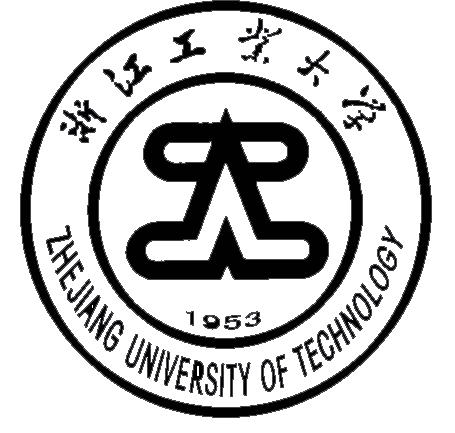


**计算机网络课程设计报告**



学 号 202203150422

姓 名 吴文韬

班 级 软件工程03

组 号

提交日期 2023 年 12 月 24 日

**目 录**

[一． 课程的目的和任务 1](#_Toc9048)

[1.1课程目标 1](#_Toc21109)

[1.2 课程任务 1](#_Toc31758)

[二． 课程的基本内容和要求 1](#_Toc7457)

[2.1常用网络命令](#_2.1常用网络命令)

[2.2交换机与路由器 1](#_2.2交换机与路由器)

[2.2.1 交换机配置与交换机VLAN设计](#_2.2.1交换机及VLAN配置)

[2.2.2 路由配置与静态路由设计](#_2.2.2路由配置与静态路由设计)

[2.2.3 动态路由设计](#_2.2.3动态路由设计)

[2.2.4 网络集成设计](#_2.2.4网络集成设计)

[2.3网络编程](#_2.3网络编程)

[2.3.1 开发环境及关键配置](#_2.3.1 设计思想)

[2.3.2 设计思路](#_2.3.3 主要函数)

[2.3.3 工作流程](#_2.3.3工作流程)

[2.3.4 运行截图及结果说明](#_2.3.4 测试结果截图)

[三． 实验总结与心得体会 3](#_Toc28355)

四.图目录

[图 1 ipconfig 2](#_Toc26169)

[图 2 ipconfig/all 3](#_Toc4991)

[图 3 ipconfig/? 4](#_Toc29549)

[图 4 ipconfig/renew 5](#_Toc3922)

[图 5 ipconfig/flushdns 5](#_Toc24456)

[图 6 ipconfig/displaydns 6](#_Toc2166)

[图 7 ping/? 7](#_Toc9348)

[图 8 ping/t 8](#_Toc10967)

[图 9 ping-a 8](#_Toc16596)

[图 10 ping-n count 8](#_Toc29871)

[图 11 arp 14](#_Toc4508)

[交换机图 1 初始化 18](#_Toc12099)

[交换机图 2 终端初始化 20](#_Toc31222)

[交换机图 3 测试拓补图 21](#_Toc18791)

[交换机图 4 测试结果 23](#_Toc23887)

[交换机图 5 划分vlan 24](#_Toc17715)

[交换机图 6 加入vlan 26](#_Toc3974)

[动态路由 1 拓补图 39](#_Toc29766)

[动态路由 2 测试 40](#_Toc11489)

[动态路由 3 rip分析 41](#_Toc10106)

[网络集成 1 拓补图 42](#_Toc24013)

[网络集成 2 ip地址划分 43](#_Toc31087)

[网络集成 3 三层交换机 43](#_Toc3715)

[网络集成 4 rip配置 44](#_Toc11814)

[网络集成 5 NAT配置 45](#_Toc12992)

[网络编程 1 流程图 49](#_Toc18140)

[网络编程 2 运行结果 50](#_Toc16811)

1. **课程的目的和任务**

## 1.1课程目标

课程教学目标1： 理解计算机网络体系结构和工作原理，掌握网络测试与故障检测的基本方法。

课程教学目标2： 通过自学熟练掌握Cisco Packet Tracer等现代主流网络仿真工具，实现基于仿真工具的的协议数据分析，认识网络技术和工具的发展现状。

课程教学目标3：针对网络工程问题设计实验方案，能够按照实验方案实施仿真实验，采集和整理数据，并用所学网络知识对实验数据进行分析、处理和解释的能力。

## 1.2 课程任务

1. 掌握计算机网络相关命令原理及应用，能够运用网络命令进行网络故障检测分析。

在仿真环境下，掌握网络设备交换机和路由器的基本操作，掌握现代网络工具的现状及发展。

1. 掌握交换机VLAN设置及验证方法。
2. 掌握在路由器上实现静态路由配置方法
3. 掌握在路由器上实现RIP或OSPF协议配置方法。
4. 基于网络工程任务需求，构建系统方案、包括子网划分、交换机、路由器的配置等，能够实施实验、处理数据并分析实验结果。
5. 基于网络编程的特定任务需求进行socket编程，设计处理流程、数据结构、实施实验并分析实验结果
6. **课程的基本内容和要求**

# 2.1常用网络命令

## 2.1.1 ipconfig

功能：显示各种TCP/IP信息包括IP地址，子网掩码，缺省网关值、虚拟机和无线网卡等的基本信息，并刷新或释放DNS和DHCP的缓存或内容。

命令：ipconfig

作用：显示各个接口的连接状况和相关信息（IP 地址、子网掩码等）

命令截图如下：



图 1ipconfig

命令ipconfig/all

作用：显示设备TCP/IP的详细信息

命令截图如下：



图 2ipconfig/all

命令：ipconfig/？

作用：显示帮助，即用法

命令截图如下：

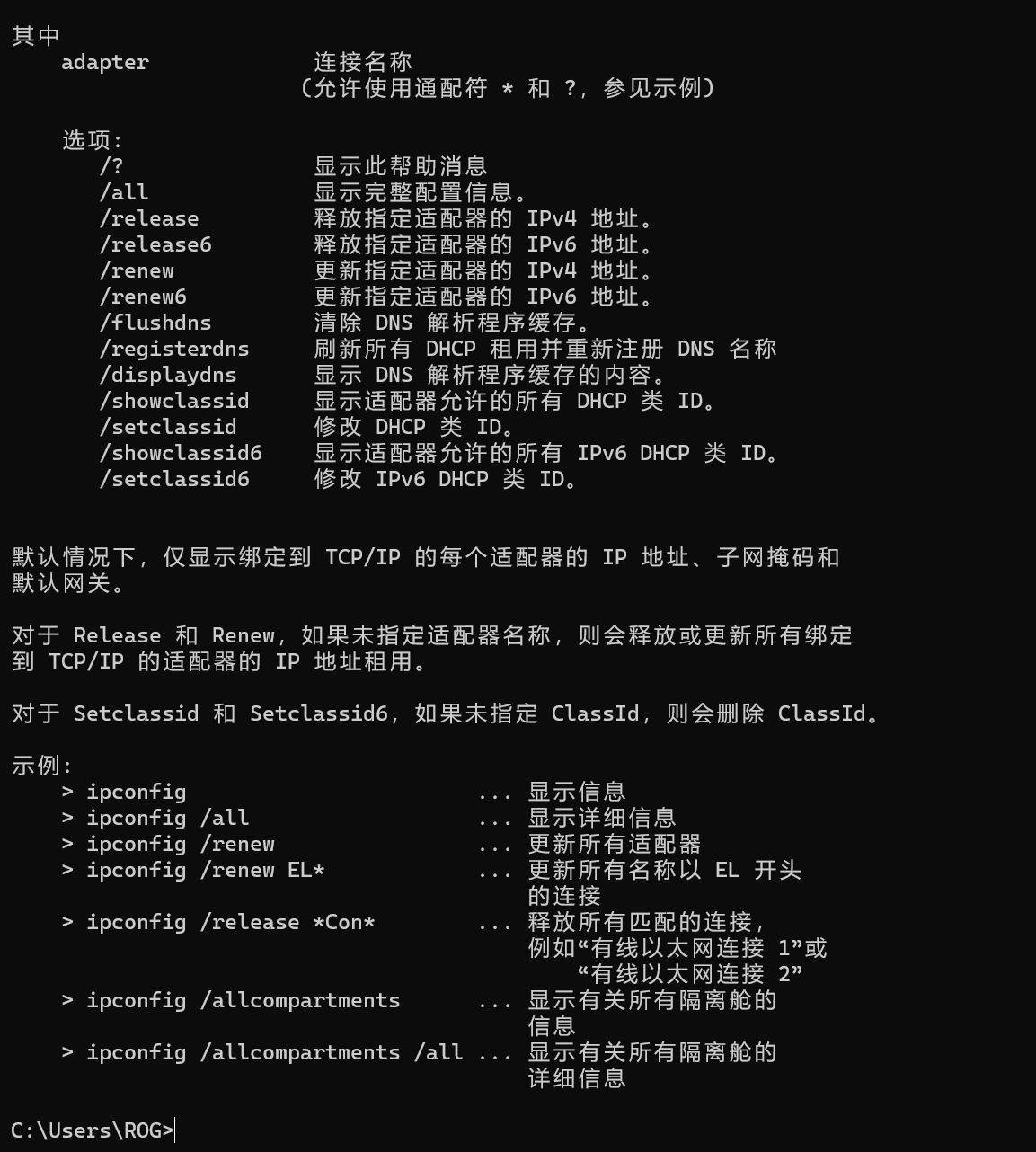


图 3ipconfig/?

命令：ipconfig/renew

作用：手动向服务器刷新请求

命令截图如下：

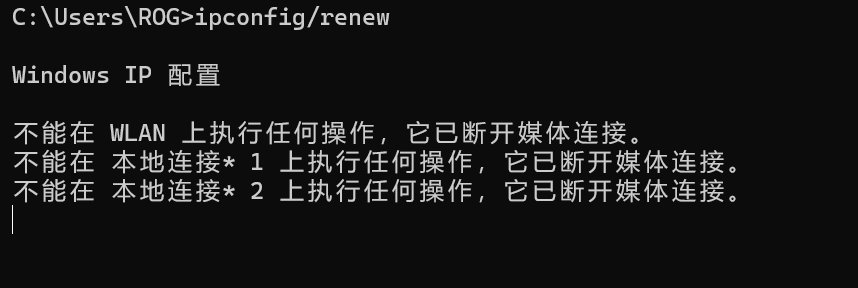


图 4ipconfig/renew

命令：ipconfig/flushdns

作用：刷新DNS解析缓存

命令截图如下：



图 5ipconfig/flushdns

命令：ipconfig/displaydns

作用：显示dns内容

命令截图如下：



图 6ipconfig/displaydns

## 2.1.2 ping

功能：一种网络诊断工具，利用ICMP用于测试TCP/IP参数是否设置正确等

用法：如ping/？命令截图所示：



图 7ping/?

