

浙江大学

本科实验报告

课程名称： 计算机网络基础

实验名称： 使用模拟软件组建互联网络

姓 名： 沈子衿

学 院： 计算机学院

系： 软件工程

专 业： 软件工程

学 号： 3160104734

指导教师： 董玮

2018 年 12 月 1 日

浙江大学实验报告

实验名称: 使用模拟软件组建互联网络 实验类型: 设计实验

同组学生: 林宇翔 实验地点: 计算机网络实验室

一、 实验目的:

- 学习掌握 PacketTracer 模拟软件的使用方法
- 学习掌握交换机、路由器的配置方法
- 学习掌握 VLAN 的工作原理, 以及如何配置 VLAN
- 学习掌握 IP 路由的工作原理, 以及如何设置静态路由表

二、 实验内容

- PacketTracer 模拟软件是 Cisco 公司推出的学习交换和路由设置的模拟软件。
- 分别采用以下方式组建网络, 测试连通性, 产生模拟数据包, 观察网络数据包流向
 - ✓ 使用 HUB、无线 AP 和 PC 机搭建局域网,
 - ✓ 使用单个交换机和 PC 机搭建局域网并配置 VLAN, 观察网络数据包流向
 - ✓ 使用多个交换机和 PC 机搭建局域网并配置 VLAN 中继, 观察网络数据包流向
 - ✓ 使用多个路由器连接多个局域网, 并配置静态路由

三、 主要仪器设备

- 联网的 PC 机
- PacketTracer 模拟软件

四、 操作方法与实验步骤

- 安装 PacketTracer 模拟软件

Part 1. 组网

- 使用 1 个 HUB 和 5 个 PC 机搭建第 1 个局域网, 并使用子网地址 10.1.0.0/8
- 使用 1 个无线 AP 和 3 个 PC 机搭建第 2 个局域网, 并使用子网地址 10.2.0.0/8
- 使用 1 个交换机和 3 个 PC 机搭建第 3 个局域网, 并使用子网地址 10.3.0.0/8
- 使用 1 个交换机和 3 个 PC 机搭建第 4 个局域网, 并使用子网地址 10.4.0.0/8
- 使用第 5 个交换机, 将 4 个局域网连接起来
- 使用 Ping 命令查看各个网络之间的连通性
- 修改第 1、2 局域网的子网掩码为 16 位, 再次查看各个网络之间的连通性
- 修改第 3、4 局域网的子网掩码为 16 位, 再次查看各个网络之间的连通性
- 每组建一次网络, 在 PC 上产生模拟数据包, 跟踪数据包的流向

Part 2. VLAN

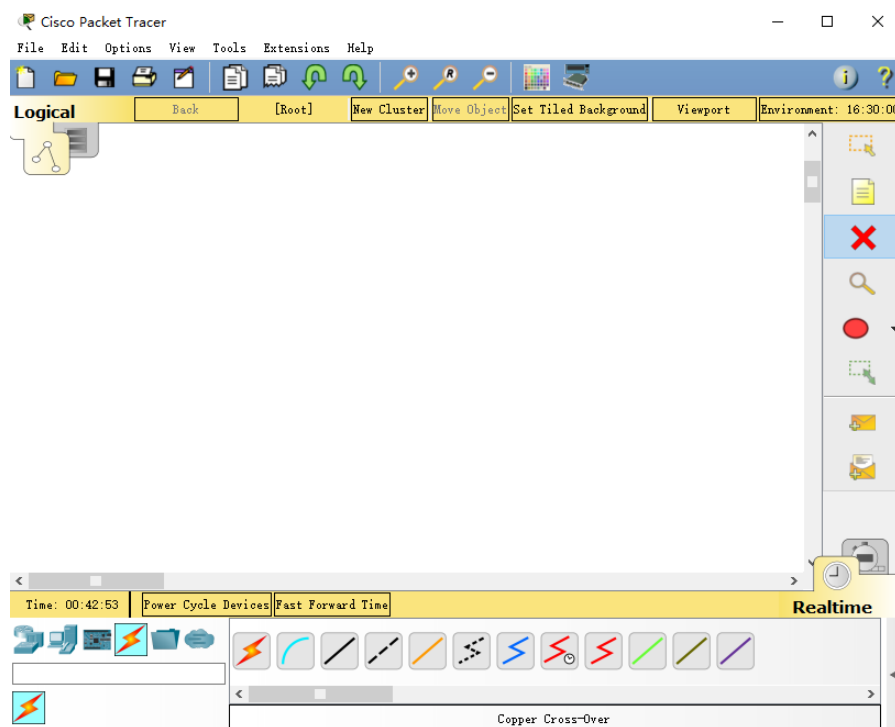
- 将 AP 和 HUB 换成交换机，并在 4 个局域网交换机上划分出 2 个 VLAN，让 PC 机属于不同 VLAN
- 使用 Ping 命令查看各个网络的联通性
- 修改 4 个局域网的子网地址，给不同组的 VLAN 分配不同的子网地址，再次查看各个网络之间的联通性
- 在第 5 个交换机（互联交换机）上设置 VLAN，使不同局域网内某个 VLAN 组的 PC 之间能够互通
- 在第 5 个交换机（互联交换机）上启用 VLAN Trunk，使不同局域网内相同 VLAN 组的 PC 之间都能够互通
- 产生模拟数据包，通过模拟软件跟踪数据包的流向

Part 3. 路由

- 将第 5 个交换机删除，每个局域网分别设立一个路由器
- 给各个路由器创建 2 个子接口，并分配合适的 IP 地址，使得同一局域网内，不同 VLAN 的 PC 之间能够互通
- 使用第 5 台路由器分别连接 4 个局域网的路由器
- 启用动态路由协议 RIP，使得不同子网的 PC 之间能够互通
- 关闭动态路由协议 RIP，给各个路由器设置正确的静态路由，使得不同子网的 PC 之间能够互通
- 产生模拟数据包，通过模拟软件跟踪数据包的流向

五、 实验数据记录和处理

- 运行 PacketTracer 模拟软件，界面上由哪些部分组成，分别有什么作用？

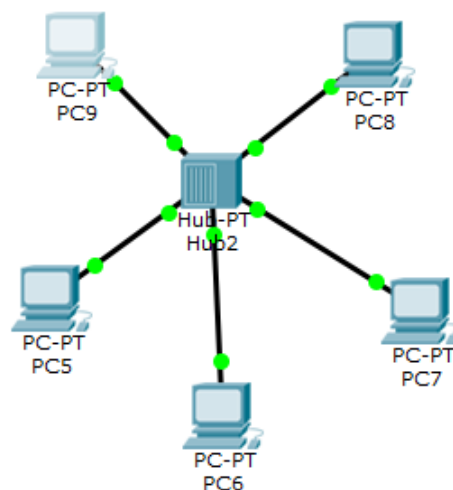


打开 PacketTracer，可以看到主界面上有上、下和右三栏。下方设备列表区可以选择设备并将它们拖动到中央的编辑区域内。对设备分为六种：Network Devices, End Devices, Components, Connections, Miscellaneous, Mutiluser Connection； 上方菜单栏可对文件进行保存、打印。还可以调整界面视图；右方拓扑工作区可以对对象执行复制、剪切和删除等编辑操作。总体逻辑还是很清晰的。

