**РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ ИМЕНИ ПАТРИСА ЛУМУМБЫ**

**Факультет физико-математических и естественных наук**

**Кафедра теории вероятностей и кибербезопасности**

ОТЧЕТ

по лабораторной работе № 6

Дисциплина «Администрирование сетевых подсистем»

*Тема «Установка и настройка системы управления базами данных MariaDB»*

Студент: Щербак Маргарита Романовна

Ст. билет: 1032216537

Группа: НПИбд-02-21

**МОСКВА**

2023 г.

# Цель работы

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

# Задание

1. Установить необходимые для работы MariaDB пакеты.

2. Настроить в качестве кодировки символов по умолчанию utf8 в базах данных.

3. В базе данных MariaDB создать тестовую базу addressbook, содержащую таблицу city с полями name и city, т.е., например, для некоторого сотрудника указан город, в котором он работает.

4. Создать резервную копию базы данных addressbook и восстановить из неё данные.

5. Написать скрипт для Vagrant, фиксирующий действия по установке и настройке базы данных MariaDB во внутреннем окружении виртуальной машины server. Соответствующим образом внести изменения в Vagrantfile.

**Выполнение**

**1. Установка MariaDB**

1. Загрузила свою ОС и перешла в рабочий каталог с проектом, затем запустила виртуальную машину server с помощью команды vagrant up server (рис.1.1).

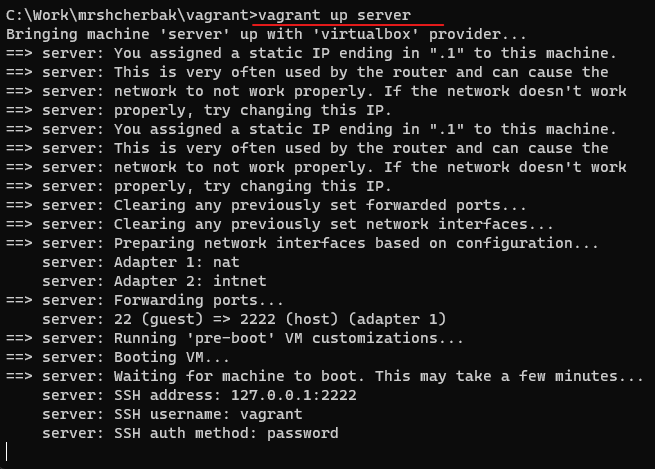


Рис.1.1. Запуск виртуальной машины Server

2. На виртуальной машине server вошла под своим пользователем и открыла терминал. Перешла в режим суперпользователя: sudo –i. Установила необходимые для работы с базами данных пакеты: dnf -y install mariadb mariadb-server (рис.1.2).

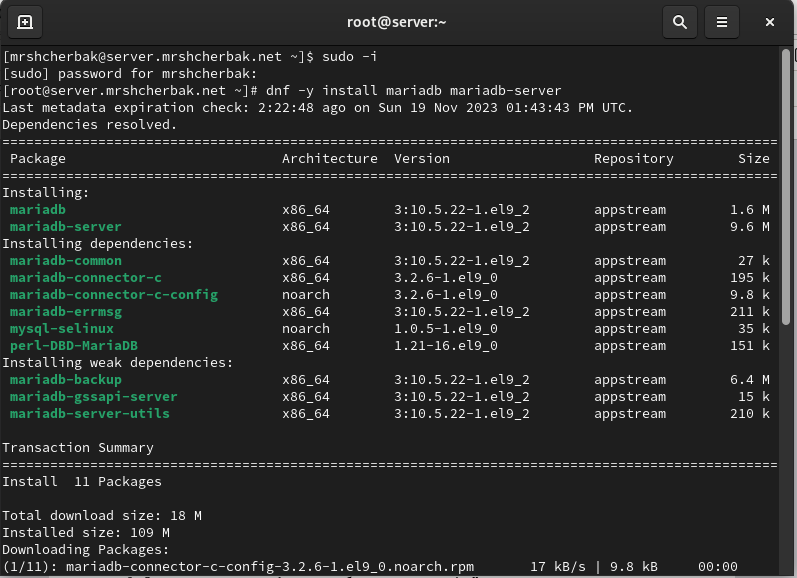


Рис.1.2. Установка пакетов для работы с базами данных

3. Просмотрела конфигурационные файлы mariadb в каталоге /etc/my.cnf.d и в файле /etc/my.cnf (рис.1.3 – рис.1.10).

Файл auth\_gssapi.cnf:

[mariadb]: секция конфигурации, которая определяет, что следующие настройки относятся к MariaDB.

#plugin-load-add=auth\_gssapi.so: закомментированная строка, что означает, что эта строка в настоящее время не активна. Она используется для загрузки плагина auth\_gssapi.so, который отвечает за поддержку аутентификации GSSAPI.

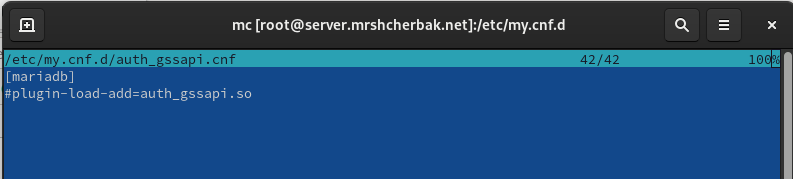


Рис.1.3. Просмотр конфигурационного файла auth\_gssapi.cnf

Файл client.cnf:

[client]: секция конфигурации предназначена для параметров, связанных с клиентской частью MariaDB/MySQL. Эти параметры могут влиять на поведение командной строки и инструментов, использующих соединение с базой данных.

[client-mariadb]: дополнительная секция конфигурации, которая позволяет определить параметры, которые будут применяться только к клиенту MariaDB, а не к общему клиенту MySQL.

В файле отсутствуют конкретные параметры. Фактические параметры расположены ниже каждой секции и определяют конкретные настройки клиента.

