

# Лабораторная работа №6

Администрирование сетевых подсистем

---

Ищенко Ирина НПИбд-02-22

Российский университет дружбы народов, Москва, Россия

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

## Выполнение лабораторной работы

---

```
root@server:~  
[ioithenko@server ~]$ sudo -i  
[sudo] password for ioithenko:  
[root@server ~]# dnf -y install mariadb mariadb-server  
Extra Packages for Enterprise Linux 9 - x86_64 22 kB/s | 39 kB 00:01  
Extra Packages for Enterprise Linux 9 - x86_64 1.2 MB/s | 23 MB 00:19  
Rocky Linux 9 - BaseOS 7.9 kB/s | 4.1 kB 00:00  
Rocky Linux 9 - BaseOS 1.4 MB/s | 2.3 MB 00:01  
Rocky Linux 9 - AppStream 2.8 kB/s | 4.5 kB 00:01  
Rocky Linux 9 - AppStream 1.5 MB/s | 8.0 MB 00:05  
Rocky Linux 9 - Extras 6.7 kB/s | 2.9 kB 00:00  
Dependencies resolved.  
=====
```

Package	Arch	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	1.0.10-1.el9	appstream	36 k
perl-DBD-MariaDB	x86_64	1.21-16.el9_0	appstream	151 k

Рис. 1: Установка пакетов

```
[root@server ~]# systemctl start mariadb  
[root@server ~]# systemctl enable mariadb
```

Рис. 2: Запуск ПО

```
/etc/passwd v6only:0 <->  
[root@server ~]# ss -tulpen | grep 3306  
tcp    LISTEN 0      80          *:3306      *:~         users:(("mariadb",pid=10047,fd=19)) uid:27 ino:43042 sk:14 cgroup:/system.slice/mariadb.servic  
e v6only:0 <->
```

Рис. 3: Прослушивание порта

```
[1] ... stopped ... mysql_secure_installation
[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...

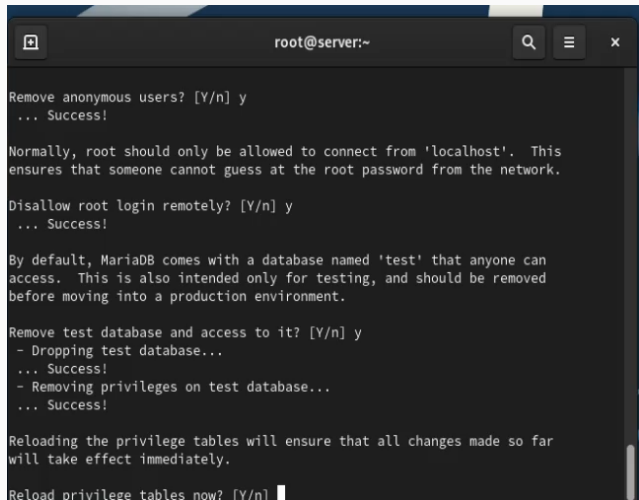
Setting the root password or using the unix_socket ensures that nobody
can log into the MariaDB root user without the proper authorisation.

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

Switch to unix_socket authentication [Y/n] n
... skipping.

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

Рис. 4: Скрипт конфигурирования безопасности

A terminal window titled 'root@server:~' with search, menu, and close buttons. It displays the output of a MariaDB security script. The script prompts for removing anonymous users, disallowing root login remotely, and removing the test database. All actions are confirmed with 'y' and succeed. It also informs about reloading privilege tables.

```
root@server:~  
Remove anonymous users? [Y/n] y  
... Success!  
  
Normally, root should only be allowed to connect from 'localhost'. This  
ensures that someone cannot guess at the root password from the network.  
  
Disallow root login remotely? [Y/n] y  
... Success!  
  
By default, MariaDB comes with a database named 'test' that anyone can  
access. This is also intended only for testing, and should be removed  
before moving into a production environment.  
  
Remove test database and access to it? [Y/n] y  
- Dropping test database...  
... Success!  
- Removing privileges on test database...  
... Success!  
  
Reloading the privilege tables will ensure that all changes made so far  
will take effect immediately.  
  
