

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра теории вероятностей и кибербезопасности

ОТЧЁТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №6

дисциплина: Администрирование сетевых подсистем

Студент: Махорин Иван Сергеевич

Студ. билет № 1032211221

Группа: НПИбд-02-21

МОСКВА

2023 г.

Цель работы:

Целью данной работы является приобретение практических навыков по установке и конфигурированию системы управления базами данных на примере программного обеспечения MariaDB.

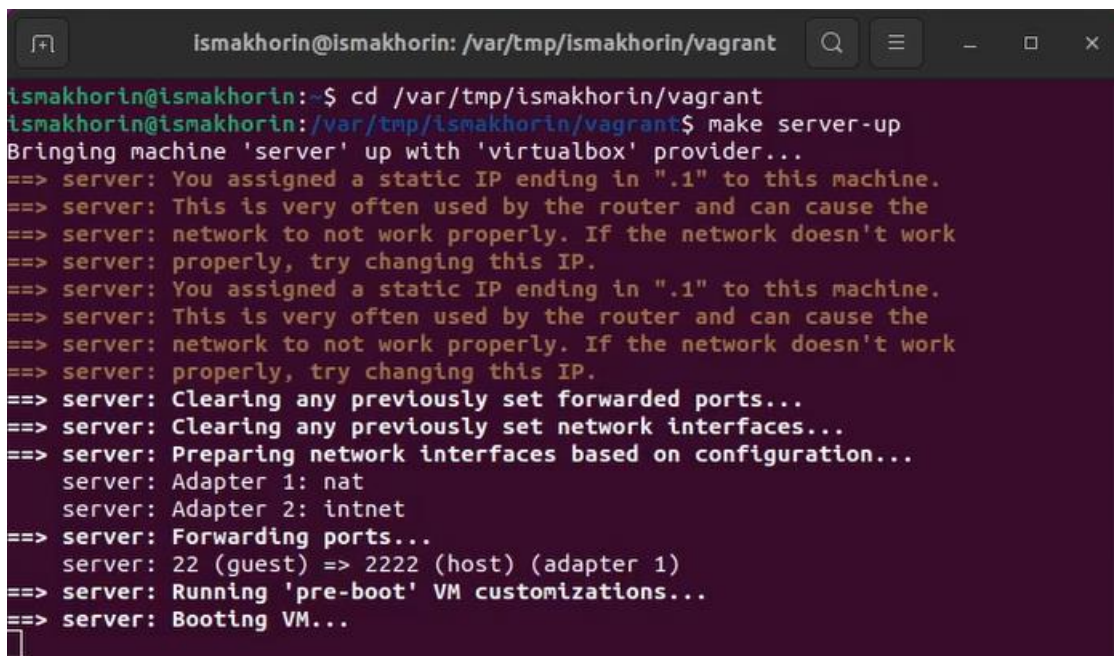
Выполнение работы:

Загрузим нашу операционную систему и перейдём в рабочий каталог с проектом:

```
cd /var/tmp/ismakhorin/vagrant
```

Далее запустим виртуальную машину server (Рис. 1.1):

```
make server-up
```



```
ismakhorin@ismakhorin: /var/tmp/ismakhorin/vagrant
ismakhorin@ismakhorin:~$ cd /var/tmp/ismakhorin/vagrant
ismakhorin@ismakhorin:/var/tmp/ismakhorin/vagrant$ make server-up
Bringing machine 'server' up with 'virtualbox' provider...
==> server: You assigned a static IP ending in ".1" to this machine.
==> server: This is very often used by the router and can cause the
==> server: network to not work properly. If the network doesn't work
==> server: properly, try changing this IP.
==> server: You assigned a static IP ending in ".1" to this machine.
==> server: This is very often used by the router and can cause the
==> server: network to not work properly. If the network doesn't work
==> server: properly, try changing this IP.
==> server: Clearing any previously set forwarded ports...
==> server: Clearing any previously set network interfaces...
==> server: Preparing network interfaces based on configuration...
server: Adapter 1: nat
server: Adapter 2: intnet
==> server: Forwarding ports...
server: 22 (guest) => 2222 (host) (adapter 1)
==> server: Running 'pre-boot' VM customizations...
==> server: Booting VM...
```

Рис. 1.1. Открытие рабочего каталога с проектом и запуск виртуальной машины server.

На виртуальной машине server войдём под нашим пользователем и откроем терминал. Далее перейдём в режим суперпользователя и установим необходимые для работы с базами данных пакеты (Рис. 1.2):