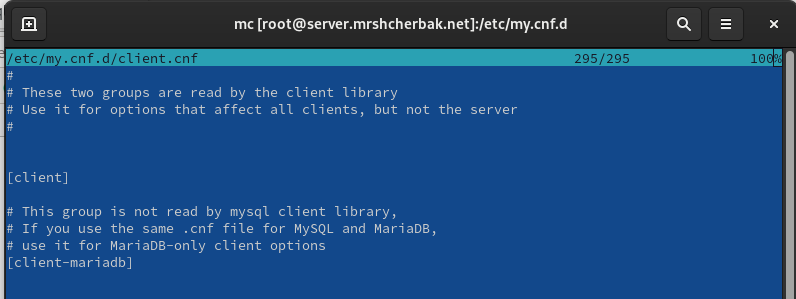


Рис.1.4. Просмотр конфигурационного файла client.cnf

Файл enable\_encryption.preset:

[mariadb]: строка указывает на начало секции конфигурации для MariaDB. Все последующие строки, до следующей секции или конца файла, относятся к настройкам MariaDB.

aria-encrypt-tables: параметр включает шифрование для таблиц, использующих Aria Storage Engine. Aria ⎯ хранилище, которое можно использовать в MariaDB.

encrypt-binlog: параметр включает шифрование для бинарных журналов (binary logs), которые записывают изменения данных в базе данных. Это может быть полезным для обеспечения конфиденциальности данных в бинарных журналах.

encrypt-tmp-disk-tables: параметр включает шифрование для временных таблиц, которые хранятся на диске.

encrypt-tmp-files: параметр включает шифрование для временных файлов, что способствует обеспечению безопасности временных данных.

loose-innodb-encrypt-log: параметр относится к InnoDB, одному из основных хранилищ данных в MariaDB. Он указывает, что шифрование следует применять к журналу InnoDB.

loose-innodb-encrypt-tables: параметр указывает, что шифрование следует применять к таблицам InnoDB.



Рис.1.5. Просмотр файла enable\_encryption.preset

Файл mariadb-server.cnf:

[mariadb-server]: обозначение секции конфигурации, где описываются настройки для MariaDB сервера.

datadir: параметр, указывающий на директорию, в которой хранятся данные базы данных.

socket: указывает путь к сокету, через который происходит взаимодействие между клиентом и сервером.

log-error: путь к файлу, в который записываются сообщения об ошибках.

pid-file: путь к файлу, в котором хранятся Process ID (PID) сервера.

port: номер порта, на котором сервер слушает входящие соединения.

bind-address: указывает IP-адрес, на котором сервер слушает входящие соединения.

character-set-server: устанавливает кодировку символов по умолчанию для сервера.

collation-server: устанавливает коллацию по умолчанию для сервера. Например,

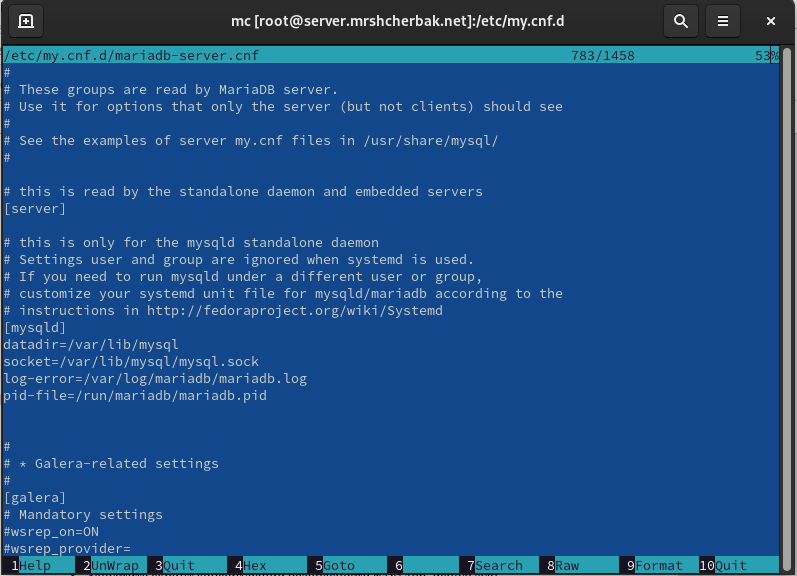


Рис.1.6. Просмотр конфигурационного файла mariadb-server.cnf

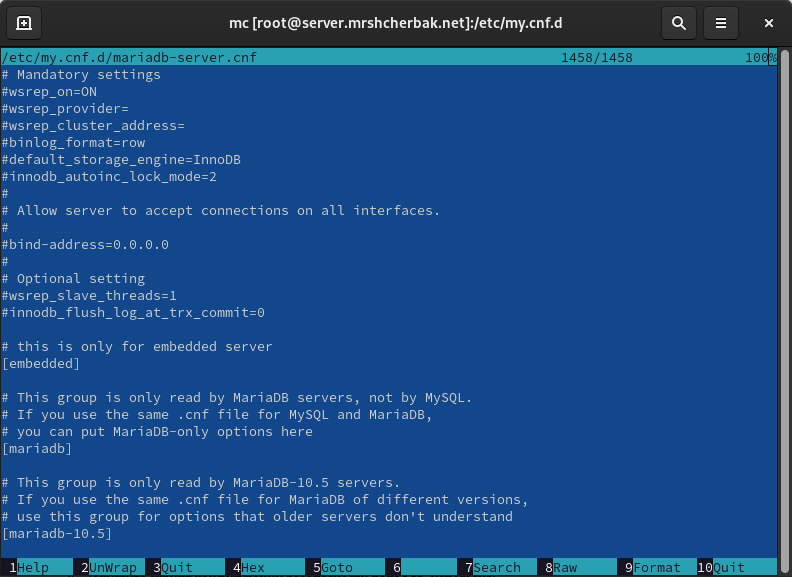


Рис.1.7. Просмотр конфигурационного файла mariadb-server.cnf

Файл mysql-clients.cnf:

[mysql]: секция конфигурации применяется к утилите mysql. Возможные параметры в этой секции могут включать определение хоста (host), пользователя (user), пароля (password), порта (port), сокета (socket), кодировки символов (default-character-set), и другие параметры, связанные с настройками подключения.

[mysql\_upgrade]: секция конфигурации может содержать параметры, связанные с

утилитой mysql\_upgrade, которая используется для обновления системы баз данных после обновления сервера MySQL/MariaDB.

[mysqladmin]: секция конфигурации применяется к утилите mysqladmin. Задаются параметры, такие как хост, пользователь, пароль и другие опции для взаимодействия с сервером.