命令：ping -t

作用：Ping 指定的主机，直到停止。要查看统计信息并继续操作输入Ctrl+Break；要停止，输入 Ctrl+C

命令截图如下：



图 8ping/t

命令：ping -a

作用：将地址解析为计算机名

命令截图如下：

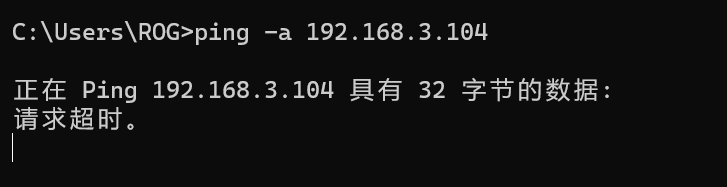


图 9ping-a

命令：ping -n count

作用：发送count个要求回复的请求

命令截图如下：



图 10ping-n count

命令：ping -l size

作用：发送缓存区大小的包

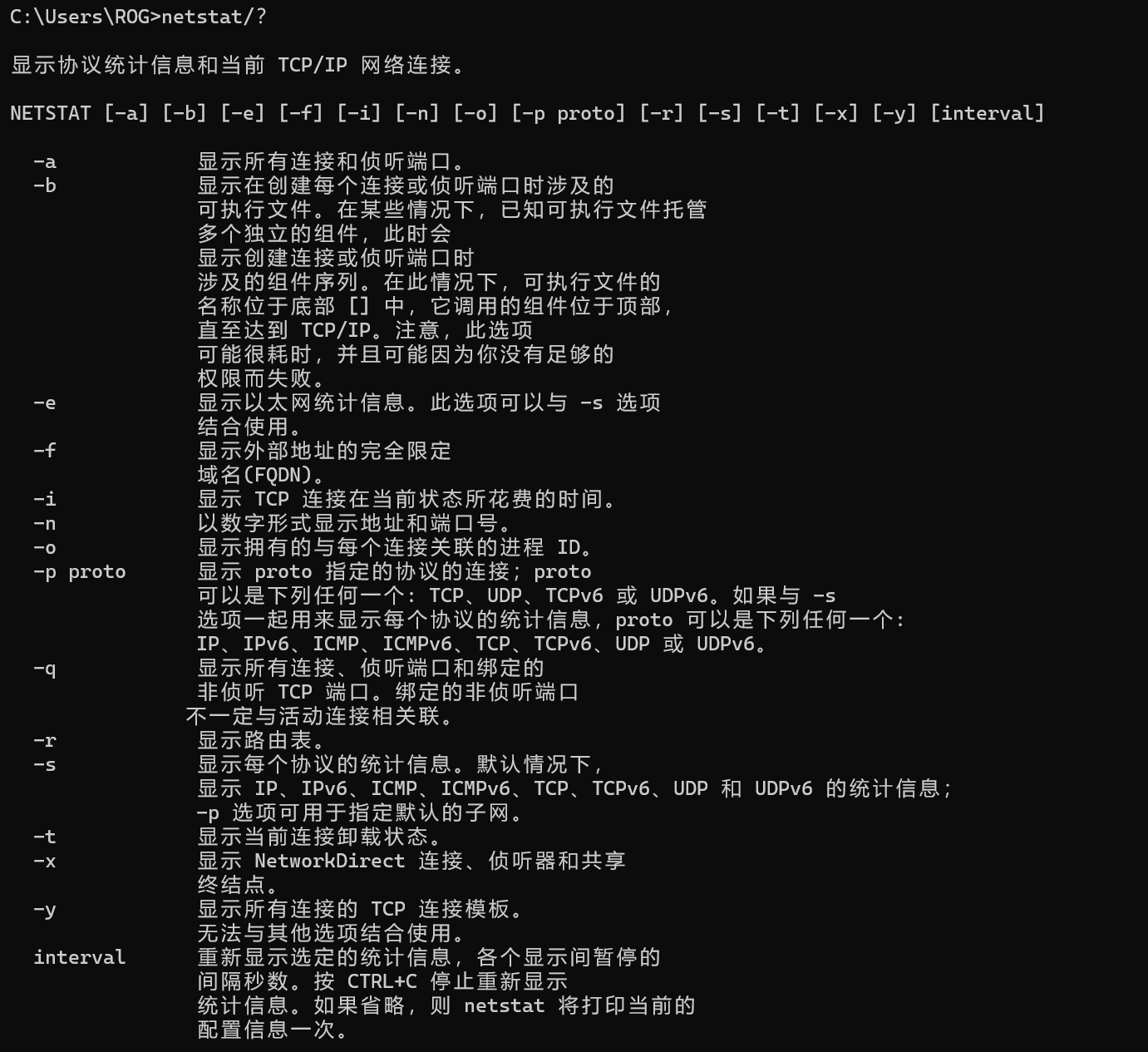
命令截图如下：



## 2.1.3 netstat

功能：用于监控TCP/IP网络，可以显示路由表、实际网络连接以及接口状态信息，让用户知道哪些网络连接正常运作。

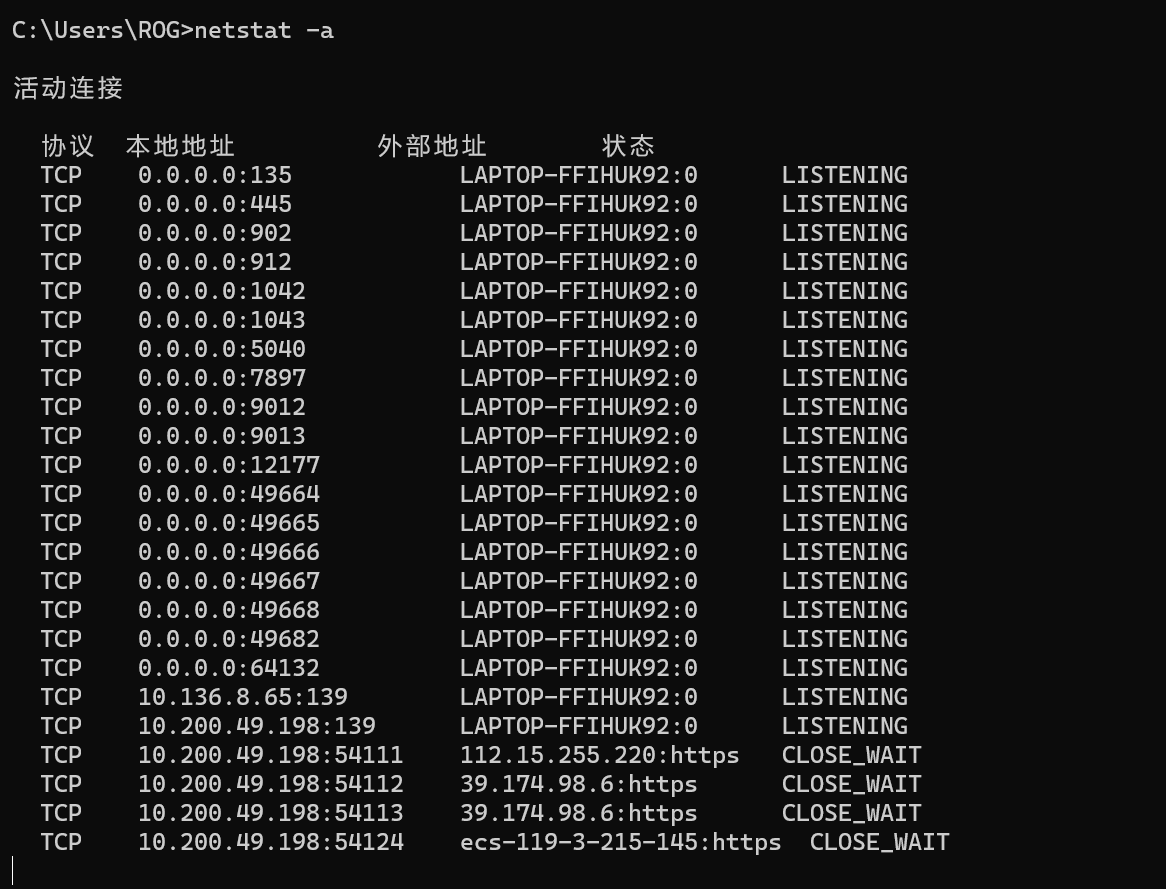
用法：如下图netstat/？指令截图所示：



命令：netstat -a

作用：显示所有socket，对运行的进程状态进行显示

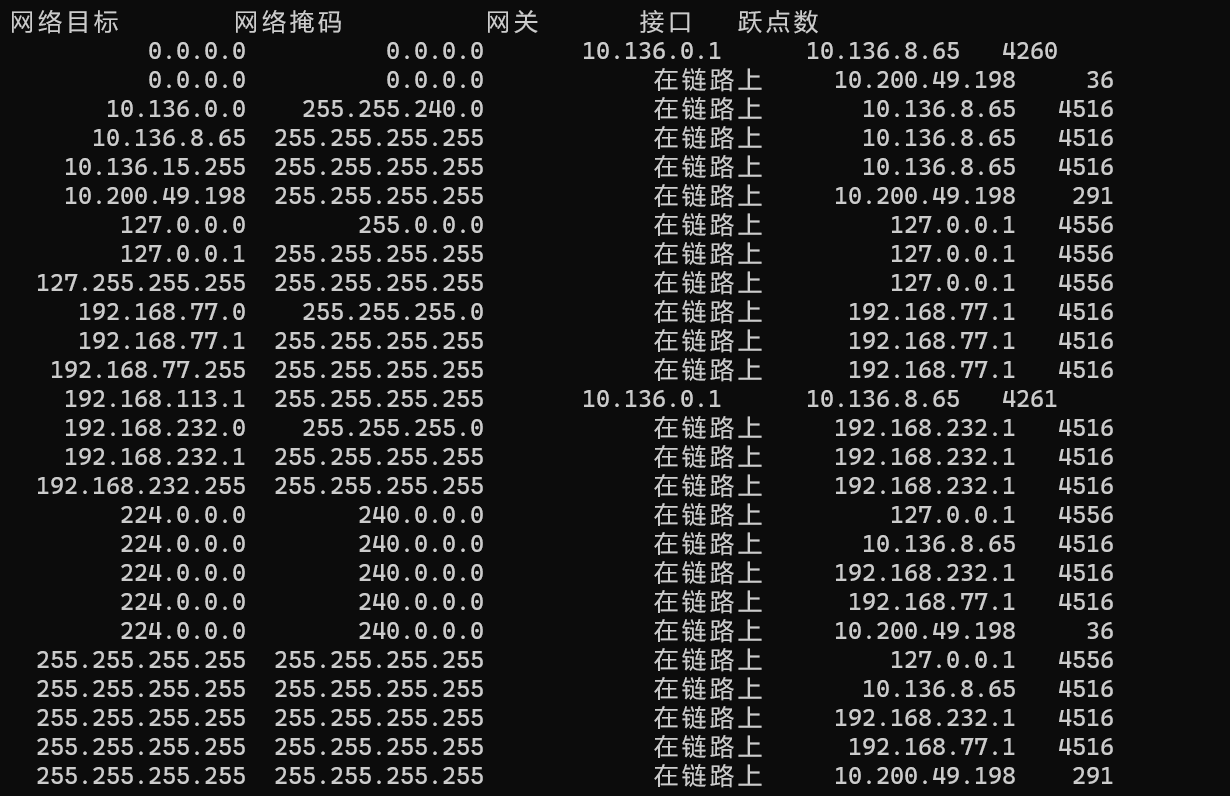
命令截图如下：



命令：netstat -r

作用：显示核心路由表

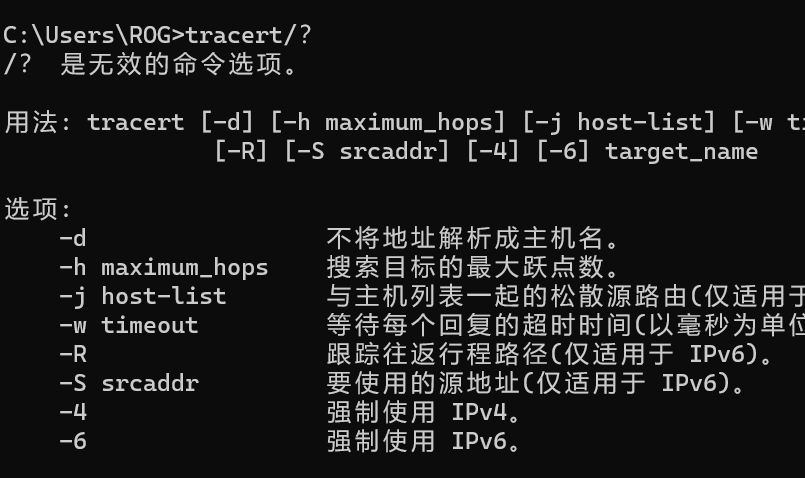
命令截图如下：



## 2.1.4 tracert

功能：是一个简单的网络诊断工具，可以列出分组经过的路由节点，以及它在IP 网络中每一跳的延迟

用法：如tracert/？指令截图所示：



命令：tracert -d

作用：不将地址解析成主机名

命令截图如下：

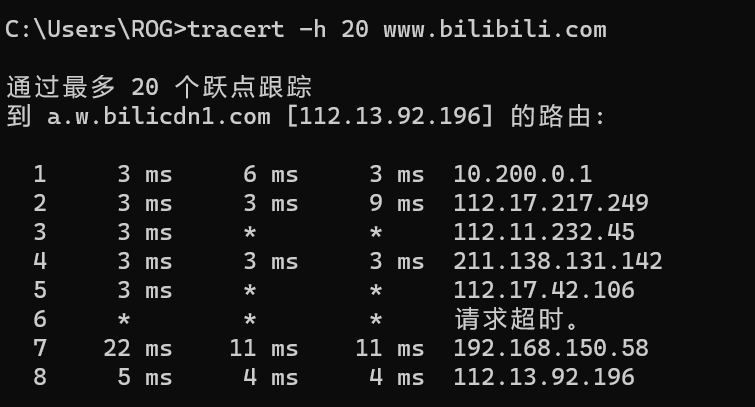


其中，从左到右的 5 条信息分别代表了“生存时间”（每途经一个路由器结点自增 1）、“三次发送的 ICMP 包返回时间”（共计3个，单位为毫秒 ms）和“途经路由器的 IP 地址”（如果有主机名，还会包含主机名）

命令：tracert-h maximum\_hops

作用：指定搜索目标的最大跃点数

命令截图如下：



## 2.1.5 arp

功能：显示或修改ARP协议使用的IP地址和MAC地址的转换表

用法：如arp/？指令截图所示：

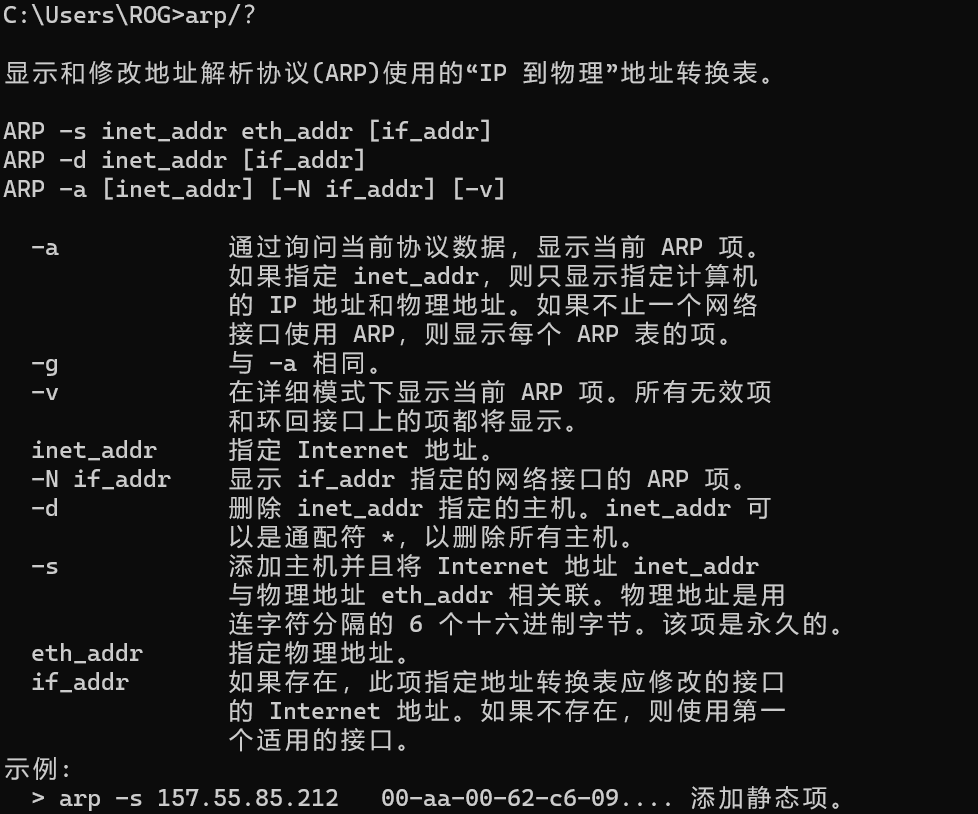
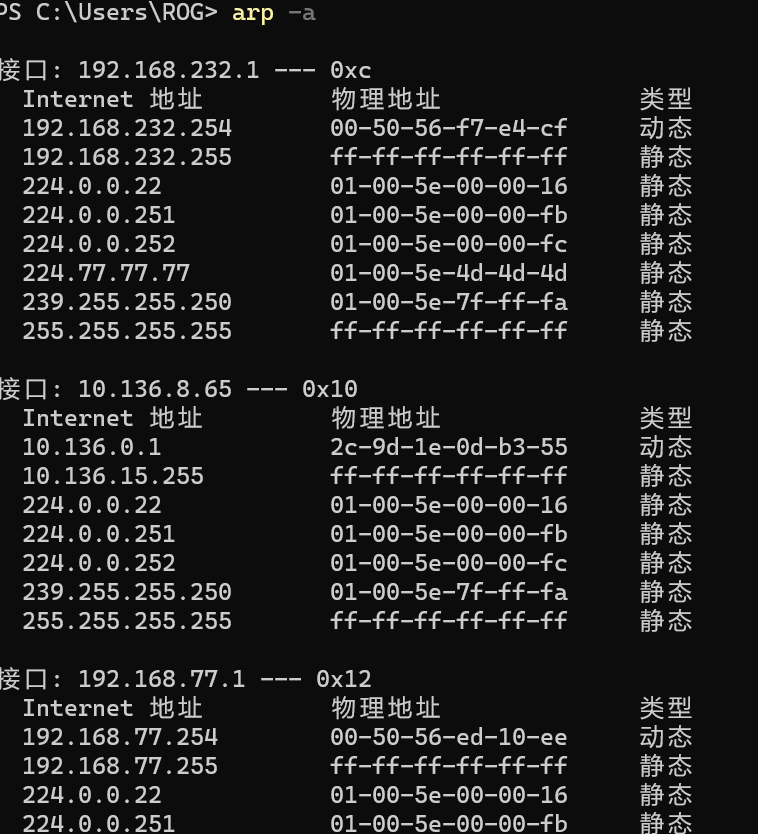


