

Министерство образования Республики Беларусь
Учреждение образования «Белорусский государственный университет
информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра информатики

Дисциплина: Информационные сети. Основы безопасности

ОТЧЁТ
к лабораторной работе №1
на тему

Локальная сеть со статической IP-адресацией

Выполнил:

студент гр. 253505
Сенько Н. С.

Проверил:

ассистент кафедры
информатики
Герчик А. В.

Минск 2025

СОДЕРЖАНИЕ

1 Цель работы.....	3
2 Ход работы.....	4
Заключение.....	5
Приложение А (обязательное) исходный код программного продукта.....	6

1 ЦЕЛЬ РАБОТЫ

Целью данной лабораторной работы является получение практических навыков в создании локальной сети с использованием статической IP-адресации. Это важный аспект в области сетевых технологий, поскольку он позволяет обеспечить стабильность и надежность соединений между устройствами в сети. В ходе работы студенты должны будут освоить основные этапы настройки сетевого оборудования, формируя навыки, которые пригодятся в профессиональной деятельности.

Создание локальной сети со статической IP-адресацией включает в себя понимание принципов работы с IP-адресами, знание порядка распределения адресов и способность настраивать сетевые устройства, такие как маршрутизаторы и коммутаторы. Это поможет в обеспечении заданной конфигурации сети и минимизации возможных конфликтов адресов.

Проверка правильности работы сети путем эмуляции передачи пакетов между устройствами позволит оценить, насколько правильно была выполнена настройка. Это также способствует лучшему пониманию процессов маршрутизации и передачи данных, что является ключевым аспектом для будущих специалистов в области информационных технологий. В результате выполнения работы студенты смогут не только понять теоретические основы создания и настройки сетей, но и приобрести ценные практические навыки.

2 ХОД РАБОТЫ

В ходе лабораторной работы была реализована локальная сеть с использованием Docker для эмуляции сетевой инфраструктуры. Работа включала в себя создание контейнеров, настройку сетевых подключений, конфигурацию маршрутизации и проверку работоспособности сети. В проекте была создана конфигурация в формате Docker Compose, включающая четыре контейнера, каждый из которых выполняет специфическую роль в сетевой инфраструктуре. Контейнеры были названы first, second, third и fourth, и каждый из них выполняет функции маршрутизации и NAT.

Сетевые настройки включали создание трех подсетей: first-net с диапазоном 172.28.1.0/24, second-net с диапазоном 172.28.2.0/24 и routers-net с диапазоном 172.28.10.0/24. Каждый контейнер получал статический IP-адрес, что упрощало маршрутизацию и связь между устройствами. Скрипт начинался с удаления существующих контейнеров и сетей для обеспечения чистой среды, после чего происходила настройка маршрутизации. В каждом контейнере выполнялась команда ip route add для добавления статических маршрутов, а также использовался iptables для настройки NAT.

В завершение лабораторной работы проводилась проверка связности между контейнерами с помощью команды ping, что подтверждало корректность настроек маршрутизации. В результате была успешно создана локальная сеть с двумя подсетями, и настройка статической маршрутизации обеспечила связь между контейнерами. Работы продемонстрировали основные принципы настройки сетевой инфраструктуры с использованием Docker, включая создание изолированных сетей и управление сетевым трафиком.

ЗАКЛЮЧЕНИЕ

В результате выполнения данной лабораторной работы студенты приобрели ценные практические навыки в области создания и управления локальными сетями с использованием статической IP-адресации. Их знания о распределении IP-адресов и настройке сетевого оборудования существенно расширились, что позволит им уверенно применять эти навыки в реальных профессиональных ситуациях. Студенты также закрепили теоретические аспекты сетевой адресации, что важно для избегания конфликтов и обеспечения стабильности сети.

Процесс эмуляции передачи пакетов между устройствами подтвердил корректность выполненных настроек и дал возможность оценить их работу в условиях, приближенных к реальным. Это обеспечило более глубокое понимание маршрутизации и процессов передачи данных, что важно для формирования компетентных специалистов в области IT-технологий.

Таким образом, лабораторная работа успешно достигла своей цели, позволив студентам сочетать теоретические знания с практическими навыками. Этот опыт будет полезен в их дальнейшей образовательной и профессиональной деятельности, закладывая фундамент для более сложных проектов и задач в области сетевых технологий.

Приложение А (обязательное)

Исходный код программного продукта

```
version: '3'

services:
  first:
    container_name: first-ubuntu
    build:
      context: ./ubuntu-iputils
    command:
      - bash
      - -c
      - |
        iptables -t nat -A POSTROUTING -s 172.28.10.0/24 -j MASQUERADE
        ip route add 172.28.2.0/24 via 172.28.1.3
        while true; do sleep 2; done
    cap_add:
      - NET_ADMIN
    sysctls:
      - net.ipv4.ip_forward=1
    networks:
      first-net:
        ipv4_address: 172.28.1.2

  second:
    container_name: second-ubuntu
    build:
      context: ./ubuntu-iputils
    command:
      - bash
      - -c
      - |
        iptables -t nat -A POSTROUTING -s 172.28.10.0/24 -j MASQUERADE
        ip route add 172.28.2.0/24 via 172.28.10.3
        while true; do sleep 2; done
    cap_add:
      - NET_ADMIN
    sysctls:
      - net.ipv4.ip_forward=1
    networks:
      first-net:
        ipv4_address: 172.28.1.3
      routers-net:
        ipv4_address: 172.28.10.2

  third:
    container_name: third-ubuntu
    build:
      context: ./ubuntu-iputils
    command:
      - bash
      - -c
      - |
        iptables -t nat -A POSTROUTING -s 172.28.10.0/24 -j MASQUERADE
        ip route add 172.28.1.0/24 via 172.28.10.2
        while true; do sleep 2; done
    cap_add:
```

```

    - NET_ADMIN
  sysctls:
    - net.ipv4.ip_forward=1
  networks:
    second-net:
      ipv4_address: 172.28.2.3
    routers-net:
      ipv4_address: 172.28.10.3

fourth:
  container_name: fourth-ubuntu
  build:
    context: ./ubuntu-iputils
  command:
    - bash
    - -c
    - |
      ip route add 172.28.1.0/24 via 172.28.2.3
      while true; do sleep 2; done
  cap_add:
    - NET_ADMIN
  sysctls:
    - net.ipv4.ip_forward=1
  networks:
    second-net:
      ipv4_address: 172.28.2.2

networks:
  first-net:
    driver: bridge
    ipam:
      config:
        - subnet: 172.28.1.0/24
          gateway: 172.28.1.1

  second-net:
    driver: bridge
    ipam:
      config:
        - subnet: 172.28.2.0/24
          gateway: 172.28.2.1

  routers-net:
    driver: bridge
    ipam:
      config:
        - subnet: 172.28.10.0/24
          gateway: 172.28.10.1

```