
华中科技大学计算机学院

《计算机通信与网络》实验报告

班级 计卓 2101 姓名 王彬 学号 U202112071

项目	Socket 编程 (40%)	数据可靠传输协议设计 (20%)	CPT 组网 (20%)	平时成绩 (20%)	总分
得分					

教师评语：

教师签名：

给分日期：

目 录

实验三 基于 CPT 的组网实验	1
1.1 环境	1
1.2 实验要求	1
1.3 基本部分实验步骤说明及结果分析	3
1.4 综合部分实验设计、实验步骤及结果分析	19
1.5 其它需要说明的问题	25
心得体会与建议	26
2.1 心得体会	26
2.2 建议	26

实验三 基于 CPT 的组网实验

1.1 环境

我们在本次 CPT 组网实验使用的环境如下：

硬件配置：

- 1) 处理器：Intel(R) Core(TM) i5-7300HQ CPU @ 2.50GHz 2.50 GHz；
- 2) 机带 RAM：8.00 GB (7.87 GB 可用)；

系统软件组件：

- 1) 系统版本：Windows 10 家庭版；
- 2) 操作系统内部版本：19045.3570；

第三方软件：

- 1) 组网软件版本：Cisco Packet Tracer 6.0；

1.2 实验要求

本实验需要使用 Cisco Packet Tracer 仿真软件完成对应组网实验。组网实验分为两项内容，分别是基本部分的实验和综合部分实验内容。

1.2.1 基本部分

基本部分将使用两张拓扑结构图配合完成实验，如下图 1.1 和图 1.2 所示。

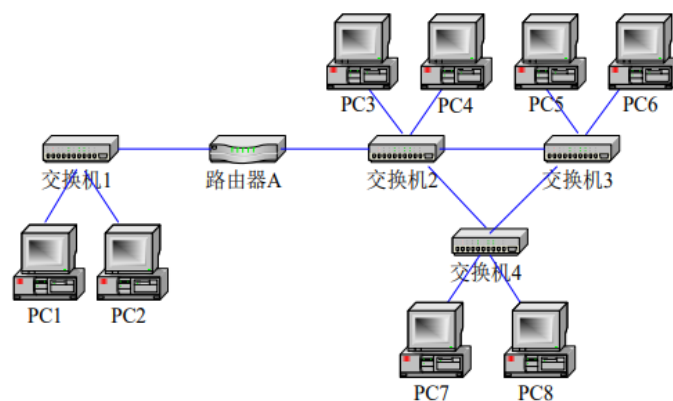


图 1.1 实验一拓扑结构图

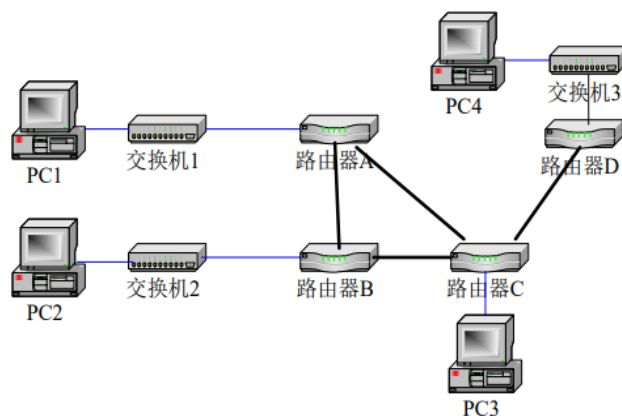


图 1.2 实验二拓扑结构图

第一项实验——IP 地址规划与 VLAN 分配实验：

✧ 使用仿真软件描述网络拓扑图 1.1。

✧ 基本内容 1

- 将 PC1、PC2 设置在同一个网段，子网地址是：192.168.0.0/24;
- 将 PC3~PC8 设置在同一个网段，子网地址是：192.168.1.0/24;
- 配置路由器，使得两个子网的各 PC 机之间可以自由通信。

✧ 基本内容 2

- 将 PC1、PC2 设置在同一个网段，子网地址是：192.168.0.0/24;
- 将 PC3、PC5、PC7 设置在同一个网段，子网地址是：192.168.1.0/24;
- 将 PC4、PC6、PC8 设置在同一个网段，子网地址是：192.168.2.0/24;
- 配置交换机 1、2、3、4，使得 PC1、PC2 属于 VLAN2，PC3、PC5、PC7 属于 VLAN3，PC4、PC6、PC8 属于 VLAN4;
- 测试各 PC 之间的连通性，并结合所学理论知识进行分析;
- 配置路由器，使得拓扑图上的各 PC 机之间可以自由通信，结合所学理论对你的路由器配置过程进行详细说明。

第二项实验——路由配置实验

✧ 使用仿真软件描述网络拓扑图 1.2

✧ 基本内容 1

- 将 PC1 设置在 192.168.1.0/24 网段;
- 将 PC2 设置在 192.168.2.0/24 网段;
- 将 PC3 设置在 192.168.3.0/24 网段;
- 将 PC4 设置在 192.168.4.0/24 网段
- 设置路由器端口的 IP 地址
- 在路由器上配置 RIP 协议，使各 PC 机能互相访问

✧ 基本内容 2

- 将 PC1 设置在 192.168.1.0/24 网段;

-
- 将 PC2 设置在 192.168.2.0/24 网段;
 - 将 PC3 设置在 192.168.3.0/24 网段;
 - 将 PC4 设置在 192.168.4.0/24 网段
 - 设置路由器端口的 IP 地址
 - 在路由器上配置 OSPF 协议, 使各 PC 机能互相访问

✧ 基本内容 3

- 在基本内容 1 或者 2 的基础上, 对路由器 A 进行访问控制配置, 使得 PC1 无法访问其它 PC, 也不能被其它 PC 机访问。
- 在基本内容 1 或者 2 的基础上, 对路由器 A 进行访问控制配置, 使得 PC1 不能访问 PC2, 但能访问其它 PC 机

1.2.2 综合部分

综合部分对应实验需要对某申请了 211.69.4.0/22 地址块的学校进行组网。该学校有 4 个学院, 1 个图书馆, 3 个学生宿舍。同时, 每个学院有 20 台主机, 图书馆有 100 台主机, 每个学生宿舍拥有 200 台主机。

组网需求:

- ✧ 图书馆能够无线上网
- ✧ 学院之间可以相互访问
- ✧ 学生宿舍之间可以相互访问
- ✧ 学院和学生宿舍之间不能相互访问
- ✧ 学院和学生宿舍皆可访问图书馆。

实验任务要求:

- ✧ 完成网络拓扑结构的设计并在仿真软件上进行绘制(要求具有足够但最少的设备, 不需要考虑设备冗余备份的问题)
- ✧ 根据理论课的内容, 对全网的 IP 地址进行合理的分配
- ✧ 在绘制的网络拓扑结构图上对各类设备进行配置, 并测试是否满足组网需求, 如有无法满足之处, 请结合理论给出解释和说明

1.3 基本部分实验步骤说明及结果分析

1.3.1 IP 地址规划与 VLAN 分配实验的步骤及结果分析

对于第一项实验, 我们绘制的 CPT 等效网络拓扑图如图 1.3 所示。其中, 使用路由器 Router_A 连接左右两个子网, 右侧子网用三台交换机与六台主机形成三角互联。基本部分的各主机的 IP 地址均已在实验要求中给出, 故此处不再赘述。

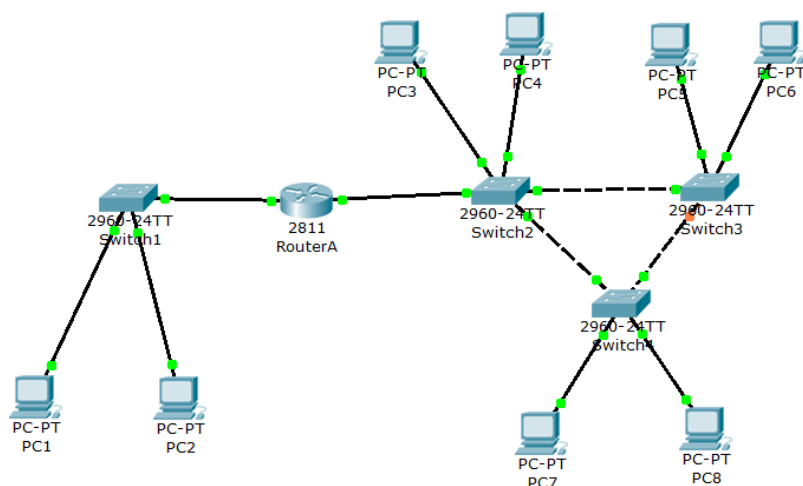


图 1.3 实验一拓扑结构实现

(一)基本内容 1

1) 配置各个主机的 IP 地址。

对于 PC1、PC2, 它们分配至同一个网段 192.168.0.0/24, 因此它们的网关均设为 192.168.0.1, 之后我们设置其以太网接口的 IP 逻辑地址和子网掩码。我们以 PC1 为例, 我们对其端口适配器分配 IP 地址 192.168.0.2, 子网掩码设置为 255.255.255.0, 其快速以太网接口 FastEthernet0 的配置如图 1.4 所示。对于 PC2, 我们分配的 IP 地址为 192.168.0.3。

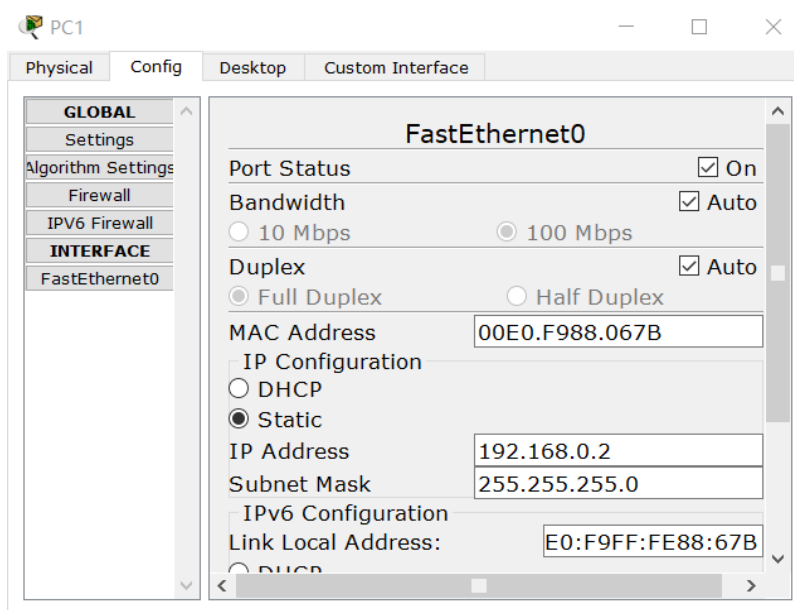


图 1.4 主机快速以太网端口配置

我们随后对本任务中的其它主机的网关、适配器 IP 地址和子网掩码进行类似的配置。配置的方式类似, 即处于同一网段的分配不同的 IP 地址; 同时, 子网 1 的网关地址均设置为 192.168.0.1, 子网 2 的网关地址均设置为 192.168.1.1, 并且由于本实验中 IP 地址均取 24 位有效地址, 因而子网掩码均为 255.255.255.0。