[mysqlbinlog]: секция конфигурации может содержать параметры для утилиты mysqlbinlog, которая используется для чтения бинарных логов MySQL.

[mysqlcheck]: секция конфигурации применяется к утилите mysqlcheck, предназначенной для проверки и ремонта таблиц в базе данных.

[mysqldump]: секция конфигурации применяется к утилите mysqldump, используемой для создания резервных копий баз данных MySQL/MariaDB.

[mysqlshow]: секция конфигурации может содержать параметры для утилиты mysqlshow, которая используется для отображения информации о базах данных, таблицах и т. д.

[mysqlslap]: секция конфигурации применяется к утилите mysqlslap, предназначенной для тестирования производительности MySQL/MariaDB.

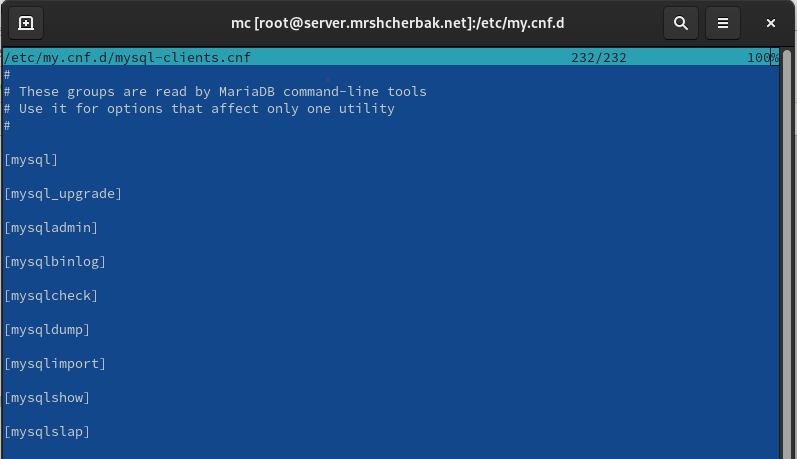


Рис.1.8. Просмотр конфигурационного файла mysql-clients.cnf

Файл spider.cnf:

plugin-load-add: параметр, который указывает серверу MySQL/MariaDB загружать

дополнительные плагины. В данном случае, ha\_spider является именем плагина.

ha\_spider: название плагина, который представляет собой Spider Storage Engine.

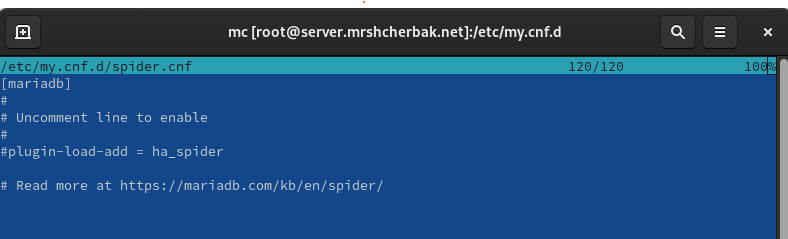


Рис.1.9. Просмотр конфигурационного файла spider.cnf

Файл my.cnf:

[client-server]: секция конфигурации предназначена для параметров, связанных с клиентской частью MariaDB/MySQL. Эти параметры могут влиять на поведение командной строки и инструментов, использующих соединение с базой данных.

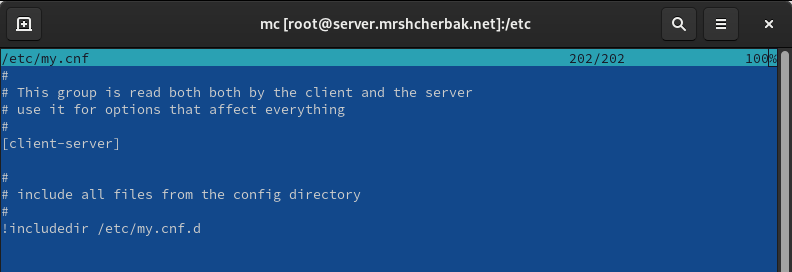


Рис.1.10. Просмотр конфигурационного файла my.cnf

4. Запустила и включила программное обеспечение mariadb. Убедилась, что mariadb прослушивает (увидела процесс mysqld, прослушивающий порт 3306). Запустила скрипт конфигурации безопасности mariadb, используя команду mysql\_secure\_installation. С помощью запустившегося диалога и путём выбора [Y/n] установила пароль для пользователя root базы данных, отключила удалённый корневой доступ и удалила тестовую базу данных и любых анонимных пользователей. Данные действия представлены на рис.1.11 – рис.1.12.