图 11arp

命令：arp-a

作用：显示每个接口的ARP表的项

命令截图如下：

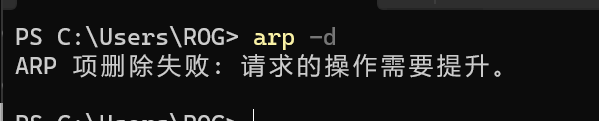


命令：arp -d

作用：删除 inet\_addr 指定的主机。inet\_addr 可以是通配符 \*，以删除所有主机

（权限需要是管理员）

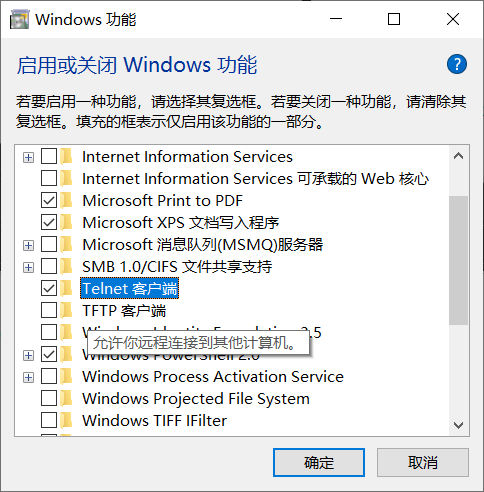
命令截图如下：



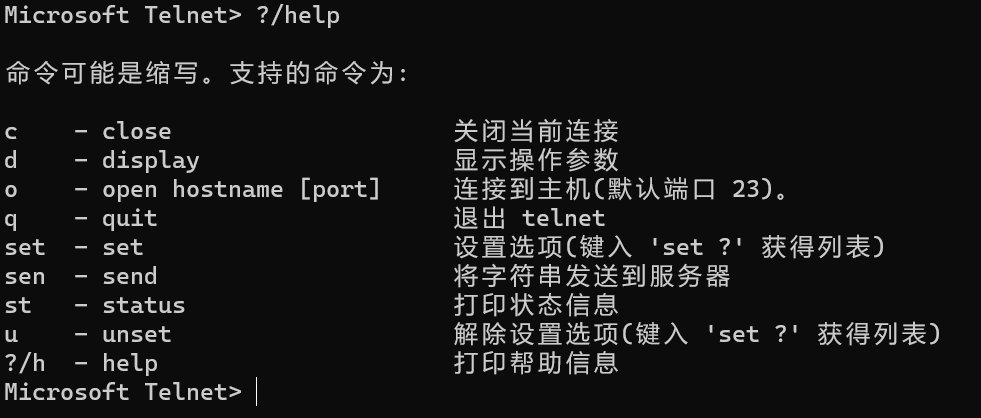
## 2.1.6 telnet

功能：用于远程登录到网络中的计算机，并以命令行的方式远程管理计算机。

注意：先开启telnet客户端才能使用



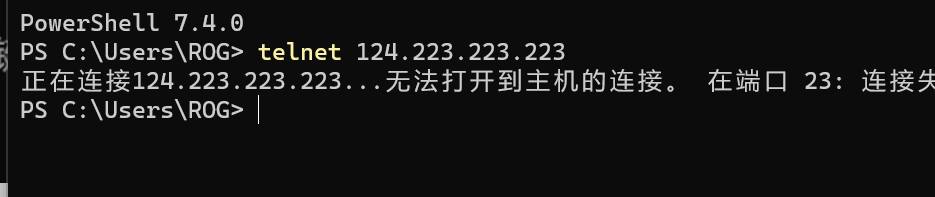
用法：如？/help指令所示：



命令：telnet ip地址

作用：远程连接

命令截图如下：



# 2.2交换机与路由器

## 2.2.1交换机及VLAN配置

**交换机命令**

交换机相关命令：

用户模式： Switch>

特权模式 ： Switch#

全局配置模式： Switch(config)

端口模式 ： Switch(config-if)#

交换机常用的配置命令行：

enable 进入特权模式（一般简写为 en）

config t 进入全局配置模式（一般简写为conf t）

interface fa 0/1 进入交换机某个端口模式（一般简写为int fa0/1）

exit 返回到上级模式（一般简写为ex）

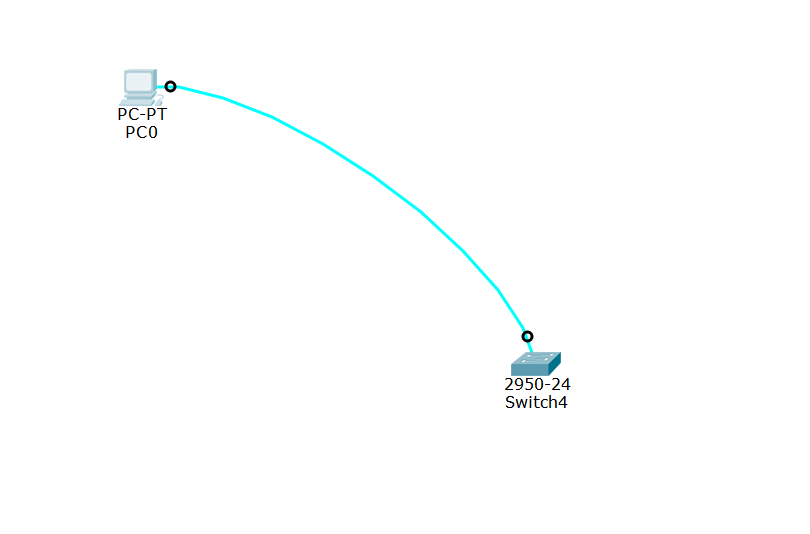
end 从全局以下模式返回到特权模式(一般简写为en)

Switchport access vlan x 将端口接入已经设置好的vlan

Switch mode trunk 将其连接模式改为trunk

**交换机初始化配置**

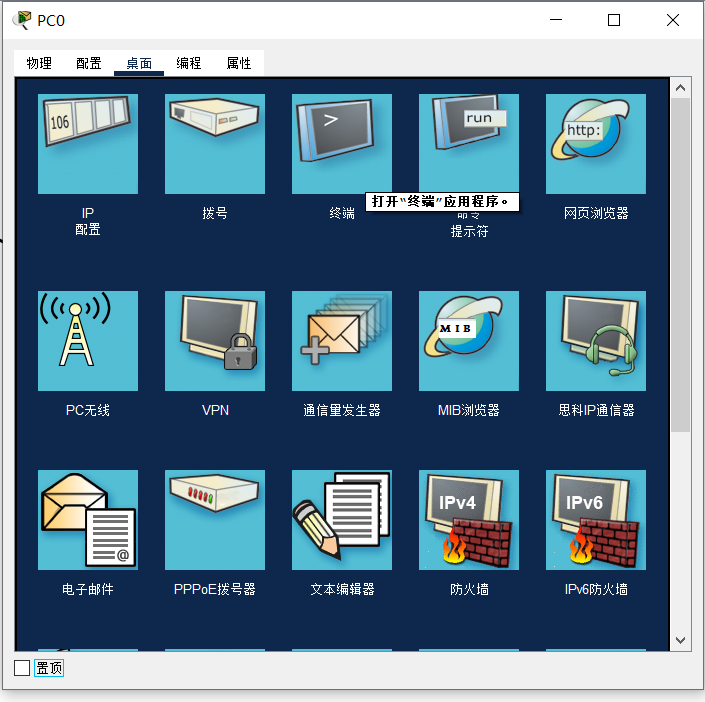
**1.控制台连线**



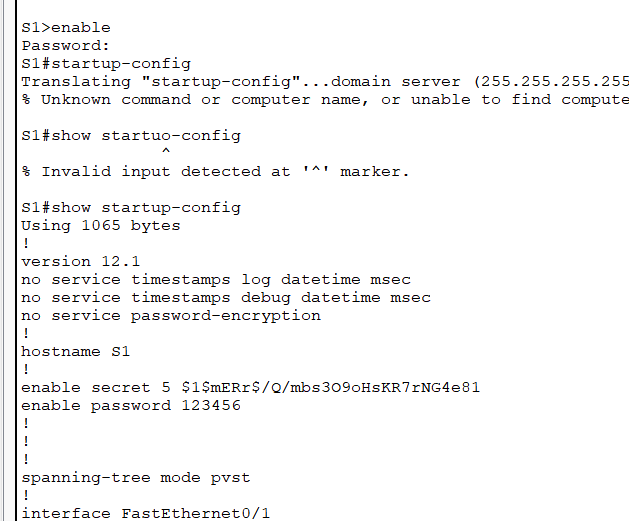
交换机图 1初始化

其中，pc0的rs232 连接 switch0 的console

**2.终端选择**



**3.交换机初始化配置**



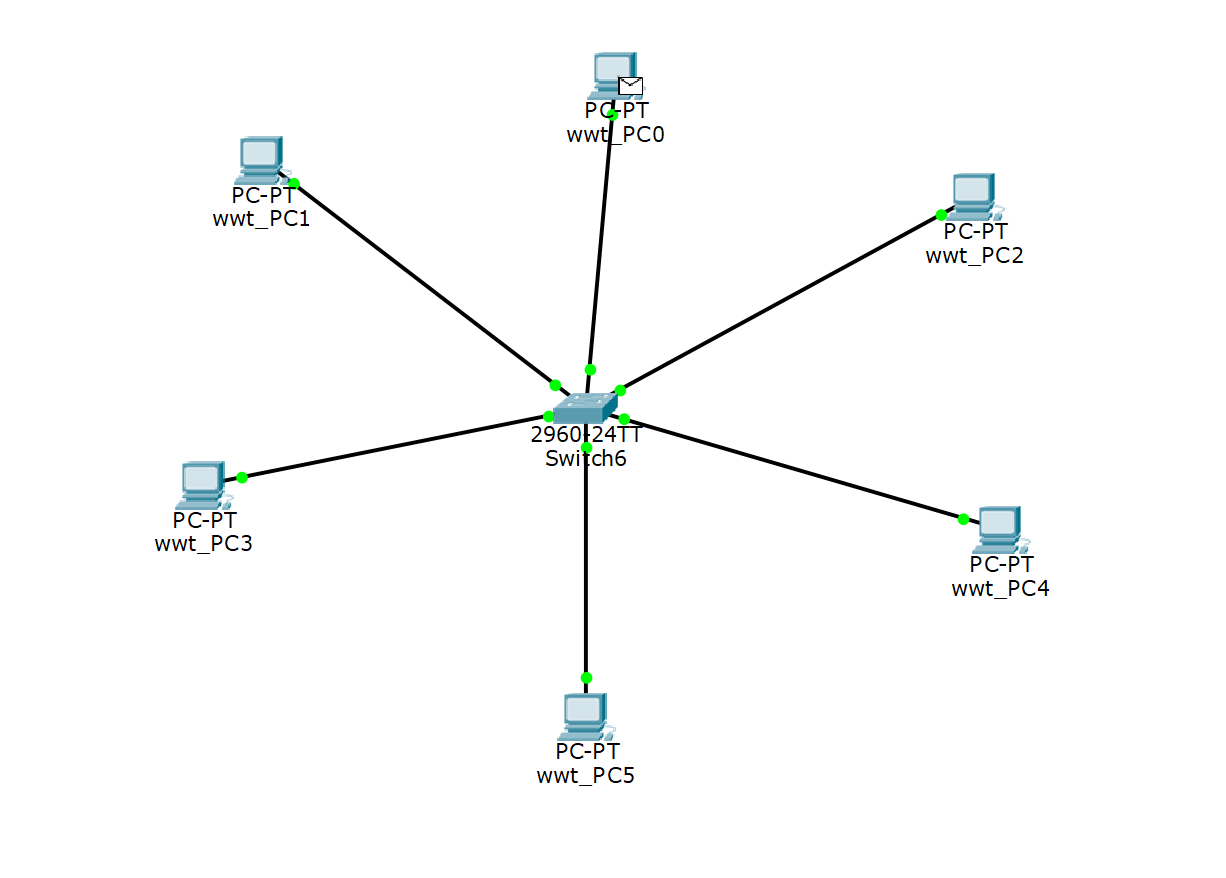
交换机图 2终端初始化

进入enable后，输入setup，初始化配置行用户名和密码的配置，特权模式密码以及IP端口的配置，然后reload，进行show startup-config命令展示配置信息。

## 交换机及VLAN配置

**(1) 初始配置及测试**

连线，端口为Fa/ pc号+1



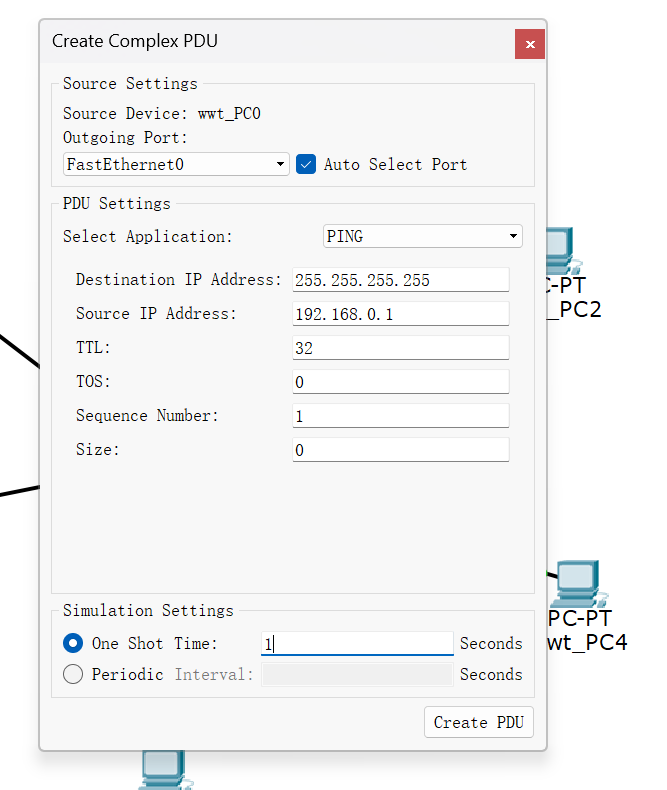
交换机图 3测试拓补图

如图所示，图中的六个主机分别对应交换机开放的六个主机，都为默认vlan1

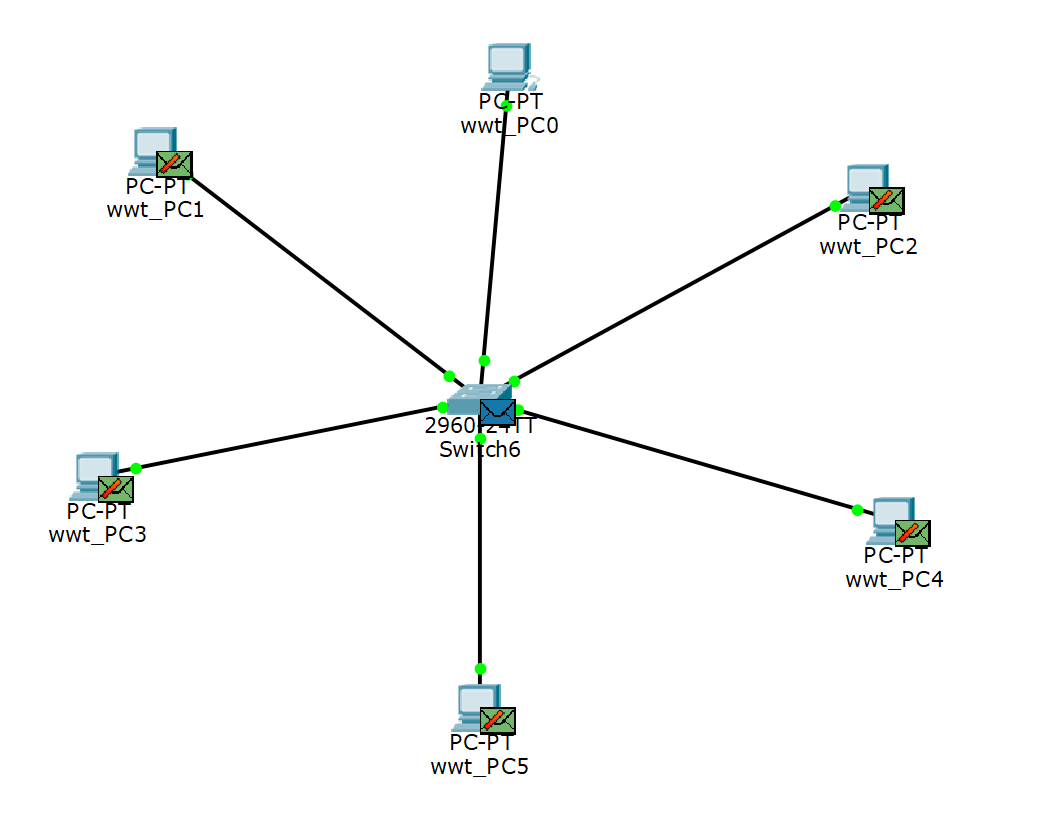
0、1 IP：192.168.0.