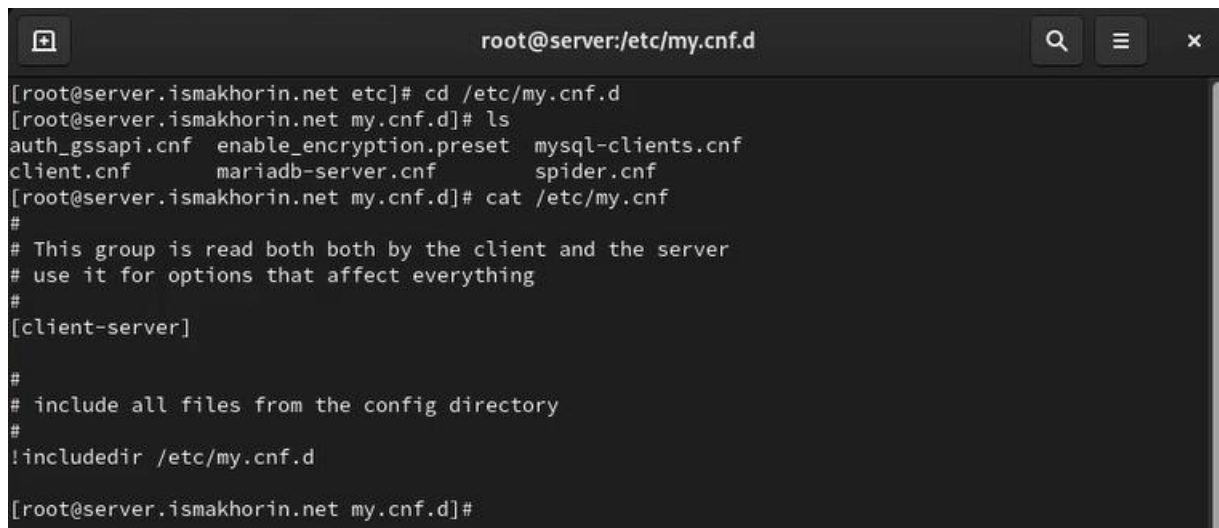
```

root@server:~
[ismakhorin@server.ismakhorin.net ~]$ sudo -i
[sudo] password for ismakhorin:
[root@server.ismakhorin.net ~]# dnf -y install mariadb mariadb-server
Last metadata expiration check: 1:35:08 ago on Thu 16 Nov 2023 12:26:49 PM UTC.
Dependencies resolved.
=====
Package                                Architecture Version                                Repository                                Size
=====
Installing:
mariadb                                x86_64      3:10.5.22-1.el9_2                        appstream                                1.6 M
mariadb-server                          x86_64      3:10.5.22-1.el9_2                        appstream                                9.6 M
Installing dependencies:
mariadb-common                          x86_64      3:10.5.22-1.el9_2                        appstream                                27 k
mariadb-connector-c                     x86_64      3.2.6-1.el9_0                            appstream                                195 k
mariadb-connector-c-config              noarch      3.2.6-1.el9_0                            appstream                                9.8 k
mariadb-errmsg                           x86_64      3:10.5.22-1.el9_2                        appstream                                211 k
mysql-selinux                           noarch      1.0.5-1.el9_0                            appstream                                35 k
perl-DBD-MariaDB                        x86_64      1.21-16.el9_0                            appstream                                151 k
perl-Sys-Hostname                       x86_64      1.23-480.el9                             appstream                                17 k
Installing weak dependencies:
mariadb-backup                          x86_64      3:10.5.22-1.el9_2                        appstream                                6.4 M
mariadb-gssapi-server                   x86_64      3:10.5.22-1.el9_2                        appstream                                15 k
mariadb-server-utils                    x86_64      3:10.5.22-1.el9_2                        appstream                                210 k

Transaction Summary
=====
Install 12 Packages

```

Просмотрим конфигурационные файлы mariadb в каталоге `/etc/my.cnf.d` и в файле `/etc/my.cnf` (Рис. 1.3):

A terminal window titled 'root@server:/etc/my.cnf.d' with search, menu, and close icons in the top right. The terminal shows the following commands and output:

```
[root@server.ismakhorin.net etc]# cd /etc/my.cnf.d
[root@server.ismakhorin.net my.cnf.d]# ls
auth_gssapi.cnf  enable_encryption.preset  mysql-clients.cnf
client.cnf       mariadb-server.cnf        spider.cnf
[root@server.ismakhorin.net 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.ismakhorin.net my.cnf.d]#
```

Рис. 1.3. Просмотр конфигурационных файлов mariadb в каталоге /etc/my.cnf.d и в файле /etc/my.cnf.

Для запуска и включения программного обеспечения mariadb используем:

```
systemctl start mariadb
```

```
systemctl enable mariadb
```

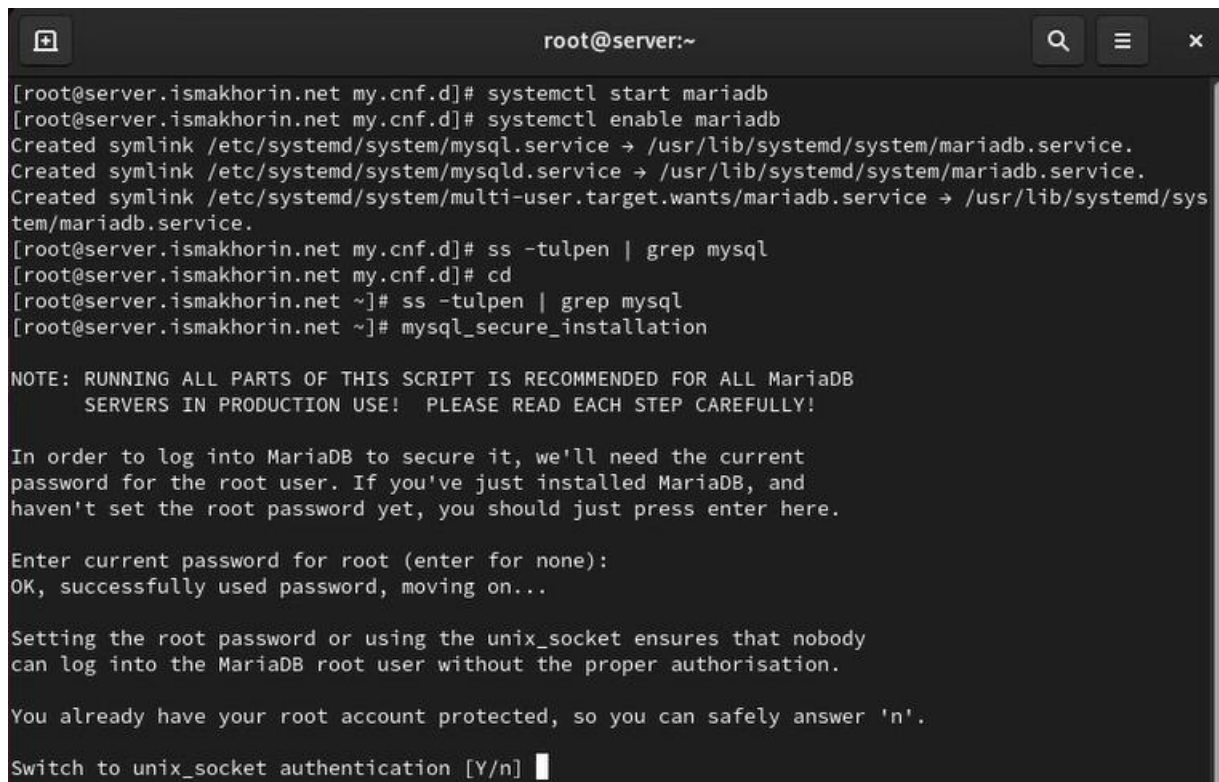
Убедимся, что mariadb прослушивает порт, используя:

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

Теперь мы видим процесс mysqld, прослушивающий порт 3306. После чего запустим скрипт конфигурации безопасности mariadb, используя:

```
mysql_secure_installation
```

С помощью запустившегося диалога и путём выбора [Y/n] установим пароль для пользователя root базы данных, отключим удалённый корневой доступ и удалим тестовую базу данных и любых анонимных пользователей (Рис. 1.4):

A terminal window titled 'root@server:~' with search, menu, and close icons in the top right. The terminal shows the following commands and output:

```
[root@server.ismakhorin.net my.cnf.d]# systemctl start mariadb
[root@server.ismakhorin.net my.cnf.d]# 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@server.ismakhorin.net my.cnf.d]# ss -tulpen | grep mysql
[root@server.ismakhorin.net my.cnf.d]# cd
[root@server.ismakhorin.net ~]# ss -tulpen | grep mysql
[root@server.ismakhorin.net ~]# mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
      SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
haven't set the root password yet, you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password or using the unix_socket ensures that nobody
can log into the MariaDB root user without the proper authorisation.

You already have your root account protected, so you can safely answer 'n'.

Switch to unix_socket authentication [Y/n]
```

Рис. 1.4. Запуск и включение программного обеспечения mariadb, проверка прослушивания порта, запуск скрипта конфигурации безопасности mariadb.

Для входа в базу данных с правами администратора базы данных введём:

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

После чего посмотрим список команд MySQL, введя \h (Рис. 1.5):

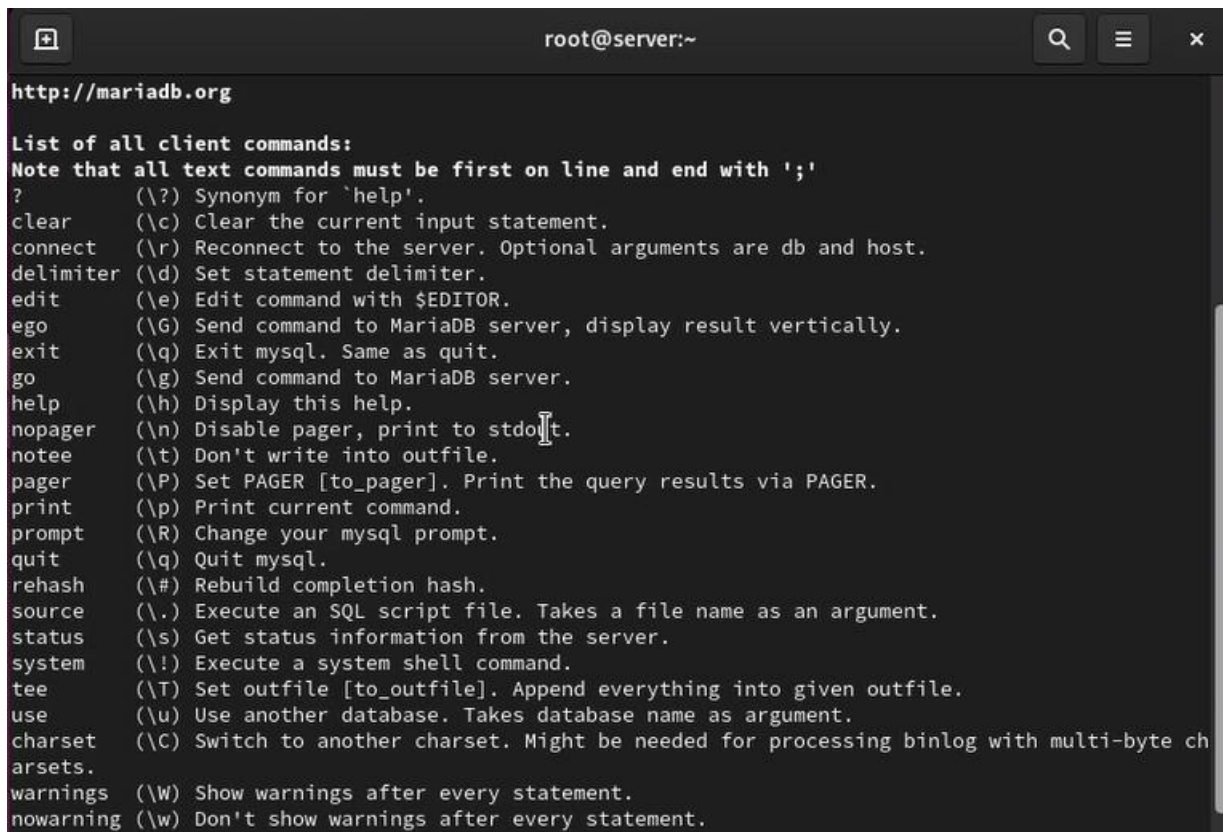
A terminal window titled 'root@server:~' with search, menu, and close icons in the top right. The terminal displays the URL 'http://mariadb.org' and a 'List of all client commands:'. A note states: 'Note that all text commands must be first on line and end with ';''. A list of commands follows, each with a shorthand and a description. The commands listed are: '? (\?) 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.', '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.', 'notee (\t) Don't write into outfile.', '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.', '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.', 'charset (\C) Switch to another charset. Might be needed for processing binlog with multi-byte characters.', 'warnings (\W) Show warnings after every statement.', and 'nowarning (\w) Don't show warnings after every statement.'

Рис. 1.5. Вход в базу данных с правами администратора базы данных и просмотр списка команд MySQL.

