

Министерство образования Республики Беларусь  
Учреждение образования  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОТЕХНИКИ

Факультет компьютерного проектирования  
Кафедра проектирования информационно-компьютерных систем

**ОТЧЁТ**  
к лабораторной работе №3 на тему  
**«ПОЛУЧЕНИЕ БАЗОВЫХ НАВЫКОВ ПО НАСТРОЙКЕ БАЗЫ  
ДАННЫХ В ОПЕРАЦИОННОЙ СИСТЕМЕ LINUX»**  
по дисциплине «ОСМУ»

Выполнил студент:

К.Г. Хоменок

Проверил:

А.Д. Станкевич

Минск 2024

**Цель работы:** получение базовых навыков по настройке базы данных в операционной системе Linux.

## 1 Порядок выполнения работы

С использованием VirtualBox или VMware Workstation настроить две виртуальные машины А и Б с минимальными параметрами (1 CPU / 2 Gb RAM / 10 Gb HDD(SSD) / NIC).

Для выполнения лабораторной работы было выбрано VMWare Workstation Pro 17.X по причине своей стабильности и удобному пользованию. На рисунке 1 показано, что две виртуальные машины были созданы.

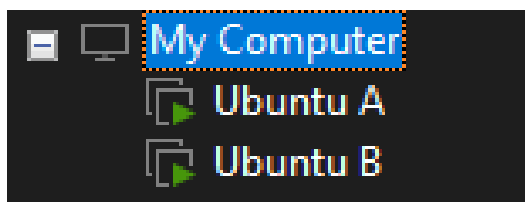


Рисунок 1 – Две виртуальные машины А и Б на базе Ubuntu

На виртуальной машине А установить:

- операционную систему GNU Linux Ubuntu Server 22.04 LTS (имя администратора должно совпадать с транслитерацией фамилии студента);
- установить обновления на операционную систему (рекомендуется);
- настроить сетевой адрес в пространстве подсети 192.168.0XX.0YY/24 (где XX - последние 2 цифры номера группы, YY - последние 2 цифры номера студента по списку);
- установить с использованием Docker или вручную СУБД MariaDB;
- создать базу данных с названием - транслитерация фамилии студента;
- создать в СУБД пользователя с именем - транслитерация имени студента, имеющего полные привилегии в созданной базе данных и возможность удалённого подключения.

В качестве операционной системы на обе виртуальные машины была поставлена GNU Linux Ubuntu Server 22.04, а также были установлены обновления:

```
sudo apt update -y && sudo apt upgrade -y
```

После чего необходимо настроить сетевой адрес в пространстве сети, который у виртуальной машины А должен соответствовать 192.168.1.25/24, а у виртуальной машины Б – 192.168.1.125/24.

На этом этапе необходимо обратиться к инструменту VMWare. Необходимо выключить виртуальные машины и зайти в меню Virtual Network Editor. Поскольку был выбран NAT, то необходимо поменять пул DHCP-адресом начиная с 192.168.1.3 и заканчивая 192.168.1.254, при этом оставив маску 255.255.255.0 (/24). Возникает вопрос, что делать с DHCP, но его можно не трогать и дальше будет рассказано почему. Отредактированная конфигурация DHCP на NAT в Virtual Network Editor показана на рисунке 2 и 3.