2、3IP：192.168.1.

4、5IP：192.168.2.



测试结果是正确的，所有pc都能收到IP数据报



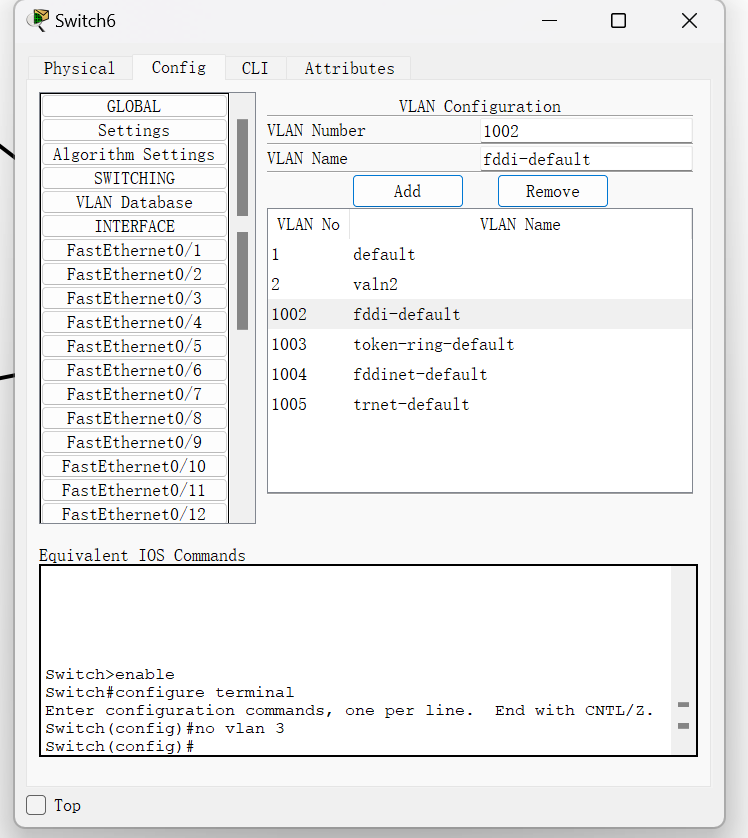
交换机图 4测试结果

**(2) 划分VLAN**

**1.新建vlan2，3**

此处的实现将面板实现和命令行实现都列出，

该实现的面板实现为：



交换机图 5划分vlan

同理，增加vlan3.

该实现的命令行实现为：

创建一个新的 VLAN

Switch>enable //进入特权模式

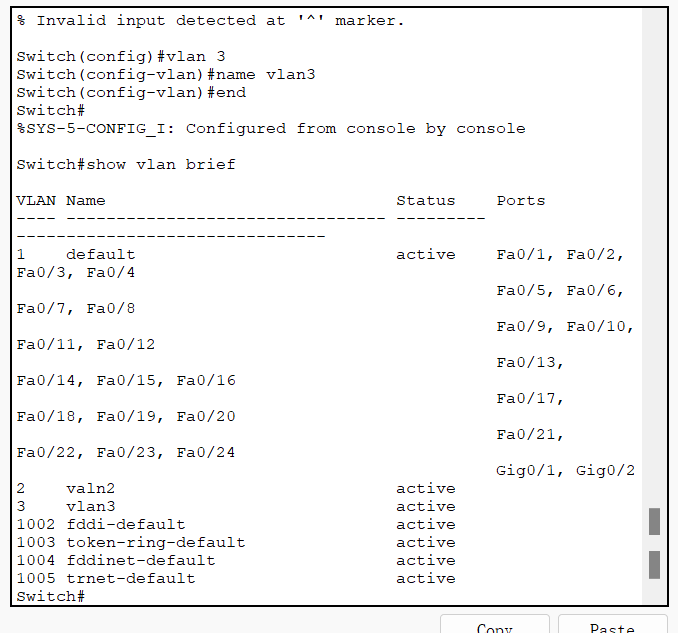
Switch#config terminal //进入全局配置

Switch(config)#vlan 3 //new 一个新的vlan

Switch(config-vlan)#name vlan3 //命名

Switch(config-vlan)#end

Switch#show vlan brief //查看 vlan 信息

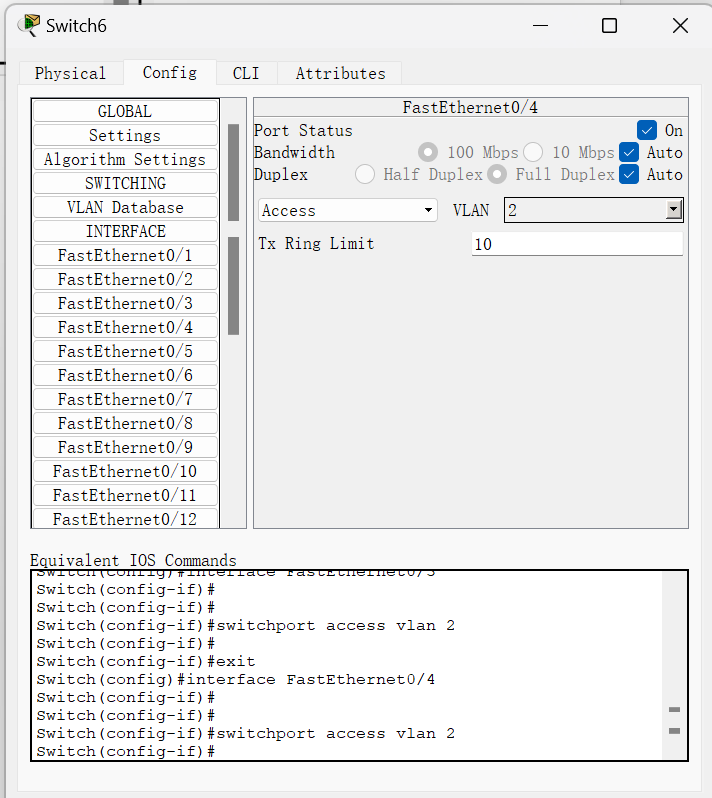


**2.将端口加入VLAN**

本文将端口fa3，4加入vlan2，fa5，6加入vlan3

此处的实现将面板实现和命令行实现都列出，

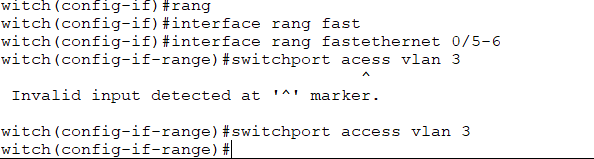
该实现的面板实现为：



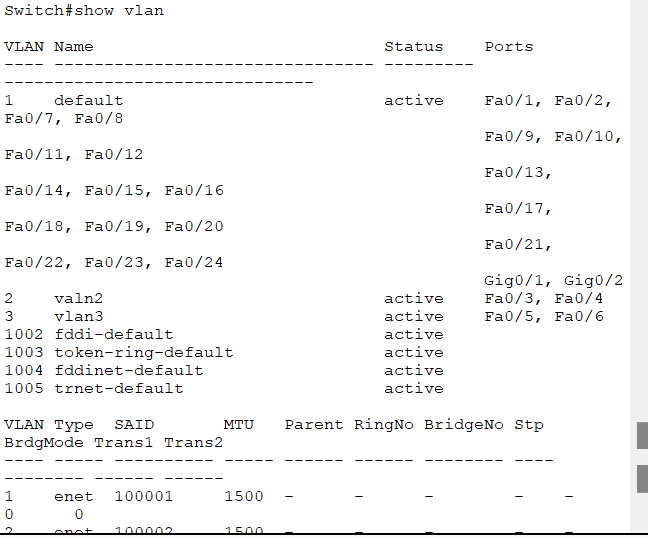
交换机图 6加入vlan

该实现的命令行实现为：

此处将5，6加入 vlan3



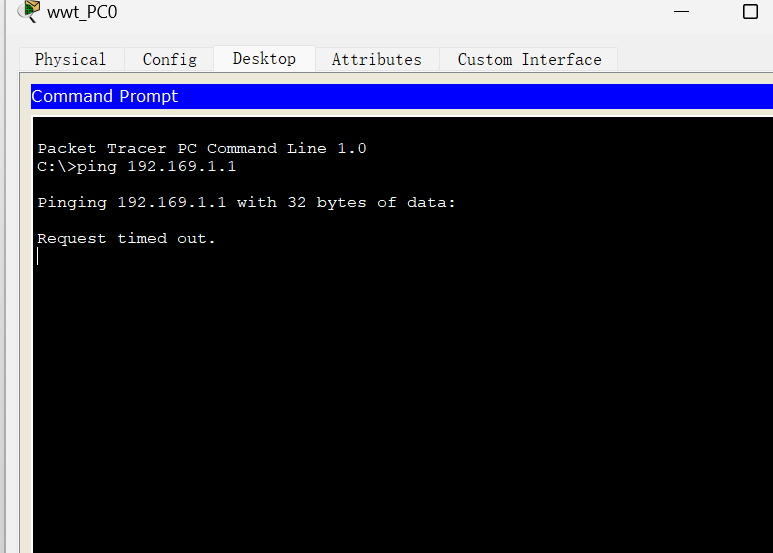
此时的端口状态



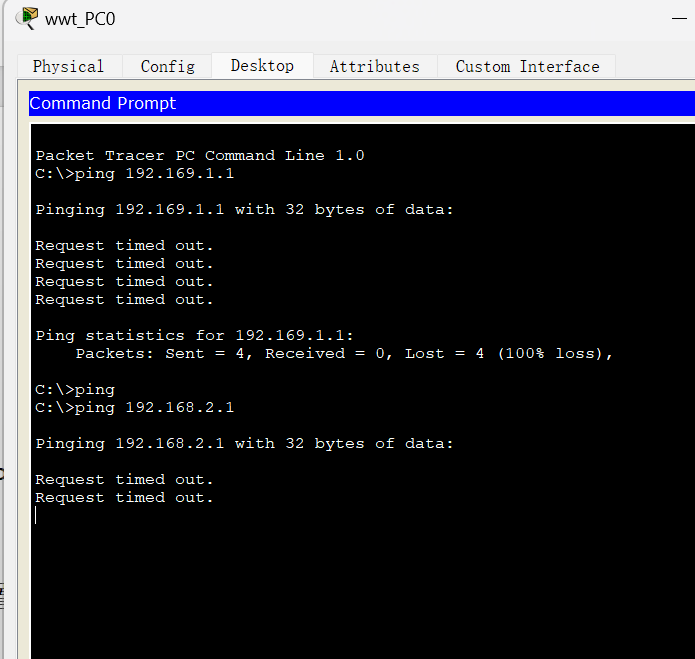
**(3) 测试新的vlan**

依旧从pc0发送ip数据报，发现不同的vlan之间不互通。

Pc0 -> pc2



Pc0 ->pc5



## 2.2.2路由配置与静态路由设计

**(1) 路由器相关命令**

进入：config terminal/memory/network

配置网络时常采用的命令：copy 和 load

1. 标识：hostname 标识名
2. 启动标识：banner 启动标识
3. 接口：interface 端口号
4. 密码：line 0 6

login

passwd

enable password/secret

1. 接口：

1）配置端口

interface 端口号

clock rate 时钟速率（64000）在来配置serial串口的，起同步作用，这个时钟频率和bandwidth带宽相关，单位是位每秒bps，带宽多大就多大

bandwidth 设置带宽（缺省 56）在串口中配置

media-type 介质类型 在以太网口上 fa

no shutdown //不关闭

write memory

2）检验端口

show interfaces

show controllers

6． 查看邻居路由

show cdp interface

show cdp neighbors [detail]