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

SHOW DATABASES;

Для выхода из интерфейса интерактивной оболочки MariaDB введём (Рис. 1.6):

exit;


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

MariaDB [(none)]> exit;
Bye
[root@server.ismakhorin.net ~]#
```

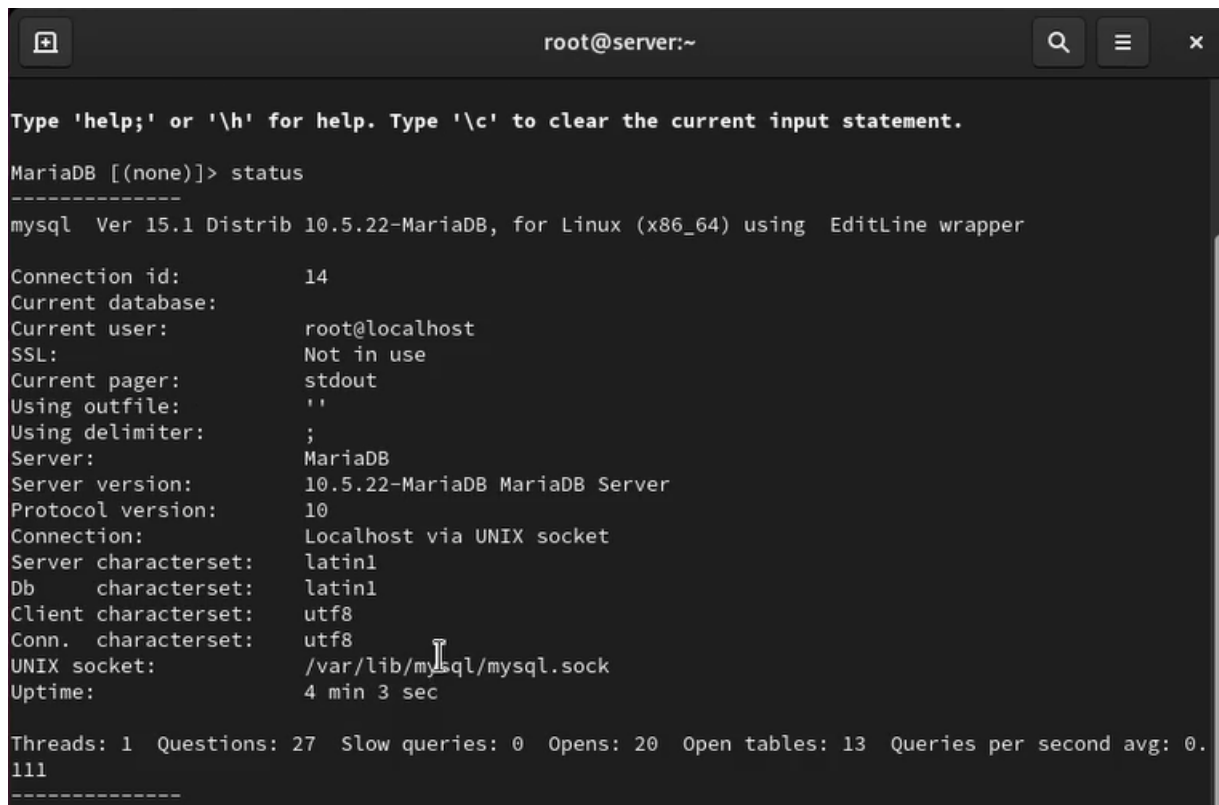
Рис. 1.6. Отображение доступных в настоящее время баз данных и выход из интерфейса интерактивной оболочки MariaDB.

Войдём в базу данных с правами администратора:

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

Для отображения статуса MariaDB введём из приглашения интерактивной оболочки MariaDB (Рис. 2.1):

```
status
```

A terminal window titled 'root@server:~' showing the output of the 'status' command in MariaDB. The output displays various connection and server details.

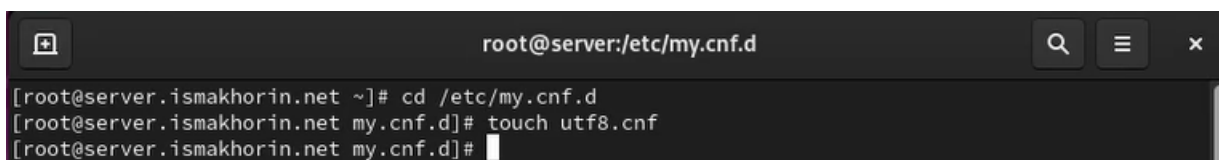
```
root@server:~  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
MariaDB [(none)]> status  
-----  
mysql Ver 15.1 Distrib 10.5.22-MariaDB, for Linux (x86_64) using EditLine wrapper  
  
Connection id:          14  
Current database:         
Current user:           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:                 4 min 3 sec  
  
Threads: 1  Questions: 27  Slow queries: 0  Opens: 20  Open tables: 13  Queries per second avg: 0.111  
-----
```

Рис. 2.1. Вход в базу данных с правами администратора, отображение статуса MariaDB.

В каталоге /etc/my.cnf.d создадим файл utf8.cnf (Рис. 2.2):

```
cd /etc/my.cnf.d
```

```
touch utf8.cnf
```

A terminal window titled 'root@server:/etc/my.cnf.d' showing the execution of 'cd' and 'touch' commands to create the utf8.cnf file.

```
root@server:/etc/my.cnf.d  
[root@server.ismakhorin.net ~]# cd /etc/my.cnf.d  
[root@server.ismakhorin.net my.cnf.d]# touch utf8.cnf  
[root@server.ismakhorin.net my.cnf.d]#
```

Рис. 2.2. Создание файла utf8.cnf в каталоге /etc/my.cnf.d.

Откроем его на редактирование и укажем в нём следующую конфигурацию (Рис. 2.3):

```
[client]
```



```
default-character-set = utf8
```

```
[mysqld]
```

```
character-set-server = utf8
```



Рис. 2.3. Открытие файла на редактирование и указание в нём конфигурации.

Перезапустим MariaDB (Рис. 2.4):