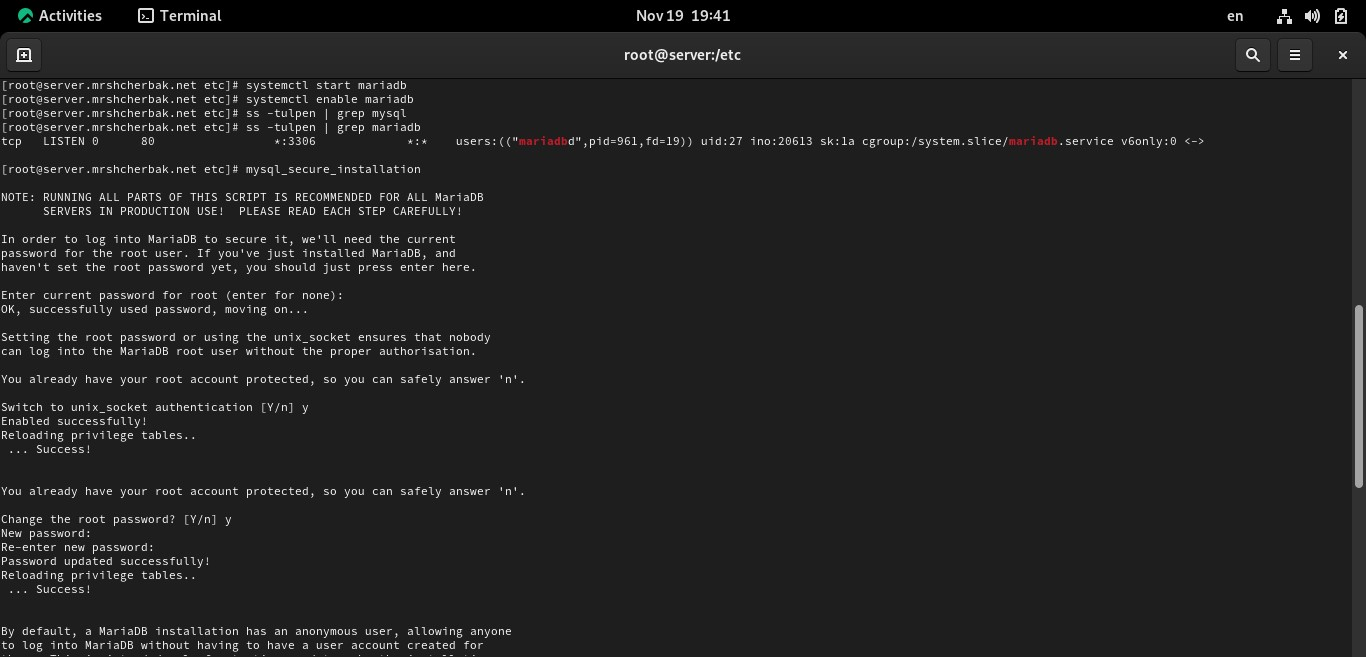


Рис.1.11. Выполнение действий

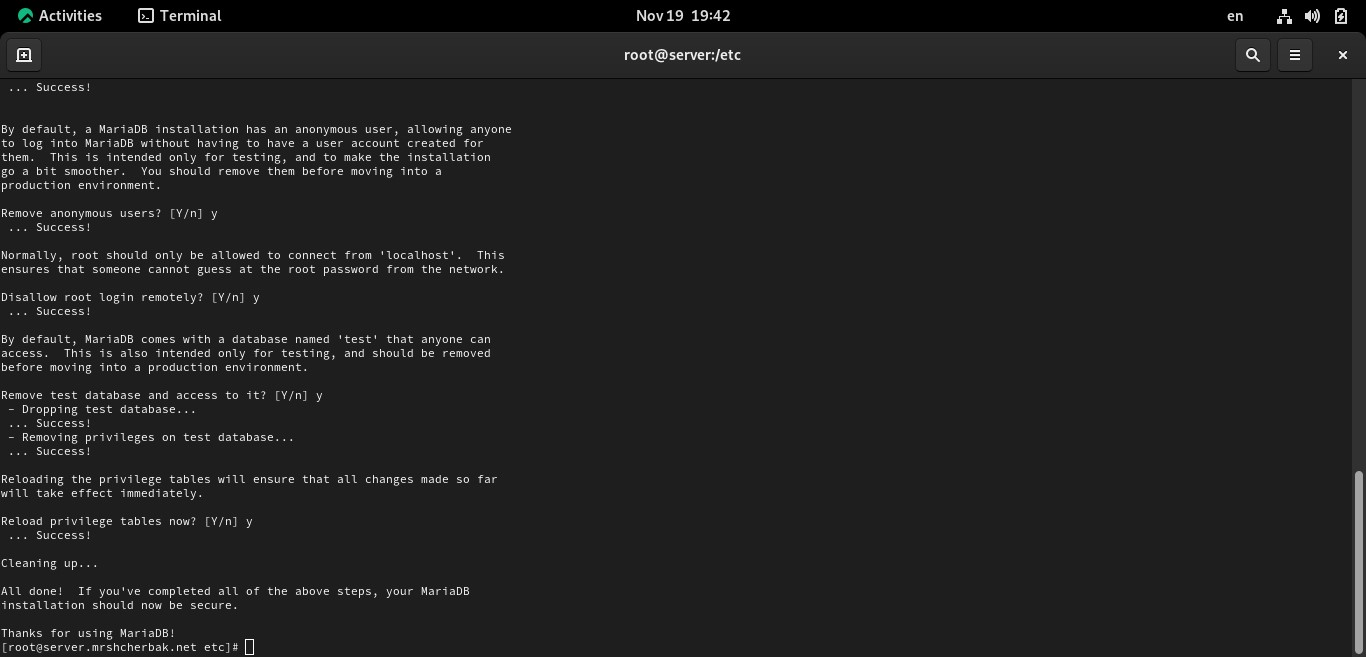


Рис.1.12. Выполнение действий

5. Для входа в базу данных с правами администратора базы данных ввела mysql -u root –p. Просмотрела список команд MySQL, введя \h (рис.1.13).



Рис.1.13. Просмотр команд

6. Из приглашения интерактивной оболочки MariaDB для отображения доступных в настоящее время баз данных ввела MySQL-запрос (рис.1.14). Вышла из интерфейса интерактивной оболочки MariaDB.

information\_schema , mysql и performance\_schema ⎯ это системные базы данных, предоставляемые MySQL и MariaDB для управления метаданными, хранения системной информации и предоставления статистики о производительности.

* information\_schema предоставляет информацию о структуре баз данных, таблиц, столбцов, индексов и прочих объектов в системе. Это метаданные, которые можно использовать для анализа структуры баз данных.
* mysql содержит данные, необходимые для управления пользователями, привилегиями доступа и другими настройками системы. Эта база данных играет важную роль в обеспечении безопасности и управлении пользователями баз данных.
* performance\_schema содержит данные о производительности системы, такие как счетчики, статистика, информация о событиях и т.д. Позволяет администраторам и разработчикам анализировать производительность запросов и других операций.

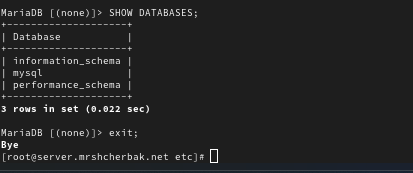


Рис.1.14. Вывод о базах данных

**2. Конфигурация кодировки символов**

1. Вошла в базу данных с правами администратора: mysql -u root –p. Для отображения статуса MariaDB ввела из приглашения интерактивной оболочки MariaDB: status.

На рис.2.1 изображен вывод команды STATUS в командной оболочке MariaDB. Данная информация полезна для мониторинга текущего состояния сервера и его нагрузки.

mysql Ver 15.1 Distrib 10.5.22-MariaDB, for Linux (x86\_64) using EditLine wrapper: сообщение о версии MariaDB. В данном случае, используется версия 10.5.22.

