

Отчет по лабораторной работе 6

Генералов Даниил, НПИбд-01-21, 1032202280

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	15
5	Контрольные вопросы	16

Список иллюстраций

3.1	mariadb	7
3.2	mariadb	8
3.3	mariadb	9
3.4	mariadb	10
3.5	mariadb	11
3.6	mariadb	12
3.7	mariadb	13
3.8	vagrant	14

Список таблиц

1 Цель работы

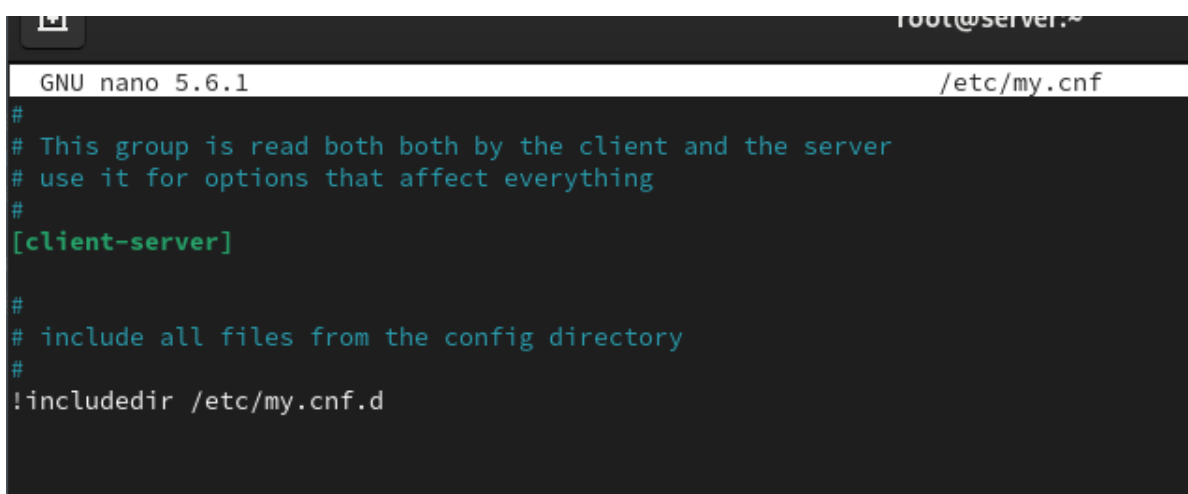
Приобретение практических навыков по установке и конфигурированию системы управления базами данных на примере программного обеспечения MariaDB.

2 Задание

1. Установите необходимые для работы MariaDB пакеты (см. раздел 6.4.1).
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 (см. раздел 6.4.1).
4. Создайте резервную копию базы данных addressbook и восстановите из неё данные (см. раздел 6.4.1).
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 (см. раздел 6.4.5).

3 Выполнение лабораторной работы

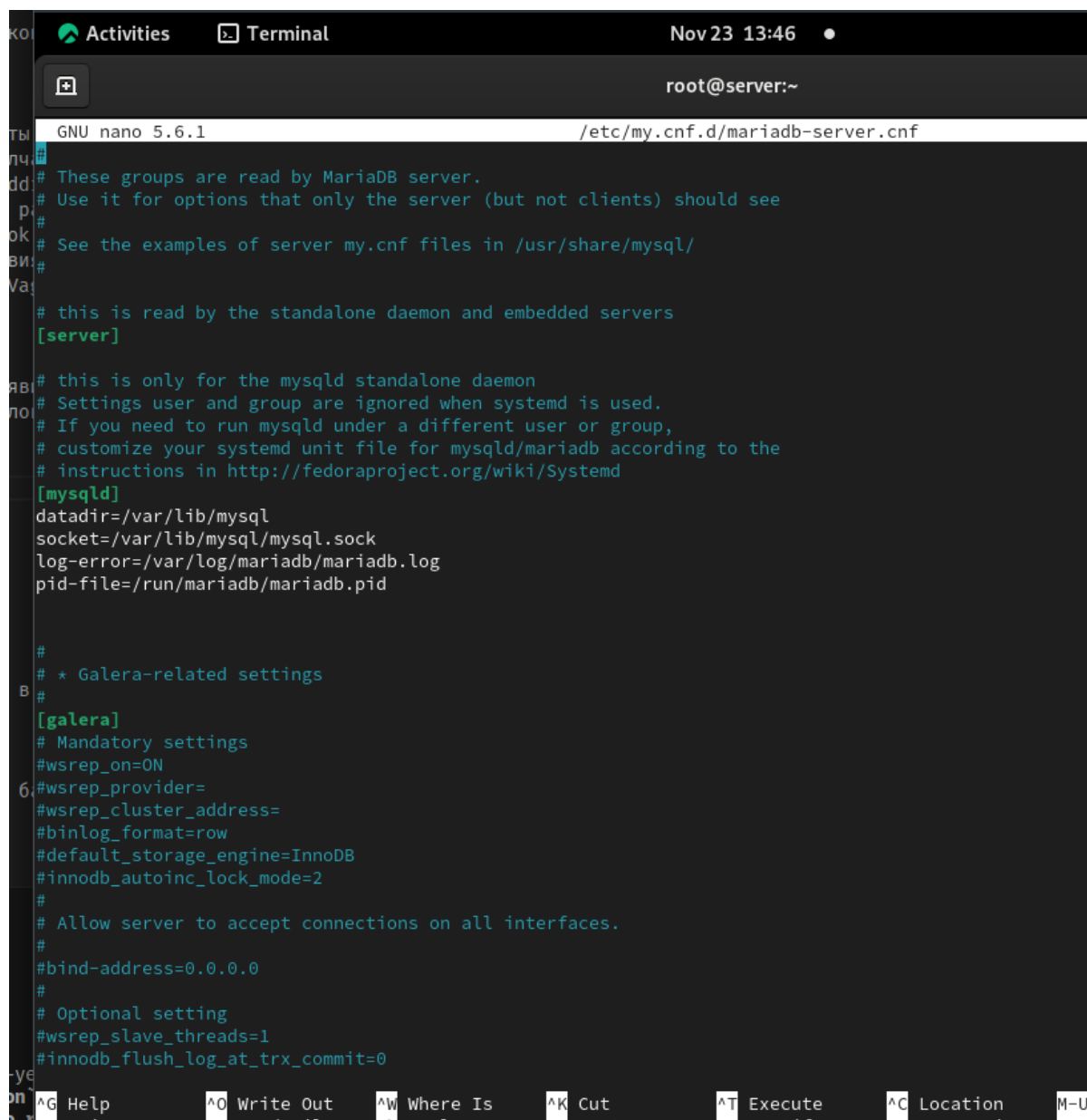
После установки пакетов mariadb и mariadb-server появились файлы конфигурации MariaDB. Файл `/etc/my.cnf` лишь содержит include других файлов



```
GNU nano 5.6.1 /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
```