```
systemctl restart mariadb
```

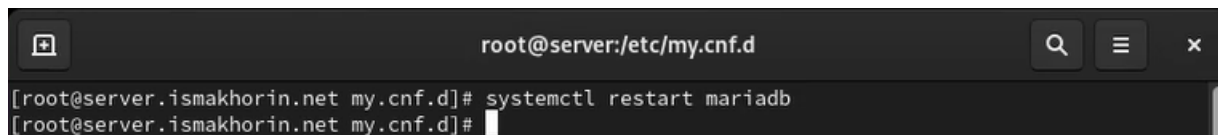
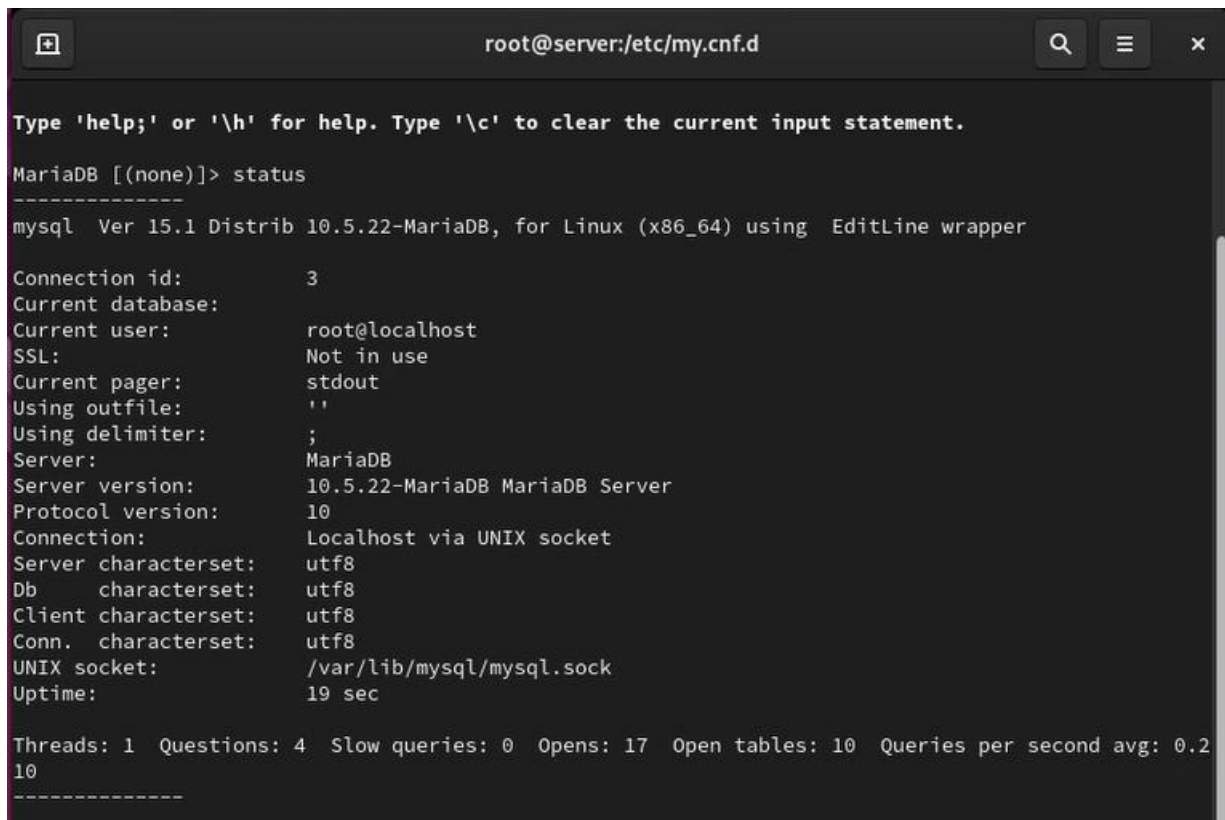


Рис. 2.4. Перезапуск MariaDB.

Войдём повторно в базу данных с правами администратора и посмотрим статус MariaDB для проверки изменений (рис. 2.5):

A terminal window titled 'root@server:/etc/my.cnf.d' with search, menu, and close icons in the title bar. The prompt is 'MariaDB [(none)]>'. The user enters 'status'. The output shows MySQL version 15.1 Distrib 10.5.22-MariaDB, for Linux (x86_64) using EditLine wrapper. It lists connection details: id 3, current database (none), user root@localhost, SSL not in use, pager stdout, outfile '', delimiter ';', server MariaDB, version 10.5.22-MariaDB MariaDB Server, protocol version 10, connection via UNIX socket, and character sets utf8 for server, db, client, and connection. The UNIX socket is /var/lib/mysql/mysql.sock and uptime is 19 sec. At the bottom, it shows 1 thread, 4 questions, 0 slow queries, 17 opens, 10 open tables, and an average of 0.2 queries per second.

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

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 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:         10.5.22-MariaDB MariaDB Server
Protocol version:       10
Connection:             Localhost via UNIX socket
Server characterset:    utf8
Db characterset:        utf8
Client characterset:    utf8
Conn. characterset:     utf8
UNIX socket:            /var/lib/mysql/mysql.sock
Uptime:                 19 sec

Threads: 1  Questions: 4  Slow queries: 0  Opens: 17  Open tables: 10  Queries per second avg: 0.2
10
-----
```

Рис. 2.5. Вход в базу данных с правами администратора и просмотр статуса MariaDB для проверки изменений.

Войдём в базу данных с правами администратора:

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

Создадим базу данных с именем addressbook:

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

Теперь перейдём к базе данных addressbook:

```
USE addressbook;
```

Отообразим имеющиеся в базе данных addressbook таблицы:

```
SHOW TABLES;
```

Создадим таблицу city с полями name и city:

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

И заполним несколько строк таблицы некоторыми данными по аналогии в соответствии с синтаксисом MySQL:

```
INSERT INTO city(name,city) VALUES ('Иванов','Москва');
```

В частности, добавим в базу сведения о Петрове и Сидорове (рис. 3.1):

Петров, Сочи

Сидоров, Дубна

```
root@server:/etc/my.cnf.d
[root@server.ismakhorin.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: 10.5.22-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)]> CREATE DATABASE addressbook CHARACTER SET utf8 COLLATE
-> ↵ utf8_general_ci;
ERROR 1273 (HY000): Unknown collation: '↵'
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 (0.029 sec)

MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Иванов','Москва');
Query OK, 1 row affected (0.025 sec)

MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Петров','Сочи');
```

Рис. 3.1. Вход в базу данных с правами администратора, создание базы данных с именем addressbook, открытие базы данных addressbook, отображение имеющиеся в базе данных addressbook таблицы. Создание таблицы city с полями name и city и заполнение таблицы некоторыми данными в соответствии с синтаксисом MySQL.