show cdp entry routerA

7． IP Address 配置

Ip address 网络地址 掩码

Ip host 主机名 address

Ip domain-lookup nsap

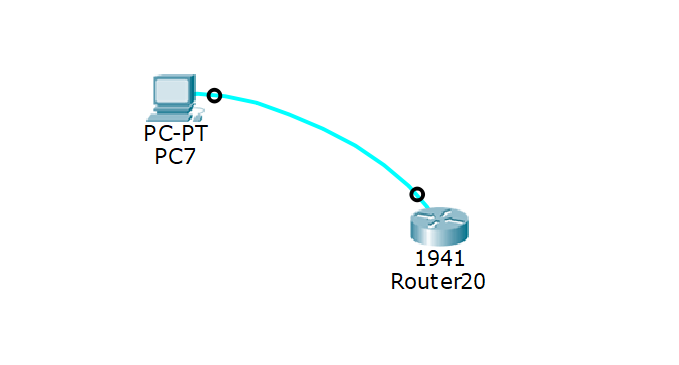
Show hosts

Ping 主机名/IP 地址

**(2) 基于console路由器基础配置**

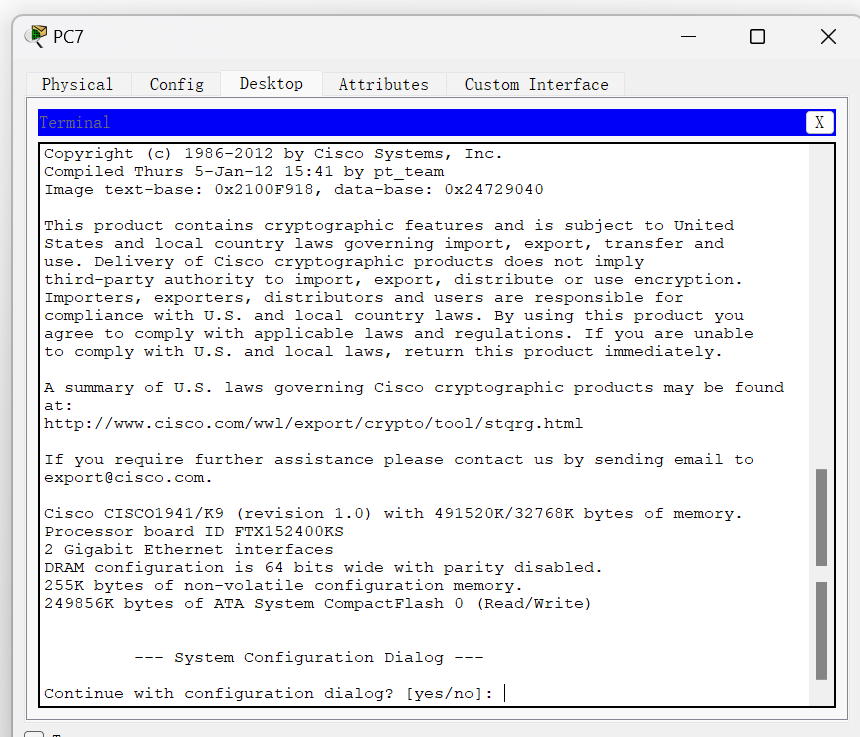
**1．接线**

利用一台主机一台路由器做简单的接线，利用控制台线将它们相连。

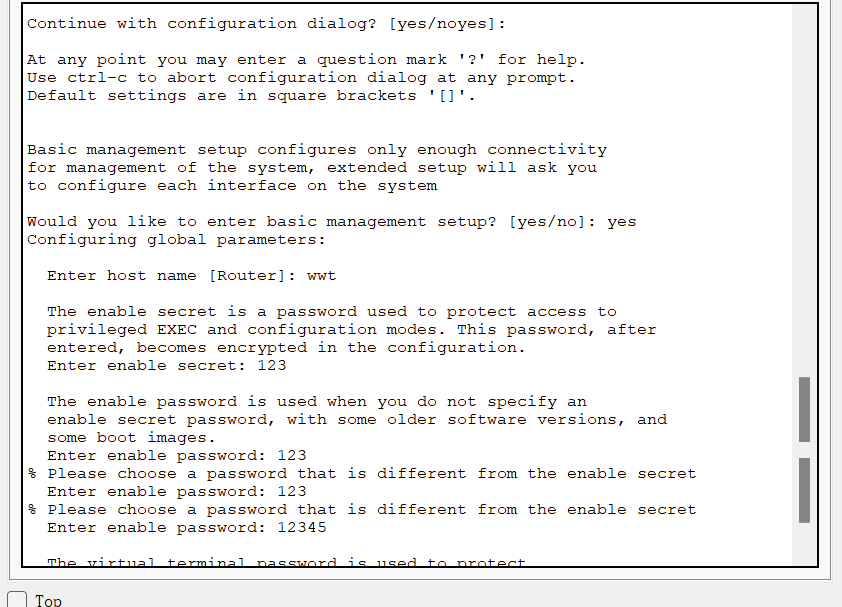


**2．配置**

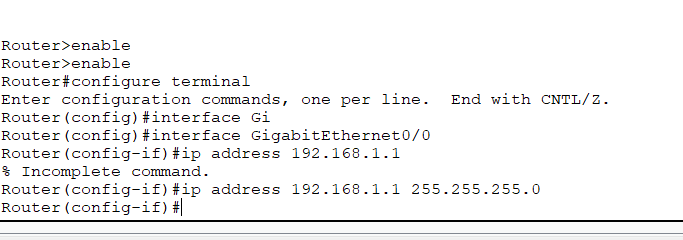
进入pc0的终端



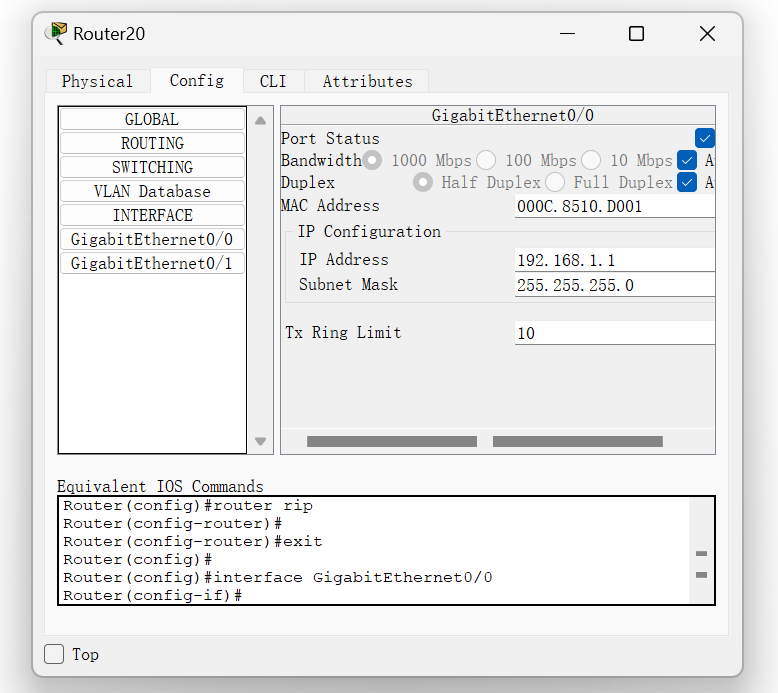
依旧是路由器名以及相关密码的填写



然后是端口ip

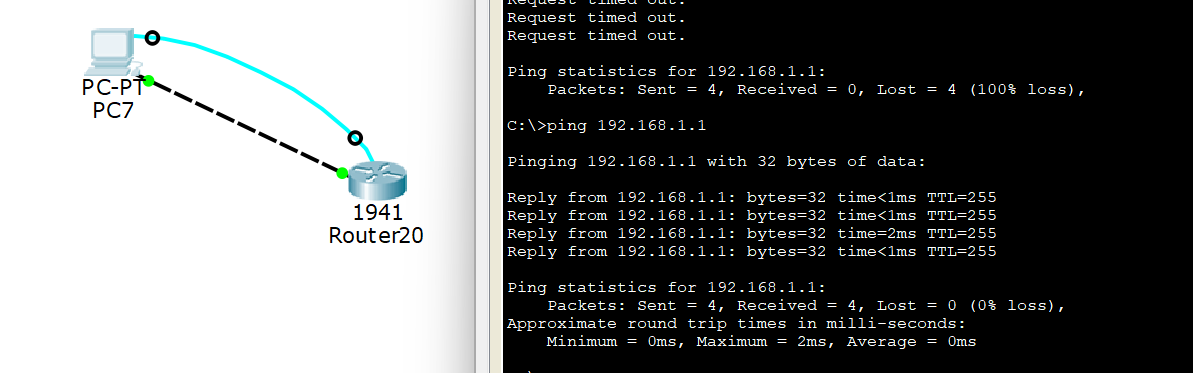


**3．更新并测试**



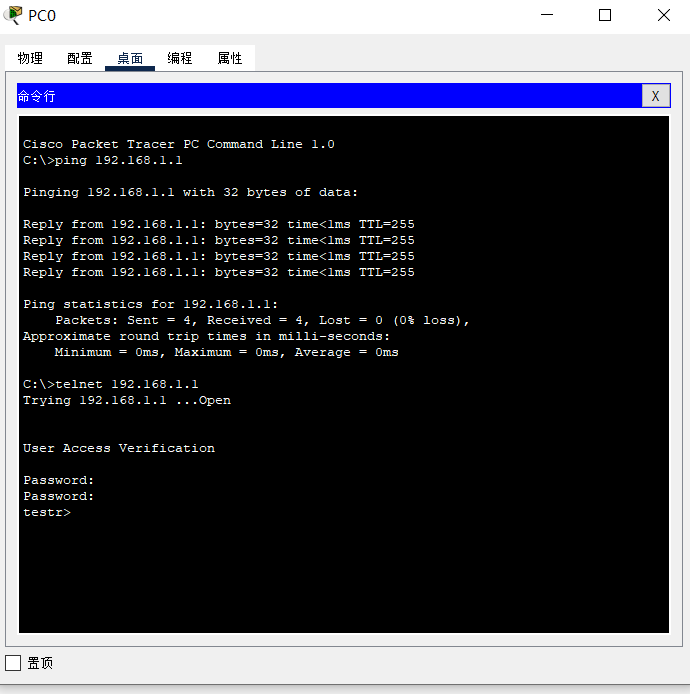
首先设置pc0的ip地址为192.168.1.x（因为路由器配置的ip是192.168.1.1）

Pc0尝试ping路由器，ping通



**4．telnet登录测试**

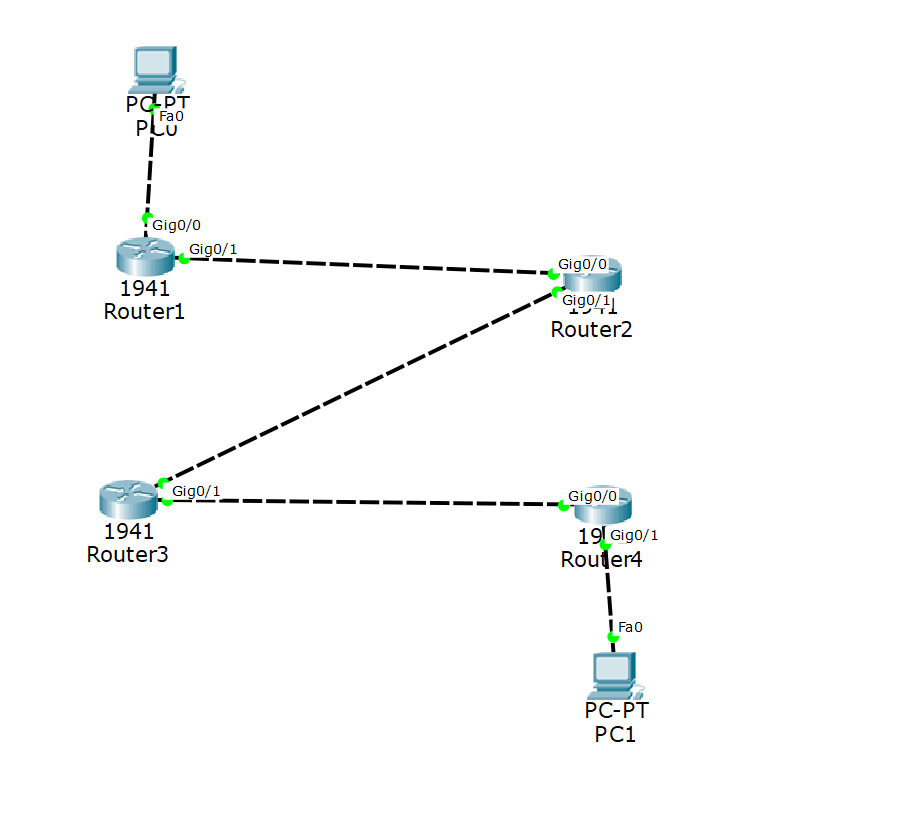
输入之前创建的密码，登录成功



## 静态路由配置

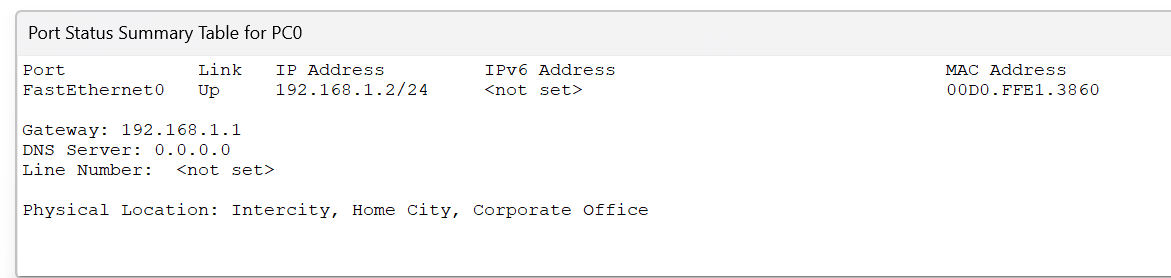
**(1) 线路配置及ip设置**

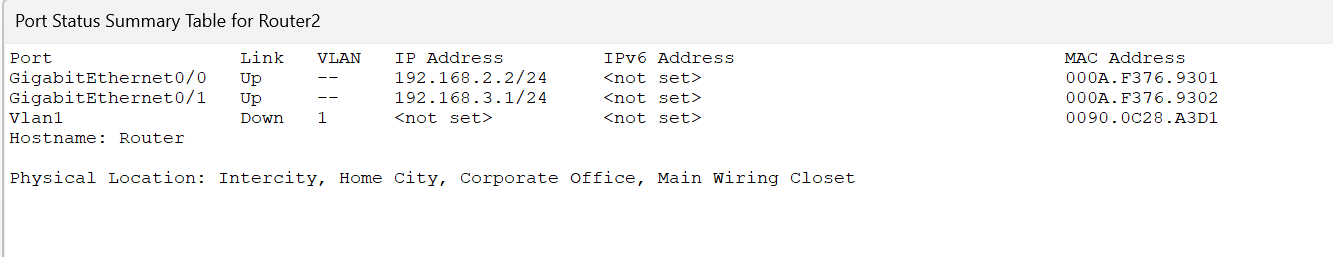
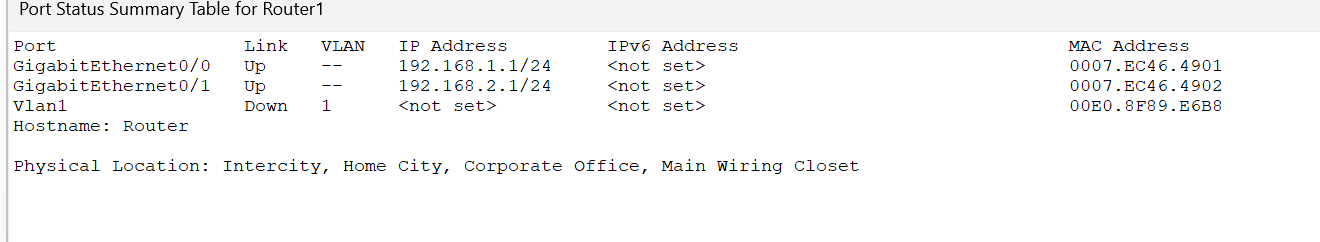
首先是进行线路配置，并把端口打开，如下图所示：