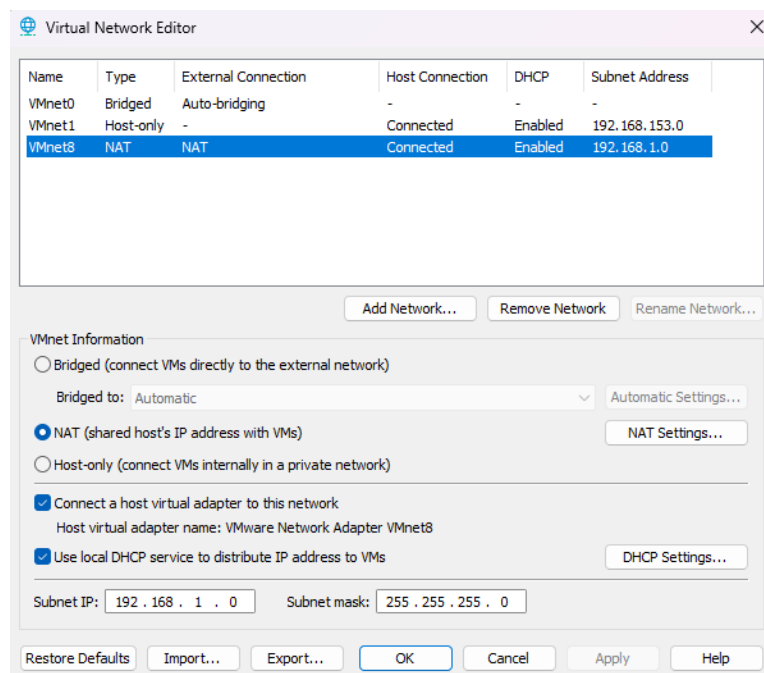


Рисунок 2 – Редактирование настроек VNE

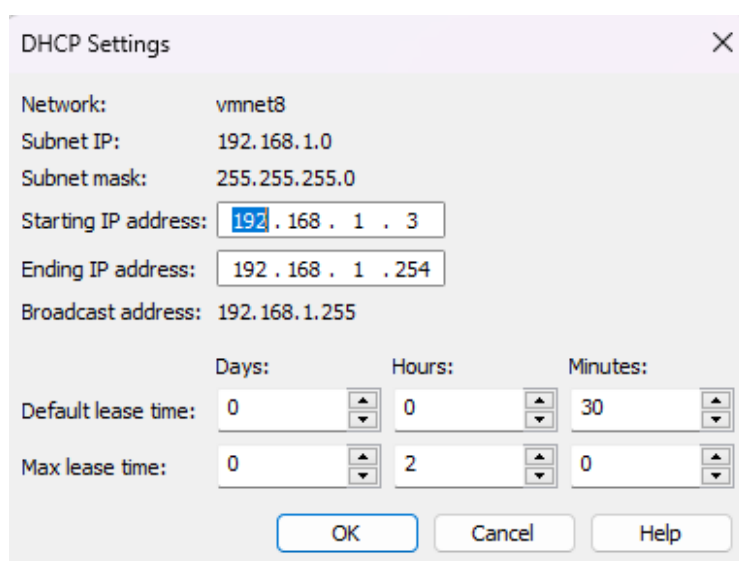


Рисунок 3 – Редактирование настроек VNE

Теперь IP-адрес виртуальных машин будет начинаться не с 192.168.2.17 и по 192.168.2.254, а с 192.168.1.3, что как раз и необходимо.

Далее, чтобы выдать статичный IP-адрес необходимо обратиться к конфигурации **netplan**. Проверяем IP-адрес:

```
hostname -I  
192.168.1.3 172.17.0.1 172.19.0.1
```

Как видно из результата, виртуальная машина А вообще не соответствует требованиям. Для этого редактируем конфигурацию netplan:

```
sudo nano /etc/netplan/00-installer-config.yaml
```

При открытии конфига будет следующее содержимое:

```
network:  
  ethernets:  
    ens33:  
      dhcp4: true  
  version: 2
```

ДНСР не нужен, поскольку требование, – статичный адрес. Именно поэтому на виртуальной машине А (192.168.1.25/24) конфигурацию заменяем на:

```
network:  
  ethernets:  
    ens33:  
      dhcp4: false  
      addresses:  
        - 192.168.1.25/24  
      routes:  
        - to: default  
          via: 192.168.1.2  
      nameservers:  
        addresses: [8.8.8.8, 8.8.4.4]  
  version: 2
```

А на виртуальной машине Б (192.168.1.125/24) заменяем на:

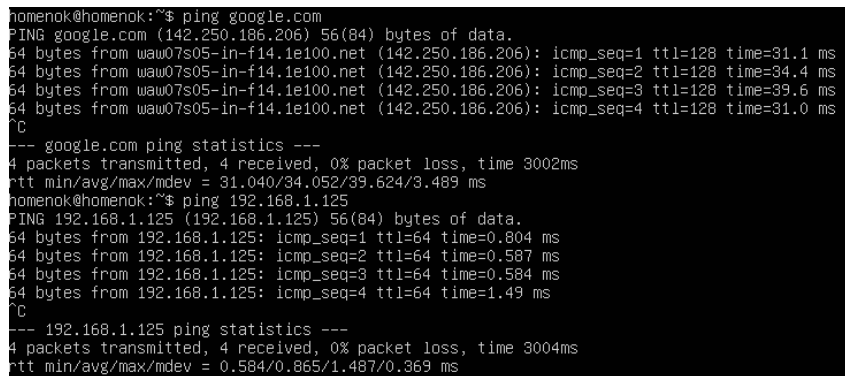
```
network:  
  ethernets:  
    ens33:  
      dhcp4: false  
      addresses:
```

```

- 192.168.1.125/24
routes:
- to: default
  via: 192.168.1.2
nameservers:
  addresses: [8.8.8.8, 8.8.4.4]
version: 2