2) 配置路由器

我们对路由器的快速以太网接口进行配置。对于接口配置的 FastEthernet0/0，该以太网接口与交换机 1 直接相连，而交换机 1 与 PC1 和 PC2 相连，是通向子网 192.168.0.0/24 的接口。同时，IP 地址 192.168.0.0 为保留字，因此该路由器对应适配器 IP 为 192.168.0.1。配置的表示如图 1.5 所示。

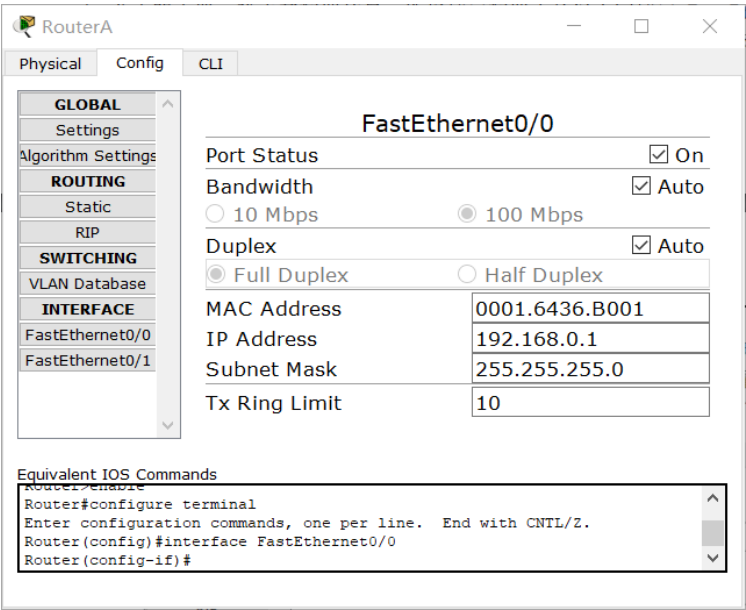


图 1.5 路由器 IP 及子网掩码配置

综上所述，我们对主要的接口分配的 IP 地址和子网掩码如表 1.1 所示。由于本实验中子网有效位均为 24 位，故子网掩码均为 255.255.255.0。对于网关，子网 1 所属主机均设置为 192.168.0.1，子网 2 所属主机均设置为 192.168.1.1。

主机名	接口名	IP 地址	子网掩码
RouterA	Fa0/0	192.168.0.1	255.255.255.0
	Fa0/1	192.168.1.1	255.255.255.0
PC1	Fa0	192.168.0.2	255.255.255.0
PC2	Fa0	192.168.0.3	255.255.255.0
PC3	Fa0	192.168.1.2	255.255.255.0
PC4	Fa0	192.168.1.3	255.255.255.0
PC5	Fa0	192.168.1.4	255.255.255.0
PC6	Fa0	192.168.1.5	255.255.255.0
PC7	Fa0	192.168.1.6	255.255.255.0
PC8	Fa0	192.168.1.7	255.255.255.0

表 1.1 各主机及路由器接口 IP 地址和子网掩码设置

3) 网段内部连通性测试

在 Cisco Packet Tracer 软件中，测试主机连通性通常有两种方法。其一是进入主机桌面中

的命令提示符，使用 Ping 指令尝试和其它 IP 地址的主机建立通信关系，期待 ICMP 报文回复；其二是使用实时仿真(Simulation)界面，对实际报文的传递进行模拟。我们下面对不同网段，以及网段之间进行连通性测试。

a) 网段 1 连通性测试

我们进入 PC1 的命令提示符界面，如图 1.6 所示，使用命令行从 PC1 对 PC2 进行 Ping 操作。如下图所示，Ping 操作收到了 4 份长度为 32 字节的反馈报文，发送 4 份，接收 4 份，丢失率为 0%，这说明 PC1 与 PC2 连通。之后，我们还从 PC2 对 PC1 进行 Ping 操作，结果相同，这表明该网段(192.168.0.0/24)是联通的。

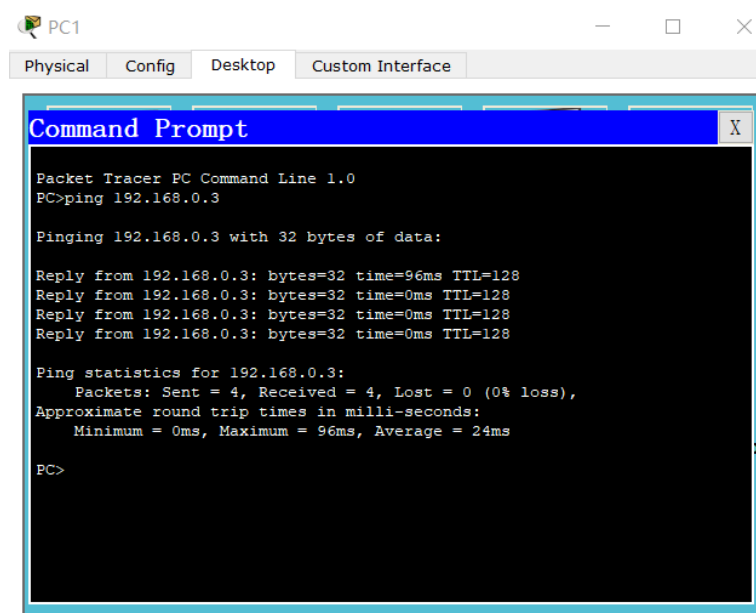


图 1.6 左侧网段连通测试

b) 网段 2 连通性测试

我们对网段 2 的六台主机进行仿真模拟，最后得到的 PDU List Window 如图 1.7 所示，可见网段 2 中所有的主机均能正常通信。

Fire	Last Status	Source	Destination	Type	Color	Time (sec)	Periodic	Num	Edit	Delete
	Successful	PC3	PC4	ICMP		0.000	N	0	(edit)	(delete)
	Successful	PC3	PC5	ICMP		0.000	N	1	(edit)	(delete)
	Successful	PC3	PC8	ICMP		0.000	N	2	(edit)	(delete)
	Successful	PC7	PC5	ICMP		0.000	N	3	(edit)	(delete)
	Successful	PC6	PC4	ICMP		0.000	N	4	(edit)	(delete)

图 1.7 右侧网段内部互联测试

c) 网段 1、2 互联测试

继续对两个网段进行互联测试。我们打开模拟仿真界面，分别从 PC2 向 PC3、PC1 向 PC6、PC1 向 PC7 发送 ICMP 报文，发现可以正常连通，如图 1.8 所示。

Fire	Last Status	Source	Destination	Type	Color	Time (sec)	Periodic	Num	Edit	Delete
	Successful	PC2	PC3	ICMP		0.000	N	0	(edit)	(delete)
	Successful	PC1	PC6	ICMP		0.000	N	1	(edit)	(delete)
	Successful	PC1	PC7	ICMP		0.000	N	2	(edit)	(delete)

图 1.8 两个网段互相连通测试

这说明两个网段之间也可以进行自由通信。

综上所述，我们实现了两个网段内部和网段之间的通信，符合实验要求。每个网段内部通过交换机连接，交换机可以将报文转发至目标主机处，无需经过路由器；而在两个网段之间的通信，则需要路由器将报文转发至目标网段中，再由目标网段中的交换机转发至目的 PC 机。

(二)基本内容 2

1) 修改 PC 机的 IP 配置

我们在基本内容 1 的基础上,将本内容所要求的 PC4、PC6 和 PC8 编入子网 192.168.2.0/24, 因此我们需要配置它们的网关和 IP 地址。如图 1.9(左)所示,PC6 的网关被重写为 192.168.2.1, 而如图 1.9 (右), 其快速以太网接口 IP 地址应当修改为当前新网段所分配的 IP 地址。

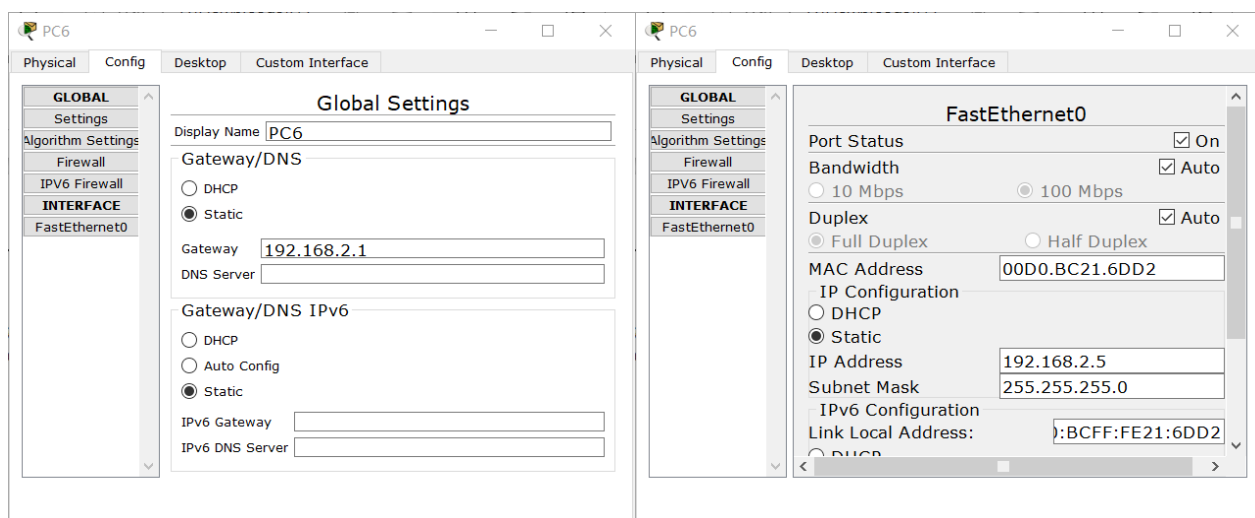


图 1.9 主机配置修改

2) 配置交换机 VLAN

我们对交换机的 VLAN 设置进行更改。由于交换机 1 连接 PC1、PC2，这两个主机都属于 VLAN2；而交换机 2、3、4 所连接的主机既有属于 VLAN3 的，又有属于 VLAN4 的，因此交换机 1 的 VLAN 数据库中仅需要包含 VLAN2，而交换机 2、3、4 的 VLAN 数据库则需要同时包含 VLAN3 和 VLAN4。例如，交换机 2 的 VLAN 配置界面如图 1.10 所示。

