

# **Лабораторная работа №6**

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

Андреева Софья Владимировна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Задание</b>	<b>5</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
3.1	Установка MariaDB . . . . .	6
3.2	Конфигурация кодировки символов . . . . .	13
3.3	Создание базы данных . . . . .	15
3.4	Резервные копии . . . . .	17
<b>4</b>	<b>Контрольные вопросы</b>	<b>20</b>
<b>5</b>	<b>Выводы</b>	<b>22</b>

## Список иллюстраций

3.1	Установка пакетов для MariaDB . . . . .	6
3.2	/etc/my.cnf.d . . . . .	11
3.3	/etc/my.cnf . . . . .	11
3.4	Проверка прослушивания порта 3306 . . . . .	12
3.5	Конфигурация безопасности mariadb . . . . .	12
3.6	Списки команд MySQL и баз данных . . . . .	13
3.7	Статус MariaDB . . . . .	14
3.8	Статус MariaDB после добавления конфигураций . . . . .	15
3.9	Содержимое таблицы city в базе данных addressbook . . . . .	16
3.10	Создание пользователя и предоставление ему прав для работы с addressbook . . . . .	16
3.11	Просмотр списка баз данных и таблиц базы данных addressbook . . . . .	17
3.12	Создание окружения для внесения изменений в настройки окружающей среды . . . . .	18
3.13	Содержание mysql.sh . . . . .	18
3.14	Изменение файла Vagrantfile . . . . .	19

# 1 Цель работы

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

## 2 Задание

1. Установите необходимые для работы MariaDB пакеты.
2. Настройте в качестве кодировки символов по умолчанию utf8 в базах данных.
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.
4. Создайте резервную копию базы данных addressbook и восстановите из неё данные.
5. Напишите скрипт для Vagrant, фиксирующий действия по установке и настройке базы данных MariaDB во внутреннем окружении виртуальной машины server. Соответствующим образом внести изменения в Vagrantfile.

## 3 Выполнение лабораторной работы

### 3.1 Установка MariaDB

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

```
cd /var/tmp/svandreeva/vagrant
```

Затем запустим виртуальную машину server:

```
make server-up
```

Установим необходимые для работы с базами данных пакеты(рис. fig. 3.1):

```
[root@server.svandreeva.net ~]# dnf -y install mariadb mariadb-server
Extra Packages for Enterprise Linux 10 - x86_64                               8.5 kB/s | 17 kB  00:01
Extra Packages for Enterprise Linux 10 - x86_64                               8.3 MB/s | 4.8 MB 00:00
Rocky Linux 10 - BaseOS                                                         2.6 kB/s | 3.9 kB 00:01
Rocky Linux 10 - BaseOS                                                         16 MB/s | 19 MB  00:01
Rocky Linux 10 - AppStream                                                      12 kB/s | 3.9 kB 00:00
Rocky Linux 10 - AppStream                                                      1.3 MB/s | 418 kB 00:01 ET
```

Рис. 3.1: Установка пакетов для MariaDB

Посмотрим конфигурационные файлы mariadb в каталоге /etc/my.cnf.d и в файле /etc/my.cnf. Рассмотрим файлы в каталоге /etc/my.cnf.d.

В файле auth\_gssapi.cnf содержится закомментированная строка для загрузки плагина аутентификации gssapi:

```
#plugin-load-add=auth_gssapi.so
```

В файле enable\_encryption.preset содержится базовый шаблон для шифрования.

[mariadb]

aria-encrypt-tables \\ опция для шифрования таблиц

encrypt-binlog \\ опция шиврования бинарных журналов сервера

encrypt-tmp-disk-tables \\ опция шифрования временных таблиц на диске

encrypt-tmp-files \\ опция шифрования временных файлов

loose-innodb-encrypt-log \\ опция включения шифрования логов механизма innodb

loose-innodb-encrypt-tables \\ опция включения шифрования таблиц с механизма innodb

В файле mysql-clients.cnf хранятся опции, читаемые разными перечисленными в файле группами.