Сделаем следующий MySQL-запрос:

```
SELECT * FROM city;
```

Теперь создадим пользователя для работы с базой данных addressbook и зададим для него пароль:

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

Предоставим права доступа созданному пользователю ismakhorin на действия с базой данных addressbook (просмотр, добавление, обновление, удаление данных):

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

Обновим привилегии (права доступа) базы данных addressbook:

```
FLUSH PRIVILEGES;
```

Просмотрим общую информацию о таблице city базы данных addressbook:

```
DESCRIBE city;
```

Выйдем из окружения MariaDB (рис. 3.2):

```
Quit
```

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

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

MariaDB [addressbook]> CREATE USER ismakhorin@'%' IDENTIFIED BY '123456';
Query OK, 0 rows affected (0.022 sec)

MariaDB [addressbook]> GRANT SELECT,INSERT,UPDATE,DELETE ON addressbook.* TO ismakhorin@'%';
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 (0.002 sec)

MariaDB [addressbook]> quit
```

Рис. 3.2. MySQL-запрос, создание пользователя для работы с базой данных addressbook, предоставление прав доступа созданному пользователю ismakhorin на действия с базой данных addressbook, обновление привилегии базы данных addressbook, просмотр общей информации о таблице city базы данных addressbook и выход из окружения MariaDB.

Посмотрим список баз данных:

```
mysqlshow -u root -p
```

Отдельно посмотрим список таблиц базы данных addressbook (рис. 3.3):

```
mysqlshow -u root -p addressbook
```

или

```
mysqlshow -u ismakhorin -p addressbook
```

```
root@server:/etc/my.cnf.d
[root@server.ismakhorin.net my.cnf.d]# mysqlshow -u root -p
Enter password:
+-----+
| Databases |
+-----+
| addressbook |
| information_schema |
| mysql |
| performance_schema |
+-----+
[root@server.ismakhorin.net my.cnf.d]# mysqlshow -u root -p addressbook
Enter password:
Database: addressbook
+-----+
| Tables |
+-----+
| city |
+-----+
[root@server.ismakhorin.net my.cnf.d]# mysqlshow -u ismakhorin -p addressbook
Enter password:
Database: addressbook
+-----+
| Tables |
+-----+
| city |
+-----+
[root@server.ismakhorin.net my.cnf.d]#
```

Рис. 3.3. Просмотр списка баз данных и списка таблиц базы данных addressbook.

На виртуальной машине server создадим каталог для резервных копий:

```
mkdir -p /var/backup
```

Сделаем резервную копию базы данных addressbook:

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

Сделаем сжатую резервную копию базы данных addressbook:

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

Сделаем сжатую резервную копию базы данных addressbook с указанием даты создания копии:

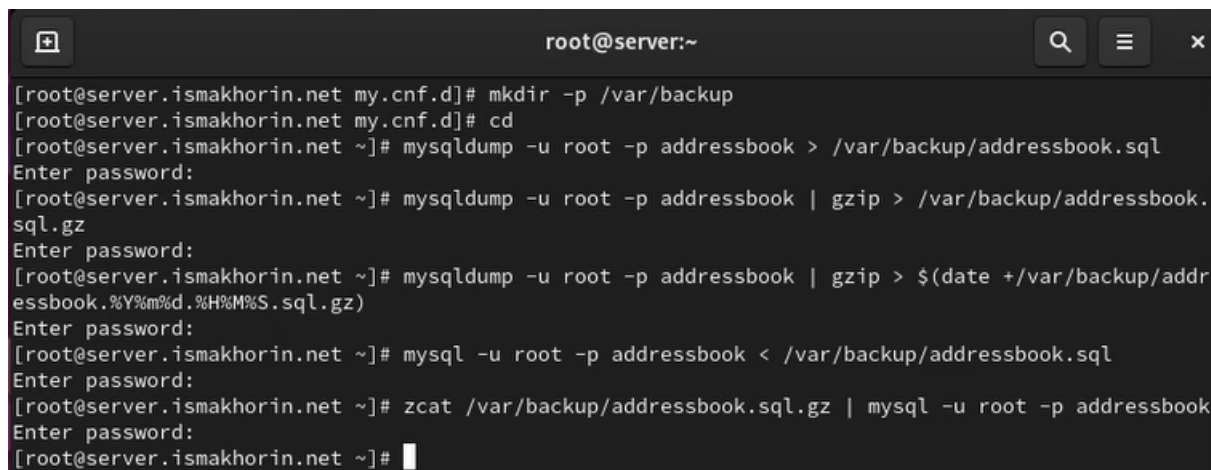
```
mysqldump -u root -p addressbook | gzip > $(date
+/var/backup/addressbook.%Y%m%d.%H%M%S.sql.gz)
```

Восстановим базу данных addressbook из резервной копии:


```
mysql -u root -p addressbook < /var/backup/addressbook.sql
```

Восстановим базу данных addressbook из сжатой резервной копии (рис. 4):

```
zcat /var/backup/addressbook.sql.gz | mysql -u root -p addressbook
```



```
root@server:~
[root@server.ismakhorin.net my.cnf.d]# mkdir -p /var/backup
[root@server.ismakhorin.net my.cnf.d]# cd
[root@server.ismakhorin.net ~]# mysqldump -u root -p addressbook > /var/backup/addressbook.sql
Enter password:
[root@server.ismakhorin.net ~]# mysqldump -u root -p addressbook | gzip > /var/backup/addressbook.sql.gz
Enter password:
[root@server.ismakhorin.net ~]# mysqldump -u root -p addressbook | gzip > $(date +%Y%m%d.%H%M%S).sql.gz
Enter password:
[root@server.ismakhorin.net ~]# mysql -u root -p addressbook < /var/backup/addressbook.sql
Enter password:
[root@server.ismakhorin.net ~]# zcat /var/backup/addressbook.sql.gz | mysql -u root -p addressbook
Enter password:
[root@server.ismakhorin.net ~]#
```

Рис. 4. Создание каталога для резервных копий, создание резервной копии базы данных addressbook, создание сжатой резервной копии базы данных addressbook, создание сжатой резервной копии базы данных addressbook с указанием даты создания копии, восстановление базы данных addressbook из резервной копии, восстановление базы данных addressbook из сжатой резервной копии.

На виртуальной машине server перейдём в каталог для внесения изменений в настройки внутреннего окружения /vagrant/provision/server/, создадим в нём каталог mysql, в который поместим 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/
```