-----Part 1. 组网-----

- 使用 1 个 HUB 和 5 个 PC 机搭建第 1 个局域网，并使用子网地址 10.1.0.0/8 给每个 PC 配置 IP 地址。在其中一台 PC 上创建数据包，目标地址设置为另外一台 PC，观察数据包的流向。

将五个 PC 机和 1 各 1 个 Hub 拖入编辑区域，使用“自动选择连接方式”进行连接，之后依次将五台 PC 的 IP 地址配置为 10.1.0.1-10.1.0.5，子网掩码为 255.0.0.0。



Physical Config Desktop Attributes Software/Services

IP Configuration

IP Configuration

☐ DHCP ☒ Static

IP Address: 10.1.0.1

Subnet Mask: 255.0.0.0

Default Gateway:

DNS Server: 0.0.0.0

IPv6 Configuration

☐ DHCP ☐ Auto Config ☒ Static

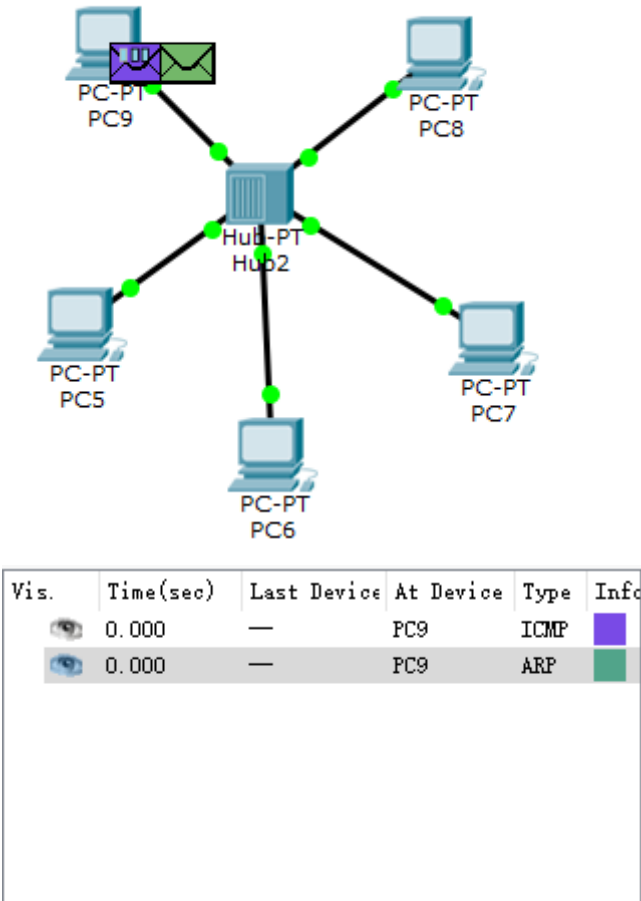
IPv6 Address: /

Link Local Address: FE80::290:CFF:FE62:5228

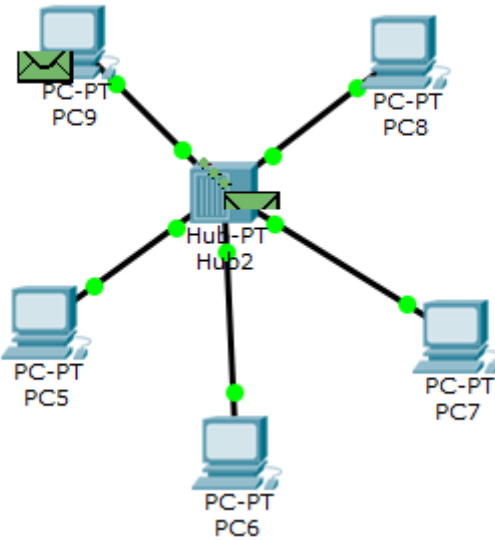
IPv6 Gateway:

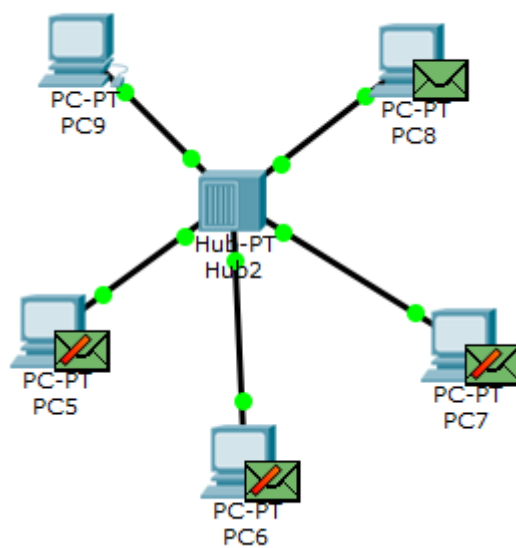
IPv6 DNS Server:

在 PC9 (10.1.0.1) 上生成一个数据包发向 PC8 (10.1.0.5)，将模式调整为 `simulate`，观察到如下场景：

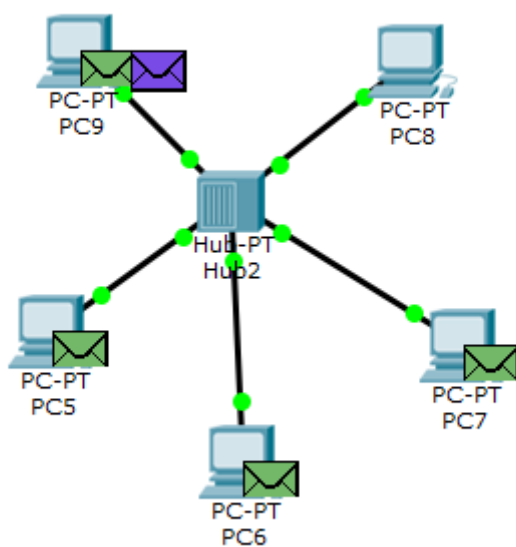


点击 `capture forward`，继续调试，观察路径





发现包发给了子网中的所有 PC，但 IP 地址不匹配的都被拒收了。



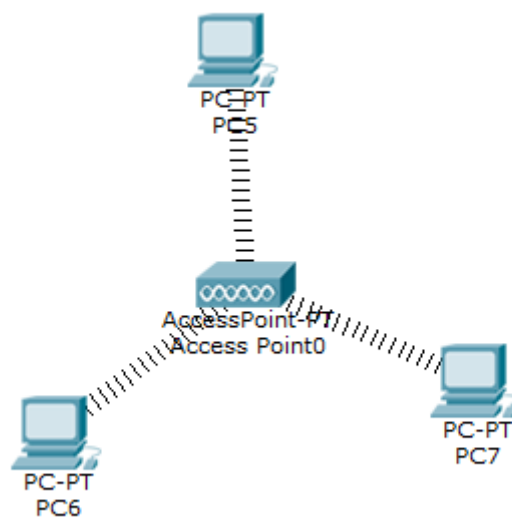
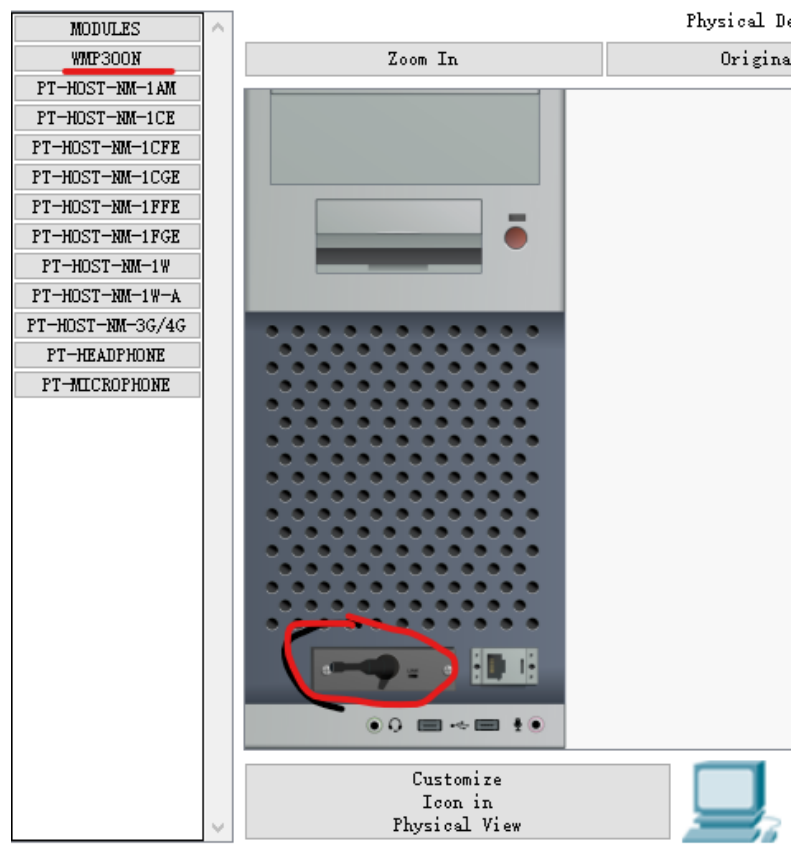
之后 PC8 会答复一个包，这个包同样被广播，但只有 PC9 接收了它。

Vis.	Time(sec)	Last Device	At Device	Type	I
	3.026	Hub2	PC5	ICMP	
	3.026	Hub2	PC7	ICMP	
	3.026	Hub2	PC8	ICMP	
	3.027	PC8	Hub2	ICMP	
	3.028	Hub2	PC6	ICMP	
	3.028	Hub2	PC5	ICMP	
	3.028	Hub2	PC7	ICMP	
	3.028	Hub2	PC9	ICMP	

这之后，PC9 重复以上操作 4 次，被捕获的包情况如上。

- 使用 1 个无线 AP 和 3 个 PC 机搭建第 2 个局域网，并使用子网地址 10.2.0.0/8 给每个 PC 配置 IP 地址。在其中一台 PC 上创建数据包，目标地址设置为另外一台 PC，观察数据包的流向。

首先将三台 PC 机的网卡更换为无线网卡

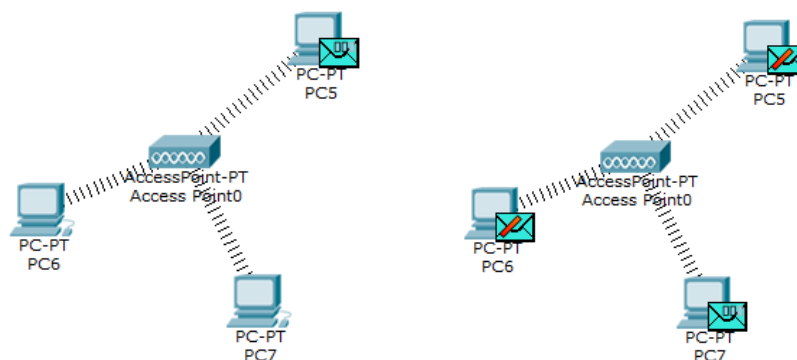


此时尚未 IP 地址，发现无线 AP 已同三台 PC 都自动建立了连接。

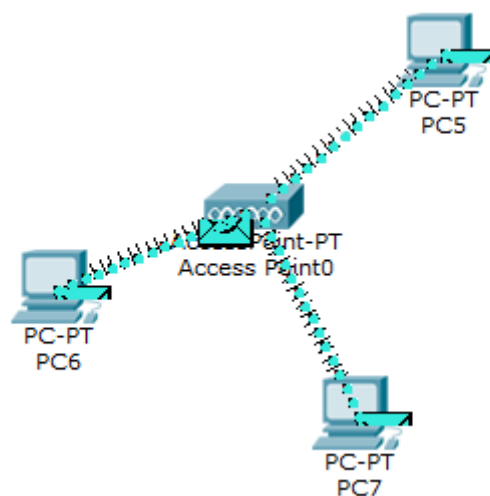
```
Pinging 10.2.0.2 with 32 bytes of data:  
  
Reply from 10.2.0.2: bytes=32 time=0ms TTL=128  
Reply from 10.2.0.2: bytes=32 time=3ms TTL=128  
Reply from 10.2.0.2: bytes=32 time=3ms TTL=128  
Reply from 10.2.0.2: bytes=32 time=4ms TTL=128  
  
Ping statistics for 10.2.0.2:  
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss)  
Approximate round trip times in milli-seconds:  
    Minimum = 0ms, Maximum = 4ms, Average = 2ms  
  
PC>
```

尝试 ping，可以 ping 通。

这里尝试跟踪从 PC5 到 PC7 的包：



以及从 PC7 返回 PC5 的包

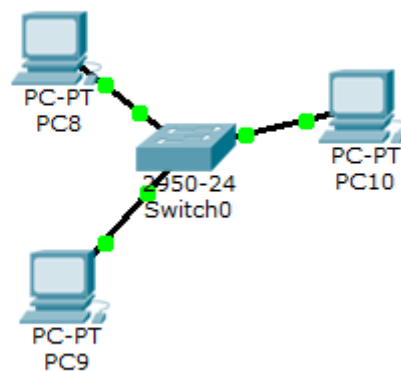


0.006	PC5	Access ...	ICMP
0.007	--	Access ...	ICMP
0.008	Access R..	PC6	ICMP
0.008	Access R..	PC7	ICMP
0.008	Access R..	PC5	ICMP
0.010	--	PC7	ICMP
0.011	PC7	Access ...	ICMP