Connection id: 10: уникальный идентификатор текущего соединения.

Current database: имя текущей активной базы данных (в данном случае не указано,

поэтому ни одна база данных не выбрана).

Current user: root@localhost: имя текущего пользователя и адрес, с которого произведено подключение.

SSL: Not in use: указывает, что SSL не используется для текущего соединения.

Current pager: stdout: текущий вывод настроен на стандартный вывод (stdout).

Server: MariaDB: указывает, что сервер использует MariaDB.

Server version: 10.5.22-MariaDB MariaDB Server: версия MariaDB сервера.

Protocol version: версия протокола, который используется для обмена данными между клиентом и сервером.

Connection: Localhost via UNIX socket: текущее соединение происходит локально через UNIX сокет.

Server characterset: latinl: кодировка символов, установленная на сервере.

Db characterset: latinl : кодировка символов для текущей базы данных.

Client characterset: utf8: кодировка символов, используемая клиентом приложения.

Conn. characterset: utf8: кодировка символов для текущего соединения.

UNIX socket: /var/Lib/mysql/mysql.sock: путь к UNIX сокету, используемому для

соединения с сервером.

Uptime: 11 min 5 sec: время работы сервера с момента последнего запуска.

Threads: 2 : количество активных потоков.

Questions: 27 : общее количество запросов, выполненных сервером.

Slow queries: 0: количество медленных запросов.

Opens: 20: количество открытых соединений.

Open tables: 13: количество открытых таблиц.

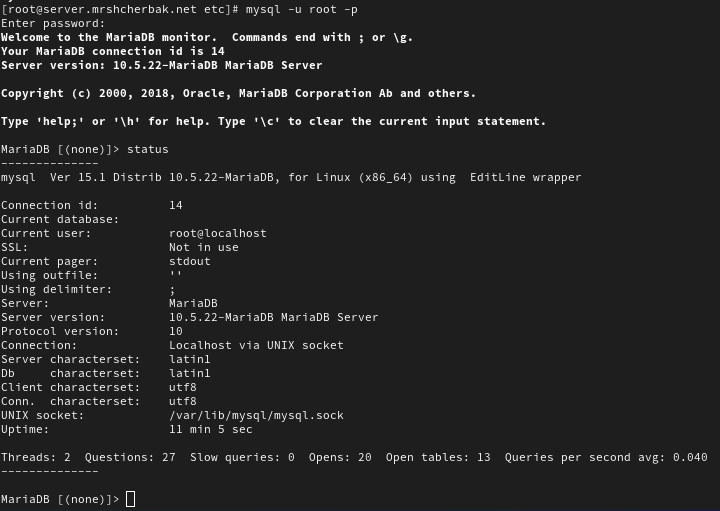


Рис.2.1. Вывод команды STATUS

2. В каталоге /etc/my.cnf.d создала файл utf8.cnf и, открыв его на редактирование, указала в нём конфигурацию, представленную на рис.2.2. Перезапустила MariaDB с помощью команды systemctl restart mariadb (рис.2.3).

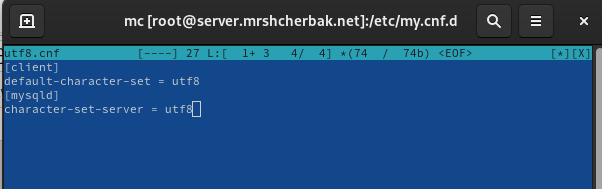


Рис.2.2. Редактирование файла utf8.cnf

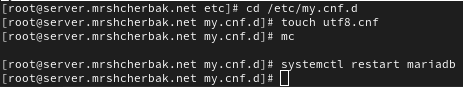


Рис.2.3. Создание файла и перезапуск MariaDB

3. Вошла в базу данных с правами администратора и посмотрела статус MariaDB (рис.2.4).

Изменилась информация:

Connection ID стал 3

Server characterset стал utf8

Db characterset стал utf8

Threads: 1 : количество активных потоков.

Questions: 4 : общее количество запросов, выполненных сервером.

Opens: 17: количество открытых соединений.

Open tables: 10: количество открытых таблиц.

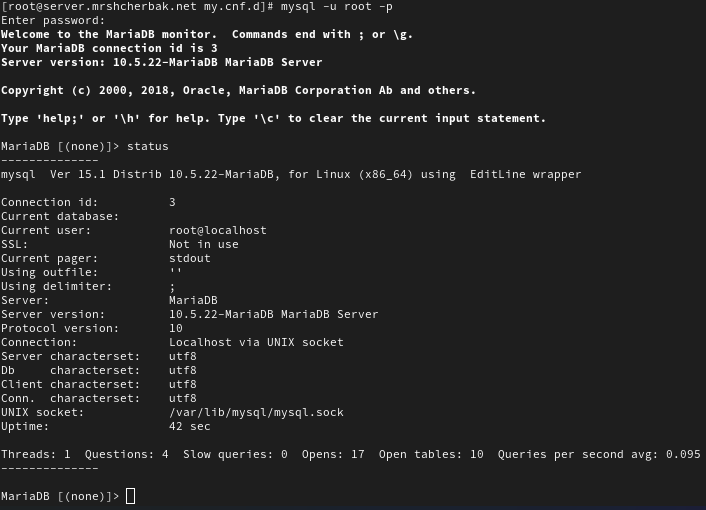


Рис.2.4. Вывод команды STATUS

**3. Создание базы данных**

1. Вошла в базу данных с правами администратора: mysql -u root –p. Создала базу данных с именем addressbook и перешла к базе данных addressbook.Отобразила имеющиеся в базе данных addressbook таблицы и создала таблицу city с полями name и city, после чего заполнила несколько строк таблицы некоторыми данными.

Сделала MySQL-запрос. Действия представлены на рис.3.1. Вывод строк и столбцов из таблицы с именем city, которую я заполняла.