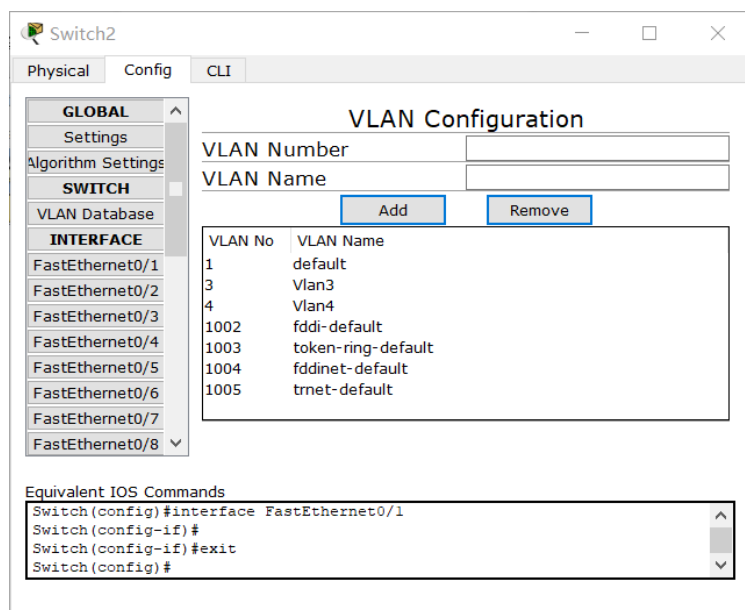


图 1.10 交换机 VLAN 配置

在添加 VLAN 数据库之后，我们接着对交换器的不同端口做链路划分。以太网交换机的链路有两种类型，一种为主干链路（Trunk），另一种为专用链路（Access），主干链路可以将所有通过的数据包均尽可能地传输，而专用链路只能通过具有相应 VLAN 标识的数据包。因此，如图 1.11 所示，交换机 2 的快速以太网端口 Fa0/1 与 PC3 相连，而 PC3 所分配的 VLAN 号为 3，故端口 Fa0/1 应选用专用链路，并且设置为 VLAN3 专用链路。同理，如图 1.12，端口 Fa0/2 与 PC4 相连，故而选择 VLAN4 专用链路。对于端口 Fa0/3~5，它们或者与其它交换机相连，或者与路由器相连，因此使用主干链路。

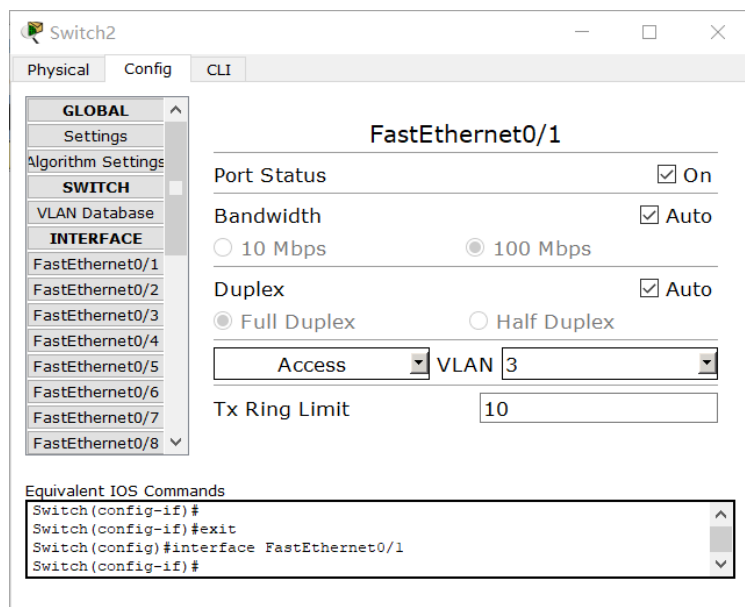


图 1.11 交换机 2 端口 1 VLAN 配置

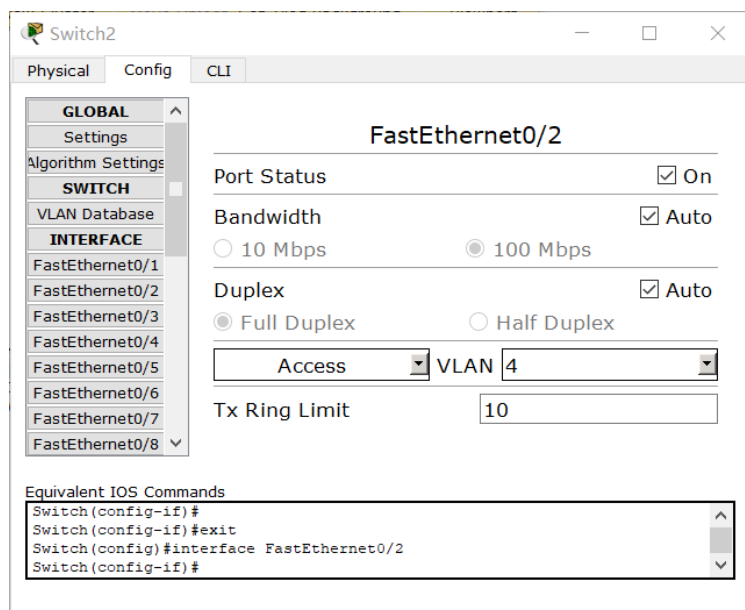


图 1.12 交换机 2 端口 2 VLAN 配置

3) 测试访问

在划分完 VLAN 后,我们分别测试同一 VLAN 内部和不同 VLAN 之间的连通性。如图 1.13 所示,首先通过 PC2 (192.168.0.2) 尝试发送 ICMP 报文至 PC1 (192.168.0.3),发现可通,因此同一网段内通信可以互达;而 PC2 尝试与 PC3 (192.168.1.2) 建立联系则不可行,这是因为它们不属于同一个网段,如果不经路由配置是不可通信的。

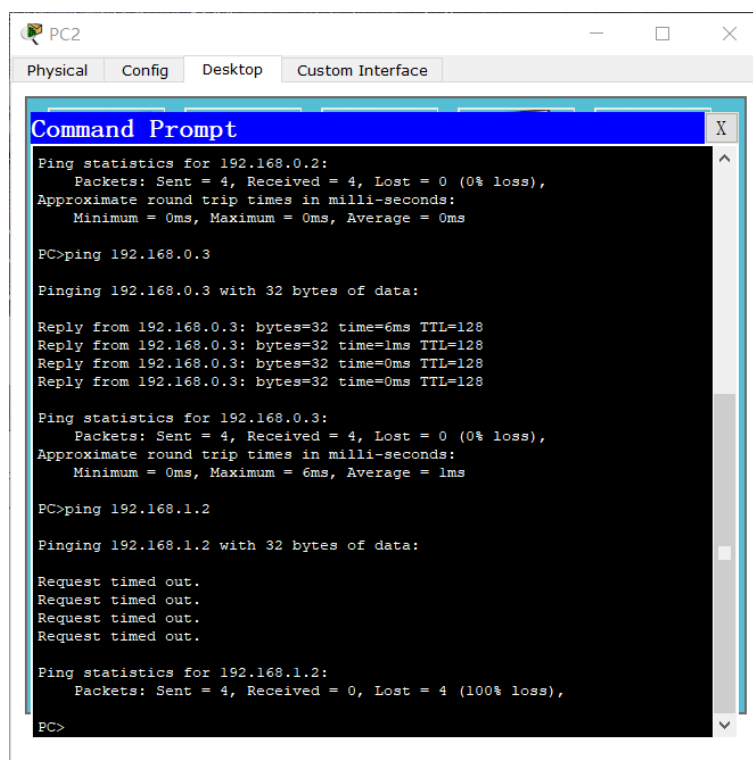


图 1.13 测试访问（不可通信）

为了减少偶然性，我们还从 PC3 出发尝试 Ping 子网 3 及子网 4 的其它主机。如图 1.14 所示，PC3（VLAN3）分别与 PC7（VLAN3）、PC8（VLAN4）通信，结果是 PC3 与 PC7 互通，而与 PC8 不互通。这两组实验说明，我们划分出的三个网段之间无法互通，但是网段内部却可以通信。

