный файл предназначен для настройки узла MariaDB в кластере Galera. Однако, большинство критических настроек Galera в настоящее время закомментированы. Это означает, что:

1. Сервер MariaDB, скорее всего, запустится как автономный (stand-alone) сервер, а не как часть кластера Galera, если эти настройки не активированы в другом месте или если Galera не будет явно инициализирован вручную.
2. Для полноценной работы в кластере Galera администратору потребуется раскомментировать и правильно задать значения для `wsrep_on`, `wsrep_provider`, `wsrep_cluster_address`, а также, по лучшим практикам, `binlog_format`, `default_storage_engine` и `innodb_autoinc_lock_mode`.
3. Настройка `bind-address` также важна для кластера, чтобы узлы могли общаться между собой по сети.

6. Для запуска и включения программного обеспечения `mariadb` используйте:

```
systemctl start mariadb
```

```
systemctl enable mariadb
```

```
[root@server.mvchigladze.net ~]#  
systemctl enable mariadb
```

Рисунок 1:4 - Запускаем программное обеспечение

7. Убедитесь, что mariadb прослушивает порт, используя

```
ss -tulpen | grep mysql
```

Вы должны увидеть процесс mysqld, прослушивающий порт 3306.

```
services - /usr/lib/systemd/system/mariadb.service  
[root@server.mvchigladze.net etc]# ss -tulpen | grep mysql  
[root@server.mvchigladze.net etc]#
```

Рисунок 1:5 - Запускаем программное обеспечение

8. Запустите скрипт конфигурации безопасности mariadb, используя:

```
mysql_secure_installation
```

С помощью запущившегося диалога и путём выбора [Y/n] установите пароль для пользователя root базы данных (обратите внимание, что это не пользователь root операционной системы), отключите удалённый корневой доступ и удалите тестовую базу данных и любых анонимных пользователей.

```
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] n  
... skipping.  
  
Reloading the privilege tables will ensure that all changes made so far  
will take effect immediately.  
  
Reload privilege tables now? [Y/n] n  
... skipping.  
  
Cleaning up...  
  
All done! If you've completed all of the above steps, your MariaDB  
installation should now be secure.  
  
Thanks for using MariaDB!
```

Рисунок 1:6 - Запускаем скрипт безопасности

9. Для входа в базу данных с правами администратора базы данных введите

`mysql -u root -p`

```
Enter password.
```

```
Welcome to the MariaDB monitor.  Commands end with ; or \g.
```

```
Your MariaDB connection id is 11
```

```
Server version: 10.11.11-MariaDB MariaDB Server
```

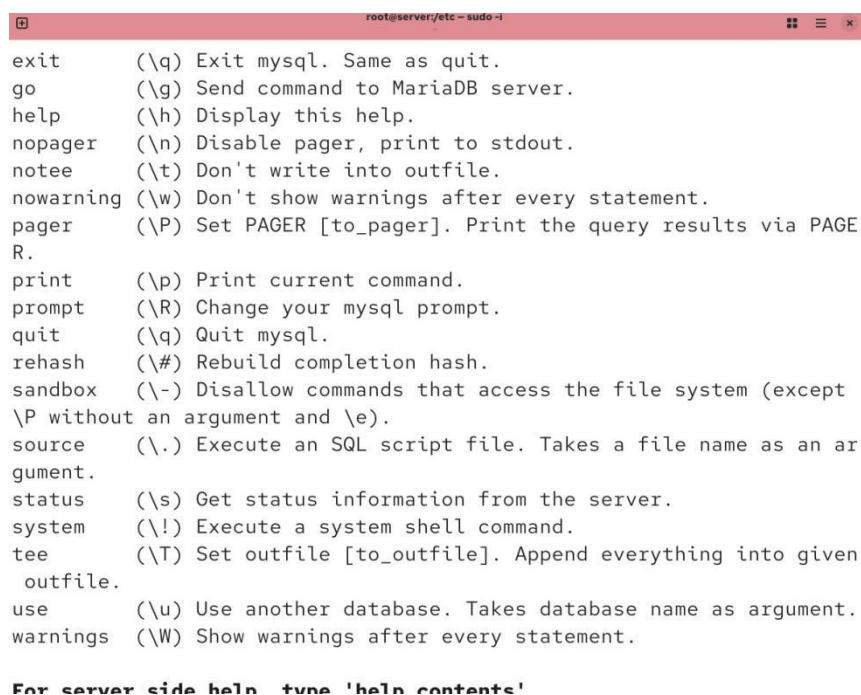
```
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.
```