Рис. 3.1: mariadb

Среди файлов `/etc/my.cnf.d` самый интересный – `mariadb-server.conf`. Он содержит путь к папке с данными, сокету для общения с сервером, PID-файлу и файлу логов.



```
GNU nano 5.6.1 /etc/my.cnf.d/mariadb-server.cnf
#
# 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
```

Рис. 3.2: mariadb

После этого мы запускаем сервер, проверяем, что он слушает соединения, а затем запускаем скрипт для настройки безопасности сервера, где указываем пароль и удаляем тестовые БД и пользователей.


```

root@server ~]# nano /etc/my.cnf.d/mariadb-server.cnf
root@server ~]# systemctl enable --now 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 ~]# ss -tulpen | grep mysql
root@server ~]# ss -tulpen | grep maria
tcp        LISTEN 0      80          *:3306      *:3306      users:((("mariadb",pid=9676,fd=16)) uid:27 ino:42012 sk:13 cgroup=)
*:system.slice/mariadb.service v6only:0 <->
root@server ~]# 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

```

Рис. 3.3: mariadb

Теперь на сервере есть только системные базы данных – mysql, information_schema и performance_schema, которые содержат генерированную информацию про состояние сервера.

```

[root@server ~]# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 13
Server version: 10.5.22-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)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
+-----+
3 rows in set (0.001 sec)

MariaDB [(none)]> exit;
Bye
[root@server ~]#

```

Рис. 3.4: mariadb

Затем мы изменяем сервер, чтобы он использовал UTF-8 кодировку. Изначально он использовал latin1, что можно увидеть в выводе команды status. Она также выводит другую информацию о текущем соединении – например, что оно происходит через UNIX-сокеты и не использует SSL.

```
root@server:~
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:           5 min 5 sec

Threads: 1  Questions: 29  Slow queries: 0  Opens: 20  Open tables: 13  Queries per second avg
-----

MariaDB [(none)]> exit
Bye
[root@server ~]# cat > /etc/my.cnf.d/utf8.cnf
[client]
default-character-set = utf8
[mysqld]
character-set-server = utf8
[root@server ~]# systemctl restart mariadb
[root@server ~]# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 3
Server version: 10.5.22-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)]> 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
```