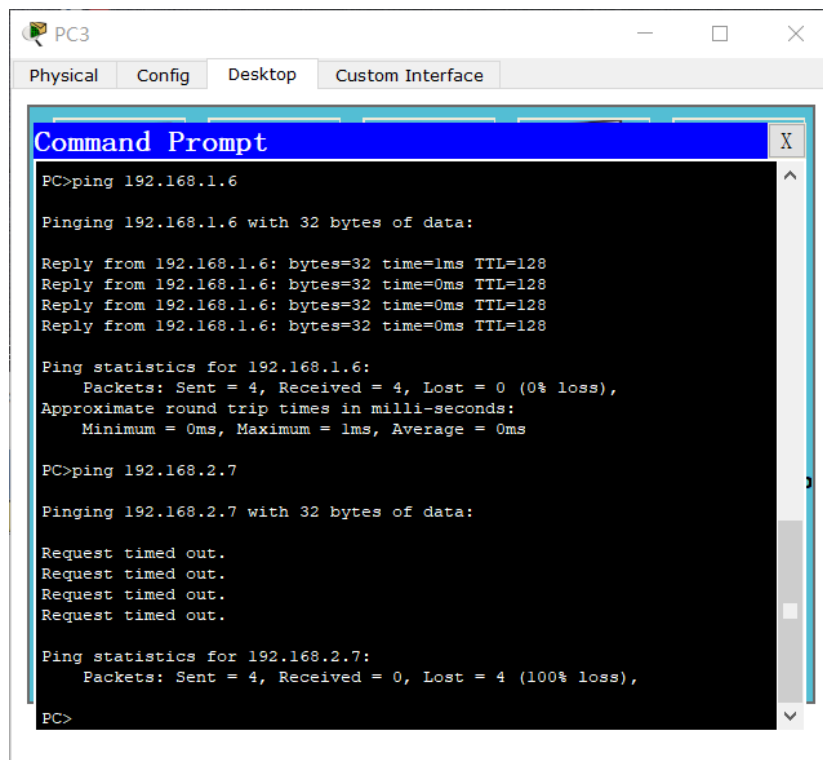


图 1.14 网段内部可通但网段间不可通

4) 路由器配置

网段之间无法互相通信，是因为路由器的接口未进行配置，仅依靠交换机无法转发不同网段的报文。我们对路由器 A 的端口进行配置。如图 1.15，使用命令行设置路由器 A 的不同接口；其中，由于路由器 A 仅有两个端口 Fa0/0 和 Fa0/1，因此需要对与 PC3~8 相连的接口 Fa0/1 下拉子接口 Fa0/1.1 和 Fa0/1.2。我们对 Fa0/0.1 配置其 IP 地址为 192.168.0.1/24，对 Fa0/1.1 配置为 192.168.1.1/24，而对 Fa0/1.2 配置为 192.168.2.1/24。

这些配置无法在图形化界面中进行，需要在命令行中 Router(config)下对路由器实现配置。其配置代码如下：

```
Router(config)# int fa0/1.1
Router(config-subif)# encap dot1q 3
Router(config-subif)# ip addr 192.168.1.1 255.255.255.0
Router(config-subif)# exit
```

首先我们创建子端口 Fa0/1.1，其次我们通过指令“encap dot1q 3”为该端口指定 3 号 VLAN

链路，再设置其 IP 地址为 192.168.1.1/24。最终设置结果如图 1.15 所示。

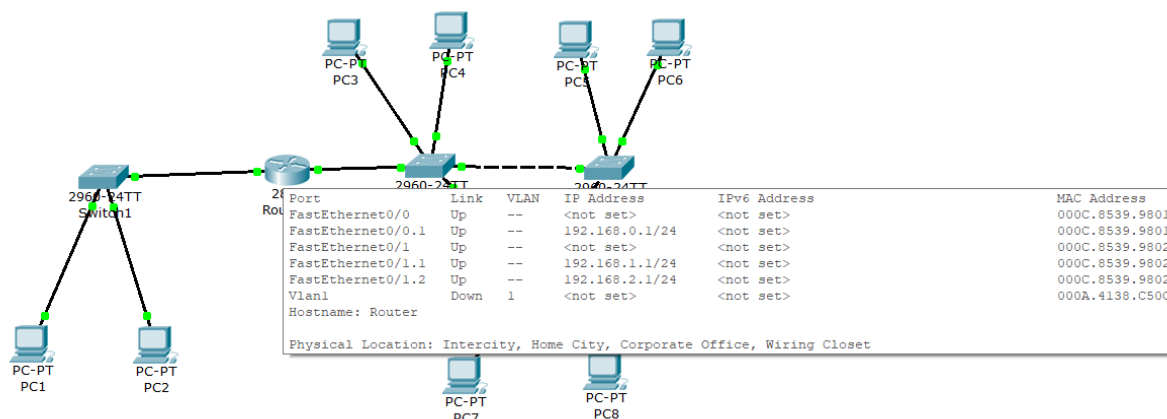


图 1.15 子端口配置状态

我们使用 RIP 路由协议对路由器进行配置。如图 1.16 所示，对路由器 A 连通的三个子网网段加入 RIP 数据库，同时将三个 VLAN 加入 VLAN 数据库中。

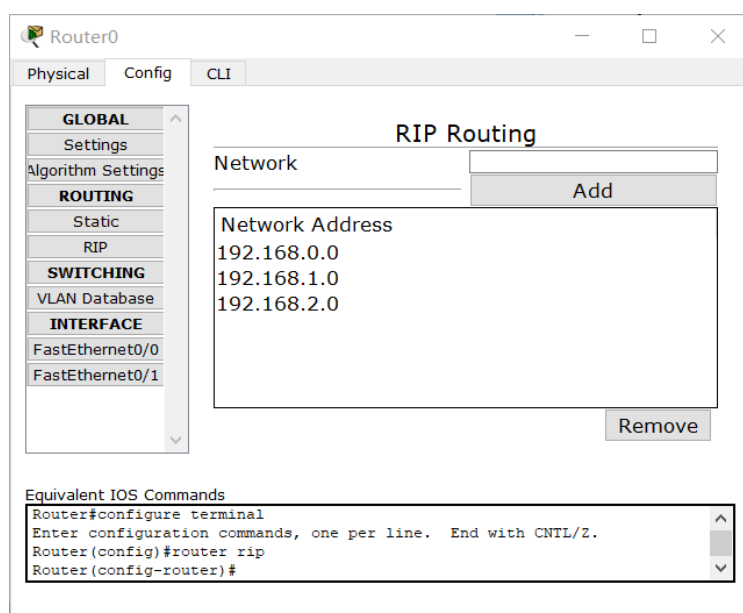


图 1.16 RIP 协议数据库写入

5) 连通测试

我们打开模拟界面，对网络中的任意两台主机进行互联测试。PC1、PC2 属于 VLAN2，PC3、PC5、PC7 属于 VLAN3，PC4、PC6、PC8 属于 VLAN4。我们对网段内部和网段间（除了 PC3->PC5 和 PC2->PC1 之外的全部）进行连通测试，测试结果如图 1.17 所示，可见在配置完路由器之后，任意两台网络中的主机均能相互连通。

Fire	Last Status	Source	Destination	Type	Color	Time (sec)	Periodic	Num	Edit	Delete
	Successful	PC3	PC4	ICMP		0.000	N	0	(edit)	(delete)
	Successful	PC3	PC6	ICMP		0.000	N	1	(edit)	(delete)
	Successful	PC1	PC3	ICMP		0.000	N	2	(edit)	(delete)
	Successful	PC3	PC5	ICMP		0.000	N	3	(edit)	(delete)
	Successful	PC8	PC3	ICMP		0.000	N	4	(edit)	(delete)
	Successful	PC2	PC1	ICMP		0.000	N	5	(edit)	(delete)
	Successful	PC2	PC5	ICMP		0.000	N	6	(edit)	(delete)

图 1.17 互联测试

6) 结果分析

我们对测试结果进行分析。原本未对路由器进行设置时，网段 192.168.0.1/24 与另外两个网段无法相互通信，这是因为路由器端口未分配 IP 地址，路由器收到该网段报文时不清楚应当转发的位置，因此路由器会忽略接收到的报文。

在交换机进行 VLAN 划分后，每个 VLAN 形成一个逻辑的虚拟局域网。在配置路由 VLAN 后，通过路由的交换功能实现不同 VLAN 的互联。例如，PC1 向 PC8 传递报文，则通过交换机 1 转发至路由器 A，路由器 A 再向目标子网转发。

我们必须在路由器上对 VLAN 进行配置，一个在某虚拟局域网下的主机才能够通过路由器向其它虚拟局域网的主机发送信息，这样的数据包才会被路由器识别和转发。如果仅仅通过交换机，即使两个 VLAN 直接相连，一个网段的数据包也会因为无法通过另一 VLAN 的专用链路，而使得数据包无法被另一个网段的主机接收。

1.3.2 路由配置实验的步骤及结果分析

对于第二项实验，即路由配置实验，我们 CPT 中绘制的等效拓扑图如图 1.18 所示。该拓扑结构包括 4 台 PC 机，3 台交换机和 4 台路由器，它们通过不同类型的链路相连。

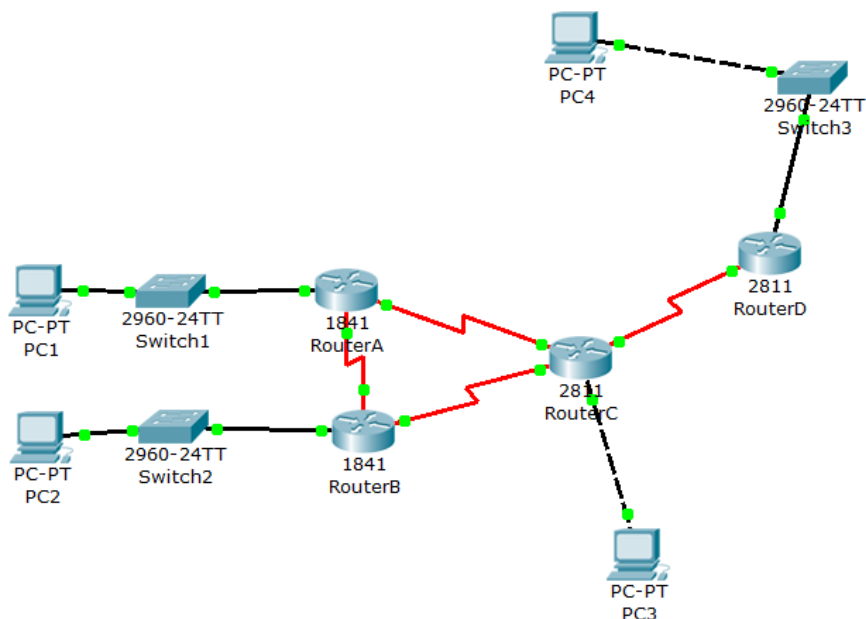


图 1.18 实验二拓扑结构实现

(一)基本内容 1

1) 配置网络边缘

我们根据实验要求，分别将 PC1~4 设置在四个不同的网段。主机配置部分因在 1.2 小节有过充分的说明，故此处不再赘述。如图 1.18，我们的拓扑图中使用了四个路由器，其中路由器 A 与网段 192.168.1.0 相连，路由器 B 与网段 192.168.2.0 相连，路由器 C 与网段 192.168.3.0 相连，而路由器 D 则与网段 192.168.4.0 相连。我们依据不同的网段为各个主机的端口分配 IP 地址，同时，对于路由器直接通向网络边缘的端口，我们也为其分配的 IP 地址：路由器 A 的 Fa0/0 端口分配 192.168.1.1，路由器 B 的 Fa0/0 端口分配 192.168.2.1，路由器 C 的 Fa0/0 端口分配 192.168.3.1，路由器 D 的 Fa0/0 端口分配 192.168.4.1。图 1.19 为路由器 C 与网络边缘相连的接口参数。