[mysql]

[mysql\_upgrade]

[mysqladmin]

[mysqlbinlog]

[mysqlcheck]

[mysqldump]

[mysqlimport]

[mysqlshow]

[mysqlslap]

В файле client.cnf хранятся опции, читаемые клиентскими программами MeSQL и MariaDB.

```
# These two groups are read by the client library
# Use it for options that affect all clients, but not the server
#
```

```
[client]
```

```
# This group is not read by mysql client library,
# If you use the same .cnf file for MySQL and MariaDB,
# use it for MariaDB-only client options
[client-mariadb]
```

В файле mariadb-server.cnf хранятся опции, читаемые сервером MariaDB. Сначала указан раздел, который читает только сервер. Затем раздел [mysqld], читаемый автономным демоном mysqld, здесь уже указана директория для данных, сокет-файл, лог-файл ошибок, pid-файл. Затем раздел настроек сервера с поддержкой galera cluster, после место для раздела встроенного сервера, далее два раздела для серверов MariaDB.

```
#
# These groups are read by MariaDB server.
# Use it for options that only the server (but not clients) should see
#
# See the examples of server my.cnf files in /usr/share/mysql/
#
```

```
# this is read by the standalone daemon and embedded servers
[server]
```

```
# this is only for the mysqld standalone daemon
```



```

# Settings user and group are ignored when systemd is used.
# If you need to run mysqld under a different user or group,
# customize your systemd unit file for mysqld/mariadb according to the
# instructions in http://fedoraproject.org/wiki/Systemd

[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 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,
```

```
# use this group for options that older servers don't understand
```

```
[mariadb-10.5]
```

В файле spider.cnf хранятся конфигурации для механизма хранения Spider. Можно увидеть закомментированную строчку для добавления плагина ha\_spider.

```
[mariadb]
```

```
#
```

```
# Uncomment line to enable
```

```
#
```

```
#plugin-load-add = ha_spider
```

```
# Read more at https://mariadb.com/kb/en/spider/
```

В файле /etc/my.cnf хранятся конфигурации, которые читаются и сервером, и клиентом. Последняя строчка включает все конфигурации из предыдущего рассматриваемого каталога /etc/my.cnf.d.

```
#
```

```
# 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.svandreeva.net ~]# ls /etc/my.cnf.d
auth_gssapi.cnf  enable_encryption.preset  mysql-clients.cnf  provider_lz4.cnf  provider_snappy.cnf
client.cnf      mariadb-server.cnf        provider_bzip2.cnf  provider_lzo.cnf  spider.cnf
root@server.svandreeva.net ~]#
```

Рис. 3.2: /etc/my.cnf.d

```
root@server.svandreeva.net ~]# 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.svandreeva.net ~]#
```

Рис. 3.3: /etc/my.cnf

Для запуска и включения программного обеспечения mariadb используем:

```
systemctl start mariadb
systemctl enable mariadb
```

Убедимся, что mariadb прослушивает порт(рис. fig. 3.4)

```
[root@server.svandreeva.net ~]# systemctl start mariadb
[root@server.svandreeva.net ~]# systemctl enable mariadb
Created symlink '/etc/systemd/system/mysql.service' → '/usr/lib/systemd/system/mariadb.service'.
Created symlink '/etc/systemd/system/mysql.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.svandreeva.net ~]# ss -tulpen | grep mysql
[root@server.svandreeva.net ~]# netstat -tulpen | grep 3306
tcp        0      0 0.0.0.0:3306 0.0.0.0:*        LISTEN      27      48538      6258/mariadb
tcp6       0      0 :::3306     :::*             LISTEN      27      48539      6258/mariadb
```

Рис. 3.4: Проверка прослушивания порта 3306

Запустите скрипт конфигурации безопасности mariadb, и 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(fig. 3.5):

```
[root@server.svandreeva.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] y
Enabled successfully!
Reloading privilege tables..
... Success!

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

Change the root password? [Y/n] y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!

By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them. This is intended only for testing, and to make the installation
go a bit smoother. You should remove them before moving into a
production environment.
```

Рис. 3.5: Конфигурация безопасности mariadb

Войдем в базу данных с правами администратора базы данных. Просмотрим список команд MySQL, введя \h. Из приглашения интерактивной оболочки

MariaDB для отображения доступных в настоящее время баз данных введем MySQL-запрос `SHOW DATABASES;`. В системе есть три базы данных: `mysql`, `information_schema`, `performance_schema`(рис. fig. 3.6):

```
?          (\\?) 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.
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'

MariaDB [(none)]> SHOW DATABASES
-> ^C
MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
4 rows in set (0,001 sec)
```

Рис. 3.6: Списки команд MySQL и баз данных

Для выхода из интерфейса интерактивной оболочки MariaDB введем `exit`.