Рис. 3.5: mariadb

Теперь мы используем командную строку, чтобы создать базу данных, таблицу внутри нее, и записи внутри нее. После этого мы можем получить эти записи обратно с помощью SELECT-запроса, а также создать пользователя и дать ему разрешения на доступ к этой базе данных.

```
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]> INSERT INTO city VALUES ('Иванов', 'Москва') ('Петров','Сочи') ('Сидоров','Дубна');
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near ' ('Петров','Сочи') ('Сидоров','Дубна')' at line 1
MariaDB [addressbook]> INSERT INTO city VALUES ('Иванов', 'Москва'), ('Петров','Сочи'), ('Сидоров','Дубна');
Query OK, 3 rows affected (0.007 sec)
Records: 3 Duplicates: 0 Warnings: 0

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

MariaDB [addressbook]> CREATE USER dmgeneralov@'%' IDENTIFIED BY 'dmgeneralov';
Query OK, 0 rows affected (0.004 sec)

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

Рис. 3.6: mariadb

После этого можно закрыть эту консоль и попробовать использовать нового пользователя и увидеть, что база данных и таблица остались.

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

Рис. 3.7: mariadb

Наконец, мы делаем резервные копии базы данных, проверяем их восстановления, а затем экспортируем все настройки в Vagrant.

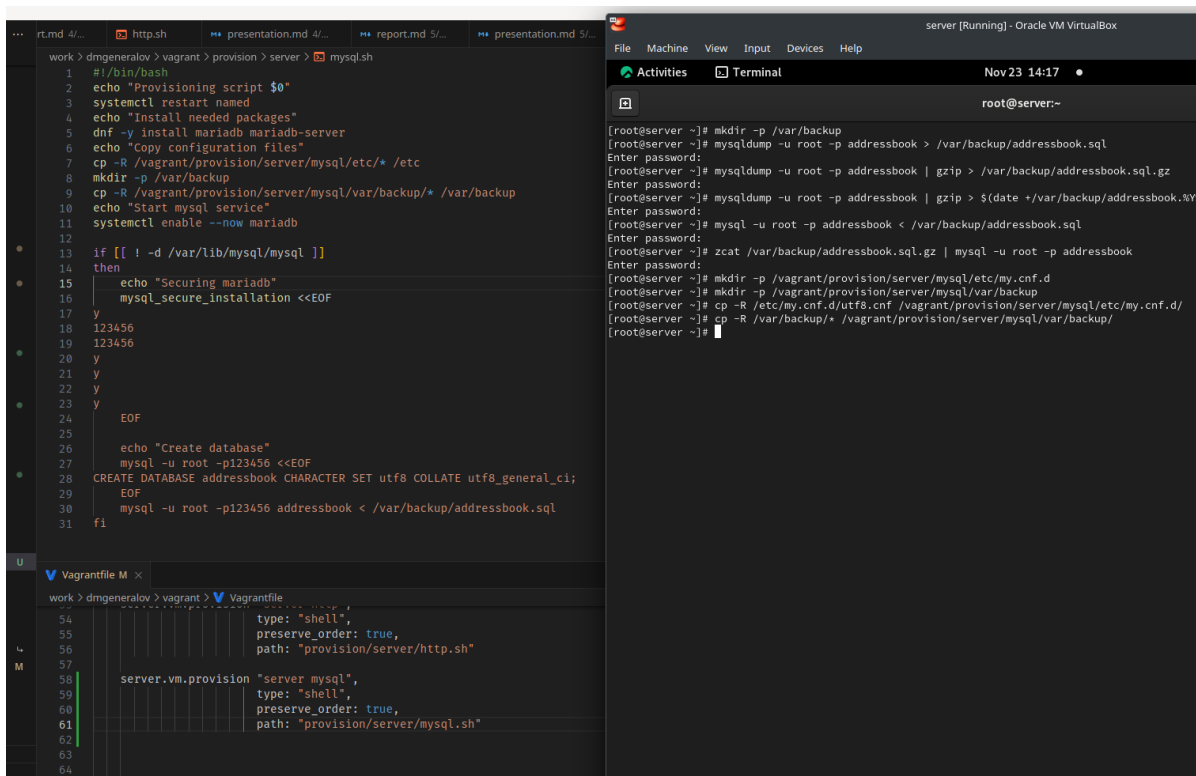


Рис. 3.8: vagrant

4 Выводы

Я получил опыт настройки и использования СУБД MariaDB.

5 Контрольные вопросы

1. Какая команда отвечает за настройки безопасности в MariaDB?

Скрипт `mysql_secure_installation` настраивает часто используемые настройки безопасности, вроде root-пароля, удаления тестовых пользователей и разрешения доступа root только через UNIX-сокеты.

2. Как настроить MariaDB для доступа через сеть?

Нужно создать пользователей кроме root, разрешить им доступ через сеть, а затем настроить сервер, чтобы он слушал по TCP-сокету, и разрешить доступ к порту 3306 в firewall.

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

`SHOW DATABASES` или `mysqlshow`

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

`SHOW TABLES` или `mysqlshow database`

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

`DESCRIBE table`

6. Какая команда позволяет узнать, какие записи доступны в таблице?

`SELECT * FROM table`

7. Как удалить запись из таблицы?

```
DELETE FROM table WHERE id=1
```

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

Они находятся в `/etc/my.cnf.d`, и с помощью них можно настраивать поведение серверов и клиентов. Например, там мы настроили, чтобы они использовали кодировку UTF-8, но там также можно указать путь к файлам базы данных, а также настройки подключения (TCP, UNIX-сокеты, SSL...)

9. Где располагаются файлы с базами данных MariaDB?

В пути, указанном в настройках под ключом `mysqld.datadir`; по умолчанию это `/var/lib/mysql`.

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

```
mysqldump -u root -p database > /tmp/database-backup.sql
```

```
mysql -u root -p database < /tmp/database-backup.sql
```