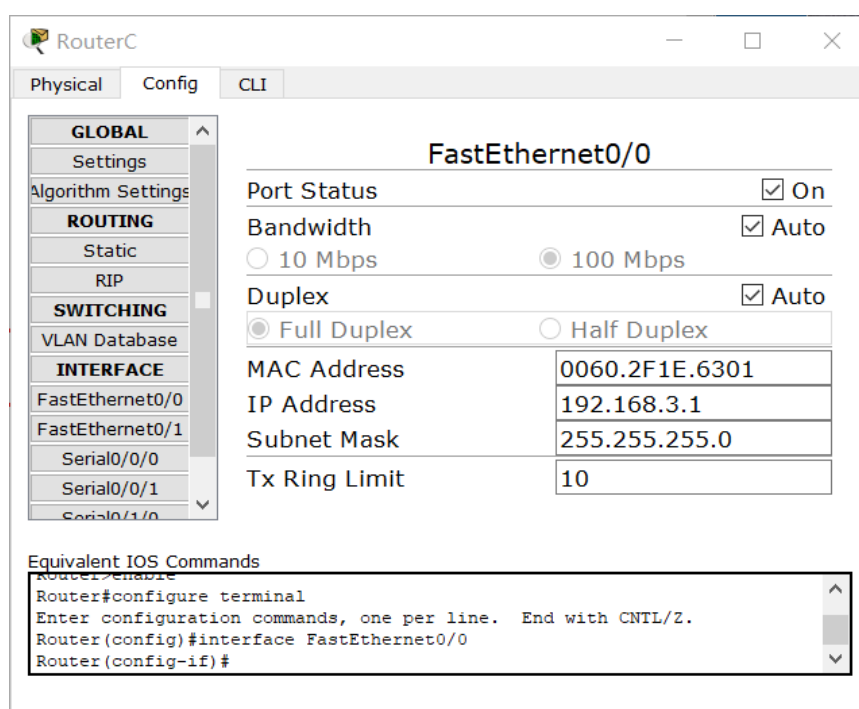


图 1.19 网络边缘路由器配置

2) 配置网络核心，及其路由器端口

首先，在配置网络核心路由器之前，我们需要为路由器添加物理设备。这是因为路由器自身只配备与以太网连接的端口，我们还需要增加路由器之间互联的串口。其操作方式为，注意到路由器不能热拔，我们需要先关闭路由器电源，再在路由器物理界面中添加模块至其空插中。如图 1.20 所示，我们为路由器 A 添加了一个“HWIC-2T”模块，这将为路由器增加两个串口。



图 1.20 网络核心路由器物理结构

增加物理插件后，我们的路由器会增加两个串口 Serial0/1/0、Serial0/1/1。串口可用于路由器之间的互联，而路由器之间互联的链路应当选择 Serial DTE。我们下面配置路由器之间互联的端口。

我们为路由器之间互联的部分分配网段。其中，路由器 A 与路由器 C 连接部分分配 192.168.5.0 网段，路由器 A 与路由器 B 连接部分分配 192.168.6.0 网段，路由器 B 与路由器 C 连接部分分配 192.168.7.0 网段，而路由器 C 与路由器 D 相连的部分则分配 192.168.8.0 网段。网段的分配示意拓扑图详见图 1.21。

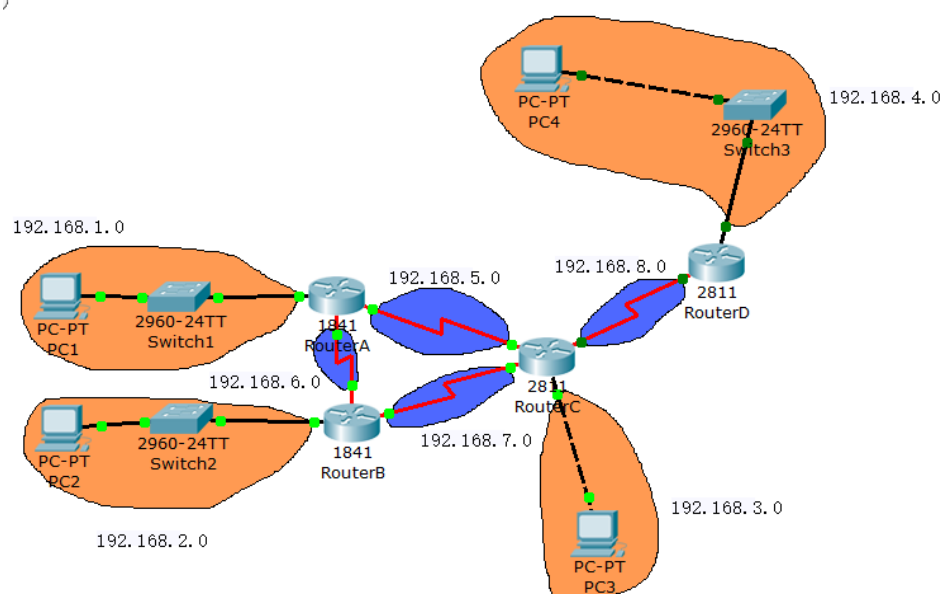


图 1.21 网段分配及 IP 地址示意

在配置串口线路时，我们还需要确保同一链路所连接的两个端口的时钟频率相同。因此，如图 1.22 所示，我们对所有的串口（Serial0/x/x）的时钟频率均设置为 2000000。

Serial0/0/0

Port Status	<input checked="" type="checkbox"/> On
Clock Rate	2000000 ▾
Duplex	<input checked="" type="radio"/> Full Duplex

图 1.22 链路时钟频率

3) 通过 RIP 协议令各 PC 机互相连通

RIP 协议是基于距离矢量算法的路由协议，可以在自治系统内维护路由器自身到每一个目的网络地址的距离记录。该协议仅和相邻路由器交换信息，因此其 RIP 选路表中应当维护其每个端口 IP 所在网段。例如，对于路由器 C，其 RIP 选路表如图 1.23 所示。由于此路由器与路由器 A、路由器 B、路由器 D 和网段 3 相邻，因此它直接相邻的网段有 4 个，分别是 192.168.3.0、192.168.5.0、192.168.7.0、192.168.8.0，这四个网段都应在选路表中写入。

RIP Routing

Network	
	Add

Network Address

192.168.3.0

192.168.5.0

192.168.7.0

192.168.8.0

图 1.23 RIP 选路写入

类似地，我们配置其它路由节点的 RIP 选路表。在完成主机网关及 IP 配置后、配置路由器节点后，我们完成了本网络的基本配置。下面我们对网络进行连通测试。

4) 连通测试

如图 1.24 所示，我们对任意两台主机进行通信，包括 PC2 和 PC3、PC4 和 PC3、PC4 和 PC1、PC1 和 PC2；同时，我们还对路由器进行了连通测试，发现路由器和主机之间也可达。通过这次测试，我们认为该组网下任意两台主机均可进行互相通信，符合我们的实验预期。













Fire	Last Status	Source	Destination	Type	Color	Time (sec)	Periodic	Num	Edit	Delete
	Successful	PC3	RouterC	ICMP		0.000	N	0	(edit)	(delete)
	Successful	RouterB	PC2	ICMP		0.000	N	1	(edit)	(delete)
	Successful	PC2	PC3	ICMP		0.000	N	2	(edit)	(delete)
	Successful	PC4	PC3	ICMP		0.000	N	3	(edit)	(delete)
	Successful	PC4	PC1	ICMP		0.000	N	4	(edit)	(delete)
	Successful	PC1	PC2	ICMP		0.000	N	5	(edit)	(delete)

图 1.24 RIP 协议连通性测试

(二)基本内容 2

1) 主机及路由器的网关、IP 地址配置等均和基本内容 1 一致，因而此处不再赘述。

2) OSPF 协议配置

OSPF 协议是另一种链路状态路由协议，我们下面对其进行配置。我们使用命令行对路由器进行 OSPF 协议配置。例如，对于路由器 A，如图 1.25 所示，首先删除其 RIP 协议记录，再对其 OSPF 采取标号为 1 的协议写入。路由器 A 具有三个端口，分别与网段 192.168.1.0、192.168.5.0、192.168.6.0 相连，因此在 OSPF 记录中写入这三个网段，其子网掩码的反码均为 0.0.0.255。

其所使用到的指令如下，最后一行指令为配置生效指令。

```
Router(config)# router ospf 1
Router(config-router)# network 192.168.1.0 0.0.0.255 area 0
Router(config-router)# network 192.168.5.0 0.0.0.255 area 0
Router(config-router)# network 192.168.6.0 0.0.0.255 area 0
Router(config-router)# end

Router# copy run startup
```

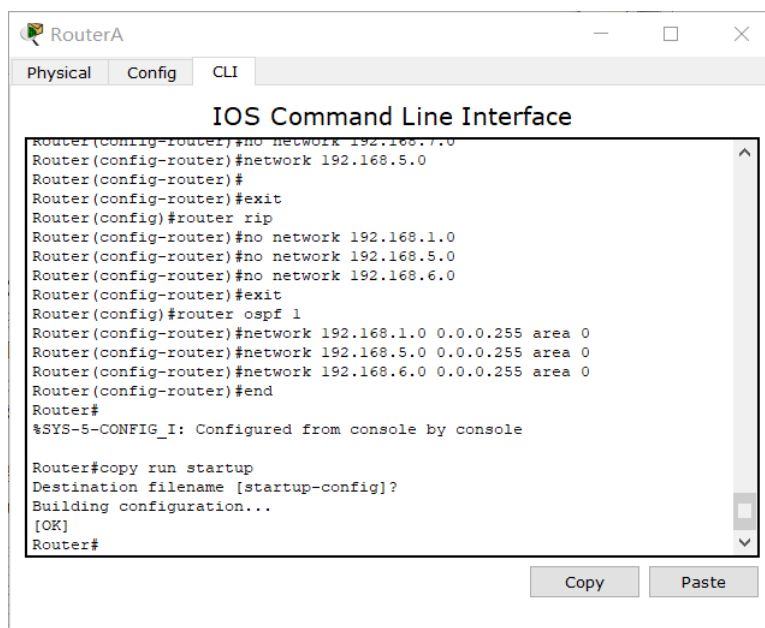


图 1.25 OSPF 命令行配置

在对其余路由器进行类似操作后，我们的路由器配置即可完成。我们下面对本实验内容所完成的网络进行测试，验证其连通性。

3) 连通测试

如图 1.26 所示，我们任意选取了不同网段的一对节点进行通信测试。结果任意两台主机均可实现通信，这说明我们的 OSPF 协议配置成功。

Fire	Last Status	Source	Destination	Type	Color	Time (sec)	Periodic	Num	Edit	Delete
	Successful	PC2	PC3	ICMP		0.000	N	0	(edit)	(delete)
	Successful	PC1	PC4	ICMP		0.000	N	1	(edit)	(delete)
	Successful	PC1	PC2	ICMP		0.000	N	2	(edit)	(delete)
	Successful	PC3	PC4	ICMP		0.000	N	3	(edit)	(delete)
	Successful	PC2	PC4	ICMP		0.000	N	4	(edit)	(delete)

