Лабораторная работа №6 (Установка и настройка системы управления базами данных MariaDB)

Шубина София Антоновна

Содержание

# 1 Цель работы

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

# 2 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 восстановите 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.

# 3 Выполнение лабораторной работы

**Установка MariaDB**

Загрузим операционную систему и перейдем в рабочий каталог с проектом: cd C:\work\sashubina\vagrant

Запустим виртуальную машину server: vagrant up server

На виртуальной машине server войдем под своим пользователем и откроем терминал. Перейдем в режим суперпользователя: sudo -i Установим необходимые для работы с базами данных пакеты:

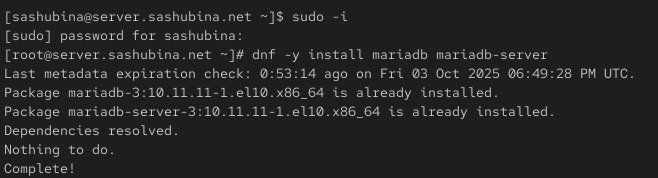


Рис. 1: Установка пакетов

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

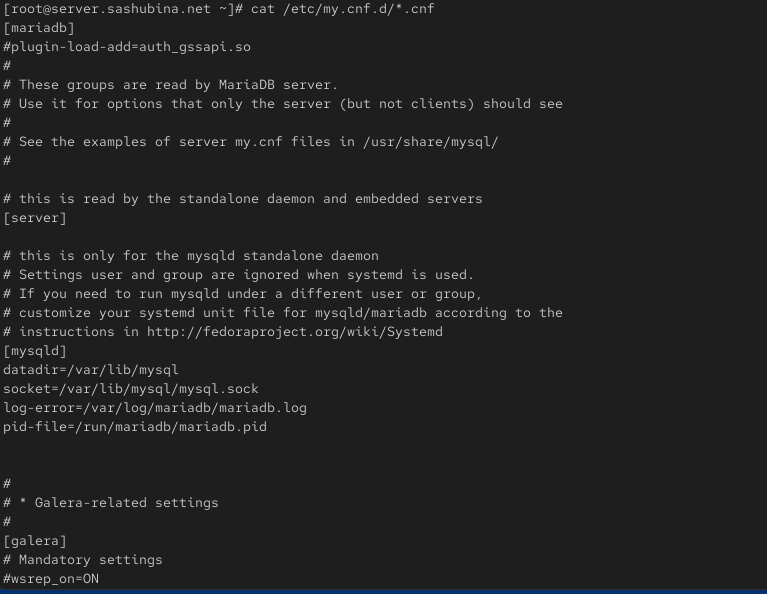


Рис. 2: Каталог /etc/my.cnf.d

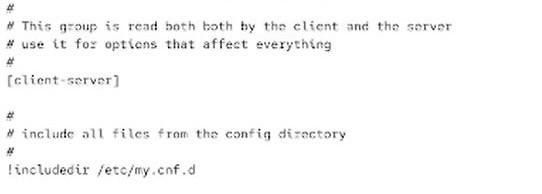


Рис. 3: Файл /etc/my.cnf

(/etc/my.cnf):

Основной конфигурационный файл MySQL/MariaDB

Содержит секцию [client-server] для общих настроек клиента и сервера

Включает конфигурационные файлы из директории /etc/my.cnf.d

Минимальная базовая конфигурация

(/etc/my.cnf.d/~.cnf):

Файл из дополнительной конфигурационной директории

Содержит разделы для MariaDB, сервера и демона mysqld

Определены стандартные пути: datadir, socket, логи и PID-файл

Включены настройки для Galera кластеризации

Имеются закомментированные опции для загрузки плагинов

Для запуска и включения программного обеспечения mariadb используем следующие команды:

Убедимся, что mariadb прослушивает порт 3306:

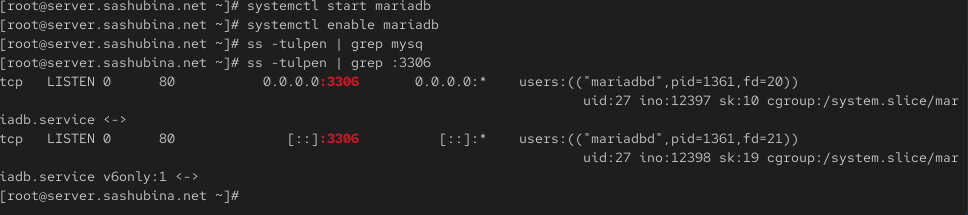


Рис. 4: Поиск нужного процесса

Запустим скрипт конфигурации безопасности mariadb. С помощью запустившегося диалога и путём выбора [Y/n] установим пароль для пользователя root базы данных (обратите внимание, что это не пользователь root операционной системы), отключим удалённый корневой доступ и удалим тестовую базу данных и любых анонимных пользователей.