Reload privilege tables now? [Y/n] █
```

Рис. 5: Скрипт конфигурирования безопасности



For server side help, type 'help contents'

MariaDB [(none)]> SHOW DATABASES;

Database
information_schema
mysql
performance_schema

3 rows in set (0.002 sec)

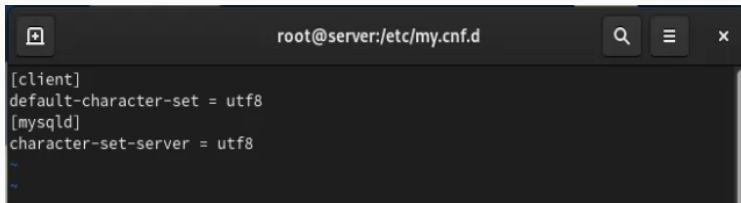
MariaDB [(none)]> exit;

Bye

Рис. 6: Отображение БД

```
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 wrap  
per  
  
Connection id:          11  
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:                 11 min 49 sec
```

Рис. 7: Статус

A terminal window with a dark background. The title bar at the top shows a window icon, the text 'root@server:/etc/my.cnf.d', a search icon, a menu icon, and a close icon. The terminal content shows two configuration sections: '[client]' with 'default-character-set = utf8' and '[mysqld]' with 'character-set-server = utf8'. There are some faint blue lines at the bottom of the terminal.

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

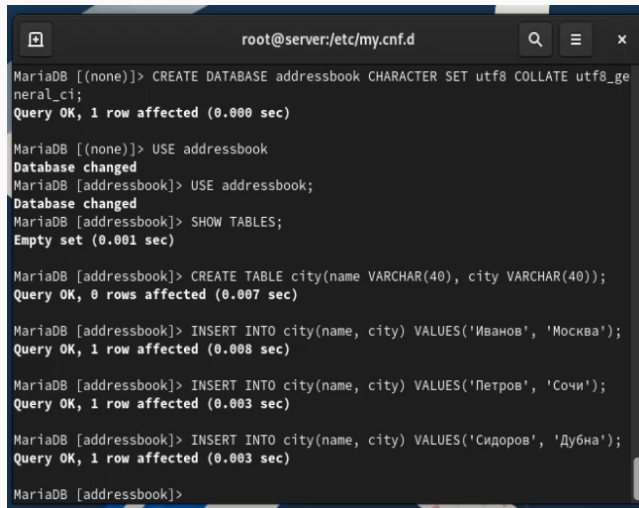
Рис. 8: Отображение БД

```
-----
mysql Ver 15.1 Distrib 10.5.22-MariaDB, for Linux (x86_64) using EditLine wrap
per

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:                  20 sec

Threads: 1  Questions: 4  Slow queries: 0  Opens: 17  Open tables: 10  Queries p
er second avg: 0.200
```

Рис. 9: Статус



```
root@server:/etc/my.cnf.d
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]> USE addressbook;
Database changed
MariaDB [addressbook]> SHOW TABLES;
Empty set (0.001 sec)

MariaDB [addressbook]> CREATE TABLE city(name VARCHAR(40), city VARCHAR(40));
Query OK, 0 rows affected (0.007 sec)

MariaDB [addressbook]> INSERT INTO city(name, city) VALUES('Иванов', 'Москва');
Query OK, 1 row affected (0.008 sec)

MariaDB [addressbook]> INSERT INTO city(name, city) VALUES('Петров', 'Сочи');
Query OK, 1 row affected (0.003 sec)

MariaDB [addressbook]> INSERT INTO city(name, city) VALUES('Сидоров', 'Дубна');
Query OK, 1 row affected (0.003 sec)

MariaDB [addressbook]>
```

Рис. 10: Создание и заполнение таблицы

```
MariaDB [addressbook]> SELECT * FROM city;
```

name	city
Иванов	Москва
Петров	Сочи
Сидоров	Дубна

3 rows in set (0.001 sec)

Рис. 11: Вывод таблицы

```
root@server:/etc/my.cnf.d
MariaDB [addressbook]> CREATE USER ioithenko@'%' IDENTIFIED BY 'password';
Query OK, 0 rows affected (0.005 sec)

MariaDB [addressbook]> GRANT SELECT,INSERT,UPDATE,DELETE ON addressbook.* TO ioi
thenko@'%';
Query OK, 0 rows affected (0.003 sec)

MariaDB [addressbook]> FLUSH PRIVILIGES;
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 'PRI
VILIGES' at line 1
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 (0.001 sec)

MariaDB [addressbook]>
```

Рис. 12: Создание пользователя, права доступа

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

Рис. 13: Список БД



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

[root@server my.cnf.d]# mysqlshow -u ioithenko -p addressbook
Enter password:
Database: addressbook
+-----+
| Tables |
+-----+
| city  |
+-----+
```

Рис. 14: Список таблиц БД

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

Рис. 15: Копии БД

```
[root@server my.cnf.d]# mysqldump -u root -p addressbook | gzip > $(date +%Y%m%d.%H%M%S).sql.gz
ackup/addressbook.%Y%m%d.%H%M%S.sql.gz)
Enter password:
[root@server my.cnf.d]# ls /var/backup
addressbook.20241007.163359.sql.gz  addressbook.sql  addressbook.sql.gz
[root@server my.cnf.d]#
```