```

После чего применяем настройки конфигурации на обоих виртуальных машинах и проверяем доступ к интернету и связь между ними (рисунок 3).



```

homenok@homenok:~$ ping google.com
PING google.com (142.250.186.206) 56(84) bytes of data:
64 bytes from waw07s05-in-f14.1e100.net (142.250.186.206): icmp_seq=1 ttl=128 time=31.1 ms
64 bytes from waw07s05-in-f14.1e100.net (142.250.186.206): icmp_seq=2 ttl=128 time=34.4 ms
64 bytes from waw07s05-in-f14.1e100.net (142.250.186.206): icmp_seq=3 ttl=128 time=39.6 ms
64 bytes from waw07s05-in-f14.1e100.net (142.250.186.206): icmp_seq=4 ttl=128 time=31.0 ms
^C
--- google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3002ms
rtt min/avg/max/mdev = 31.040/34.052/39.624/3.489 ms
homenok@homenok:~$ ping 192.168.1.125
PING 192.168.1.125 (192.168.1.125) 56(84) bytes of data:
64 bytes from 192.168.1.125: icmp_seq=1 ttl=64 time=0.804 ms
64 bytes from 192.168.1.125: icmp_seq=2 ttl=64 time=0.587 ms
64 bytes from 192.168.1.125: icmp_seq=3 ttl=64 time=0.584 ms
64 bytes from 192.168.1.125: icmp_seq=4 ttl=64 time=1.49 ms
^C
--- 192.168.1.125 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 0.584/0.865/1.487/0.369 ms

```

Рисунок 3 – Проверка работоспособности конфигурации

Как видно из рисунка, маршрут проходит в Интернет, и виртуальная машина А может связаться с виртуальной машиной Б (поскольку они в одной подсети).

Далее выберем простой, удобный и наиболее рациональный способ разворачивания MariaDB, – docker-compose.

Написанный docker-compose:

```

version: '3.8'

services:
  mariadb:
    image: mariadb
    container_name: mariadb
    restart: always
    environment:
      MYSQL_DATABASE: homenok
      MYSQL_USER: homenok
      MYSQL_PASSWORD: 123321
      MYSQL_ROOT_PASSWORD: 123321

volumes:
  - mariadb_data:/var/lib/mysql

```

```

ports:
  - "3306:3306"
networks:
  - bridge_network

networks:
  bridge_network:
    driver: bridge

```

Таким образом, виртуальная машина А готова. Перейдем к виртуальной машине Б.

На виртуальной машине Б установить:

- операционную систему GNU Linux Ubuntu Server 22.04 LTS (имя администратора должно совпадать с транслитерацией фамилии студента);
- установить обновления на операционную систему (рекомендуется);
- настроить сетевой адрес в пространстве подсети 192.168.0XX.1YY/24 (где XX - последние 2 цифры номера группы, YY - последние 2 цифры номера студента по списку);
- установить с использованием Docker или вручную любой SQL-менеджер, основанный на веб-интерфейсе (например DbGate или SQLPad);
- пробросить порты на виртуальной машине Б таким образом, чтобы веб-интерфейс был доступен в браузере родительской операционной системы.

Аналогично с SQLPad, – docker compose имеет вид:

```

version: '3.8'

services:
  sqlpad:
    image: sqlpad/sqlpad
    container_name: sqlpad
    restart: always
    environment:
      PORT: 3000
      SQLPAD_ADMIN: kirya.khomenok@mail.ru
      SQLPAD_ADMIN_PASSWORD: 123321
    ports:
      - "3000:3000"
    networks:
      - sqlpad_network

networks:
  sqlpad_network:
    driver: bridge

```

Далее проверяем работоспособность выполненных заданий и обращаемся с хостовой машины в веб-интерфейс SQL-менеджера «SQLPad», написав в адресной строке браузера адрес виртуальной машины Б (192.168.1.125/24) (рисунок 4).