图 1.26 OSPF 连通性测试

(三)基本内容 3

- 1) 在基本内容 1 的基础上，对路由器 A 进行访问控制配置，使得 PC1 无法访问其它 PC 机，且无法被其它 PC 机访问；

这需要使用更改相关路由器的访问控制列表。注意到 PC1 所属网段为 192.168.1.0/24，其仅与路由器 A 相连，因此只需要对边缘路由器 A 更改即可。如图 1.27 所示，我们对路由器 A 的快速以太网接口 FastEthernet0/0 进行配置，对其 Access-List 建立以 25 为编号，对网段 192.168.1.0/24 进行屏蔽操作，再使得该访问控制列表项对接口 Fa0/0 生效。

这样可以使得 PC1 既无法访问其它 PC 机，又无法被其它 PC 机访问，因为该网段被其边缘路由器所屏蔽。

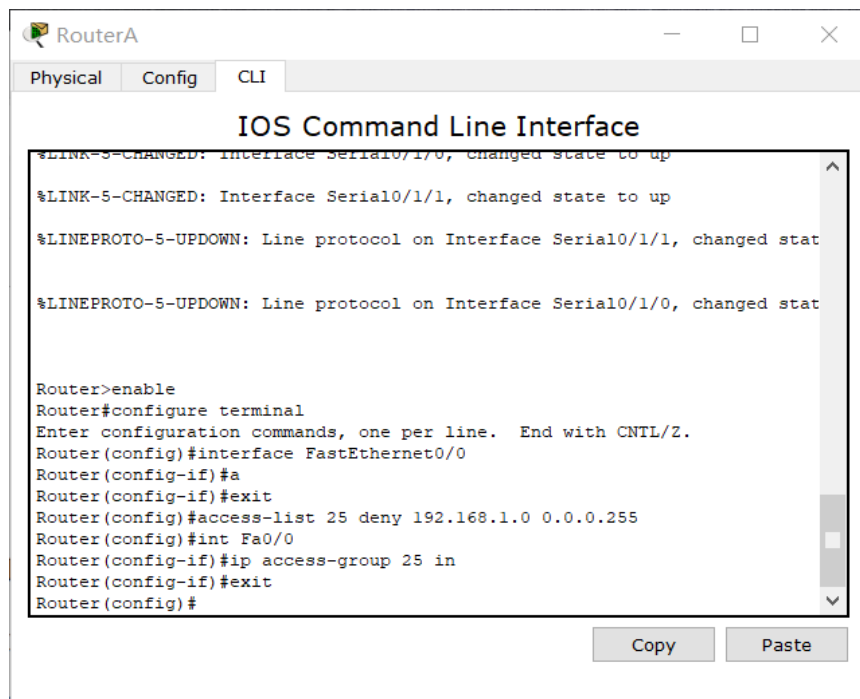


图 1.27 ACL 配置阻止 PC1 的访问与被访问

我们对于该实验进行连通测试。如图 1.28 所示，我们分别以 PC1 为信息源或数据包传输对象，与路由器 A、PC4、PC3 尝试建立连接，却均以失败告终。而网络中的其它部分不受其影响，仍然能够相互通信，例如 PC4 至路由器 C、PC2 和 PC3 仍然通信成功。这两者说明我们达到的实验效果符合预期。













Fire	Last Status	Source	Destination	Type	Color	Time (sec)	Periodic	Num	Edit	Delete
	Failed	PC1	RouterA	ICMP		0.000	N	0	(edit)	(delete)
	Failed	PC1	PC4	ICMP		0.000	N	1	(edit)	(delete)
	Failed	PC1	PC3	ICMP		0.000	N	2	(edit)	(delete)
	Failed	PC3	PC1	ICMP		0.000	N	3	(edit)	(delete)
	Successful	PC4	RouterC	ICMP		0.000	N	4	(edit)	(delete)
	Successful	PC2	PC3	ICMP		0.000	N	5	(edit)	(delete)

图 1.28 PC1 无法与其它主机实现通信，但其余主机之间则可以

- 2) 在基本内容 1 的基础上，对路由器 A 进行访问控制配置，使得 PC1 不能访问 PC2，但能够与其它 PC 机进行访问；

我们继续对路由器 A 进行 ACL 的更改。命令行中，ACL 的更改命令如下所示。第一行为在全局配置模式下，创建 ACL 表项；而第二行是在对应接口配置中，对于该接口生效某一访问控制配置。

```
Router (config)# access-list access-list-number {permit | deny } {test-conditions}
```

```
Router (config-if)# {protocol} access-group access-list-number {in | out }
```

因此，如若要使 PC1 不能访问 PC2，则应使用 deny 指令，使目标节点网段无法于本网段获得通信，而随后还要再该表项中使用 permit 指令，使得其余 PC 机仍然能够访问，相关指令如图 1.29 所示。

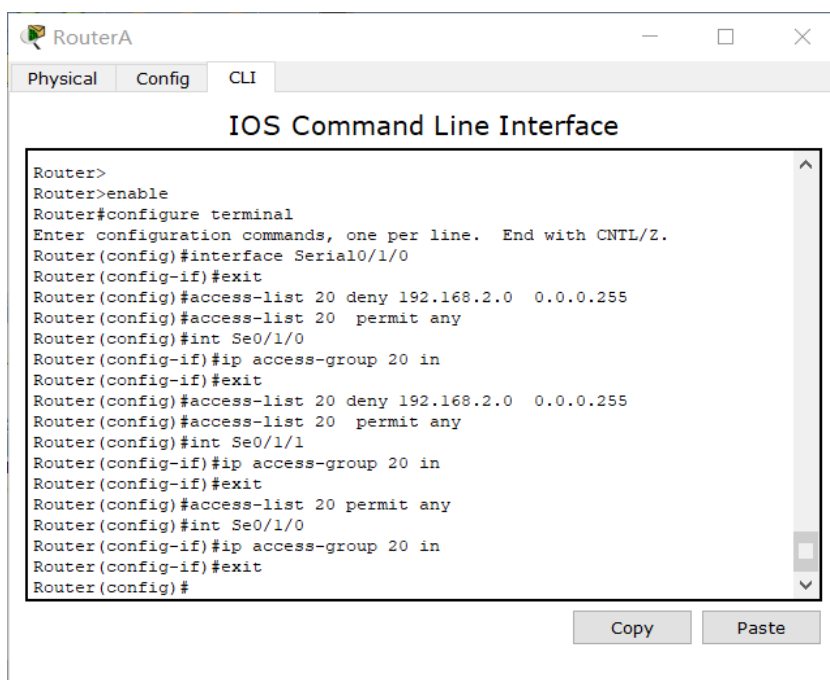


图 1.29 对于本实验内容配置 ACL 表

我们对本实验的测试结果如图 1.30 所示。对于 PC1 和 PC2 无法相互通信，而对于其它主机之间则可以通信；特别地，其它主机也可以访问 PC1（除 PC2 之外），而 PC1 也可访问除了 PC2 外的其它主机。这说明我们的实验结果是正确的。











Fire	Last Status	Source	Destination	Type	Color	Time (sec)	Periodic	Num	Edit	Delete
	Failed	PC1	PC2	ICMP		0.000	N	0	(edit)	(delete)
	Successful	PC1	PC3	ICMP		0.000	N	1	(edit)	(delete)
	Failed	PC2	PC1	ICMP		0.000	N	2	(edit)	(delete)
	Successful	PC1	PC4	ICMP		0.000	N	3	(edit)	(delete)
	Successful	PC4	PC1	ICMP		0.000	N	4	(edit)	(delete)

图 1.30 实验结果

(四)结果分析

我们分别通过 RIP 协议和 OSPF 协议对不同网段进行了组网实验。在同一个自治系统中，RIP 协议通过距离向量分布式地对路由选路进行计算，而 OSPF 则在链路状态产生变化时对路由地址及其它信息进行广播，并依据 Dijkstra 算法对最短路径进行选路计算。我们在 OSPF 配置时，需要键入“copy run startup”指令，这使得链路状态改变，从而使得路由器发送相应 ICMP 数据包并计算更改后的最短路由。

访问控制列表 ACL（Access Control List）是对路由器的逻辑控制指令，对路由器相关端口的数据包进行过滤操作。列表表项有允许 permit 指令，和阻止 deny 指令，使用这两者的结合可以实现数据包的选通。例如，在我们实现的实验中，需要结合 deny 指令（即阻止 PC2 与 PC1 所处网段的连通），同时增加 permit 指令（其余数据包仍能进入），完成对数据包的选通。

1.4 综合部分实验设计、实验步骤及结果分析

1.4.1 实验设计

(一)子网划分

本实验设计中，该学校申请到前缀为 211.69.4.0/22 的地址块。同时，该学校有 4 个学院，1 个图书馆，3 个学生宿舍，且每个学院有 20 台主机，图书馆有 100 台主机，每个学生宿舍有 200 台主机。

我们申请到的地址块共有 1024 个 IP 地址可供使用。对于学生宿舍，由于每个学生宿舍有 200 台主机，从而考虑将 211.69.5.0/24、211.69.6.0/24、211.69.7.0/24 作为三个学生宿舍的 IP 地址块，这样每个宿舍分配到 $(256-2)=254$ 个 IP 地址，满足学生宿舍需求。