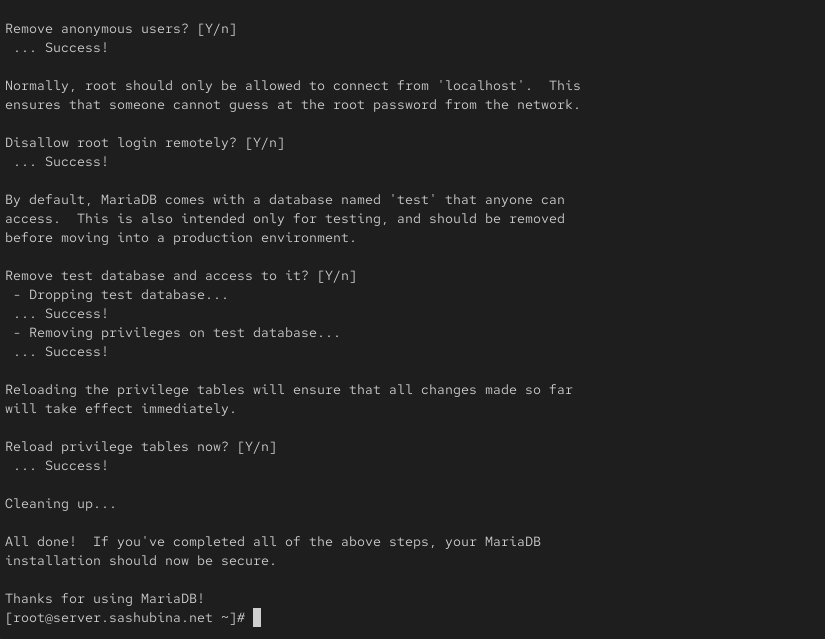


Рис. 5: Запуск скрипта конфигурации безопасности mariadb

Для входа в базу данных с правами администратора базы данных введем: mysql -u root -p Просмотрим список команд MySQL, введя \h.

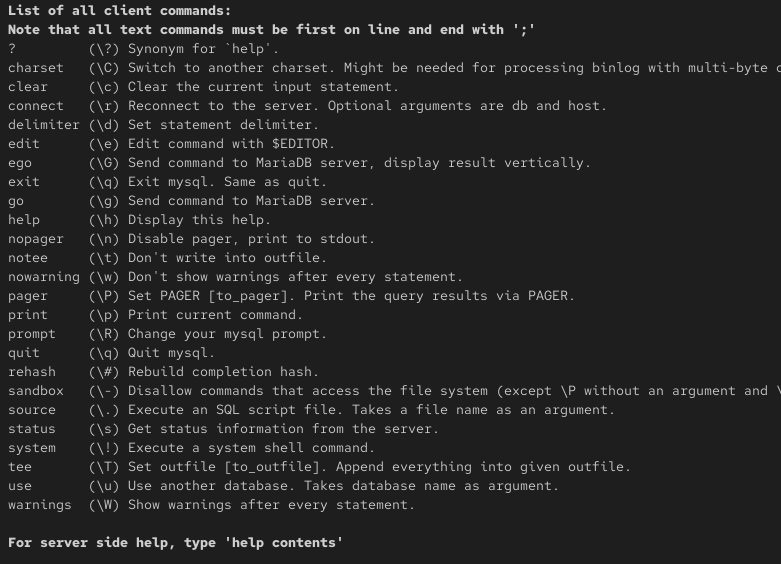


Рис. 6: список команд MySQL

Из приглашения интерактивной оболочки MariaDB для отображения доступных в настоящее время баз данных введем MySQL-запрос SHOW DATABASES; Для выхода из интерфейса интерактивной оболочки MariaDB введите exit;

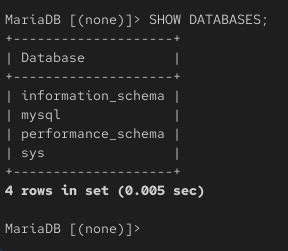


Рис. 7: Доступыне базы данных

Видно, что на данный момент есть 4 базы данных (mysql, information\_schema, perfomance\_schema, sys).