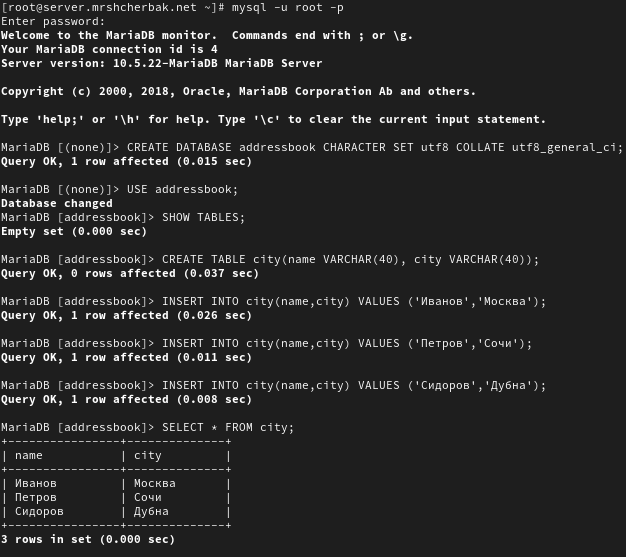


Рис.3.1. Выполнение действий

2. Создала пользователя mrshcherbak для работы с базой данных addressbook и задала для него пароль. Предоставила права доступа созданному пользователю на действия с базой данных addressbook (просмотр, добавление, обновление, удаление данных). Обновила привилегии (права доступа) базы данных addressbook и посмотрела общую информацию о таблице city базы данных addressbook, после чего вышла из окружения MariaDB. Действия представлены на рис.3.2.

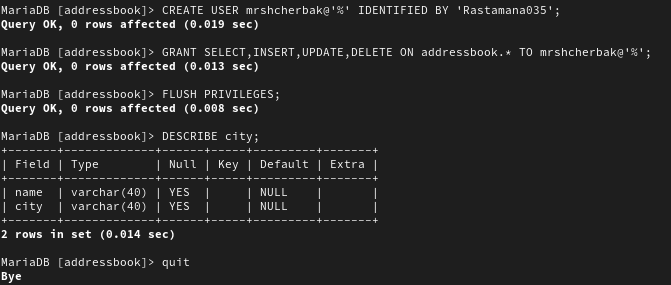


Рис.3.2. Выполнение действий

3. Просмотрела список баз данных: mysqlshow -u root –p. Просмотрела список таблиц базы данных addressbook. Действия представлены на рис.3.3.

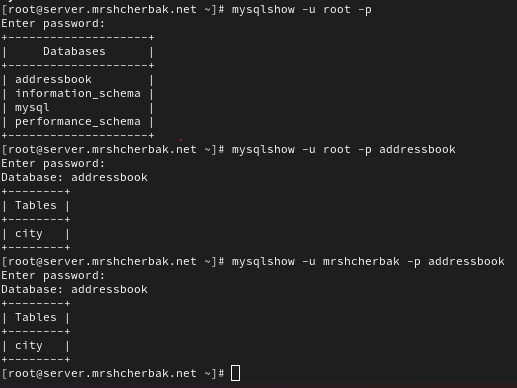


Рис.3.3. Выполнение действий

**4. Резервные копии**

1. На виртуальной машине server создала каталог для резервных копий и сделала резервную копию базы данных addressbook, после чего сделала сжатую резервную копию базы данных addressbook. Также сделала сжатую резервную копию базы данных addressbook с указанием даты создания и восстановила базу данных addressbook из резервной копии. Восстановила базу данных addressbook из сжатой резервной копии. Данные действия представлены на рис.4.1.

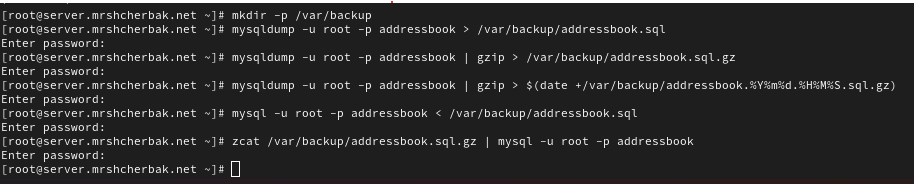


Рис.4.1. Выполнение действий

**5. Внесение изменений в настройки внутреннего окружения виртуальной машины**

1. На виртуальной машине server перешла в каталог для внесения изменений в настройки внутреннего окружения /vagrant/provision/server/, создала в нём каталог mysql, в который поместила в соответствующие подкаталоги конфигурационные файлы MariaDB и резервную копию базы данных addressbook. В каталоге /vagrant/provision/server создала исполняемый файл mysql.sh. Данные действия представлены на рис. Данные действия представлены на рис.5.1.

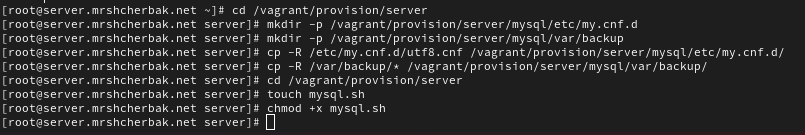


Рис.5.1. Выполнение действий

Открыв его на редактирование, прописала в нём следующий скрипт (рис.5.2). Этот скрипт повторяет произведённые мной действия по установке и настройке сервера баз данных.

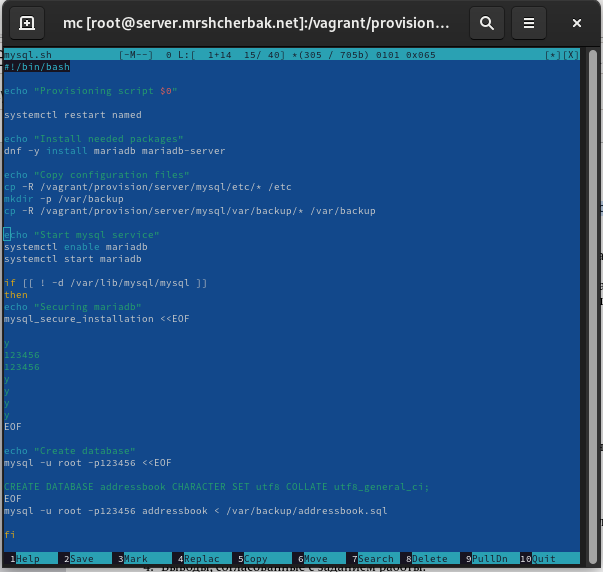


Рис.5.2. Редактирование файла mysql.sh

2. Для отработки созданного скрипта во время загрузки виртуальных машин в конфигурационном файле Vagrantfile добавила в конфигурации сервера запись (рис.5.3).