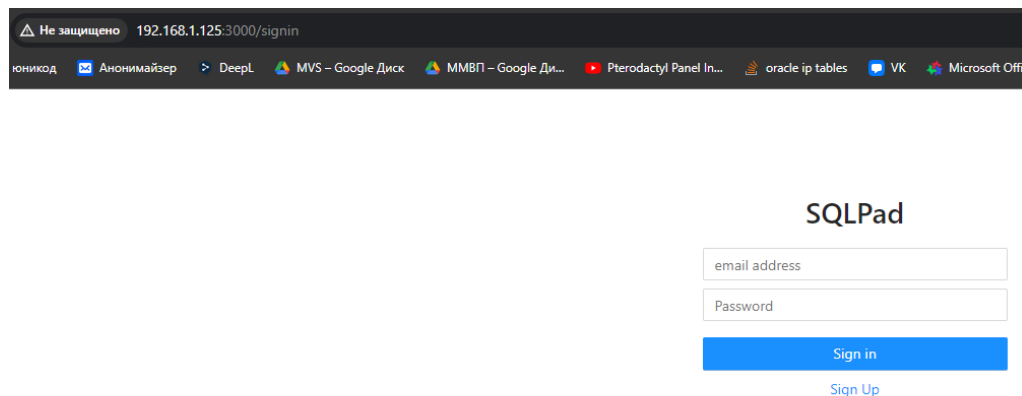


Рисунок 4 – Рабочий веб-интерфейс SQLPad

Заходим в учетную запись под теми данными, которые были явным образом написаны в переменные окружения и не зашифрованы.

После чего создаем подключение и проверяем (рисунок 5).

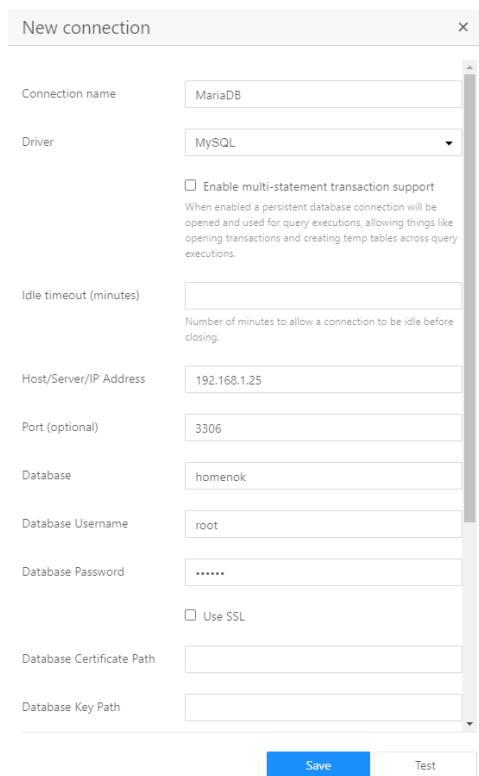


Рисунок 5 – Создание подключения

Сохраняем, после чего пробуем выполнить какой-либо запрос (рисунок 6).



Рисунок 6 – Успешная отправка запросов с хостовой машины

Как видно из рисунка, мы успешно отправляем запросы в базу данных на виртуальной машине А через веб-интерфейс SQL-менеджера на виртуальной машине Б на хостовой машине.

Попробуем выполнить запрос для создания в СУБД пользователя с именем «kiryl», имеющего полные привилегии в созданной базе данных и возможность удалённого подключения.:

```
CREATE USER 'kiryl'@'%' IDENTIFIED BY 'your_password_here';  
GRANT ALL PRIVILEGES ON homenok.* TO 'kiryl'@'%' WITH GRANT OPTION;  
FLUSH PRIVILEGES;
```

Таким образом, был создан пользователь в БД с полными привилегиями и возможностью удаленного подключения.

## 2 Вывод

В результате выполнения лабораторной работы были успешно настроены две виртуальные машины: А и Б. На виртуальной машине А была установлена операционная система GNU Linux Ubuntu Server 22.04 LTS, настроен сетевой адрес в подсети 192.168.0XX.0YY/24, установлена СУБД MariaDB с созданием базы данных и пользователя с полными привилегиями. На виртуальной машине Б также была установлена операционная система GNU Linux Ubuntu Server 22.04 LTS, настроен сетевой адрес в подсети



192.168.0XX.1YY/24, установлен веб-интерфейс SQL-менеджера и проброшены порты для доступа к нему из браузера родительской операционной системы.

Организация сети была осуществлена в рамках подсети с адресом 192.168.0XX.0/24, и SQL-менеджер виртуальной машины Б успешно подключен к базе данных на виртуальной машине А с помощью созданного пользователя.

В целом, выполнение лабораторной работы позволило ознакомиться с основами настройки виртуальных машин, установкой операционных систем, настройкой сетевых параметров, установкой и конфигурацией баз данных и веб-интерфейсов SQL-менеджеров, а также организацией взаимодействия между ними в рамках локальной сети.