**Конфигурация кодировки символов**

Войдем в базу данных с правами администратора: mysql -u root -p Для отображения статуса MariaDB введем из приглашения интерактивной оболочкиMariaDB: status

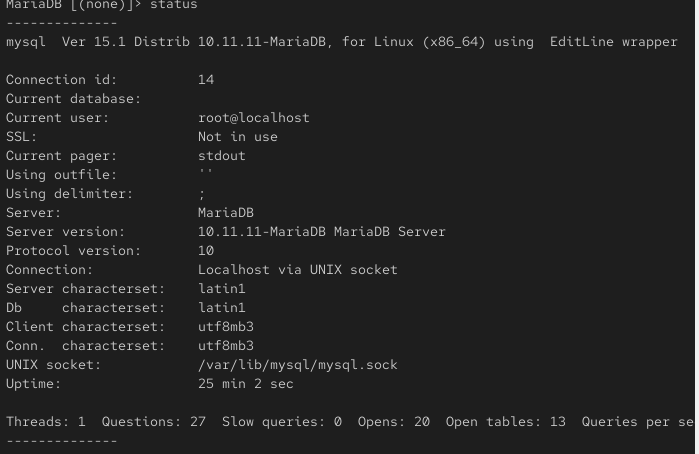


Рис. 8: Статус MariaDB

Вывелась информация, из которой мы можем узнать, например, текущую бд, текущего пользователя, сервер, версию сервера, версию протокола, набор символов сервера, базы данных и т.д.

В каталоге /etc/my.cnf.d создадим файл utf8.cnf:

cd /etc/my.cnf.d  
touch utf8.cnf

создание файла

Рис. 9: создание файла

Откроем его на редактирование и укажем в нём следующую конфигурацию:

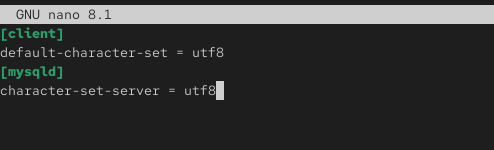


Рис. 10: Редактирование файла

Перезапустим MariaDB: systemctl restart mariadb Войдем в базу данных с правами администратора и посмотрим статус MariaDB.

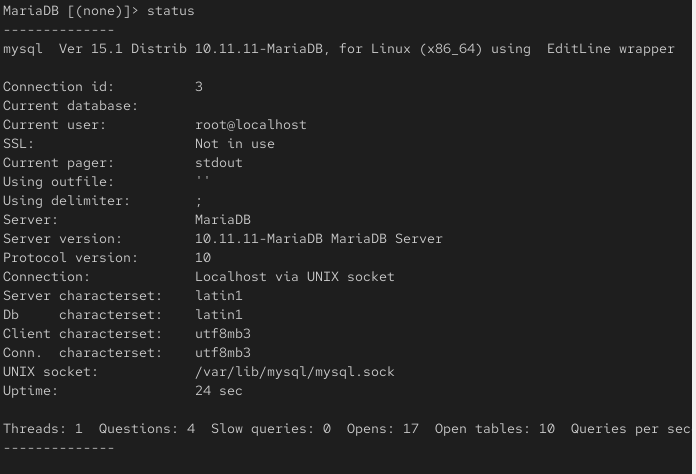


Рис. 11: Статус MariaDB

Увидим, что поменялся стандарт кодирования символов для сервера и базы данных на utf8.