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

```
MariaDB [(none)]>
```

Рисунок 1:9 - Для входа в базу данных с правами администратора вводим

10. Просмотрите список команд MySQL, введя `\h`.



```
root@server/etc ~ sudo -i
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.
notee     (\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.
quit      (\q) Quit mysql.
rehash    (\#) Rebuild completion hash.
sandbox   (\-) Disallow commands that access the file system (except \P without an argument and \e).
source    (\.) Execute an SQL script file. Takes a file name as an argument.
status    (\s) Get status information from the server.
system    (\!) Execute a system shell command.
tee        (\T) Set outfile [to_outfile]. Append everything into given outfile.
use        (\u) Use another database. Takes database name as argument.
warnings  (\W) Show warnings after every statement.

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

Рисунок 1:10 - Просмотрим список команд

11. Из приглашения интерактивной оболочки MariaDB для отображения доступных

в настоящее время баз данных введите MySQL-запрос

`SHOW DATABASES;`

В отчёте укажите, какие базы данных есть в системе.

```

For server side help, type 'help contents'

MariaDB [(none)]> show databases;
+-----+
| Database                |
+-----+
| information_schema      |
| mysql                   |
| performance_schema      |
| sys                     |
+-----+
4 rows in set (0.002 sec)

```

Рисунок 1:11 - Смотрим базы данных

Согласно изображению, в системе MariaDB присутствуют следующие базы данных:

- information_schema
- mysql
- performance_schema
- sys

12. Для выхода из интерфейса интерактивной оболочки MariaDB введите exit;

6.4.2. Конфигурация кодировки символов

1. Войдите в базу данных с правами администратора:

```
mysql -u root -p
```

```

[root@server.mvchigladze.net etc]# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 12
Server version: 10.11.11-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 s
tatement.

MariaDB [(none)]>

```

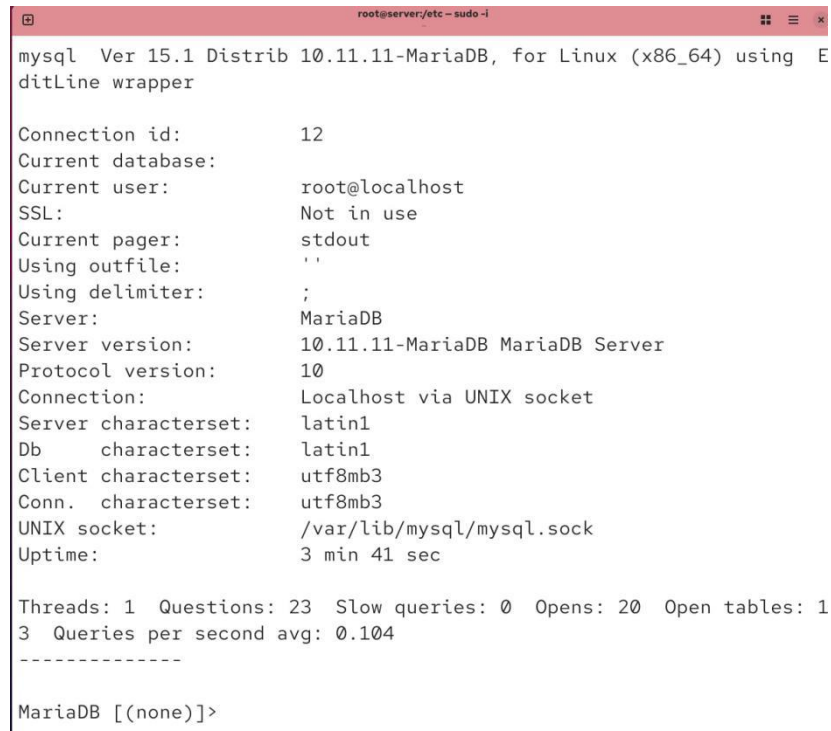
Рисунок 2:1 - Входим в базу данных

2. Для отображения статуса MariaDB введите из приглашения интерактивной оболочки

MariaDB:

status

В отчёте построчно поясните выведенную на экран информацию.