这个过程和使用有线连接的逻辑是一致的。

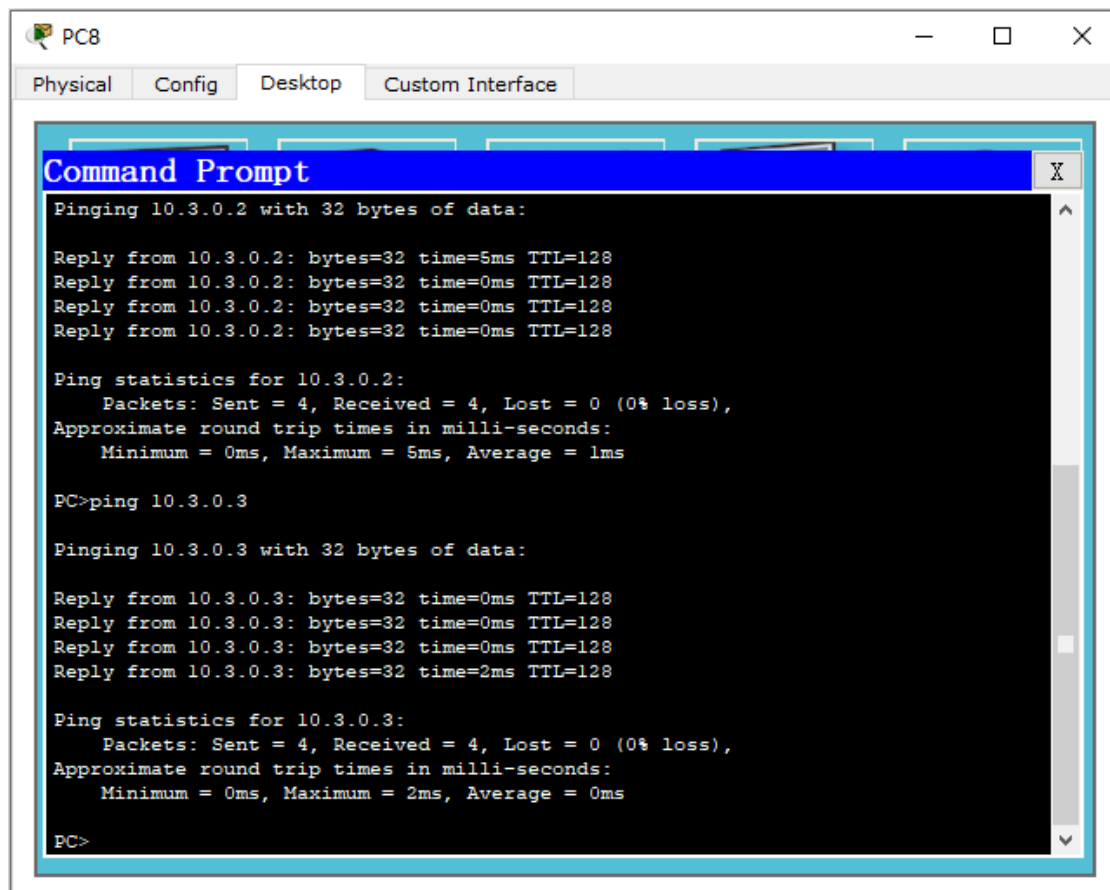
以下使用的交换机全部为二层交换机，比如 2950。

- 使用 1 个交换机和 3 个 PC 机搭建第 3 个局域网，并使用子网地址 10.3.0.0/8。给每个 PC 配置 IP 地址。在其中一台 PC 上创建数据包，目标地址设置为另外一台 PC，观察数据包的流向。

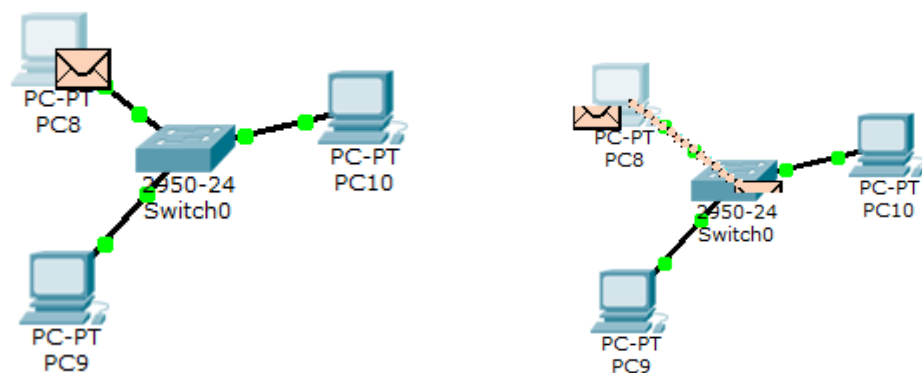


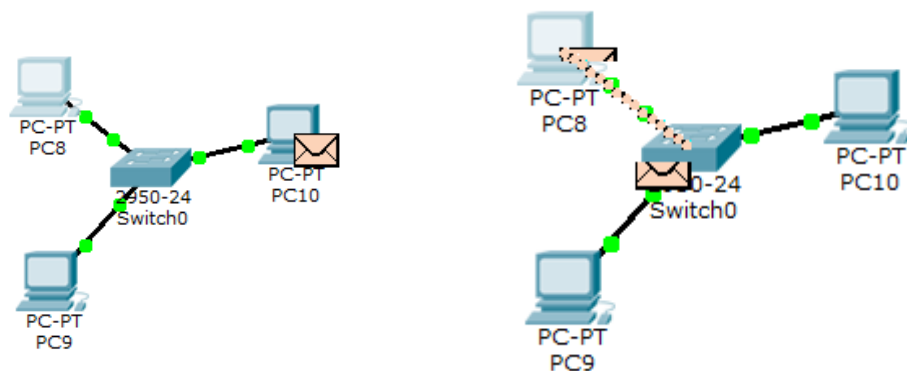
使用三台 PC 机连接 2950 交换机，配置好 IP 后测试联通无误，同 hub 连接的区别是连接好之后有短暂但明显的加载时间。

在 PC8（10.3.0.1）上创建 ping 数据包，访问 PC10（10.3.0.3）



在 simulation 模式下监测这一过程:

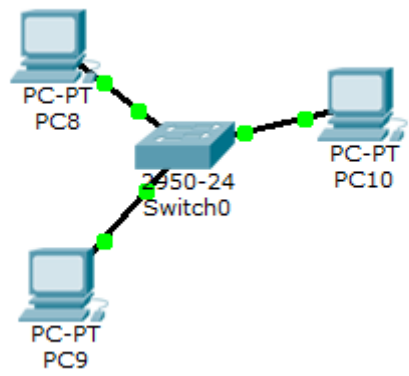




值得注意的是，交换机并没有将 hub 和无线 AP 那样将包分发到子网的各个 PC 上（这种情况下，PC 通过判断 IP 地址决定是否接收），而是定向发给了接收者 PC10，PC10 发送响应包时也是如此。这说明交换机有简单的路由功能。

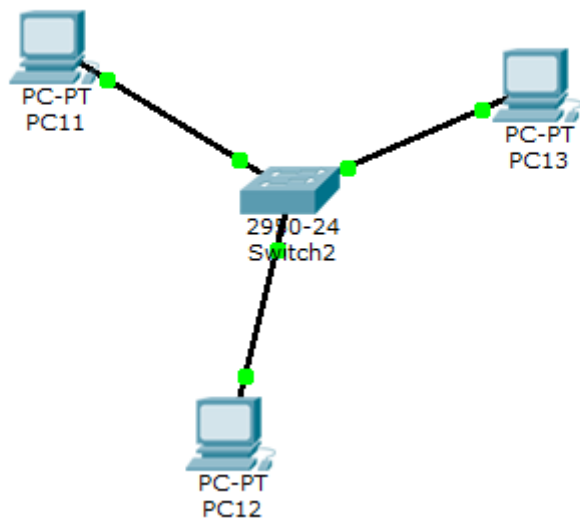
0.001	PC8	Switch0	ICMP
0.002	Switch0	PC10	ICMP
0.003	PC10	Switch0	ICMP
0.004	Switch0	PC8	ICMP
1.007	--	PC8	ICMP
1.008	PC8	Switch0	ICMP