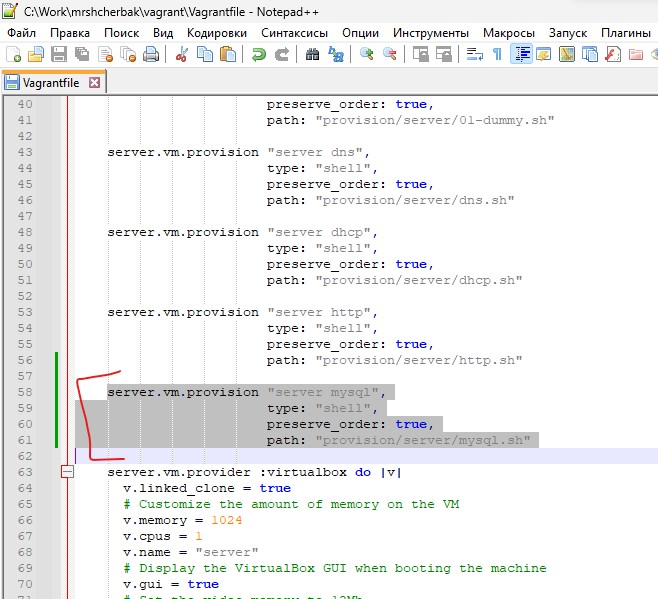


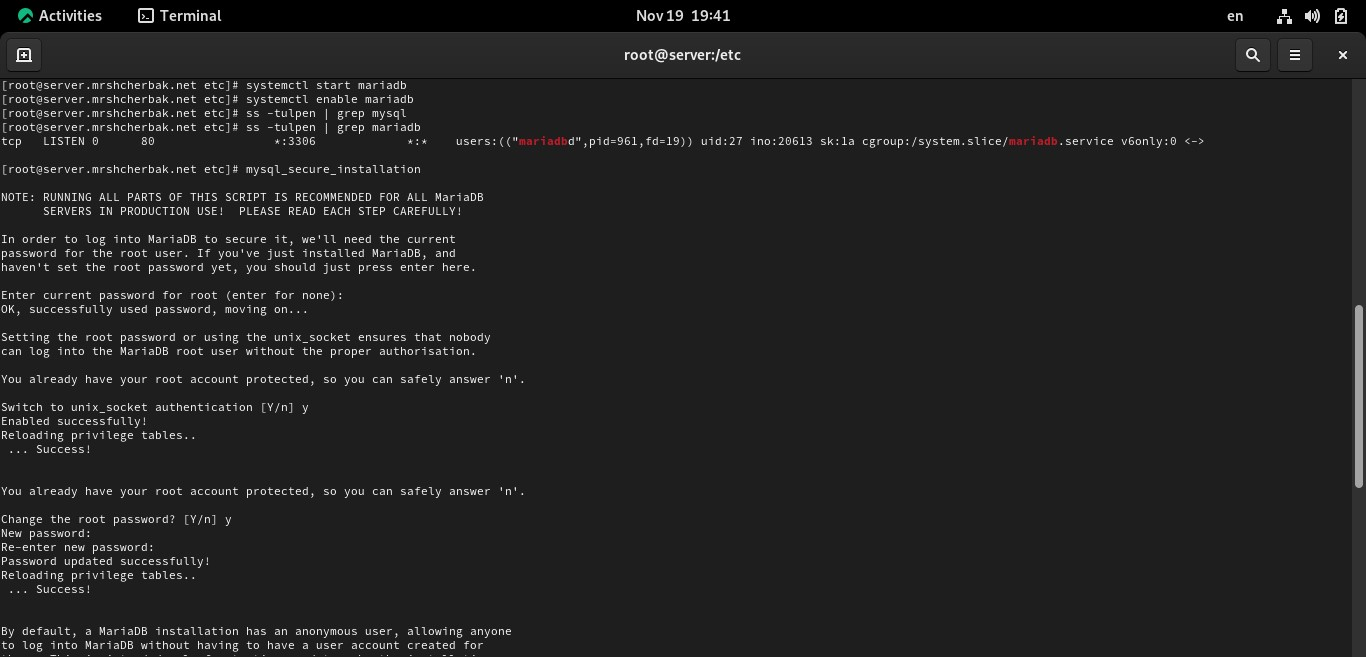
Рис.5.3. Редактирование файла Vagrantfile

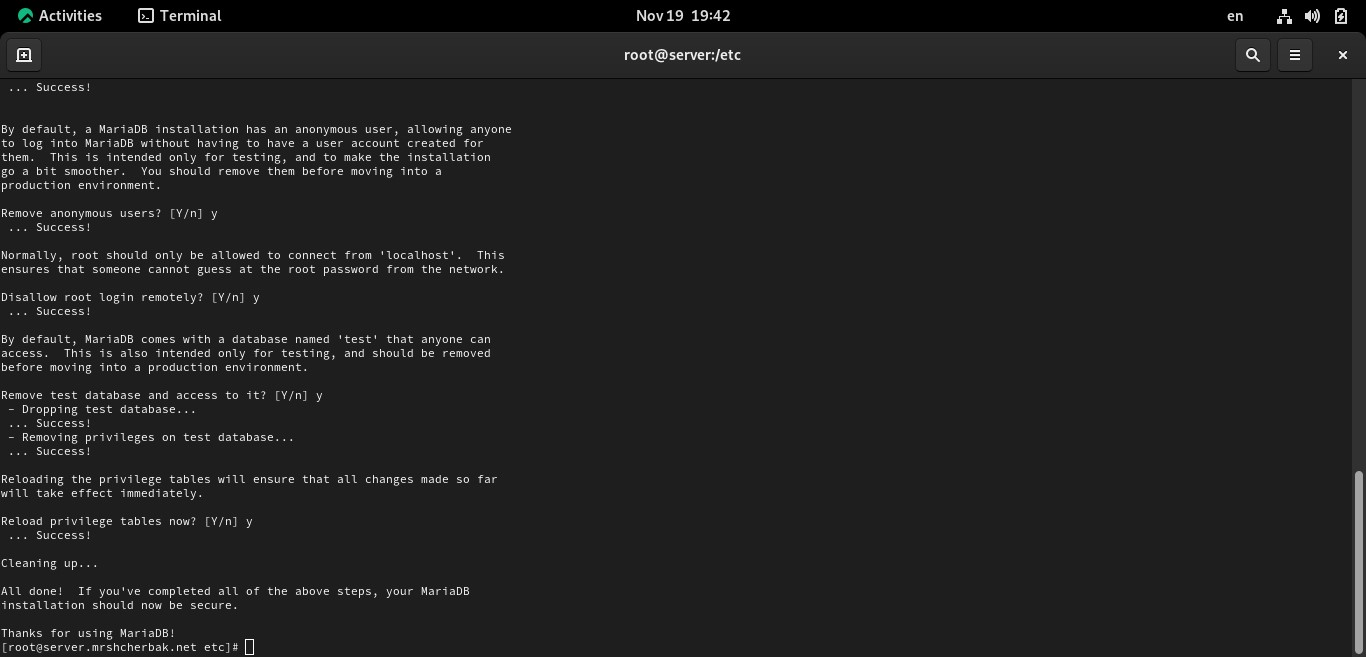
# Вывод: таким образом, в ходе выполнения л/р №6, я приобрела практические навыки по установке и конфигурированию системы управления базами данных на примере программного обеспечения MariaDB.