```
mysql Ver 15.1 Distrib 10.11.11-MariaDB, for Linux (x86_64) using E
ditline wrapper

Connection id:          12
Current database:
Current user:           root@localhost
SSL:                    Not in use
Current pager:          stdout
Using outfile:          ''
Using delimiter:        ;
Server:                 MariaDB
Server version:         10.11.11-MariaDB MariaDB Server
Protocol version:       10
Connection:             Localhost via UNIX socket
Server characterset:    latin1
Db characterset:        latin1
Client characterset:    utf8mb3
Conn. characterset:     utf8mb3
UNIX socket:            /var/lib/mysql/mysql.sock
Uptime:                 3 min 41 sec

Threads: 1  Questions: 23  Slow queries: 0  Opens: 20  Open tables: 1
3  Queries per second avg: 0.104
-----
MariaDB [(none)]>
```

Рисунок 2:2 - Для отображения вводим приглашения

3. В каталоге /etc/my.cnf.d создайте файл utf8.cnf:

cd /etc/my.cnf.d

touch utf8.cnf

Откройте его на редактирование и укажите в нём следующую конфигурацию

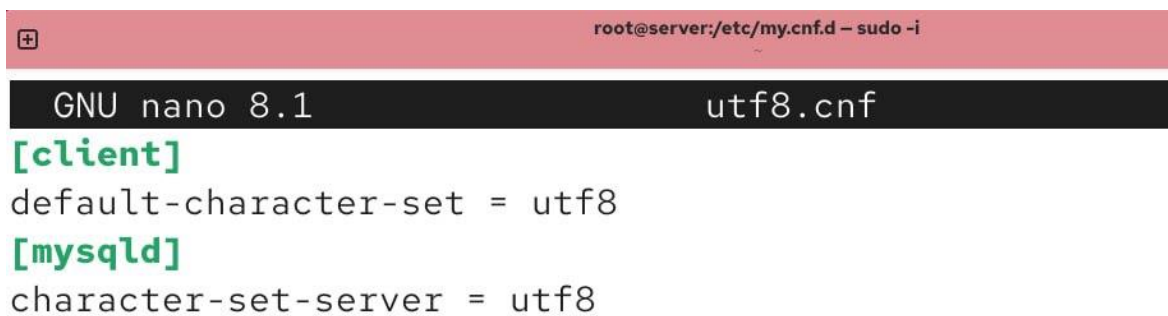
[root@server.mvchigladze.net ~]# cd /etc/my.cnf.d

touch utf8.cnf

[root@server.mvchigladze.net ~]#

[root@server.mvchigladze.net my.cnf.d]#

Рисунок 2:3 - В каталоге создаем файл



```
root@server:/etc/my.cnf.d - sudo -i
GNU nano 8.1 utf8.cnf
[client]
default-character-set = utf8
[mysqld]
character-set-server = utf8
```

Рисунок 2:4 - Редактируем файл

4. Перезапустите MariaDB:

```
systemctl restart mariadb
```

```
[root@server.mvchigladze.net my.cnf.d]# systemctl restart mariadb
[root@server.mvchigladze.net my.cnf.d]# █
```

Рисунок 2:5 - Перезапускаем базу

5. Войдите в базу данных с правами администратора и посмотрите статус MariaDB.

В отчёте поясните, что изменилось.

```

mysql Ver 15.1 Distrib 10.11.11-MariaDB, for Linux (x86_64) using
ditline wrapper

Connection id:          3
Current database:
Current user:           root@localhost
SSL:                   Not in use
Current pager:          stdout
Using outfile:          ''
Using delimiter:        ;
Server:                 MariaDB
Server version:         10.11.11-MariaDB MariaDB Server
Protocol version:       10
Connection:             Localhost via UNIX socket
Server characterset:    utf8mb3
Db characterset:        utf8mb3
Client characterset:    utf8mb3
Conn. characterset:     utf8mb3
UNIX socket:            /var/lib/mysql/mysql.sock
Jptime:                 23 sec

Threads: 1 Questions: 4 Slow queries: 0 Opens: 17 Open tables: 1
  Queries per second avg: 0.173
-----