在一个循环中被捕获的包如上图所示。



另一个值得注意的点是，在交换机网络联通后，表征其连接状态的小绿点是在不断闪动的。这说明交换机无时无刻不在通过请求子网中 PC 机的方式获取和更新路由信息。这同静态的 Hub 是不同的。

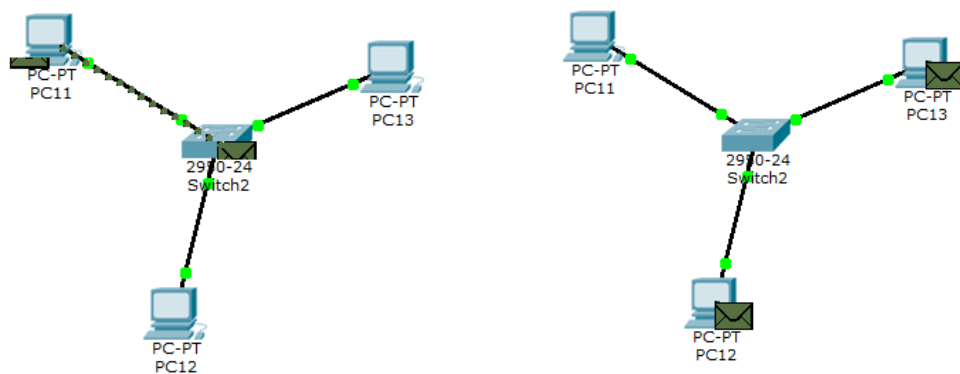
- 使用 1 个交换机和 3 个 PC 机搭建第 4 个局域网，并使用子网地址 10.4.0.0/8。给每个 PC 配置 IP 地址。在一台 PC 上再创建一个广播包，观察数据包的流向。



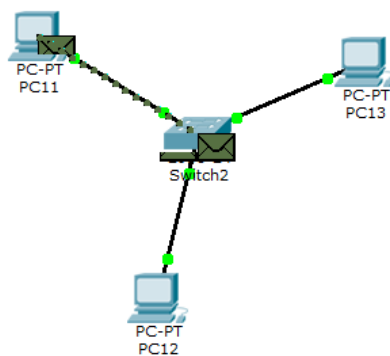
使用三台 PC 机连接 2950 交换机，配置好 IP 后测试联通无误。

```
PC>ping 10.255.255.255
```