# Контрольные вопросы

1. Какая команда отвечает за настройки безопасности в MariaDB?

Настройки безопасности в MariaDB управляются командой mysql\_secure\_installation. Так, в данной лабораторной работе я запускала скрипт конфигурации безопасности mariadb, используя команду mysql\_secure\_installation. С помощью запустившегося диалога и путём выбора [Y/n] устанавливала пароль для пользователя root базы данных, отключала удалённый корневой доступ и удаляла тестовую базу данных и любых анонимных пользователей.





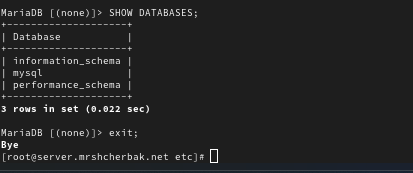
2. Как настроить MariaDB для доступа через сеть?

Для настройки доступа через сеть в MariaDB нужно отредактировать файл конфигурации (обычно my.cnf) и установить параметры, такие как bind-address и skip-networking. После изменений перезапустить MariaDB.

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

Обзор доступных баз данных в MariaDB можно получить командой SHOW DATABASES; после входа в среду оболочки.

Из приглашения интерактивной оболочки MariaDB для отображения доступных в настоящее время баз данных ввела MySQL-запрос, после чего вышла из интерфейса интерактивной оболочки MariaDB.

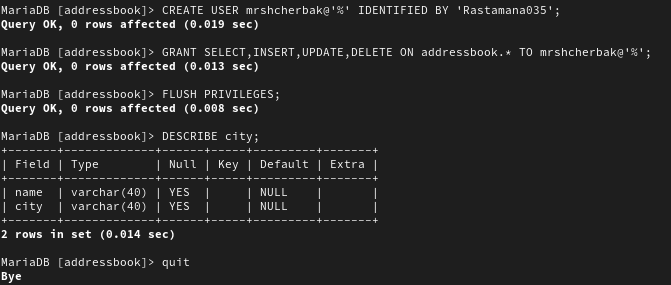


4. Какая команда позволяет узнать, какие таблицы доступны в базе данных?

Чтобы узнать доступные таблицы в базе данных, используйте команду SHOW TABLES; после выбора базы данных.

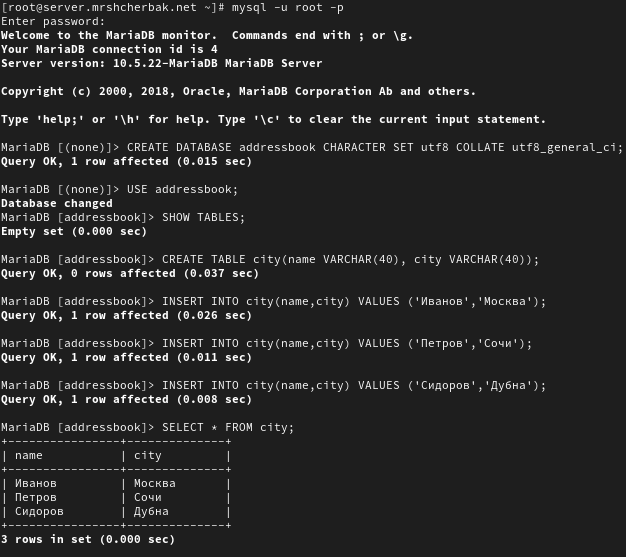
5. Какая команда позволяет узнать, какие поля доступны в таблице?

Для просмотра полей в таблице используйте команду DESCRIBE table\_name; или SHOW COLUMNS FROM table\_name;.



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

Команда SELECT \* FROM table\_name; позволяет увидеть все записи в таблице.



7. Как удалить запись из таблицы?

Удаление записи из таблицы осуществляется с помощью команды DELETE FROM table\_name WHERE condition;, где condition определяет условие для удаления.

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

Конфигурационные файлы mariadb находятся в каталоге /etc/my.cnf.d и в файле /etc/my.cnf. Также в /etc/mysql/my.cnf или /etc/mysql/mariadb.conf.d/. С их помощью можно настроить параметры, такие как порт, пути к файлам баз данных и другие аспекты конфигурации сервера.

9. Где располагаются файлы с базами данных MariaDB?

Файлы с базами данных MariaDB обычно хранятся в директории, указанной в конфигурационных файлах. По умолчанию это может быть что-то вроде /var/lib/mysql/.

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

Для создания резервной копии базы данных используйте команду mysqldump, например: mysqldump -u username -p dbname > backup.sql. Для восстановления используйте команду mysql -u username -p dbname < backup.sql.

На виртуальной машине server я создавала каталог для резервных копий и делала резервную копию базы данных addressbook, после чего делала сжатую резервную копию базы данных addressbook. Также делала сжатую резервную копию базы данных addressbook с указанием даты создания и восстанавливала базу данных addressbook из резервной копии. Восстанавливала базу данных addressbook из сжатой резервной копии.