对于学院，它们分配到的总地址块则是 211.69.4.128/25。学院 1 分配 211.69.4.128/27，学院 2 分配 211.69.4.160/27，学院 3 分配 211.69.4.192/27，学院 4 分配 211.69.4.224/27，如此每个学院均有 30 个 IP 地址可以使用，也满足组网需求。亦即，IP 地址的倒数第 6、5 位若分别为 00，则对应学院 1；类似地，01 对应学院 2，10 对应学院 3，11 对应学院 4。

对于图书馆，该设施需要 100 个 IP 地址，因此考虑分配 211.69.4.0/25。需要注意的是，我们对图书馆和学院的划分的区别是地址的倒数第 8 位，若该位为 1，则属于学院地址；若该位为 0，则归属于图书馆。

(二)网络拓扑结构设计

我们依据上述划分，对网络拓扑结构的设计如图 1.31 所示。我们使用树形的结构设计，这

样可以更加方便地进行流量控制。

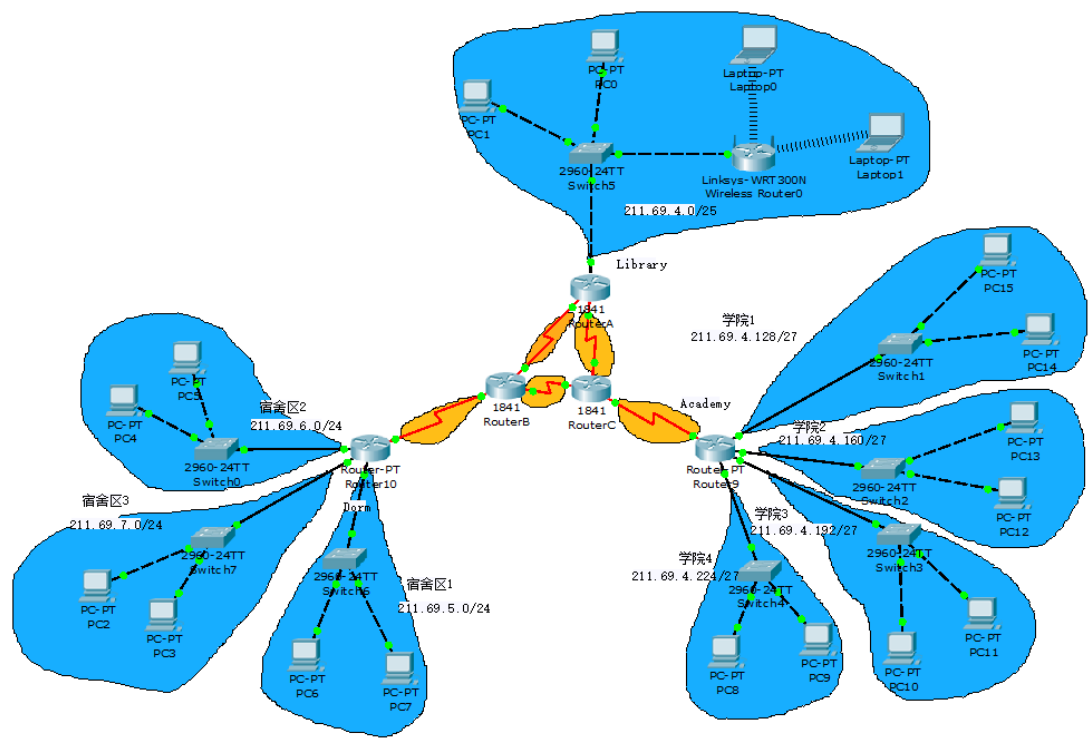


图 1.31 综合部分网络拓扑结构设计

1.4.2 实验步骤

(一) 路由器接口配置

首先我们为各个主机分配 IP 地址，此处不再赘述。其次，由于路由器的接口数不足，我们为各个路由器增加满足其需要的串口及快速以太网接口。需要注意的是，我们对路由器之间分配的串口 IP 地址为 192.168.0.0/16，这是为了防止因 IP 共用导致的路由混乱。

连接学院的路由器 9 因为其端口数量较大，如图 1.32 所示，我们使用了 Router-PT 设备以支持更多的端口插槽。



图 1.32 路由器物理组件添加

在配置完物理设备和 IP 地址后，我们需要对其路由选路进行支持。我们这里考虑使用 OSPF 协议作为网络层组网协议，如图 1.33 所示，可以通过命令行对其链路状态、连接网段进行更新。

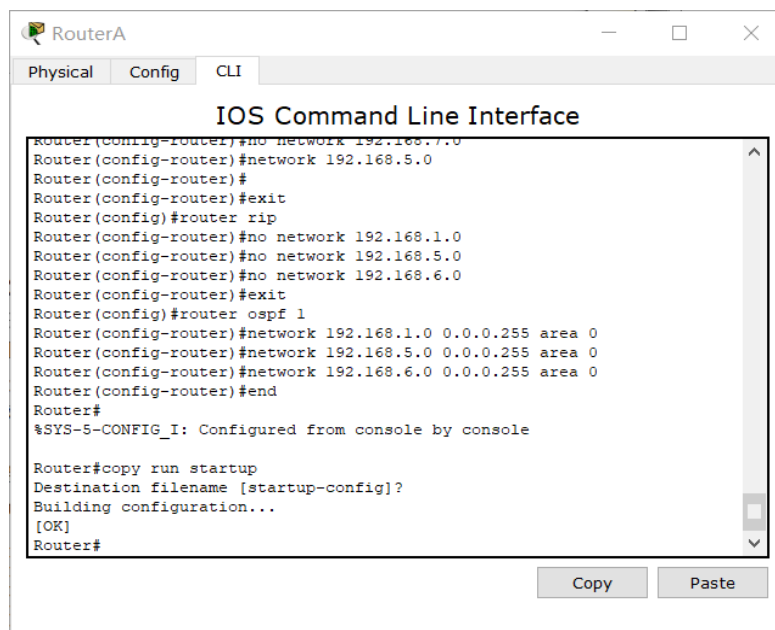


图 1.33 使用 OSPF 协议进行路由协议组网

在配置完路由协议后，我们首先尝试学院内部、宿舍区内部以及图书馆内部是否可以相互连通。如图 1.34 所示，在学院间尝试建立相互通信均可连通。实际上，我们也测试了另外两个区块内部的通信效果，也均可连通。

PDU List Window										
Fire	Last Status	Source	Destination	Type	Color	Time (sec)	Periodic	Num	Edit	Delete
	Successful	PC11	PC13	ICMP		0.000	N	0	(edit)	(delete)
	Successful	PC9	PC13	ICMP		0.000	N	1	(edit)	(delete)
	Successful	PC14	PC11	ICMP		0.000	N	2	(edit)	(delete)
	Successful	PC8	PC13	ICMP		0.000	N	3	(edit)	(delete)

图 1.34 在学院内部测试通信

而在测试不同区域的连通性时，我们发现学院的数据包无法抵达宿舍区 3。经过排查，我们发现配置的 OSPF 协议输入错误，于是用“no network 211.69.4.0 0.0.0.255 area 0”指令消除错误命令，并重新建立正确的联系。相关纠错情况如图 1.35 所示，从学院处发送数据包至宿舍区 3 的主机的模拟界面如图 1.36 所示。根据图 1.36，我们可见数据包正确地进入目的路由器节点，即将转发至目标主机。

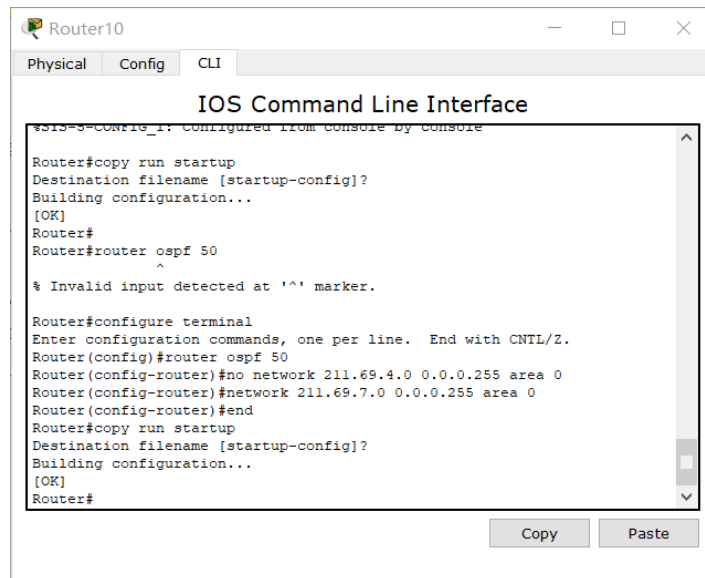


图 1.35 为 OSPF 选路纠错

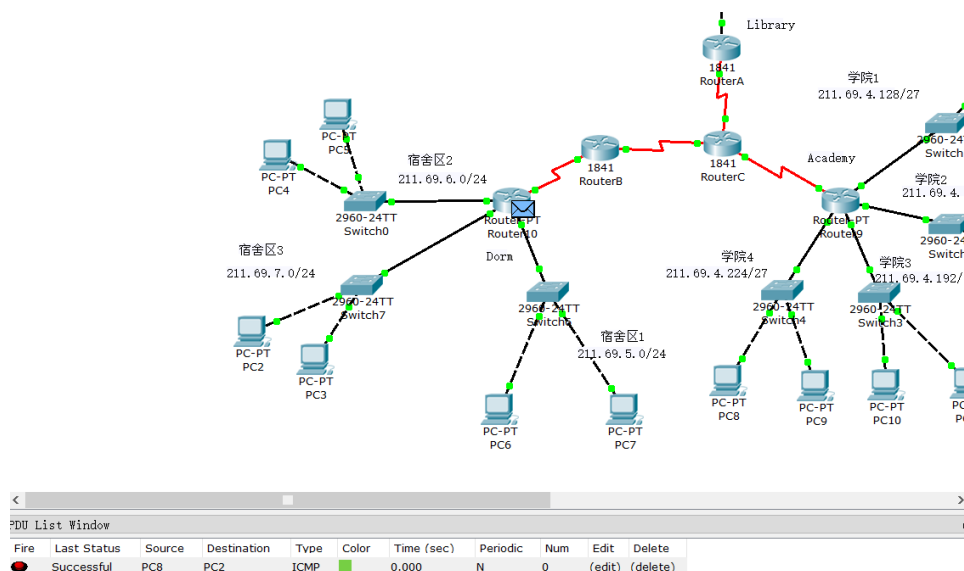


图 1.36 纠错完成示例

这样我们就完成了对基本线路的支持。我们完成了任意两个主机的相互通信，下面我们需要配置无线端口，并依据组网需求，阻挡学院和学生宿舍之间的互相访问。

(二)DHCP 服务器及无线网端配置

我们进入无线路由服务器的 GUI 界面，如图 1.37 所示，设置其 IP 地址为 211.69.4.4，子网掩码为 255.255.255.128。对于新接入的无线主机，通过 DHCP 服务器对其进行 IP 地址分配。分配的 IP 地址从 211.69.4.100 开始，最大分配 28 个用户。