**Создание базы данных**

Войдем в базу данных с правами администратора: mysql -u root -p Создадим базу данных с именем addressbook. Перейдем к базе данных addressbook. Отобразим имеющиеся в базе данных addressbook таблицы. Создадим таблицу city с полями name и city. CREATE DATABASE addressbook CHARACTER SET utf8 COLLATE utf8\_general\_ci;

Создание базы данных и таблицы

Рис. 12: Создание базы данных и таблицы

Перейдем к базе данных addressbook USE addressbook;



Рис. 13: USE addressbook

Отобразим имеющиеся в базе данных addressbook таблицы: SHOW TABLES; Создадим таблицу city с полями name и city: CREATE TABLE city(name VARCHAR(40), city VARCHAR(40));

Заполним несколько строк таблицы некоторыми данными по аналогии в соответствии с синтаксисом MySQL: Добавим в базу сведения о Петрове и Сидорове: Петров, Сочи Сидоров, Дубна

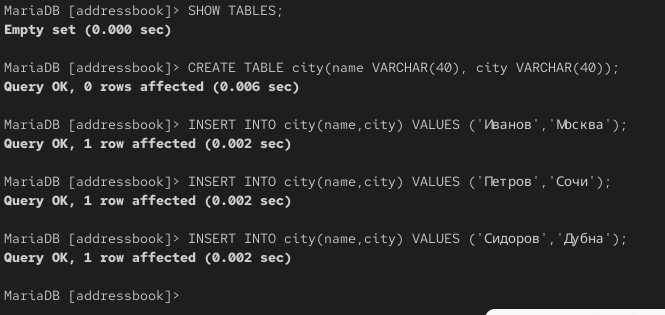


Рис. 14: заполнение строк

Сделаем следующий MySQL-запрос: SELECT \* FROM city;. Увидим, что вывелась таблица с колонками город и имя с заполненными мною данными.

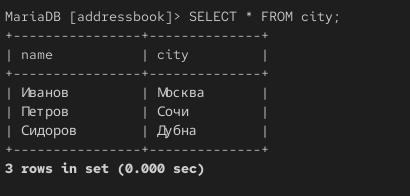


Рис. 15: SELECT \* FROM city

Создадим пользователя для работы с базой данных addressbook и зададим для него пароль: CREATE USER sashubina@'%' IDENTIFIED BY 'password';

Предоставим права доступа созданному пользователю user на действия с базой данных addressbook (просмотр, добавление, обновление, удаление данных): GRANT SELECT,INSERT,UPDATE,DELETE ON addressbook.\* TO user@'%';

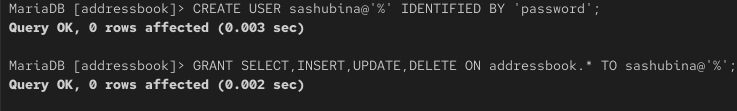


Рис. 16: создание пользователя и предоставление прав доступа

Обновим привилегии (права доступа) базы данных addressbook: FLUSH PRIVILEGES; Посмотрим общую информацию о таблице city базы данных addressbook: DESCRIBE city;

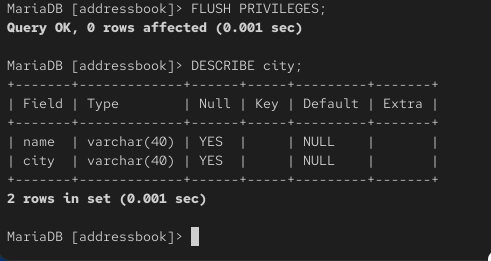


Рис. 17: обновим права доступа и посмотрим инф

Выйдем из окружения MariaDB: quit Просмотрим список баз данных: mysqlshow -u root -p

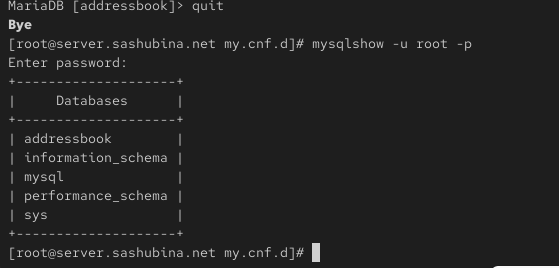


Рис. 18: Работа с базой данных

Просмотрим список таблиц базы данных addressbook: mysqlshow -u user -p addressbook

