

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 6

Установка и настройка системы

управления базами данных MariaDB

Дисциплина: Администрирование сетевых подсистем

Студент: Карташова А.С.

Группа: НФИбд-03-18

МОСКВА

2020 г.

Оглавление

Цель работы.....	2
Задачи.....	2
Ход работы.....	3
Установка MariaDB.....	3
Конфигурация кодировки символов	6
Создание базы данных	8
Резервные копии	10
Внесение изменений в настройки внутреннего окружения виртуальной машины	10
Заключение	11
Контрольные вопросы	11

Цель работы

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

Задачи

1. Установим необходимые для работы MariaDB пакеты
2. Настроим в качестве кодировки символов по умолчанию utf8 в базах данных.
3. В базе данных MariaDB создадим тестовую базу addressbook, содержащую таблицу city с полями name и city
4. Создадим резервную копию базы данных addressbook и восстановим из неё данные
5. Напишем скрипт для Vagrant, фиксирующий действия по установке и настройке базы данных MariaDB во внутреннем окружении виртуальной машины server. Соответствующим образом внесем изменения в Vagrantfile

Ход работы

Установка MariaDB

Запустим виртуальную машину, перейдем в режим суперпользователя и установим необходимые для работы с базами данных пакеты.

```
[askartashova@server.askartashova.net ~]$ sudo -i
[sudo] password for askartashova:
[root@server.askartashova.net ~]# dnf -y install mariadb mariadb-server
Last metadata expiration check: 2:20:51 ago on Mon 30 Nov 2020 02:31:31 AM UTC.
Dependencies resolved.
=====
Package                                Arch      Version                                Repository
=====
Installing:
mariadb                                x86_64    3:10.3.17-1.module_el8.1.0+257+48736ea6  AppStream
mariadb-server                          x86_64    3:10.3.17-1.module_el8.1.0+257+48736ea6  AppStream
Installing dependencies:
mariadb-common                          x86_64    3:10.3.17-1.module_el8.1.0+257+48736ea6  AppStream
mariadb-connector-c                     x86_64    3.0.7-1.el8                               AppStream
mariadb-connector-c-config              noarch    3.0.7-1.el8                               AppStream
mariadb-errmsg                          x86_64    3:10.3.17-1.module_el8.1.0+257+48736ea6  AppStream
perl-DBD-MySQL                          x86_64    4.046-3.module_el8.1.0+203+e45423dc      AppStream
Installing weak dependencies:
mariadb-backup                          x86_64    3:10.3.17-1.module_el8.1.0+257+48736ea6  AppStream
mariadb-gssapi-server                   x86_64    3:10.3.17-1.module_el8.1.0+257+48736ea6  AppStream
mariadb-server-utils                    x86_64    3:10.3.17-1.module_el8.1.0+257+48736ea6  AppStream
Enabling module streams:
mariadb                                10.3
perl-DBD-MySQL                         4.046
Transaction Summary
=====
Install 10 Packages
```

Просмотрим конфигурационные файлы mariadb в каталоге /etc/my.cnf.d и в файле /etc/my.cnf и запустим mariadb

```
/etc/my.cnf 202/202
#
# 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
```

my.cnf -конфигурационный файл Mysql. В нем находятся настройки, влияющие и на клиент и на сервер.Сейчас он работает с файлами из каталога /etc/my.cnf.d

```
/etc/my.cnf.d/auth_gssapi.cnf
[mariadb]
plugin-load-add=auth_gssapi.so
```

```
/etc/my.cnf.d/client.cnf
#
# 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]
```

```
/etc/my.cnf.d/enable_encryption.preset 763/763
#
# !include this file into your my.cnf (or any of *.cnf files in /etc/my.cnf.d)
# and it will enable data at rest encryption. This is a simple way to
# ensure that everything that can be encrypted will be and your
# data will not leak unencrypted.
#
# DO NOT EDIT THIS FILE! On MariaDB upgrades it might be replaced with a
# newer version and your edits will be lost. Instead, add your edits
# to the .cnf file after the !include directive.
#
# NOTE that you also need to install an encryption plugin for the encryption
# to work. See https://mariadb.com/kb/en/mariadb/data-at-rest-encryption/#encryption-key-management
#
[mariadb]
aria-encrypt-tables
encrypt-binlog
encrypt-tmp-disk-tables
encrypt-tmp-files
loose-innodb-encrypt-log
loose-innodb-encrypt-tables
```

```
/etc/my.cnf.d/mysql-clients.cnf
#
# These groups are read by MariaDB command-line tools
# Use it for options that affect only one utility
#
[mysql]
[mysql_upgrade]
[mysqladmin]
[mysqlbinlog]
[mysqlcheck]
[mysqldump]
[mysqlimport]
[mysqlshow]
[mysqlslap]
```

Auth_gssapi.cnf – отвечает за использование общего интерфейса службы безопасности