## 3.2 Конфигурация кодировки символов

Войдем в базу данных с правами администратора. Для отображения статуса MariaDB введем из приглашения интерактивной оболочки MariaDB:

```
status
```

В первой строчке указана версия ПО, далее перечислены следующая информация: id соединения, текущая база данных, текущей пользователь, SSL не используется, средство вывода информации, используемый файл для вывода

информации, используемый разделитель, ПО сервера и его версия, версия протокола, вид соединения, характеристики сервера, баз данных, клиента, сокет-файл и время начала работы. В последней строке указано количество потоков, вопросов, очередей, открытых файлов, таблиц и средняя скорость в секунду запросов(рис. fig. 3.7):

```
MariaDB [(none)]> exit;
Bye
[root@server.svandreeva.net ~]# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 14
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)]> status
-----
mysql Ver 15.1 Distrib 10.11.11-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:          10.11.11-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:                  8 min 58 sec

Threads: 1  Questions: 27  Slow queries: 0  Opens: 20  Open tables: 13  Queries per second avg: 0.050
```

Рис. 3.7: Статус MariaDB

В каталоге /etc/my.cnf.d создадим файл utf8.cnf:

```
cd /etc/my.cnf.d
touch utf8.cnf
```

Откроем его на редактирование и укажем в нём следующую конфигурацию:

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

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

Перезапустим MariaDB:

```
systemctl restart mariadb
```

Войдем в базу данных с правами администратора и посмотрим статус MariaDB(fig. 3.8):

```
[root@server.svandreeva.net ~]# cd /etc/my.cnf.d
[root@server.svandreeva.net my.cnf.d]# touch utf8.cnf
[root@server.svandreeva.net my.cnf.d]# vim utf8.cnf
[root@server.svandreeva.net my.cnf.d]# vim utf8.cnf
[root@server.svandreeva.net my.cnf.d]# systemctl restart mariadb
[root@server.svandreeva.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 3
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)]> status
-----
mysql Ver 15.1 Distrib 10.11.11-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:         10.11.11-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
Uptime:                 19 sec

Threads: 1  Questions: 4  Slow queries: 0  Opens: 17  Open tables: 10  Queries per second avg: 0.210
```

Рис. 3.8: Статус MariaDB после добавления конфигураций

Изменился ID подключения, увеличилось среднее время запроса, а также для соединения сервера и баз данных стала задана кодировка utf8.

### 3.3 Создание базы данных

Создадим базу данных с именем addressbook и перейдем к ней, затем отобразим имеющиеся в ней таблицы(их нет) и создадим таблицу с полями name и city, затем внесём в неё три записи. Сделаем запрос `SELECT * FROM city;`, он выводит все строки таблицы city (fig. 3.9):

```

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 (0,014 sec)

MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Иванов','Москва');
Query OK, 1 row affected (0,005 sec)

MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Петров','Сочи');
Query OK, 1 row affected (0,002 sec)

MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Сидоров','Дубна');
Query OK, 1 row affected (0,004 sec)

MariaDB [addressbook]> SELECT * FROM city;
+-----+-----+
| name      | city      |
+-----+-----+
| Иванов    | Москва   |
| Петров    | Сочи     |
| Сидоров   | Дубна    |
+-----+-----+
3 rows in set (0,000 sec)

```

Рис. 3.9: Содержимое таблицы city в базе данных addressbook

Создадим пользователя для работы с базой данных addressbook и предоставим ему права доступа, а затем обновим привелегии и посмотрим общую информацию о таблице city(рис. 3.10):

```

MariaDB [addressbook]> CREATE USER svandreeva@%' IDENTIFIED BY 'password';
Query OK, 0 rows affected (0,006 sec)

MariaDB [addressbook]> GRANT SELECT,INSERT,UPDATE,DELETE ON addressbook.* TO svandreeva@%';
Query OK, 0 rows affected (0,005 sec)

MariaDB [addressbook]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,000 sec)

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

```

Рис. 3.10: Создание пользователя и предоставление ему прав для работы с addressbook

Выйдем из окружения MariaDB и посмотрим список баз данных, а затем посмотрим список таблиц базы данных addressbook(рис. fig. 3.11):



```

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

```

Рис. 3.11: Просмотр списка баз данных и таблиц базы данных addressbook

## 3.4 Резервные копии

На виртуальной машине server создайте каталог для резервных копий /var/backup, затем сделаем резервную копию базы данных, сжатую резервную копию базы данных addressbook и её сжатую резервную копию с указанием даты, восстановим базу данных addressbook из резервной копии и из сжатой резервной копии