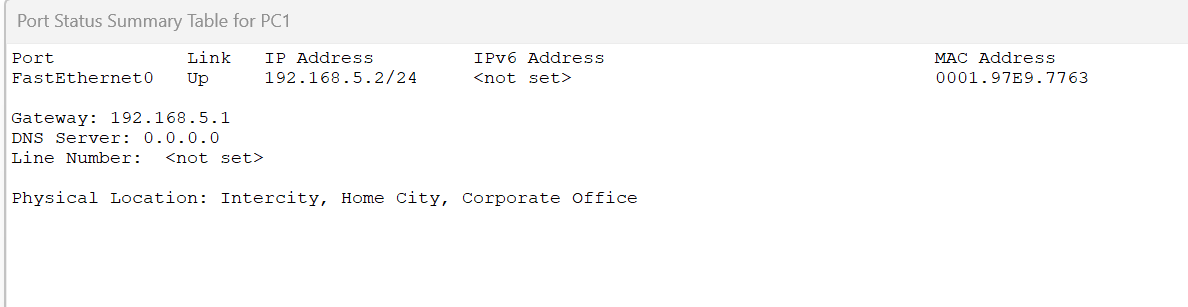
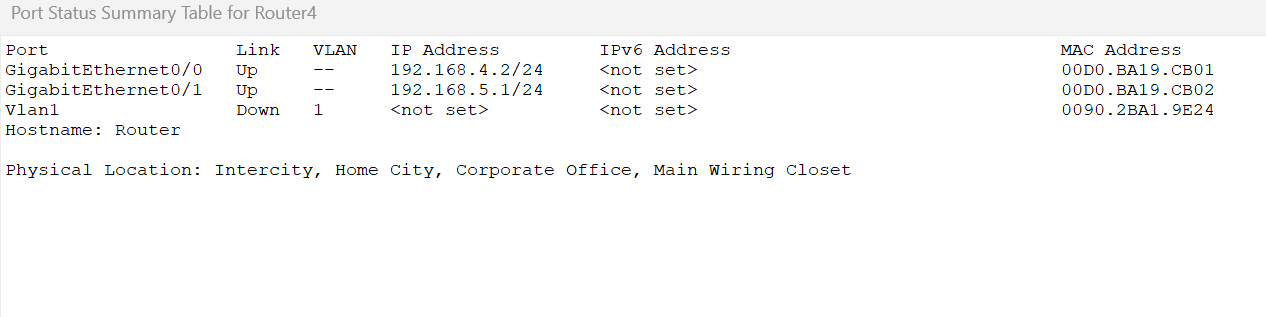
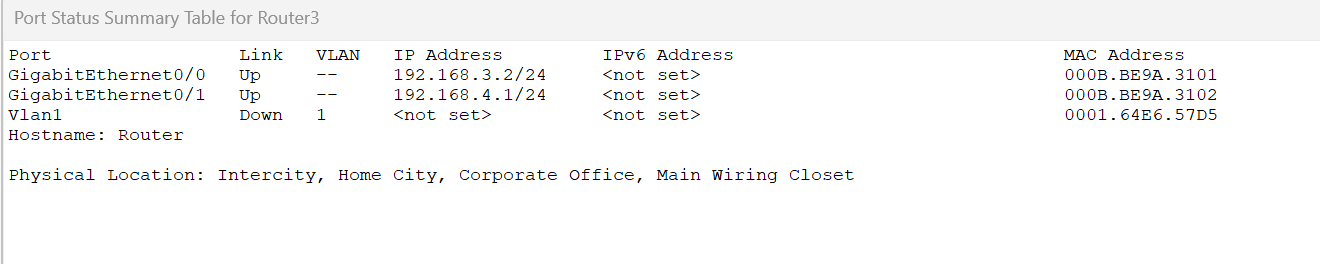


静态路由 1拓补图

拥有四个路由器，两台主机用于静态路由的配置，其中连接的线为交换线，各主机，端口的ip地址如下图所示：



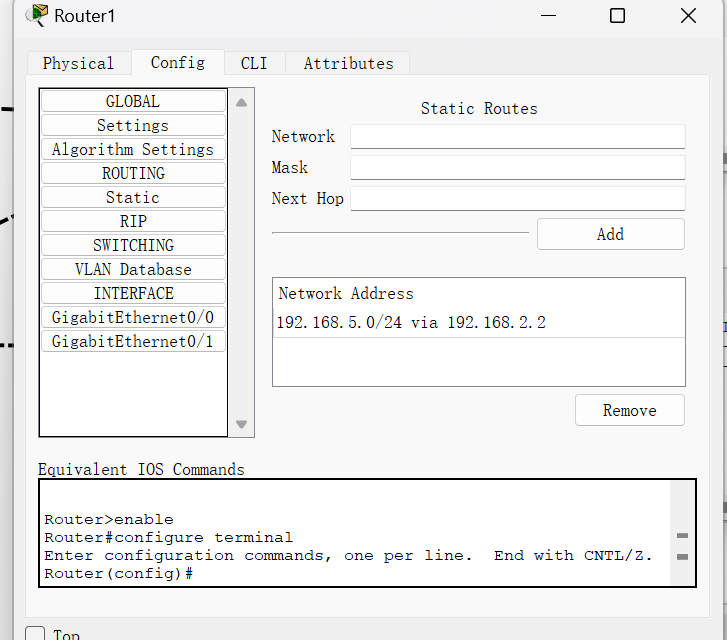


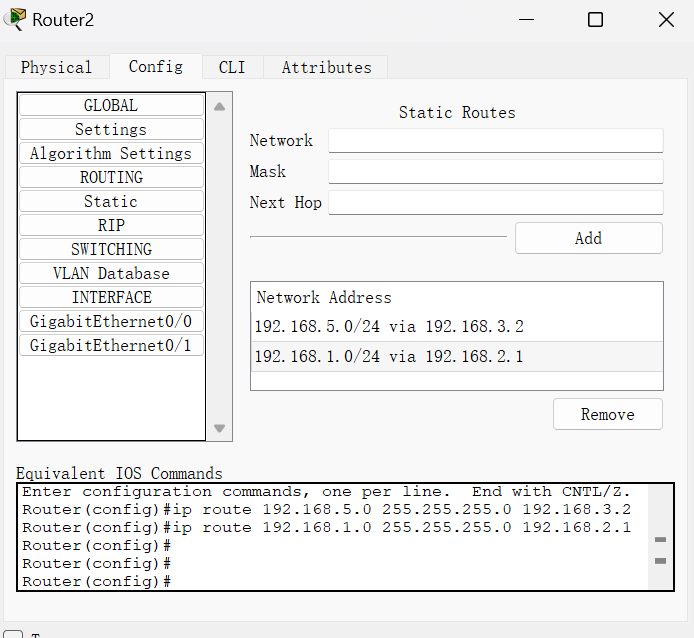


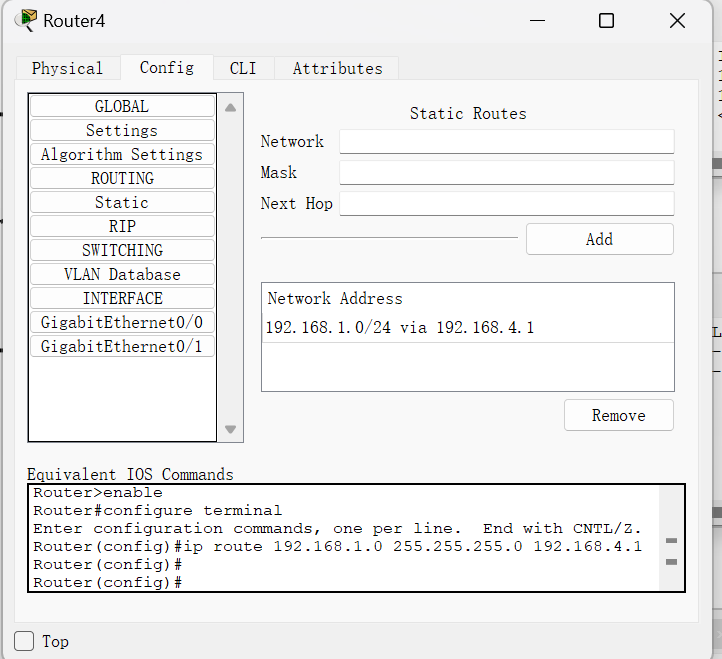
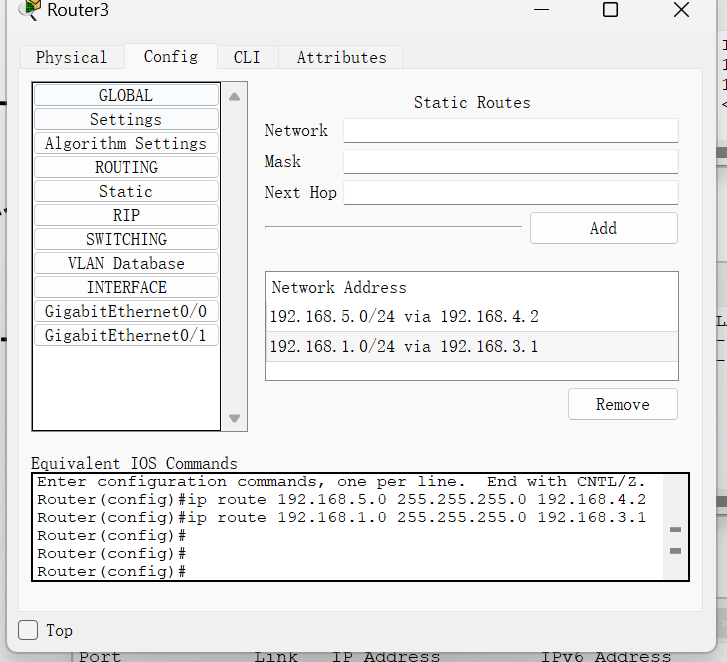
静态路由 2ip划分

**(2) 静态路由设置**

对于将PC0（192.168.1.2），以及PC1（192.168.5.2）作为目的地址，此刻设置子网掩码为255.255.255.255以代表特定主机路由，并对四个路由的下一跳端口进行依次设计，如下图所示：