Client.cnf – конфигурация только клиента

Запустим программное обеспечение и убедимся, что mariadb прослушивает порт 3306:

ss -tulpen | grep mysql

```
[root@server.askartashova.net ~]# systemctl start mariadb
[root@server.askartashova.net ~]# 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.askartashova.net ~]# ss -tulpen | grep mysql
tcp    LISTEN  0      80          *:3306          *:              users:((("mysqld",pid=19076,fd=21)) uid:27 ino:184087 sk:27 v6only:0 <->
```

Запустим скрипт конфигурации безопасности mariadb, используя:

mysql_secure_installation

```
[root@server.askartashova.net ~]# systemctl start mariadb
[root@server.askartashova.net ~]# 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.askartashova.net ~]# ss -tulpen | grep mysql
tcp    LISTEN  0      80          *:3306          *:              users:((("mysqld",pid=19076,fd=21)) uid:27 ino:184087 sk:27 v6only:0 <->
```

Войдём базу данных с правами администратора

mysql -u root -p

```
Thanks for using MariaDB!
[root@server.askartashova.net ~]# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 18
Server version: 10.3.17-MariaDB MariaDB Server

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

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
MariaDB [(none)]> \h
```

Просмотрим список команд MySQL, введя \h.

```

List of all MySQL commands:
Note that all text commands must be first on line and end with ';'
?      (?) 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 mysql server, display result vertically.
exit     (\q) Exit mysql. Same as quit.
go       (\g) Send command to mysql 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 cha
warnings  (\W) Show warnings after every statement.
nowarning (\w) Don't show warnings after every statement.

```

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

SHOW DATABASES;

```

MariaDB [(none)]> show databases
-> ;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
+-----+
3 rows in set (0.000 sec)

```

Базы данных системы:

- information_schema
- mysql
- performance_schema

Выйдем из интерфейса интерактивной оболочки

```

MariaDB [(none)]> exit
Bye
[root@server.askartashova.net ~]# S

```

Конфигурация кодировки символов

Войдем в базу данных с правами администратора:

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

Для отображения статуса MariaDB введем из приглашения интерактивной оболочки MariaDB:

Status

```
-----
mysql Ver 15.1 Distrib 10.3.17-MariaDB, for Linux (x86_64) using readline 5.1

Connection id:          19
Current database:
Current user:           root@localhost
SSL:                    Not in use
Current pager:          stdout
Using outfile:          ''
Using delimiter:        ;
Server:                 MariaDB
Server version:         10.3.17-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:                 13 min 55 sec

Threads: 7  Questions: 22  Slow queries: 0  Opens: 17  Flush tables: 1  Open tables: 11  Queries per sec
g: 0.026
-----
```

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, войдем в базу данных с правами администратора и посмотрите статус MariaDB.

```
systemctl restart mariadb
```

```

mysql Ver 15.1 Distrib 10.3.17-MariaDB, for Linux (x86_64) using readline 5.1

Connection id:          8
Current database:
Current user:           root@localhost
SSL:                    Not in use
Current pager:          stdout
Using outfile:          ''
Using delimiter:        ;
Server:                 MariaDB
Server version:         10.3.17-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:                 14 sec

Threads: 7  Questions: 4  Slow queries: 0  Opens: 17  Flush tables: 1  Open tables: 11  Queries per sec
: 0.285
-----

```

Теперь для кодирования символов и на клиенте и на сервере используется стандарт utf8

Создание базы данных

Войдем в базу данных с правами администратора:

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

Создадим базу данных с именем addressbook, перейдем к ней и отобразим имеющиеся в базе таблицы

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

```
USE addressbook;
```

```
SHOW TABLES;
```

Создадим таблицу city с полями name и city и заполним несколько строк таблицы:

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

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

Сделаем следующий MySQL-запрос:

```
SELECT * FROM city;
```

Запрос выводит все столбцы таблицы

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

```
CREATE USER askartashova@'%' IDENTIFIED BY '123456';
```

Предоставим права доступа созданному пользователю user на действия с базой данных addressbook и обновим привилегии

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

```
FLUSH PRIVILEGES;
```


Посмотрим общую информацию о таблице city базы данных addressbook и выйдем из окружения MariaDB:

```
DESCRIBE city;  
quit
```

Просмотрим список баз данных, список таблиц базы данных addressbook

```
-----  
mysql Ver 15.1 Distrib 10.3.17-MariaDB, for Linux (x86_64) using readline 5.1  
  
Connection id:          8  
Current database:        
Current user:         root@localhost  
SSL:                   Not in use  
Current pager:         stdout  
Using outfile:         ''  
Using delimiter:       ;  
Server:                MariaDB  
Server version:        10.3.17-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:                14 sec  
  
Threads: 7  Questions: 4  Slow queries: 0  Opens: 17  Flush tables: 1  Open tables: 11  Queries per sec  
: 0.285  
-----
```

```
MariaDB [addressbook]> GRANT SELECT,INSERT,UPDATE,DELETE ON addressbook.* TO askartashova@'%';  
Query OK, 0 rows affected (0.000 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    |       |  
+-----+-----+-----+-----+-----+-----+  
2 rows in set (0.001 sec)  
  
MariaDB [addressbook]> mysqlshow -u root -p  
-> exit  
-> ;  
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MariaDB se  
rver version for the right syntax to use near 'mysqlshow -u root -p  
exit' at line 1  
MariaDB [addressbook]> quit  
Bye
```