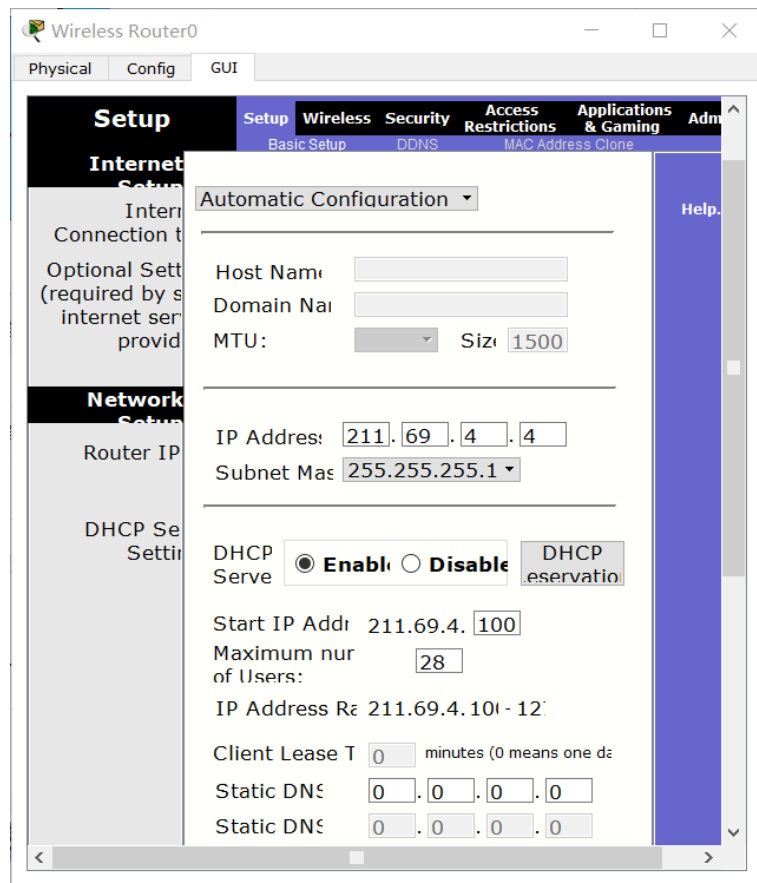


图 1.37 DHCP 服务器配置

我们继续改装 Laptop，使得其可以无线上网。我们在断电后，为笔记本添加 Linksys-WPC300N 模块，再将其接入电源，使之可以接入无线网络，再由我们的 DHCP 服务器通过申请与反馈，将 IP 地址分配至对应笔记本中。改装后的笔记本如图 1.38 所示。

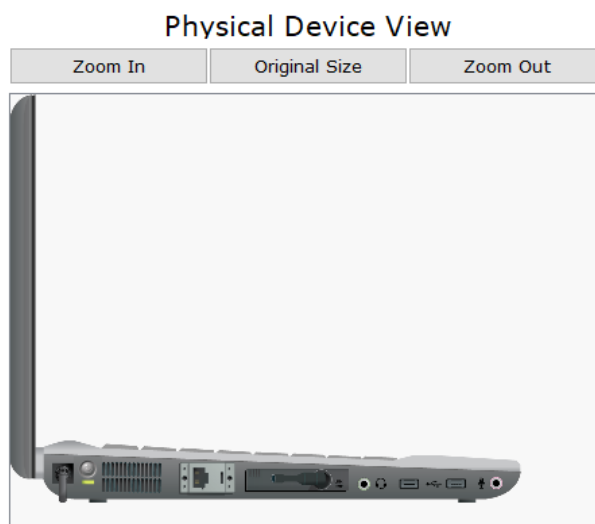


图 1.38 无线联网物理接口添加

(三)组网需求满足

由于实验要求中，需要我们对学院和宿舍进行隔离。因此我们只需要配置学院端，以及学生宿舍端，对于来自相应 IP 地址段的报文进行拒绝。例如，如图 1.39 所示，对于学院端，边缘路由器需要拒绝来自宿舍网段的数据包（211.69.5.0/24、211.69.6.0/24、211.69.7.0/24），同时对于其它网端的数据包则应接受。我们对于该路由器的 ACL 表进行配置，可以达成该目的。

```
Router#
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#access-list 35 deny 211.69.5.0 0.0.0.255
Router(config)#access-list 35 deny 211.69.6.0 0.0.0.255
Router(config)#access-list 35 deny 211.69.7.0 0.0.0.255
Router(config)#access-list 35 permit any
Router(config)#int se2/0
Router(config-if)#ip access-group 35 in
Router(config-if)#exit
Router(config)#
```

图 1.39 对学院端边缘路由进行配置

而图 1.40 则为我们配置宿舍端的边缘路由命令。该边缘路由则需要对接入串口 Se2/0 实施 ACL 表项，使得其拒绝来自 211.69.4.127/25 的数据包，而对其它数据包允许进入。

```
Router#
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#access-list 36 deny 211.69.4.128 0.0.0.127
Router(config)#access-list 36 permit any
Router(config)#int se2/0
Router(config-if)#ip access-group 36 in
Router(config-if)#exit
Router(config)#
```

图 1.40 对宿舍端边缘路由进行配置

这样我们完成了对所有实验基础要求及进阶要求的支持，下面我们验证我们的组网结果。

1.4.3 结果分析

首先我们测试不同网段内部的通信情况，如图 1.41 所示，我们以宿舍区内部为例进行测试，发现可以正常运行。这说明网段内部是可以相互通信的。









Fire	Last Status	Source	Destination	Type	Color	Time (sec)	Periodic	Num	Edit	Delete
	Successful	PC5	PC2	ICMP		0.000	N	0	(edit)	(delete)
	Successful	PC4	PC3	ICMP		0.000	N	1	(edit)	(delete)
	Successful	PC3	PC7	ICMP		0.000	N	2	(edit)	(delete)
	Successful	PC2	PC3	ICMP		0.000	N	3	(edit)	(delete)

图 1.41 网段内部通信情况

其次，我们尝试图书馆端与宿舍区、学院的通信情况。如图 1.42 所示，图书馆与学院、宿舍区均可连通，这说明实验要求可以连通的也是可连的。特别地，图书馆的无线主机也可与宿舍区相连。









Fire	Last Status	Source	Destination	Type	Color	Time (sec)	Periodic	Num	Edit	Delete
	Successful	PC1	PC5	ICMP		0.000	N	0	(edit)	(delete)
	Successful	PC0	PC2	ICMP		0.000	N	1	(edit)	(delete)
	Successful	PC6	PC1	ICMP		0.000	N	2	(edit)	(delete)
	Successful	Laptop0	PC5	ICMP		0.000	N	3	(edit)	(delete)

图 1.42 图书馆与宿舍、学院通信情况

最后我们验证学院和宿舍区的不可连通性。如图 1.43 所示，可见宿舍区和学院无法互联，这是由于路由器的接入限制。











Fire	Last Status	Source	Destination	Type	Color	Time (sec)	Periodic	Num	Edit	Delete
	Failed	PC15	PC5	ICMP		0.000	N	0	(edit)	(delete)
	Failed	PC14	PC4	ICMP		0.000	N	1	(edit)	(delete)
	Failed	PC11	PC3	ICMP		0.000	N	2	(edit)	(delete)
	Failed	PC14	PC7	ICMP		0.000	N	3	(edit)	(delete)
	Failed	PC6	PC10	ICMP		0.000	N	4	(edit)	(delete)

图 1.43 宿舍与学院通信情况

综上所述，我们验证了实验组网符合预期。特别注意到，实验中 RIP 选路只能以前 24 位作为选路地址，而本实验我们给图书馆、学院分配的地址的前三字节均为 211.69.4，因此我们选用了 OSPF 协议作为路由选路协议。之后，我们再对 ACL 表单进行了配置，从而实现了目标信息的选通。

1.5 其它需要说明的问题

暂无。

1.6 参考文献

[1] James, F. Kurose, Keith, W. Ross. 计算机网络：自顶向下方法. (第七版) 北京：机械工业出版社，2018.7

心得体会与建议

2.1 心得体会

对于 Socket 编程实验，我使用 C++ 对服务器端和客户端进行了支持。该实验主要是调用库函数以对 C/S 端的互相通信做保障，这次实验是我第一次接触网络通信，以及网页报文的获取和发布。

可靠数据传输也有一定的难度，收到了 ACK 应当怎么做，如果超时应当有什么处理。这些都在让我反思上课时学习的内容，从而对传输层有了较好的把握和认识。

CPT 组网实验是与计算机网络课程强相关的一门实验。我在做这次实验时，最初因为对网络层的理解还没有比较立体的把握，最初做起来绞尽脑汁，而且不知道为何连通不了。之后我复习了网络层的相关内容，对网络的选路协议运作有了更好的认识之后，就比较行云流水啦。

我觉得这次实验对我的进步还是挺大的，有些事情还是要实际动手去做的。所谓“**Make your hands dirty.**”，所谓“纸上得来终觉浅”，都让我从对书本的理解之外，对实际上网络的运作流程更为了解。

特别是 CPT 的“模拟 (Simulation)”功能，我觉得很有必要和学生讲解一下，这个是如何使用的。我观察数据包的传递，实在对我理解 OSPF 协议的 ICMP 数据包如何传递有太大帮助了！它的 PDU 传递的小动画，既生动，又形象，非常直观而可感！

但与此同时，该工具也会有需要反复重新尝试才能得到正确结果的情形，不稳定性较大，同时逻辑组网和实际组网仍有差距，该工具具有一定的局限。

2.2 建议

我觉得现有的实验已经挺完善的，只是许多材料我们仍然需要自己去互联网上去寻找，诸如 CPT 命令行的诸多指令。我个人的建议是，对于实验一，如果只允许学生使用 UDP（这种情形很常见），那么应用应当如何传输和获取报文呢？这可能会很有意思。