

Лабораторная работа №6

Установка и настройка системы управления базами данных MariaDB

Андреева С.В.

Группа НПИбд-01-23

Российский университет дружбы народов, Москва, Россия

Информация

- Андреева Софья Владимировна
- Группа НПИбд-01-23
- Российский университет дружбы народов

Вводная часть

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

```
[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 | 410 kB      00:01 ET
```

Рис. 1: Установка пакетов для MariaDB

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

```
[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. 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: /etc/my.cnf

Запускаем и включаем программное обеспечение mariadb Убедимся, что mariadb прослушивает порт

```
[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/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.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/mariadbd
tcp6	0	0	:::3306	:::*	LISTEN	27	48539	6258/mariadbd

Рис. 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:

Выполнение работы

```
[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
```

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:

Выполнение работы

```
?      (\?) Synonym for 'help'.
charset (\G) 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 SEDITOR.
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)

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

Выполнение работы

```
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
```


Выполнение работы

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

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

```
[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
```

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

Выполнение работы

```
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)

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

```
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. 10: Создание пользователя и предоставление ему прав для работы с addressbook

Выполнение работы

Выйдем из окружения MariaDB и посмотрим список баз данных, а затем посмотрим список таблиц базы данных addressbook

```
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 |
```

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`


```
[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 +%var/backup/addressbook.%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 +%var/backup/addressbook.%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. 12: Создание окружения для внесения изменений в настройки окружающей среды

Открыв `mysql.sh` на редактирование, пропишем в нём следующий скрипт

```
root@server:/vagrant/provision/ser

#!/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;
EOF
mysql -u root -p123456 addressbook < /var/backup/addressbook.sql
ft
~
```

Для отработки созданного скрипта во время загрузки виртуальной машины `server` в конфигурационном файле `Vagrantfile` добавим в разделе конфигурации для сервера

Выполнение работы

```
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
```

Выводы

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