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

Резервные копии

На виртуальной машине server создадим каталог для резервных копий

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

```
[root@server.askartashova.net my.cnf.d]# mkdir -p /var/backup
[root@server.askartashova.net my.cnf.d]# mysqldump -u root -p addressbook >
Enter password:
[root@server.askartashova.net my.cnf.d]# mysqldump -u root -p addressbook |
.gz
Enter password:
```

Восстановим базу данных addressbook из резервной копии, и из сжатой резервной копии.

```
[askartashova@server.askartashova.net ~]$ sudo -i
[sudo] password for askartashova:
[root@server.askartashova.net ~]# mysqldump -u root -p addressbook | gzip >
%Y%m%d.%H%M%S.sql.gz)
Enter password:
[root@server.askartashova.net ~]# mysql -u root -p addressbook < /var/backu
Enter password:
[root@server.askartashova.net ~]# zcat /var/backup/addressbook.sql.gz | mys
Enter password:
ERROR 1046 (3D000) at line 22: No database selected
[root@server.askartashova.net ~]# zcat /var/backup/addressbook.sql.gz | mys
Enter password:
[root@server.askartashova.net ~]#
```

Внесение изменений в настройки внутреннего окружения виртуальной машины

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.

В каталоге /vagrant/provision/server создадим исполняемый файл mysql.sh, пропишем в нём следующий скрипт, который повторяет произведённые действия по установке и настройке сервера баз данных.

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

```
server.vm.provision "server mysql",
type: "shell",
```

```
preserve_order: true,  
path: "provision/server/mysql.sh"
```

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

```
GNU nano 2.9.8 mysql.sh Modified  
  
#!/bin/bash  
echo "Provisioning script $0"  
systemctl restart named  
echo "Install needed packages"  
dnf -y install mariadb mariadb-server  
echo "Copy configuration files"  
cp -R /vagrant/provision/server/mysql/etc/* /etc  
mkdir -p /var/backup  
cp -R /vagrant/provision/server/mysql/var/backup/* /var/backup  
echo "Start mysql service"  
systemctl enable mariadb  
systemctl start mariadb  
if [[ ! -d /var/lib/mysql/mysql ]]  
then  
echo "Securing mariadb"  
mysql_secure_installation <<EOF  
y  
123456  
123456  
y  
y  
y  
y  
EOF  
echo "Create database"  
mysql -u root -p123456 <<EOF  
  
CREATE DATABASE addressbook CHARACTER SET utf8 COLLATE utf8_general_ci;
```

Заключение

Мы приобрели практические навыки по установке по установке и конфигурированию системы управления базами данных на примере программного обеспечения MariaDB.

Контрольные вопросы

1. Какая команда отвечает за настройки безопасности в MariaDB?

mysql_secure_installation – запускает скрипт конфигурации безопасности

2. Как настроить MariaDB для доступа через сеть?

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

`SHOW DATABASES`

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

`mysqlshow -u root -p databasename`

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

`DESCRIBE tablename`

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

`SELECT * FROM tablename;`

8. Как удалить запись из таблицы?

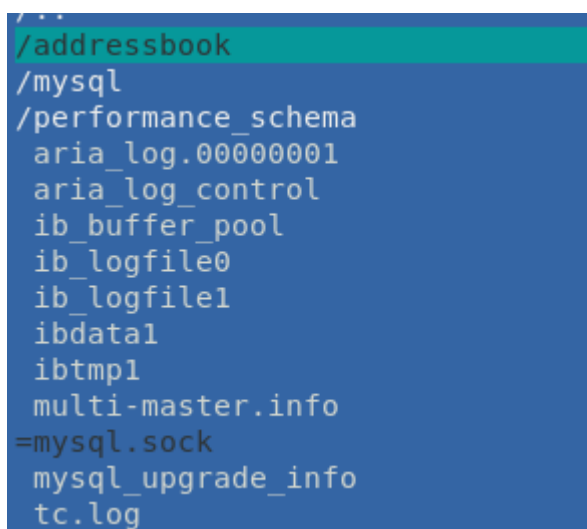
`DELETE FROM tablename WHERE deletion conditions`

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

Они расположены в каталоге `/etc/my.cnf.d` и в файле `/etc/my.cnf`. Выполнить конфигурирование клиента и сервера, подключить интерфейс службы безопасности, зашифровать базу данных

9. Где располагаются файлы с базами данных MariaDB?

В каталоге `/var/lib/mysql`



```
/.  
/addressbook  
/mysql  
/performance_schema  
aria_log.00000001  
aria_log_control  
ib_buffer_pool  
ib_logfile0  
ib_logfile1  
ibdata1  
ibtmp1  
multi-master.info  
mysql.sock  
mysql_upgrade_info  
tc.log
```

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

С помощью команд

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

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