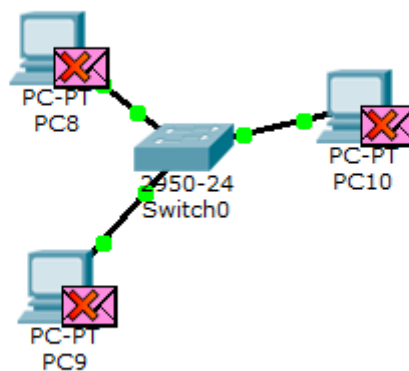
使用 PC11 创建一个广播包（ping 广播地址），观测其传输路径如下：



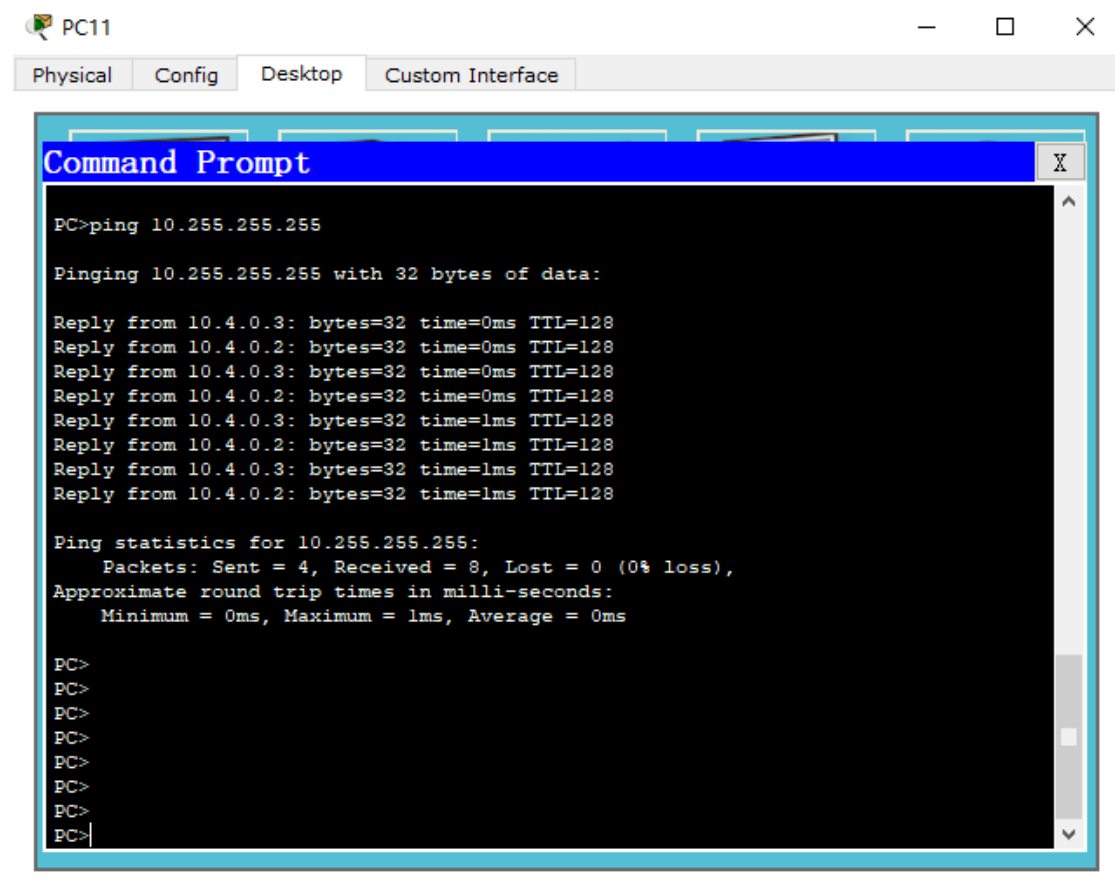
发现数据包被发送给了子网中的所有 PC，并且它们都接受了包。



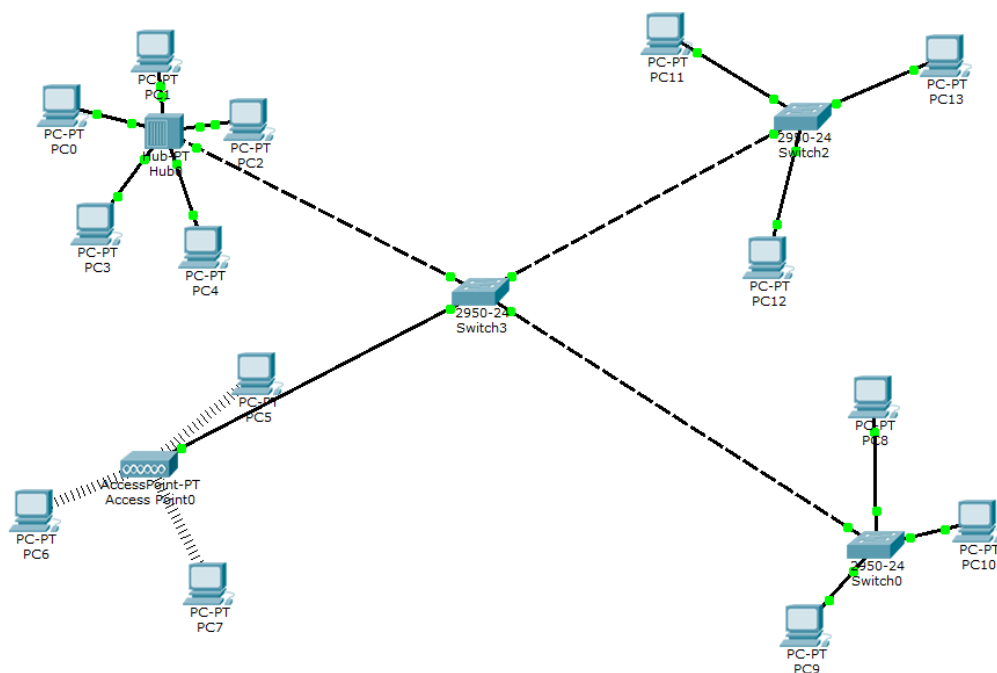
响应的时候，子网中所有的 PC 向交换机发送包，交换机将这些包统一转发为 PC11。



值得注意的是，从软件上看，由于广播地址是 10.255.255.255 的缘故（也就是说，如果联通，3 号子网从逻辑上说也在广播范围之内）此时 3 号子网的交换机貌似也向子网中的 PC “发送了包”。但考虑到 3 号和 4 号子网此时并不联通，因此包上标记有“X”，用来表明逻辑上可能存在、实际并不存在的包。



- 使用第 5 个交换机，将 4 个局域网连接起来。使用 Ping 命令检查各个局域网 PC 之间的联通性（每个局域网选取 2 台 PC 做代表）。是否都能 Ping 通？如果不通，请检查原因。



如上图，添加一台交换机，将四个子网连接在一起。稍等片刻，显示全部联通。

我们以子网 1 的 PC0/PC1，子网 2 中的 PC5/PC6，子网 3 中的 PC9, PC10 以及子网 4 中的 PC11/PC12 为例。测试结果如下：

	0	1	5	6	9	10	11	12
0	√	√	√	√	√	√	√	√
1	√	√	√	√	√	√	√	√
5	√	√	√	√	√	√	√	√
6	√	√	√	√	√	√	√	√
9	√	√	√	√	√	√	√	√
10	√	√	√	√	√	√	√	√
11	√	√	√	√	√	√	√	√
12	√	√	√	√	√	√	√	√

所有的机器都互相 ping 通了，证明在 8 位掩码下，这个小型网络是可以正常工作的。

- 把第 1、2 局域网中所有 PC 机的子网掩码从 8 位改成 16 位，再次用 Ping 检查各个 PC 之间的联通性（每个局域网选取 2 台 PC 做代表）。哪些通？哪些不通？跟踪下不通的 PC 之间的数据流，哪一个环节数据包不再流动了？原因是什么？

将 1, 2 局域网中的子网掩码更改为 16 位后，复现上述测试，结果如下：

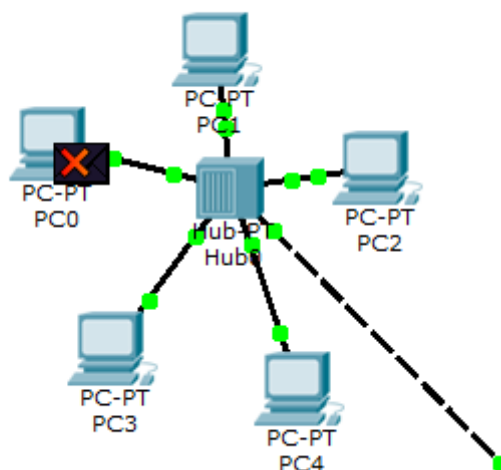
	0	1	5	6	9	10	11	12
0	√	√	×	×	×	×	×	×
1	√	√	×	×	×	×	×	×
5	×	×	√	√	×	×	×	×
6	×	×	√	√	×	×	×	×
9	×	×	×	×	√	√	√	√
10	×	×	×	×	√	√	√	√
11	×	×	×	×	√	√	√	√
12	×	×	×	×	√	√	√	√