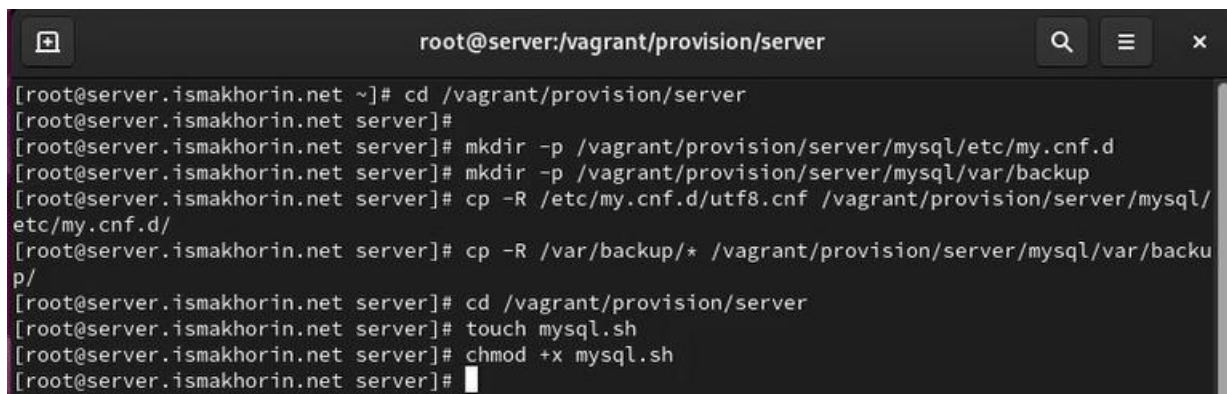
```
cp -R /var/backup/* /vagrant/provision/server/mysql/var/backup/
```

В каталоге /vagrant/provision/server создадим исполняемый файл mysql.sh (рис. 5.1):

```
cd /vagrant/provision/server
```

```
touch mysql.sh
```

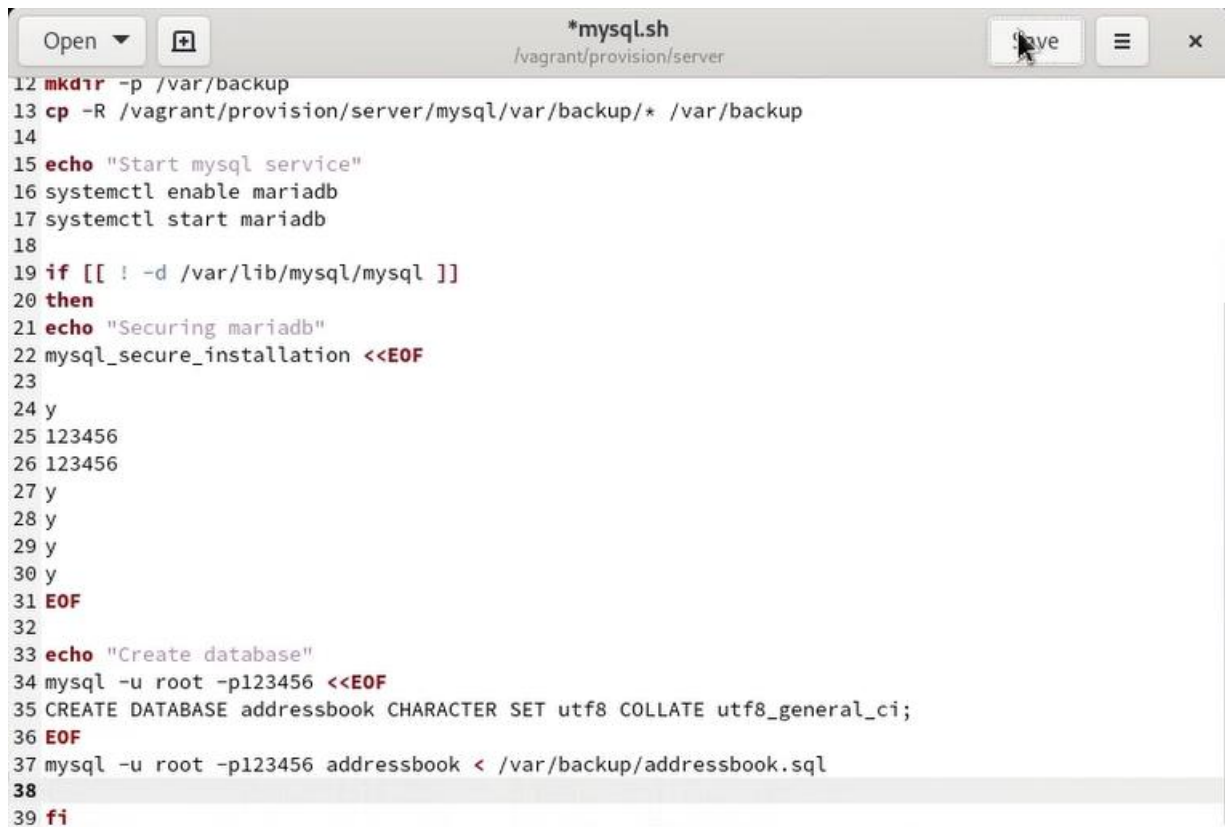
```
chmod +x mysql.sh
```



```
root@server:/vagrant/provision/server
[root@server.ismakhorin.net ~]# cd /vagrant/provision/server
[root@server.ismakhorin.net server]#
[root@server.ismakhorin.net server]# mkdir -p /vagrant/provision/server/mysql/etc/my.cnf.d
[root@server.ismakhorin.net server]# mkdir -p /vagrant/provision/server/mysql/var/backup
[root@server.ismakhorin.net server]# cp -R /etc/my.cnf.d/utf8.cnf /vagrant/provision/server/mysql/
etc/my.cnf.d/
[root@server.ismakhorin.net server]# cp -R /var/backup/* /vagrant/provision/server/mysql/var/backu
p/
[root@server.ismakhorin.net server]# cd /vagrant/provision/server
[root@server.ismakhorin.net server]# touch mysql.sh
[root@server.ismakhorin.net server]# chmod +x mysql.sh
[root@server.ismakhorin.net server]#
```

Рис. 5.1. Открытие каталога для внесения изменений в настройки внутреннего окружения /vagrant/provision/server/, создание в нём каталога mysql, в который помещаем в соответствующие подкаталоги конфигурационные файлы MariaDB и резервную копию базы данных addressbook. Создание в каталоге /vagrant/provision/server исполняемого файла mysql.sh.