MariaDB [(none)]>

```

Рисунок 2:6 - Смотрим статус

- Идентификатор соединения: был 12, теперь 3.
- Сервер charset: был latin1, теперь utf8mb3.
- Db charset: был latin1, теперь utf8mb3.
- Client charset: был utf8mb3, теперь utf8mb3.
- Conn. charset: был utf8mb3, теперь utf8mb3.
- Время работы: было 3 минуты 41 секунду, теперь 23 секунды.
- Вопросы: было 23, теперь 4.
- Открывается: было 20, теперь 17.
- Среднее количество запросов в секунду: было 0,104, теперь 0,173.

6.4.3. Создание базы данных

1. Войдите в базу данных с правами администратора:

```
mysql -u root -p
```

2. Создайте базу данных с именем addressbook:

```
CREATE DATABASE addressbook CHARACTER SET utf8 COLLATE utf8_general_ci;
```

```
MariaDB [(none)]> CREATE DATABASE addressbook CHARACTER SET utf8 COLLATE utf8_general_ci;  
Query OK, 1 row affected (0.011 sec)
```

Рисунок 3:1 - Смотрим статус

3. Перейдите к базе данных addressbook

USE addressbook;

```
Database changed  
MariaDB [addressbook]> SHOW TABLES;  
Empty set (0.001 sec)  
  
MariaDB [addressbook]> CREATE TABLE city(name VARCHAR(40), city VARCHAR(40));  
Query OK, 0 rows affected (0.030 sec)  
  
MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Иванов','Москва');  
Query OK, 1 row affected (0.015 sec)  
  
MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Петров','Сочи');  
Query OK, 1 row affected (0.003 sec)  
  
MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Сидоров','Дубна');  
Query OK, 1 row affected (0.002 sec)  
  
MariaDB [addressbook]>
```

Рисунок 3:2 - Переходим в базу данных

4. Отобразите имеющиеся в базе данных addressbook таблицы:

SHOW TABLES;

```
Database changed  
MariaDB [addressbook]> SHOW TABLES;  
Empty set (0.001 sec)  
  
MariaDB [addressbook]> CREATE TABLE city(name VARCHAR(40), city VARCHAR(40));  
Query OK, 0 rows affected (0.030 sec)  
  
MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Иванов','Москва');  
Query OK, 1 row affected (0.015 sec)  
  
MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Петров','Сочи');  
Query OK, 1 row affected (0.003 sec)  
  
MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Сидоров','Дубна');  
Query OK, 1 row affected (0.002 sec)  
  
MariaDB [addressbook]>
```

Рисунок 3:3 - Отображаем таблицу

5. Создайте таблицу city с полями name и city:

CREATE TABLE city(name VARCHAR(40), city VARCHAR(40));

```
Database changed
MariaDB [addressbook]> SHOW TABLES;
Empty set (0.001 sec)

MariaDB [addressbook]> CREATE TABLE city(name VARCHAR(40), city VARCHAR(40));
Query OK, 0 rows affected (0.030 sec)

MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Иванов', 'Москва');
Query OK, 1 row affected (0.015 sec)

MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Петров', 'Сочи');
Query OK, 1 row affected (0.003 sec)

MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Сидоров', 'Дубна');
Query OK, 1 row affected (0.002 sec)

MariaDB [addressbook]>
```

Рисунок 3:4 - Создаем таблицу

6. Заполните несколько строк таблицы некоторыми данными по аналогии в соответствии с синтаксисом MySQL:

```
Database changed
MariaDB [addressbook]> SHOW TABLES;
Empty set (0.001 sec)

MariaDB [addressbook]> CREATE TABLE city(name VARCHAR(40), city VARCHAR(40));
Query OK, 0 rows affected (0.030 sec)

MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Иванов', 'Москва');
Query OK, 1 row affected (0.015 sec)

MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Петров', 'Сочи');
Query OK, 1 row affected (0.003 sec)

MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Сидоров', 'Дубна');
Query OK, 1 row affected (0.002 sec)

MariaDB [addressbook]>
```

Рисунок 3:5 - Заполняем строки

7. Сделайте следующий MySQL-запрос:

SELECT * FROM city;

и в отчёте поясните результат его выполнения.