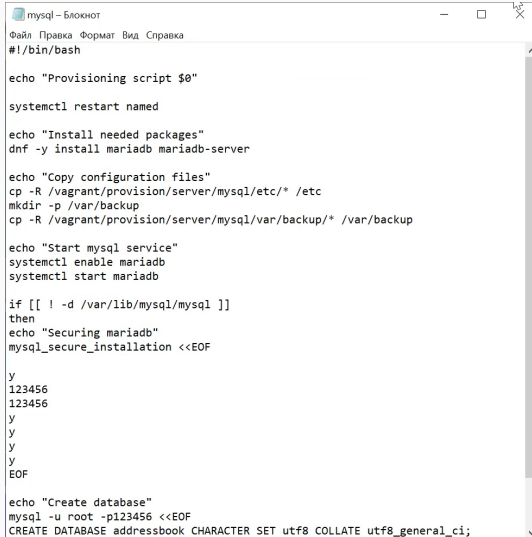
Рис. 16: Копия БД с датой

```
Enter password:  
[root@server my.cnf.d]# zcat /var/backup/addressbook.sql.gz | mysql -u root -p addressbook  
Enter password:  
[root@server my.cnf.d]#
```

Рис. 17: Восстановление БД

```
no such file or directory  
[root@server server]# mkdir -p /vagrant/provision/server/mysql/etc/my.cnf.d  
[root@server server]# cp -R /etc/my.cnf.d/utf8.cnf /vagrant/provision/server/mysql/etc/my.cnf.d/  
[root@server server]# mkdir -p /vagrant/provision/server/mysql/var/backup  
[root@server server]# cp -R /var/backup/* /vagrant/provision/server/mysql/var/backup/  
[root@server server]#
```

Рис. 18: Настройки внутреннего окружения



```
mysql - Блокнот
Файл Правка Формат Вид Справка
#!/bin/bash

echo "Provisioning script $0"

systemctl restart named

echo "Install needed packages"
dnf -y install mariadb mariadb-server

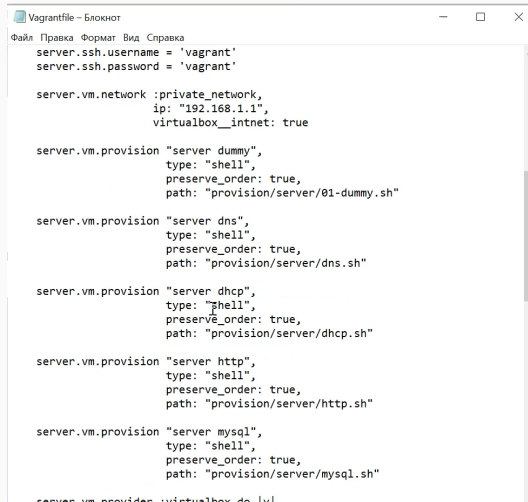
echo "Copy configuration files"
cp -R /vagrant/provision/server/mysql/etc/* /etc
mkdir -p /var/backup
cp -R /vagrant/provision/server/mysql/var/backup/* /var/backup

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;
```

Рис. 19: mysql.sh

A screenshot of a Notepad window titled "Vagrantfile - Блокнот". The window contains a Vagrantfile script. The script defines the SSH username and password for a VM, sets the network configuration (private network, IP 192.168.1.1, and virtualbox\_\_\_\_intnet: true), and defines several provisioners for the VM: "server dummy", "server dns", "server dhcp", "server http", and "server mysql". Each provisioner is configured with a type of "shell", preserve\_order: true, and a specific path to a shell script in the "provision/server/" directory. The script is written in a Ruby-like syntax with indentation.

```
Vagrantfile - Блокнот
Файл Правка Формат Вид Справка
server.ssh.username = 'vagrant'
server.ssh.password = 'vagrant'

server.vm.network :private_network,
  ip: "192.168.1.1",
  virtualbox__intnet: true

server.vm.provision "server dummy",
  type: "shell",
  preserve_order: true,
  path: "provision/server/01-dummy.sh"

server.vm.provision "server dns",
  type: "shell",
  preserve_order: true,
  path: "provision/server/dns.sh"

server.vm.provision "server dhcp",
  type: "shell",
  preserve_order: true,
  path: "provision/server/dhcp.sh"

server.vm.provision "server http",
  type: "shell",
  preserve_order: true,
  path: "provision/server/http.sh"

server.vm.provision "server mysql",
  type: "shell",
  preserve_order: true,
  path: "provision/server/mysql.sh"

server.vm.provider :virtualbox do |v|
```

Рис. 20: Vagrantfile

В ходе лабораторной работы я приобрела практических навыков по установке и конфигурированию системы управления базами данных на примере программного обеспечения MariaDB.