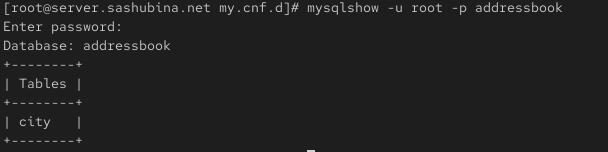


Рис. 19: список таблиц

**Резервные копии**

На виртуальной машине server создадим каталог для резервных копий: mkdir -p /var/backup Сделаем резервную копию базы данных addressbook: mysqldump -u root -p addressbook > /var/backup/addressbook.sql Сделаем сжатую резервную копию базы данных addressbook: mysqldump -u root -p addressbook | gzip > /var/backup/addressbook.sql.gz Сделаем сжатую резервную копию базы данных addressbook с указанием даты создания копии: mysqldump -u root -p addressbook | gzip > $(date +%-Y%-m%-d%-H%.M%.S).sql.gz Восстановим базу данных addressbook из резервной копии: mysql -u root -p addressbook < /var/backup/addressbook.sql Восстановим базу данных addressbook из сжатой резервной копии: zcat /var/backup/addressbook.sql.gz | mysql -u root -p addressbook

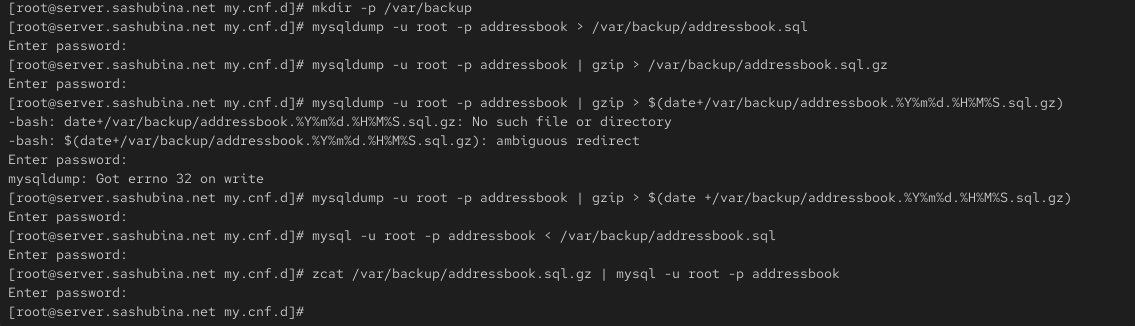


Рис. 20: Работа с резервными копиями

**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:



Рис. 21: Внесения изменений в настройки внутренного окружения

В каталоге /vagrant/provision/server создадим исполняемый файл mysql.sh:

cd /vagrant/provision/server  
touch mysql.sh  
chmod +x mysql.sh

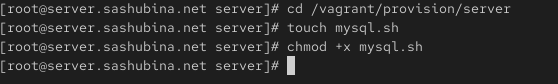


Рис. 22: Создание файла

Открыв его на редактирование, пропишем в нём следующий скрипт:

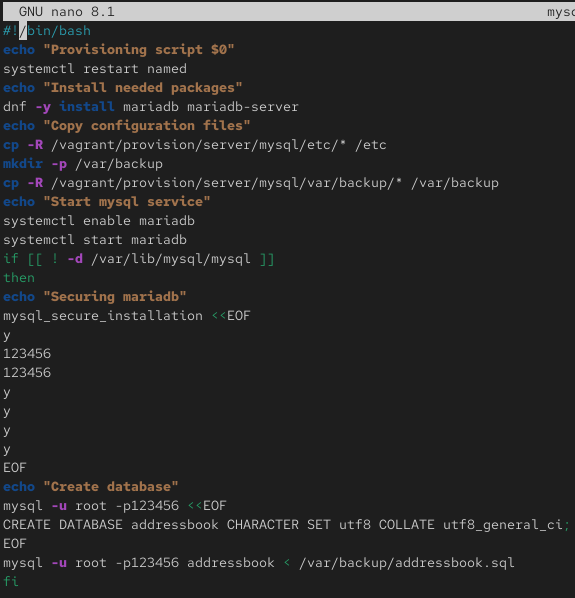


Рис. 23: Редактирование файла

Этот скрипт, по сути, повторяет произведённые действия по установке и настройке сервера баз данных.

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

server.vm.provision "server mysql",  
type: "shell",  
preserve\_order: true,  
path: "provision/server/mysql.sh"

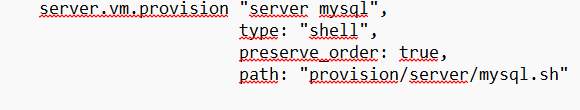


Рис. 24: Редактирование файла Vagrantfile

# 4 Выводы

В процессе выполения данной лабораторной работы я приобрела практические навыки по установке и конфигурированию системы управления базами данных на примере программного обеспечения MariaDB.