```
mysql> use addressbook;
mysql>
```

```
MariaDB [addressbook]> SELECT * FROM city;
```

```
+-----+-----+
| name      | city      |
+-----+-----+
| Иванов    | Москва    |
| Петров    | Сочи      |
| Сидоров   | Дубна     |
+-----+-----+
```

```
3 rows in set (0.001 sec)
```

```
MariaDB [addressbook]> CREATE USER mvchigladze@%' IDENTIFIED BY '123456';
```

```
Query OK, 0 rows affected (0.010 sec)
```

```
MariaDB [addressbook]>
```

Рисунок 3:6 - Заполняем строки

8. Создайте пользователя для работы с базой данных addressbook (вместо user до знака

@ используйте ваш логин) и задайте для него пароль:

```
CREATE USER user@%' IDENTIFIED BY 'password';
```

```
mysql> use addressbook;
mysql>
```

```
MariaDB [addressbook]> SELECT * FROM city;
```

```
+-----+-----+
| name      | city      |
+-----+-----+
| Иванов    | Москва    |
| Петров    | Сочи      |
| Сидоров   | Дубна     |
+-----+-----+
```

```
3 rows in set (0.001 sec)
```

```
MariaDB [addressbook]> CREATE USER mvchigladze@%' IDENTIFIED BY '123456';
```

```
Query OK, 0 rows affected (0.010 sec)
```

```
MariaDB [addressbook]>
```

Рисунок 3:7 - Создаем пользователя

9. Предоставьте права доступа созданному пользователю user на действия с базой данных addressbook (просмотр, добавление, обновление, удаление данных):

GRANT SELECT,INSERT,UPDATE,DELETE ON addressbook.* TO user@'%';

10. Обновите привилегии (права доступа) базы данных addressbook:

FLUSH PRIVILEGES;

11. Посмотрите общую информацию о таблице city базы данных addressbook:

DESCRIBE city;

```
MariaDB [addressbook]> GRANT SELECT,INSERT,UPDATE,DELETE ON addressbook.* TO mvchigladze@'%';
```

Query OK, 0 rows affected (0.002 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 (0.002 sec)

```
MariaDB [addressbook]>
```

Рисунок 3:8 - Предоставила права доступа, обновила привилегии

12. Выйдете из окружения MariaDB:

quit

13. Просмотрите список баз данных:

mysqlshow -u root -p

```

MariaDB [addressbook]> quit
Bye
[root@server.mvchigladze.net my.cnf.d]# mysqlshow -u root -p
Enter password:
+-----+
|   Databases   |
+-----+
| addressbook   |
| information_schema |
| mysql         |
| performance_schema |
| sys          |
+-----+
[root@server.mvchigladze.net my.cnf.d]#

```

Рисунок 3:9 - Вышла из окружения, посмотрела список баз данных

14. Просмотрите список таблиц базы данных addressbook:

`mysqlshow -u root -p addressbook`

или

`mysqlshow -u user -p addressbook`

```

[root@server.mvchigladze.net my.cnf.d]# mysqlshow -u root -p addressbook
Enter password:
Database: addressbook
+-----+
| Tables |
+-----+
| city   |
+-----+
[root@server.mvchigladze.net my.cnf.d]#

```

Рисунок 3:10 - Просмотрела список таблиц

6.4.5. Внесение изменений в настройки внутреннего окружения виртуальной машины

Внесение изменений в настройки внутреннего окружения

1. На виртуальной машине `server` перейдите в каталог для внесения изменений в настройки внутреннего окружения `/vagrant/provision/server/`, создайте в нём каталог `mysql`, в который поместите в соответствующие подкаталоги конфигурационные файлы MariaDB и резервную копию базы данных `addressbook`:

```

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/
-----
[root@server.mvchigladze.net my.cnf.d]# cd /vagrant/provision/server
mkdir -p /vagrant/provision/server/mysql/etc/my.cnf.d
mkdir -p /vagrant/provision/server/mysql/var/backup
[root@server.mvchigladze.net server]# cp -R /etc/my.cnf.d/utf8.cnf /v
agrant/provision/server/mysql/etc/my.cnf.d/
cp -R /var/backup/* /vagrant/provision/server/mysql/var/backup/
cp: cannot stat '/var/backup/*': No such file or directory
[root@server.mvchigladze.net server]# █

```

Рисунок 4:1 - Переходим и создаем каталог

2. В каталоге /vagrant/provision/server создайте исполняемый файл mysql.sh:

```

[root@server.mvchigladze.net server]# cd /vagrant/provision/serve
touch mysql.sh
chmod +x mysql.sh
[root@server.mvchigladze.net server]# nano

```

Рисунок 4:2 - Создаем исполняемый файл

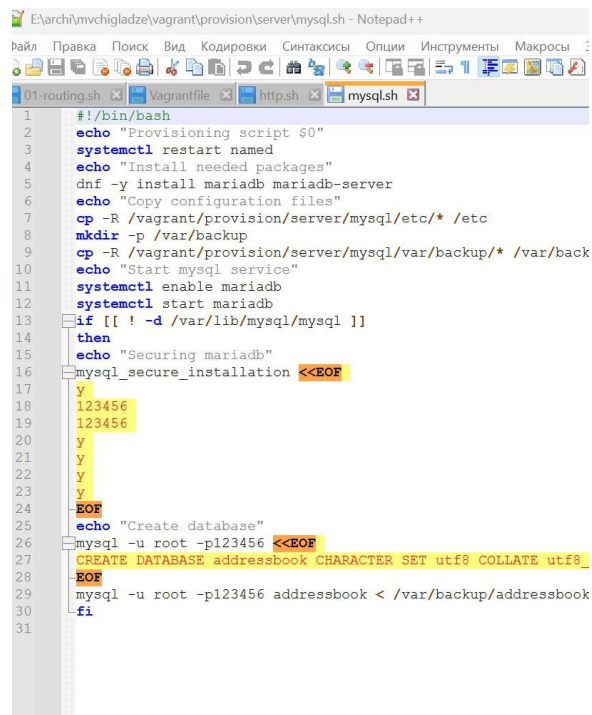
3. cd /vagrant/provision/server

touch mysql.sh

chmod +x mysql.sh

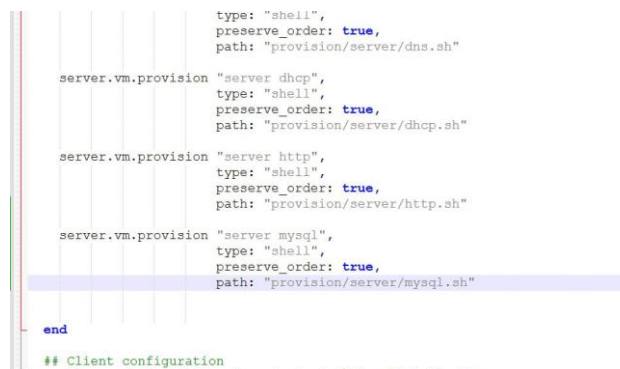
Открыв его на редактирование, пропишите в нём следующий скрипт:

Для отработки созданного скрипта во время загрузки виртуальных машин в конфигурационном файле Vagrantfile необходимо добавить в конфигурации сервера следующую запись



```
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/back
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
28     EOF
29     mysql -u root -p123456 addressbook < /var/backup/addressbook
30 fi
31
```

Рисунок 4:3 - Пишем скрипт



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

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

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

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

end

## Client configuration
```

Рисунок 4:4 - Добавляем запись

5 Выводы

В ходе выполнения лабораторной работы №6 я приобрела практические навыки по установке и конфигурированию системы управления базами данных на примере программного обеспечения MariaDB.

Список литературы

1. Barr D. Common DNS Operational and Configuration Errors: RFC / RFC Editor. —02/1996. — DOI: 10.17487/rfc1912.
2. Security-Enhanced Linux. Linux с улучшенной безопасностью: руководство пользователя / М. McAllister, S. Radvan, D. Walsh, D. Grift, E. Paris, J. Morris. — URL: https://docs-old.fedoraproject.org/ru-RU/Fedora/13/html/Security-Enhanced_Linux/index.html (дата обр.13.09.2021).
3. Systemd. — 2015. — URL: <https://wiki.archlinux.org/index.php/Systemd> (visited on 09/13/2021).
4. Костромин В. А. Утилита lsof — инструмент администратора. — URL: <http://ruslinux.net/kos.php?name=/papers/lsof/lsof.html> (дата обр. 13.09.2021).
5. Поттеринг Л. Systemd для администраторов: цикл статей. — 2010. — URL: <http://wiki.opennet.ru/Systemd> (дата обр. 13.09.2021).
6. Сайт проекта NetworkManager. — URL: <https://wiki.gnome.org/Projects/NetworkManager> (visited on 09/13/2021).
7. Сайт проекта nmcli. — URL: <https://developer.gnome.org/NetworkManager/stable/nmcli.html> (visited on 09/13/2021).