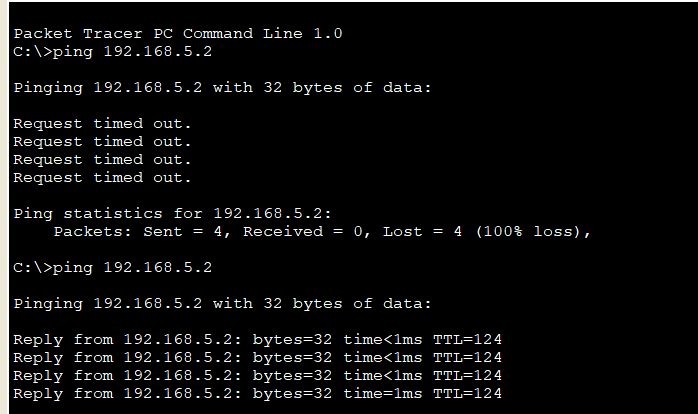


静态路由 3静态路由配置

其中，实现指令为ip route 目的网络 掩码 { 网关地址 | 接口 }

**(3) 测试**

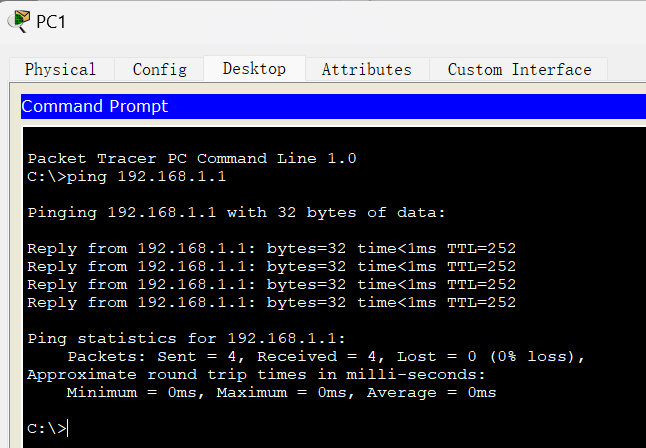
首先是pc0 ping pc1的操作，如下图所示：



第一次ping不通：事先路由不知道物理地址，需要arp，所以丢失

第二次ping通，说明配置完成

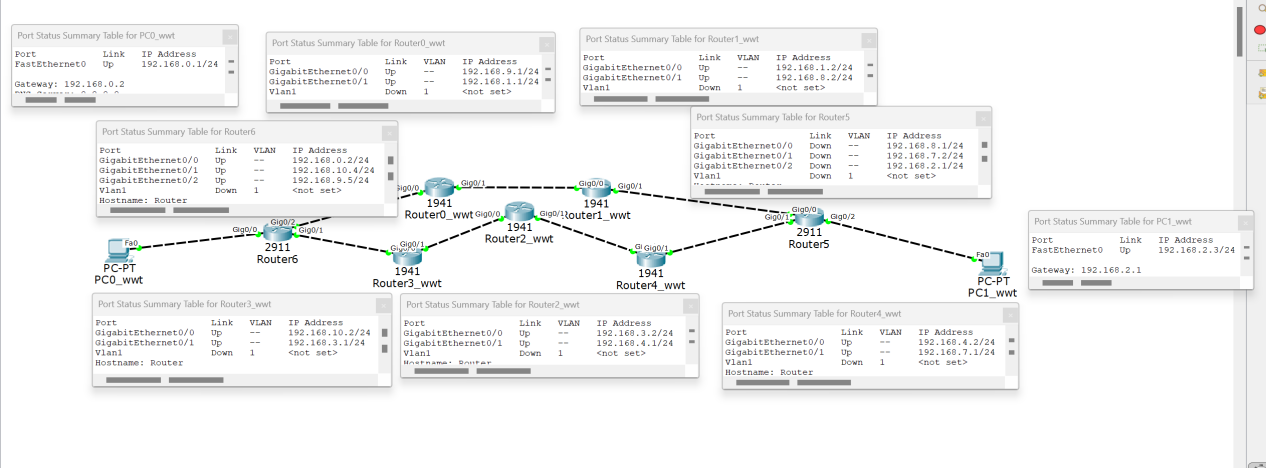
然后是进行反向ping的操作，如下图所示：



## 2.2.3动态路由设计

1. **网络拓扑图**

IP配置



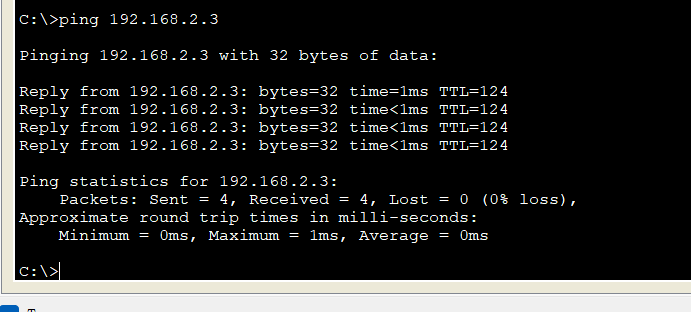
动态路由 1拓补图

1. **配置RIP**

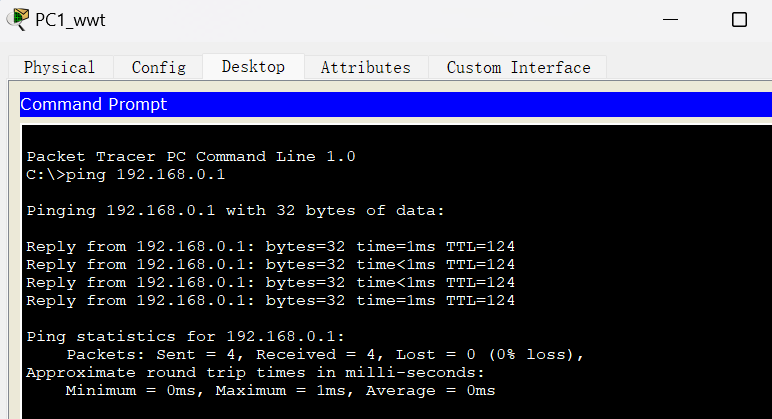
配置各个路由器rip设置为其接口的IP

1. **测试**

PC0->PC1

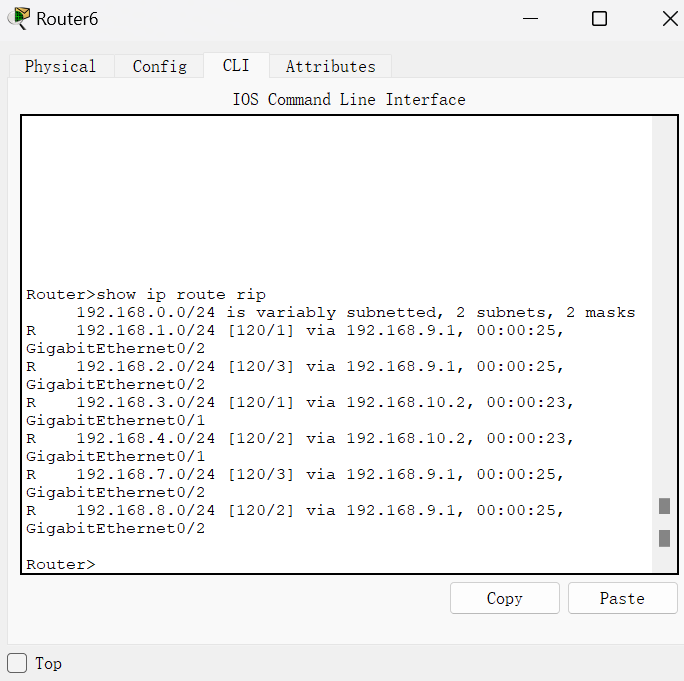


PC1->PC0



动态路由 2测试

1. **分析RIP**



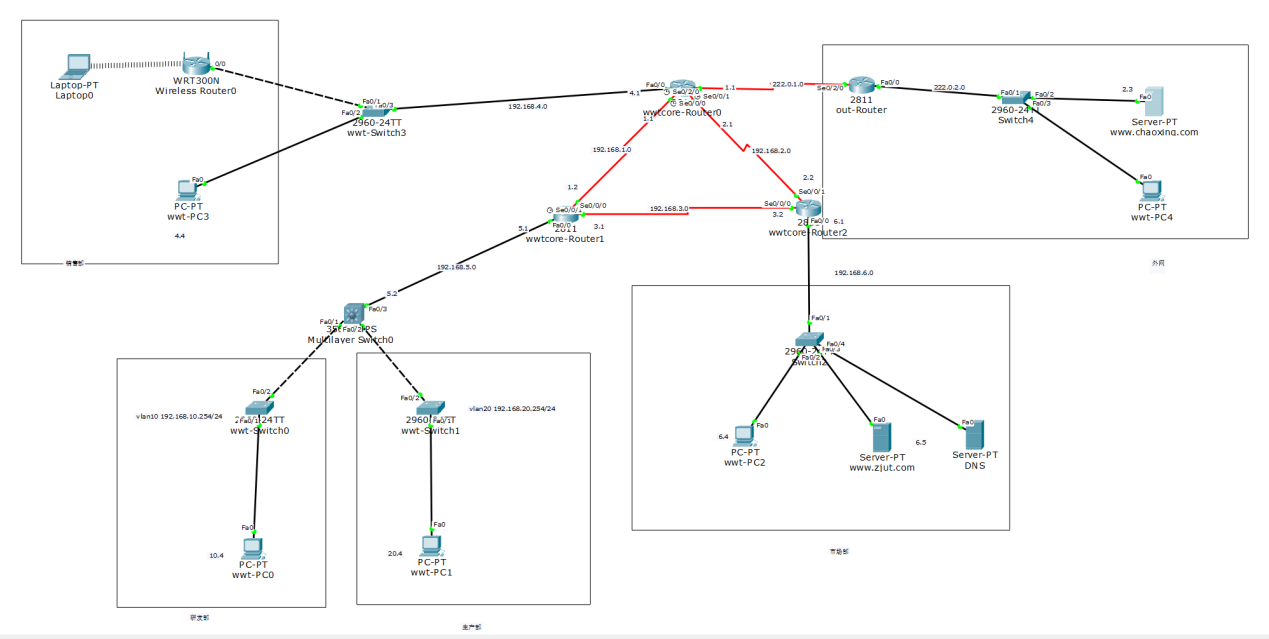
动态路由 3rip分析

选取router6，观察它的路由表，使用show ip route rip命令

可以看到，如果想要发到192.168.2.0会到192.168.9.1这个下一跳，即更短的那条路径。

## 2.2.4网络集成设计

**(1) 网络拓扑图**



网络集成 1拓补图

**(2) 简单介绍**

利用PC，二层交换机，三层交换机，路由器，服务器，等设备设计小型企业网络，目前主要完成设计的是企业内局域网。通过模拟现实场景中企业用户的上网需求。为企业划分销售部、市场部、研发部、生产部，用于满足各部门间的通信需求。

**(3) 主要实现的功能**

1、根据部门划分网段，其中生产部和研发部在一个网段中的两个vlan ，根据rip实现内部互联

2、利用三层交换机通过单臂路由的模式实现不同两个部门间PC的相互通信

3、nat地址转换与外网单向联通

4、DNS域名解析服务器部署

**(4) ip地址划分**

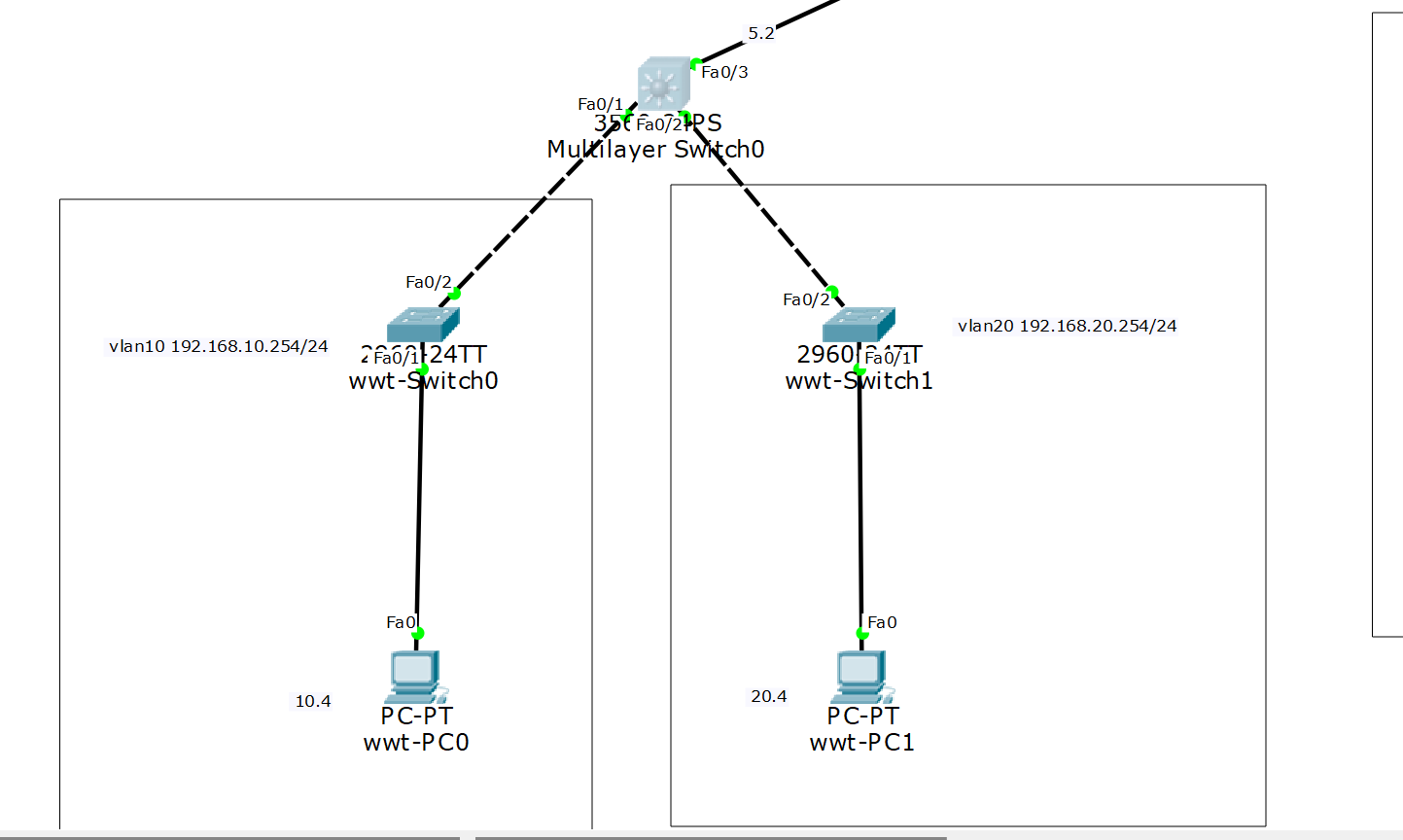
|  |  |  |
| --- | --- | --- |
| **名称** | **Ip网段** | **默认网关** |
| **销售部** | **192.168.4.0** | **192.168.4.1** |
| **研发部** | **192.168.10.0** | **192.168.10.254** |
| **生产部** | **192.168.20.4** | **192.168.20.254** |
| **市场部** | **192.168.6.0** | **192.168.6.1** |
| **外网** |  |  |
|  |  |  |

网络集成 2ip地址划分

**(5) 三层交换机vlan联通实现**

创建vlan10、20

配置（no switchport）vlan10的ip作为默认网关192.168.10.254，vlan20的ip作为默认网关192.168.20.254



网络集成 3三层交换机

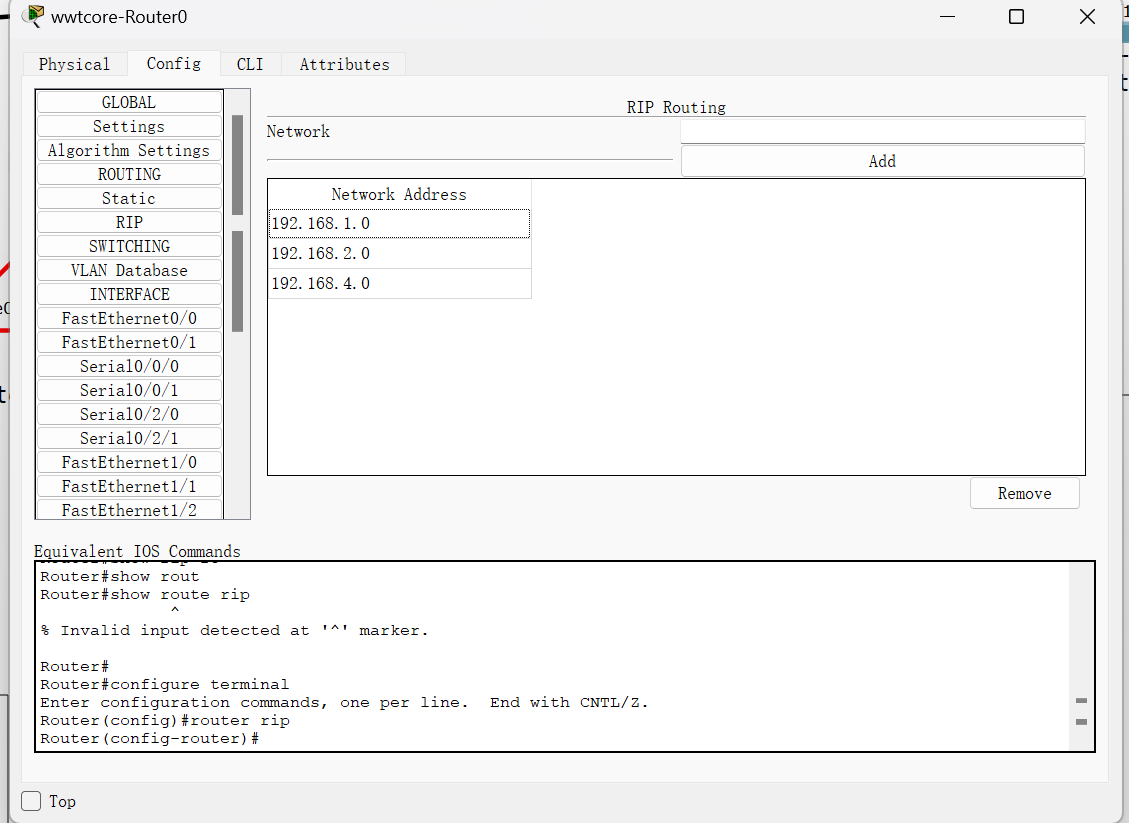
把三层交换机的f0/3口打开（no switchport），添加ip192.168.5.2。

将三层交换机f0/1和f0/2的口打开trunk模式。

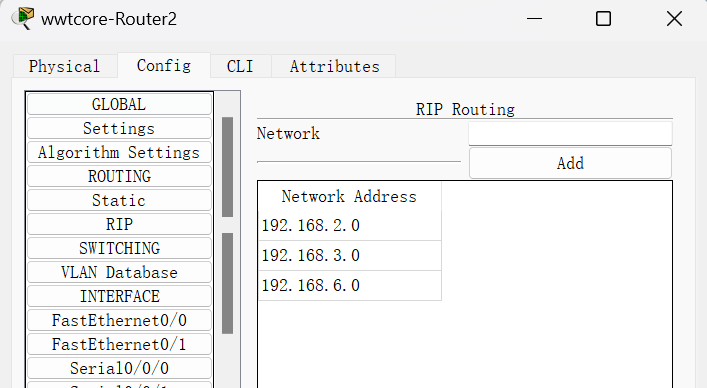
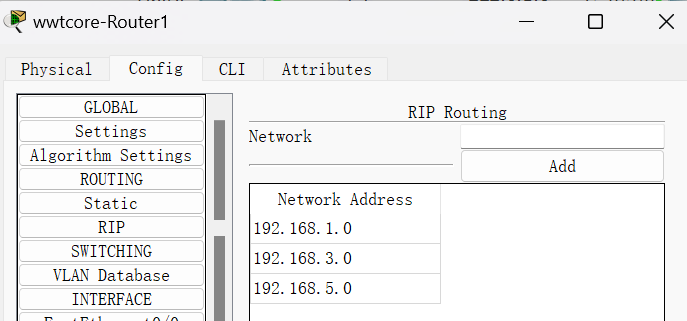
将二层交换机添加vlan10和vlan20并且将f0/2打开trunk模式，f0/1打开access模式。