На виртуальной машине server перейдем 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 и в каталоге /vagrant/provision/server создадим исполняемый файл mysql.sh(рис. fig. 3.12)

```
[root@server.svandreeva.net my.cnf.d]# cd
[root@server.svandreeva.net ~]# mkdir -p /var/backup
[root@server.svandreeva.net ~]# mysqldump -u root -p addressbook > /var/backup/addressbook.sql
Enter password:
[root@server.svandreeva.net ~]# mysqldump -u root -p addressbook | gzip > /var/backup/addressbook.sql.gz
Enter password:
[root@server.svandreeva.net ~]# mysqldump -u root -p addressbook | gzip > $(date +%Y%m%d.%H%M%S).sql.gz)
-bash: date:/var/backup/addressbook.%Y%m%d.%H%M%S.sql.gz: Нет такого файла или каталога
-bash: $(date +%Y%m%d.%H%M%S).sql.gz): ambiguous redirect
Enter password:
mysqldump: Got errno 32 on write
[root@server.svandreeva.net ~]# mysql -u root -p addressbook < /var/backup/addressbook.sql
Enter password:
[root@server.svandreeva.net ~]# zcat /var/backup/addressbook.sql.gz | mysql -u root -p addressbook
Enter password:
[root@server.svandreeva.net ~]# cd /vagrant/provision/server
[root@server.svandreeva.net server]# mkdir -p /vagrant/provision/server/mysql/etc/my.cnf.d
[root@server.svandreeva.net server]# mkdir -p /vagrant/provision/server/mysql/var/backup
[root@server.svandreeva.net server]# cp -R /etc/my.cnf.d/utf8.cnf /vagrant/provision/server/mysql/etc/my.cnf.d/
[root@server.svandreeva.net server]# cp -R /var/backup/* /vagrant/provision/server/mysql/var/backup/
[root@server.svandreeva.net server]# cd /vagrant/provision/server
[root@server.svandreeva.net server]# touch mysql.sh
```

Рис. 3.12: Создание окружения для внесения изменений в настройки окружающей среды

Открыв `mysql.sh` на редактирование, пропишем в нём следующий скрипт (fig. 3.13):

```
root@server:/vagrant/provision/server
#!/bin/bash
echo "Provisioning script $@"
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
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
```

Рис. 3.13: Содержание `mysql.sh`

Для отработки созданного скрипта во время загрузки виртуальной машины `server` в конфигурационном файле `Vagrantfile` добавим в разделе конфигурации для сервера (fig. 3.14):

```
Vagrantfile
C: > work > svandreeva > vagrant > Vagrantfile
4  Vagrant.configure("2") do |config|
24  config.vm.define "server", autostart: false do |server|
38  server.vm.provision "server_dummy",
40      preserve_order: true,
41      path: "provision/server/01-dummy.sh"
42
43  server.vm.provision "server_dns",
44      type: "shell",
45      preserve_order: true,
46      path: "provision/server/dns.sh"
47
48  server.vm.provision "server_dhcp",
49      type: "shell",
50      preserve_order: true,
51      path: "provision/server/dhcp.sh"
52  server.vm.provision "server_http",
53      type: "shell",
54      preserve_order: true,
55      path: "provision/server/http.sh"
56
57  server.vm.provision "server_mysql",
58      type: "shell",
59      preserve_order: true,
60      path: "provision/server/mysql.sh"
61
62
```

Рис. 3.14: Изменение файла Vagrantfile

## 4 Контрольные вопросы

1. Какая команда отвечает за настройки безопасности в MariaDB?

Для настройки безопасности в MariaDB используется команда “GRANT” и “REVOKE”.

2. Как настроить MariaDB для доступа через сеть?

Для доступа через сеть в MariaDB необходимо изменить файл конфигурации my.cnf, указав IP-адрес и порт, а также настроить привилегии пользователей.

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

Обзор доступных баз данных 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 используется команда “SHOW TABLES FROM database\_name;”.

5. Какая команда позволяет узнать, какие поля доступны в таблице?

Чтобы узнать доступные поля 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 используется команда “SELECT \* FROM table\_name;”.

7. Как удалить запись из таблицы?

Для удаления записи из таблицы используется команда “DELETE FROM table\_name WHERE condition;”.

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

Файлы конфигурации MariaDB обычно расположены в /etc/mysql/ или /etc/my.cnf. С их помощью можно настроить параметры сервера, безопасность, память и другие параметры.

9. Где располагаются файлы с базами данных MariaDB?

Файлы с базами данных MariaDB обычно хранятся в каталоге данных, указанном в конфигурационном файле (например, /var/lib/mysql/).

10. Как сделать резервную копию базы данных и затем её восстановить?

Для создания резервной копии базы данных используйте команду “mysqldump”, а для восстановления - команду “mysql < backup\_file.sql”.

## 5 Выводы

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