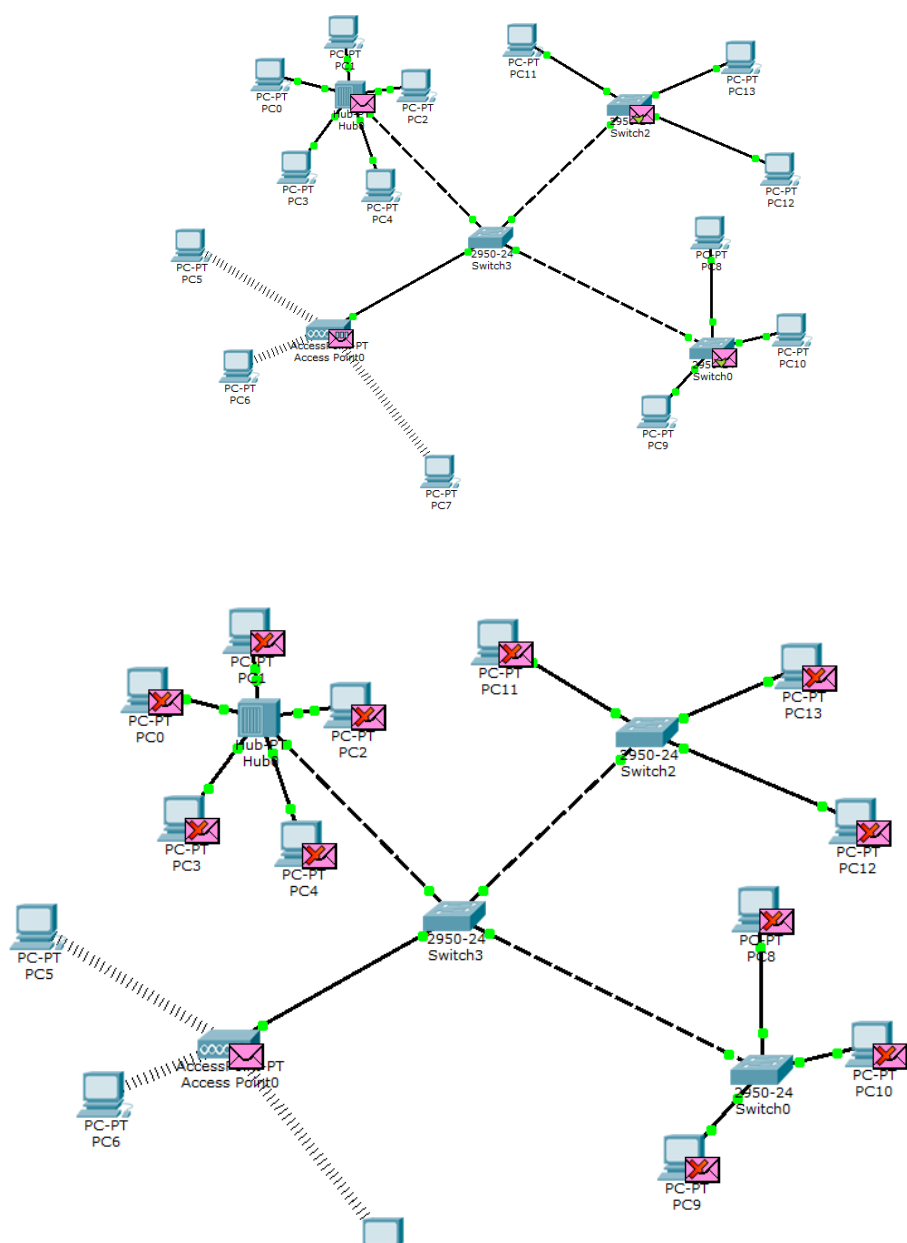
分析测试结果，可以看到如下现象：

1. 子网掩码修改不影响子网内部互相访问；
2. 1、2 子网修改掩码之后，无法再访问 3、4 子网的 PC；
3. 1、2 子网修改掩码之后，也不能互相访问；
4. 3、4 子网的 PC 在 1、2 修改掩码后，依然可以互相访问。

挑选两类不通的路径（1-3，1-2），分析原因：

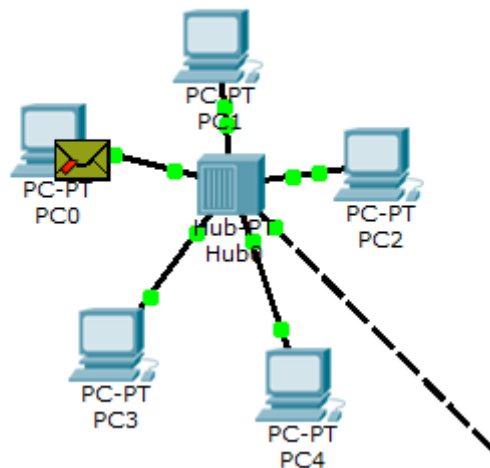
先尝试让 PC0 ping PC9，追踪包：





发现从 PC0 位置开始，数据包便不再流动了。

再尝试让 PC0 ping PC5:



发现它也是从 PC0 开始就不流动了。

- 把第 3、4 局域网的子网掩码从 8 位改成 16 位，再次用 Ping 检查各个 PC 之间的联通性（每个局域网选取 2 台 PC 做代表）。哪些通？哪些不通？

将第 3、4 局域网的子网掩码从 8 位改成 16 位后，简单测试部分子网之间的连通性。

测试 1-2 连通性如下：

```

PC>ping 10.2.0.1

Pinging 10.2.0.1 with 32 bytes of data:

Request timed out.
Request timed out.
|
  
```

发现 ping 不通。

测试 1-3 连通性如下：

```

PC>ping 10.3.0.1

Pinging 10.3.0.1 with 32 bytes of data:

Request timed out.
Request timed out.
  
```

发现 ping 不通。

测试 2-3 连通性如下：

```

PC>ping 10.3.0.1

Pinging 10.3.0.1 with 32 bytes of data:

Request timed out.
Request timed out.
  
```

发现 ping 不通。

测试 3-4 连通性如下：

```
PC>ping 10.4.0.3

Pinging 10.4.0.3 with 32 bytes of data:
```

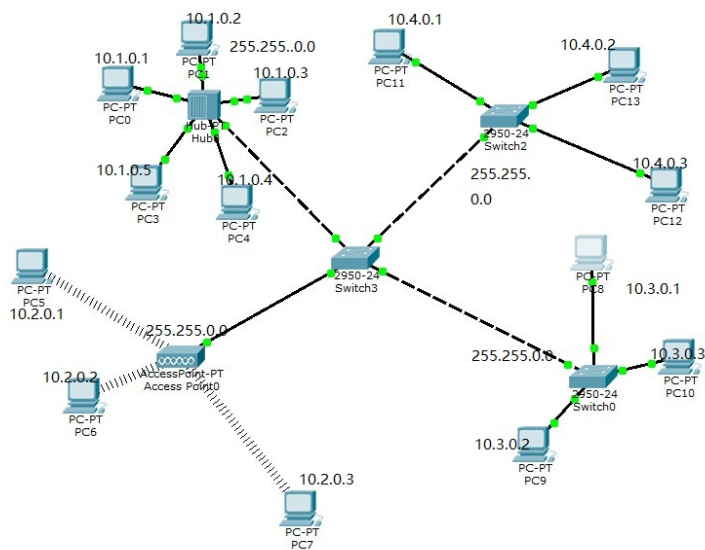
发现 ping 不通。

按照如此方法进行多次测试，得表格如下：

	0	1	5	6	9	10	11	12
0	√	√	×	×	×	×	×	×
1	√	√	×	×	×	×	×	×
5	×	×	√	√	×	×	×	×
6	×	×	√	√	×	×	×	×
9	×	×	×	×	√	√	×	×
10	×	×	×	×	√	√	×	×
11	×	×	×	×	×	×	√	√
12	×	×	×	×	×	×	√	√

发现除子网内部的 PC 机还可以互相 ping 通之外，子网间的 PC 机在修改子网掩码之后已经无法互相 Ping 通了。

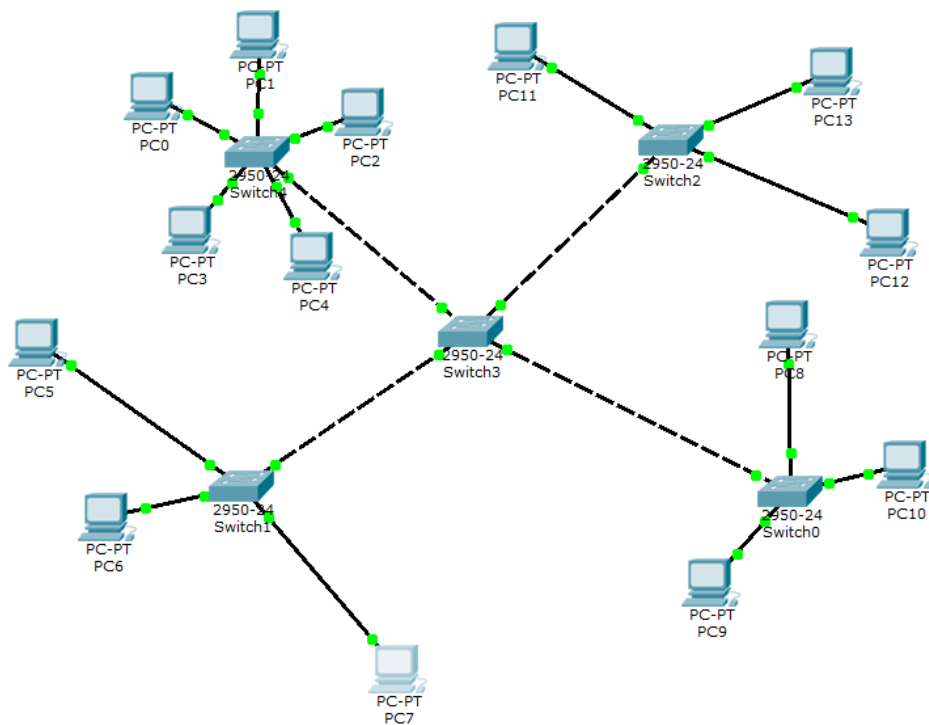
- 请把本部分的最后完整拓扑图记录在此。并标注每台 PC 的 IP 地址、子网掩码。（将实验拓扑数据保存为 part1.pkt 文件，随实验报告一起打包上交）