**(6) 通过rip实现企业网内互通**

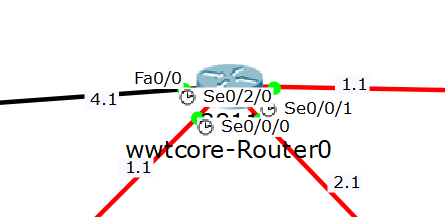
将三个核心路由器以及三层交换机添加自己所链接的网段，完成rip。



网络集成 4rip配置

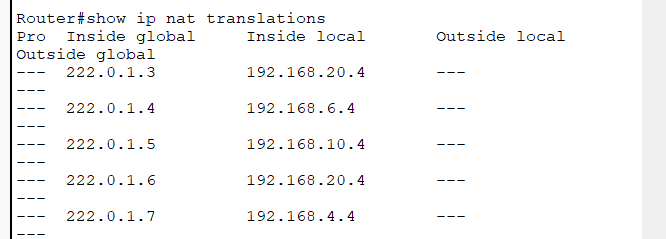


1. **添加NAT**



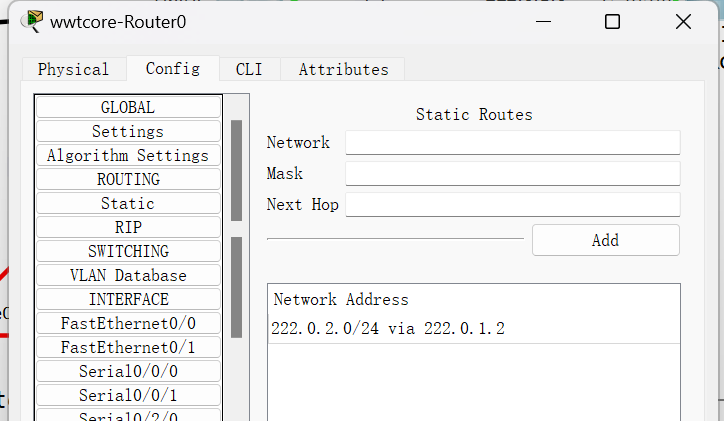
将f0/0，s0/0/0，s0/0/1设置为inside，s0/2/0设置为outside

添加映射



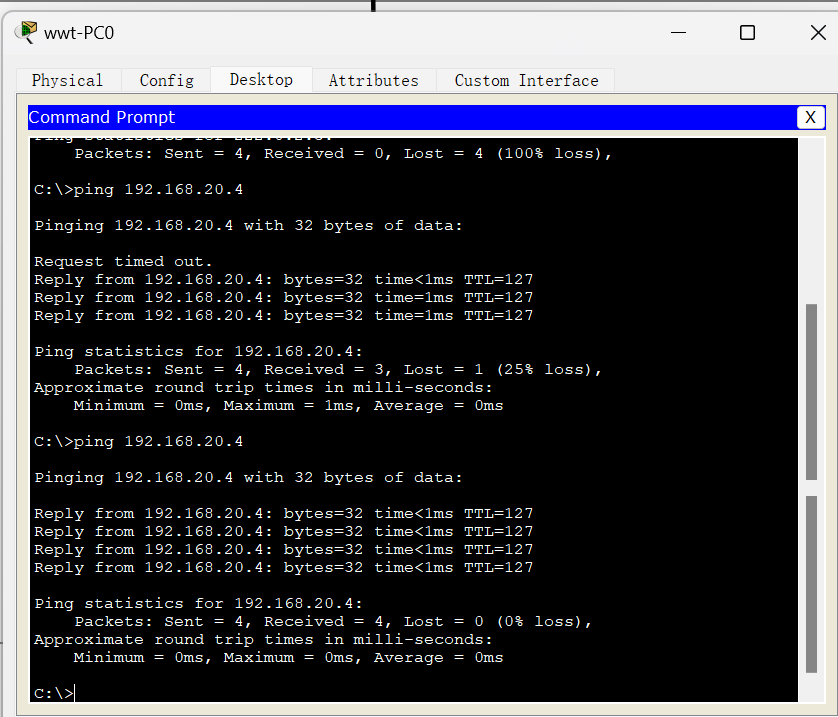
网络集成 5NAT配置

内网所有路由器配置静态路由，network 222.0.2.0，以及相应的下一跳

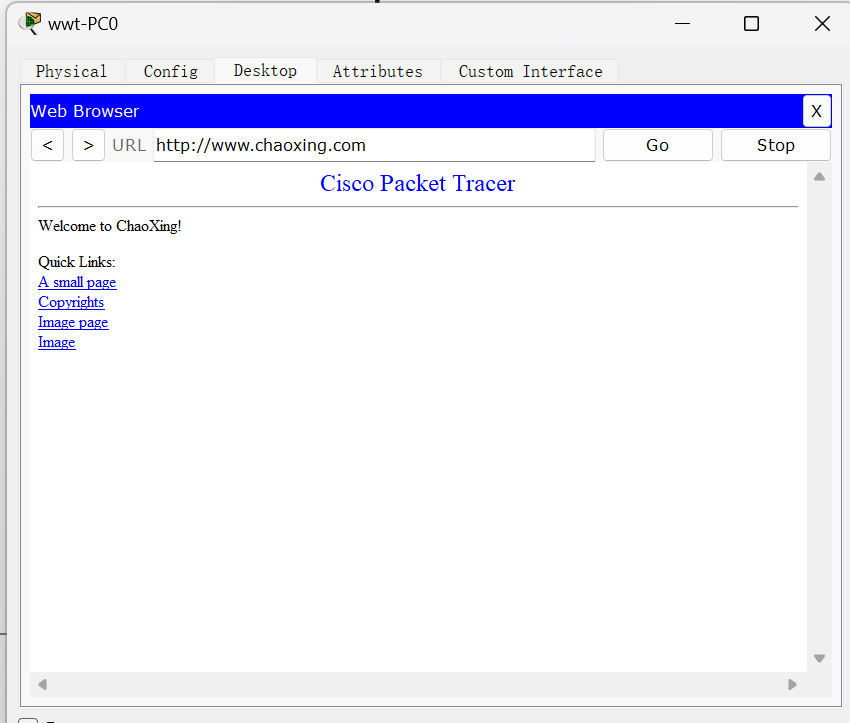
例如

1. **测试**

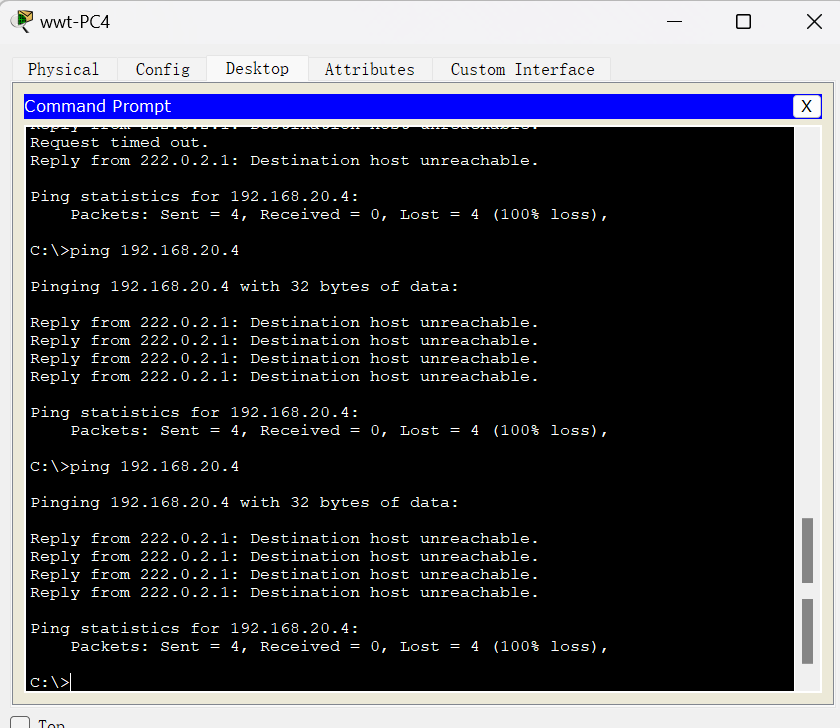
**PC0->PC1**



PC0->外网服务器使用DNS服务



外网PC->内网PC



# 2.3网络编程

## 2.3.1 开发环境及关键配置

利用java实现，安装winpcap和配置Jpcap.jar库文件，从管理员权限运行，利用jpcap的getdevicelist查看网卡列表，并通过opendevice函数绑定网卡，利用getpacket函数抓包，同时为了输出ip对ippacket进行4的锁定，接着对ippacket类型的各种属性进行输出，便能得到版本号、总长度、标志位、片偏移、协议、源地址和目的地址等内容。

通讯程序编程语言：Java 编程软件: idea

抓包工具：wincap

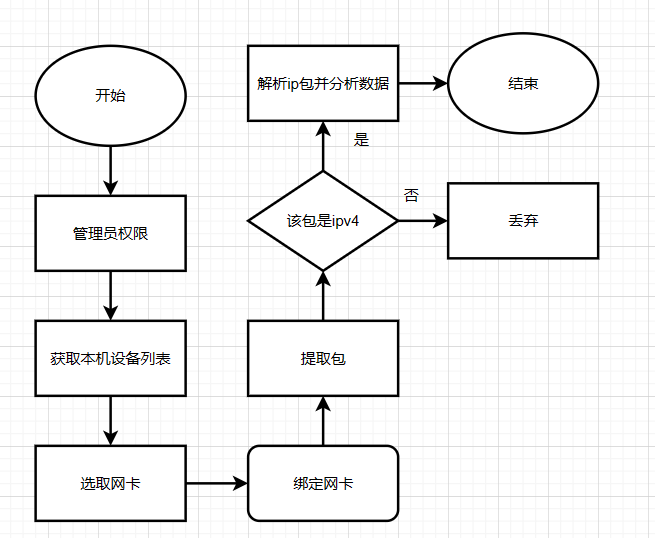
## 2.3.2 设计思路

用JpcapCaptor.getDeviceList()获取网卡列表，返回一个jpcap.NetworkInterface的数组

对网卡调用JpcapCaptor.openDevice()，返回一个JpcapCaptor类的对象

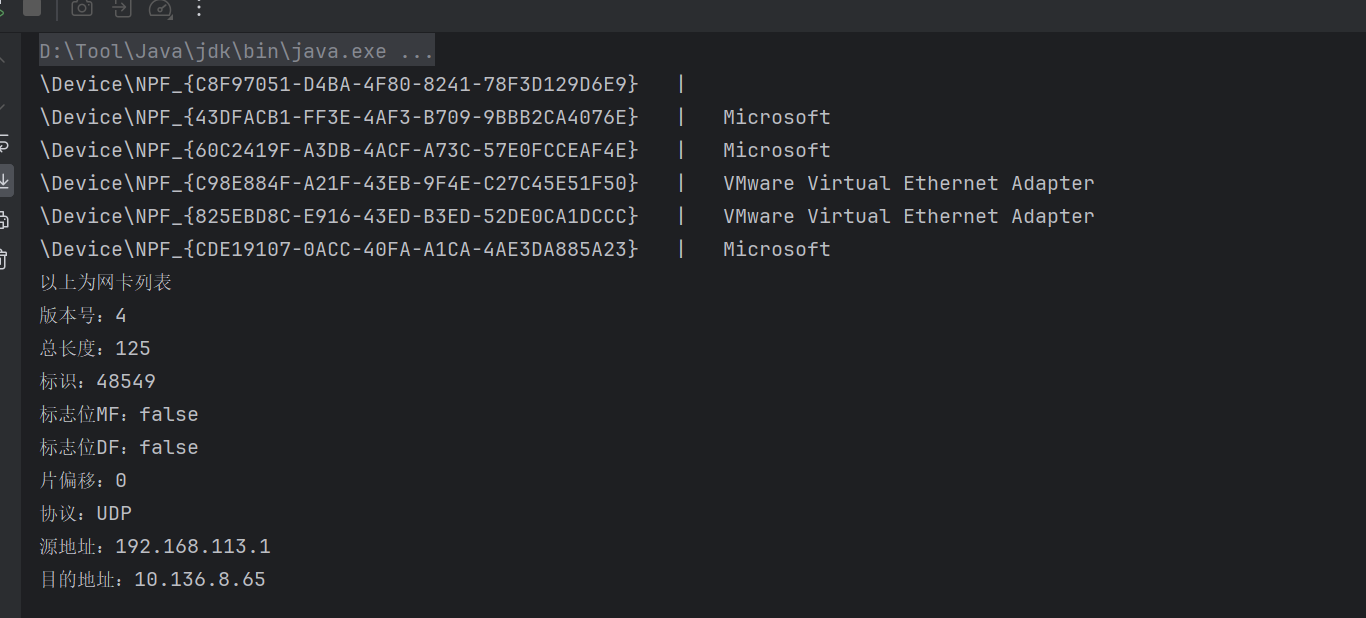
创建Packet类的对象用来接收抓到的包，利用jpcap.getPacket()函数

## 2.3.3工作流程



网络编程 1流程图

## 2.3.4 运行截图及结果说明



网络编程 2运行结果

首先是ip数据报的基本信息：

版本号：占4位，IP协议版本号, IPv4此字段值为4, IPv6此字段值为6

总长度：占16位，以字节为单位, 总长度包含IP的头部和数据部分, IP数据报

最大长度为65535字节, 但是最大不要超过MTU的长度

标识：占16位，唯一标识，一个数据报,可每发送一个数据报, 则该值加1, 如果数据报分片,则每个分片的标识都一样, 各个分片共享一。个标识号

标志位：占3位，最高位为0；最低位MF=1（0），后面还有分片（最后一个报

片）；DF=1（0），不能分片（可以分片）

片偏移：占13位，用以指出该分段的第一个数据字节在原始数据报中的偏移位

置(以8字节为单位),IP分片后每一个分组都具有自己的首部, 而且标志位相同,

但是片偏移值不同, 通过片偏移值接收端可以重新组装IP包

协议类型：占8位，指明IP层上承载的是哪个高级协议, 在分用的过程中, 协议

栈知道该交给上层的哪个协议处理, 如1为ICMP, 2为IGMP, 6为TCP, 17为UDP等.。

源地址：32位，发送方IP地址

目的地址：32位，接受方IP地址

# 实验总结与心得体会

通过实验的设计与执行，我深入理解了网络配置、VLAN设计和网络编程等关键概念。这次课程设计让我意识到理论知识与实践操作之间的重要联系，也增强了我解决实际网络问题的能力。其中很多只根据教程无法完成，还是需要结合所学，根据实际网络结构进行分析和配置。