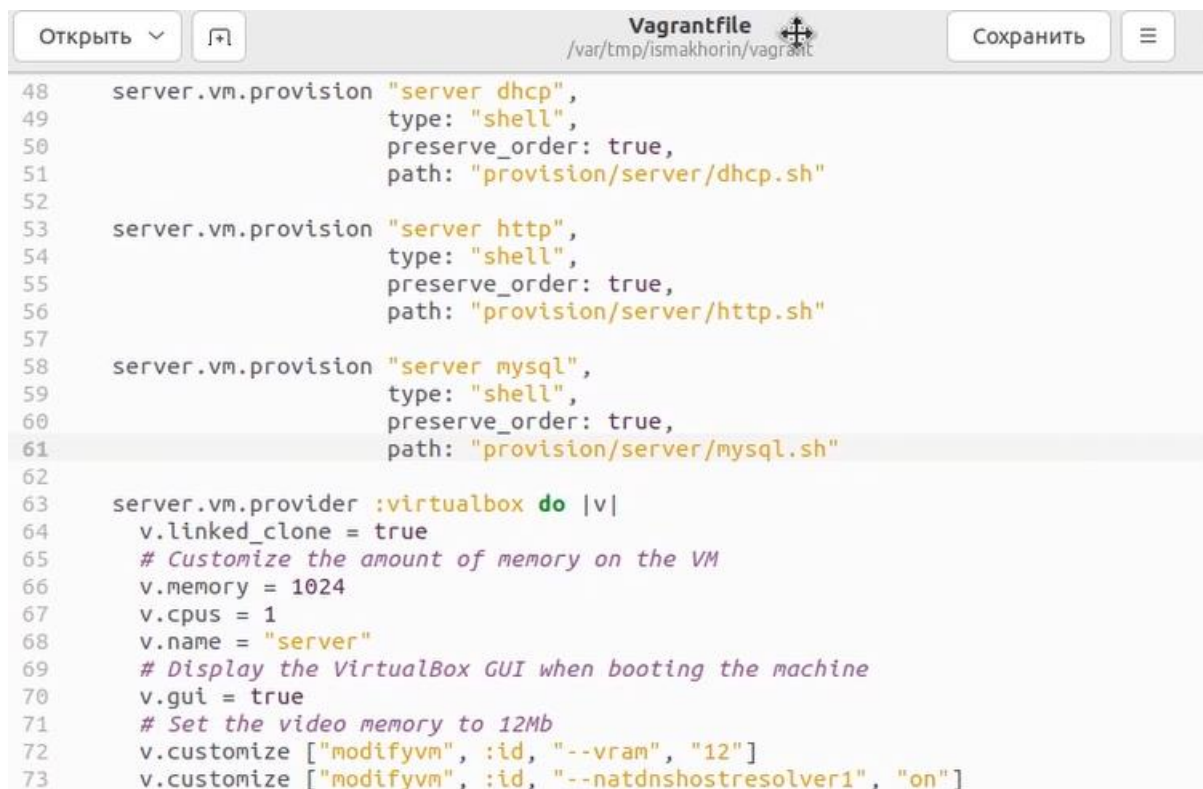
Откроем его на редактирование и пропишем в нём следующий скрипт (рис. 5.2):



```
12 mkdir -p /var/backup
13 cp -R /vagrant/provision/server/mysql/var/backup/* /var/backup
14
15 echo "Start mysql service"
16 systemctl enable mariadb
17 systemctl start mariadb
18
19 if [[ ! -d /var/lib/mysql/mysql ]]
20 then
21 echo "Securing mariadb"
22 mysql_secure_installation <<EOF
23
24 y
25 123456
26 123456
27 y
28 y
29 y
30 y
31 EOF
32
33 echo "Create database"
34 mysql -u root -p123456 <<EOF
35 CREATE DATABASE addressbook CHARACTER SET utf8 COLLATE utf8_general_ci;
36 EOF
37 mysql -u root -p123456 addressbook < /var/backup/addressbook.sql
38
39 fi
```

Рис. 5.2. Открытие исполняемого файла на редактирование и прописывание в нём скрипта.

Для отработки созданного скрипта во время загрузки виртуальных машин в конфигурационном файле Vagrantfile добавим в конфигурации сервера следующую запись (рис. 5.3):



```
48  server.vm.provision "server dhcp",
49                        type: "shell",
50                        preserve_order: true,
51                        path: "provision/server/dhcp.sh"
52
53  server.vm.provision "server http",
54                        type: "shell",
55                        preserve_order: true,
56                        path: "provision/server/http.sh"
57
58  server.vm.provision "server mysql",
59                        type: "shell",
60                        preserve_order: true,
61                        path: "provision/server/mysql.sh"
62
63  server.vm.provider :virtualbox do |v|
64    v.linked_clone = true
65    # Customize the amount of memory on the VM
66    v.memory = 1024
67    v.cpus = 1
68    v.name = "server"
69    # Display the VirtualBox GUI when booting the machine
70    v.gui = true
71    # Set the video memory to 12Mb
72    v.customize ["modifyvm", :id, "--vram", "12"]
73    v.customize ["modifyvm", :id, "--natdnshostresolver1", "on"]
```

Рис. 5.3. Добавление записи в конфигуционном файле Vagrantfile.

Вывод:

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

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? - Обзор доступных баз данных после входа в среду оболочки MariaDB можно получить с помощью команды `SHOW DATABASES;`.
4. Какая команда позволяет узнать, какие таблицы доступны в базе данных? - Для просмотра доступных таблиц в базе данных используйте команду `SHOW TABLES;`.
5. Какая команда позволяет узнать, какие поля доступны в таблице? - Чтобы узнать, какие поля доступны в таблице, используйте команду `DESCRIBE table_name;` или `SHOW COLUMNS FROM table_name;`.
6. Какая команда позволяет узнать, какие записи доступны в таблице? - Для просмотра записей в таблице можно использовать команду `SELECT * FROM table_name;`.
7. Как удалить запись из таблицы? - Для удаления записи из таблицы используйте команду `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`.**