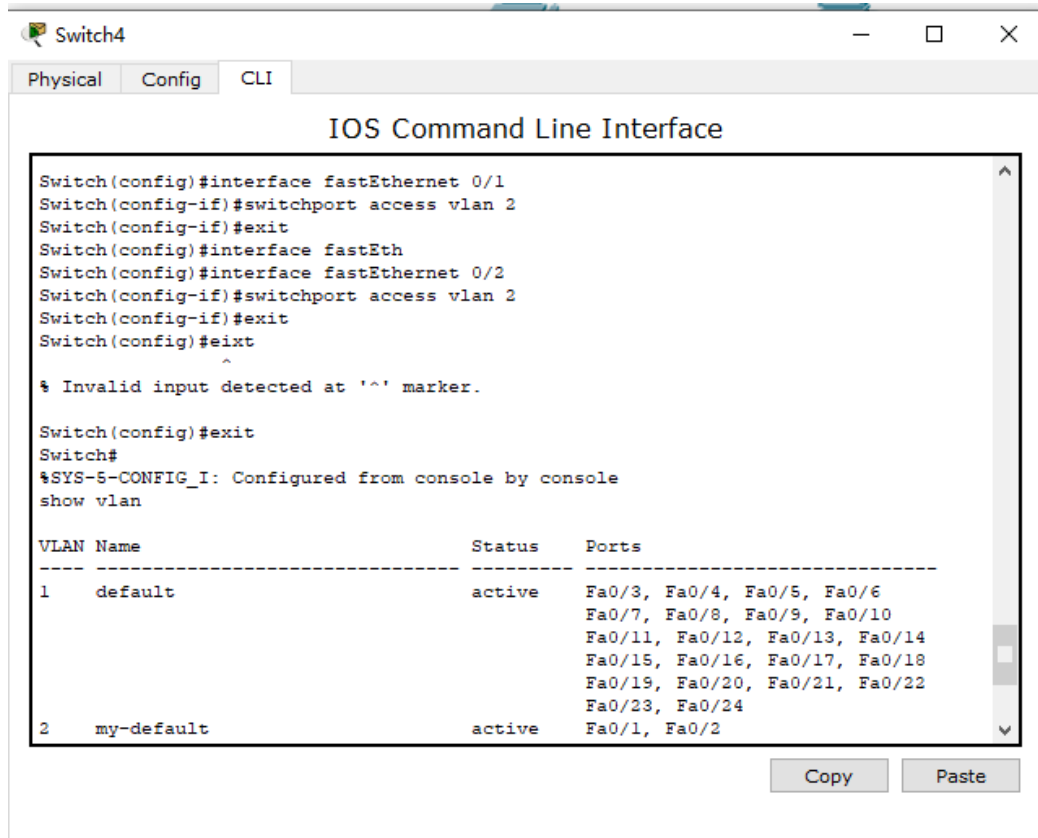
-----Part 2. VLAN-----

- 将 AP 和 HUB 都换成交换机。在第 1-4 局域网交换机上都新增 1 个 VLAN 2，让每个局域网中都有一部分 PC 机属于 VLAN 2（默认所有的 PC 都属于 VLAN 1）。使用 Ping 命令检查各个 PC 之间的联通性。哪些通？哪些不通？跟踪下不通 PC 之间的数据包，在哪一个环节数据包不再流动了？

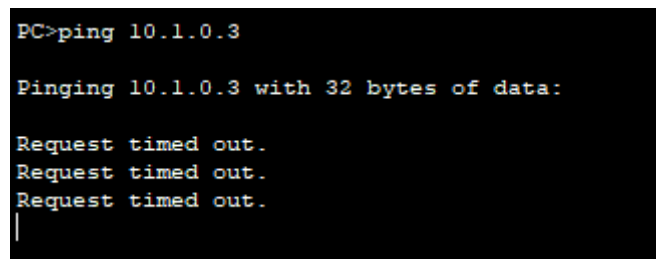
首先将 AP 和 HUB 都换成交换机：



待加载完成后，在各个交换机里新增 VLAN2，并且令端口 1 和端口 2（PC0, PC1 与交换机相连的端口，为确保无误我重连了一遍）属于 VLAN2。

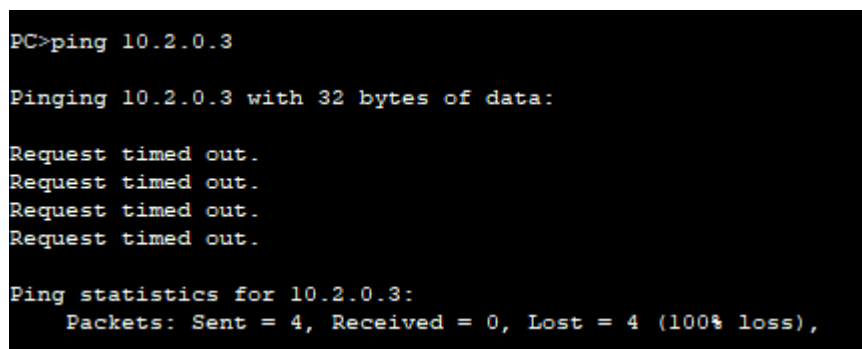


尝试使用 PC0 ping PC3:



发现 VLAN2 下的 PC0 无法 ping 通 VLAN1 下的 PC3。

尝试 ping 另一局域网下的 PC7，结果如下图所示：



发现 VLAN2 下的 PC0 无法 ping 通另一子网、VLAN1 下的 PC7。

最后尝试 ping 同为 VLAN2 的 PC1，结果如图：

```

PC>ping 10.1.0.2

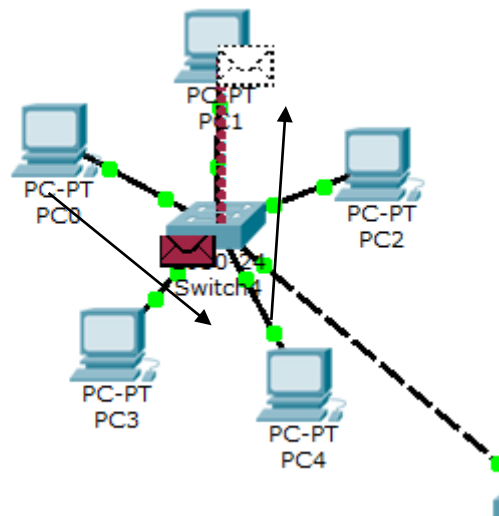
Pinging 10.1.0.2 with 32 bytes of data:

Reply from 10.1.0.2: bytes=32 time=1ms TTL=128
Reply from 10.1.0.2: bytes=32 time=0ms TTL=128
Reply from 10.1.0.2: bytes=32 time=0ms TTL=128
Reply from 10.1.0.2: bytes=32 time=0ms TTL=128

Ping statistics for 10.1.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

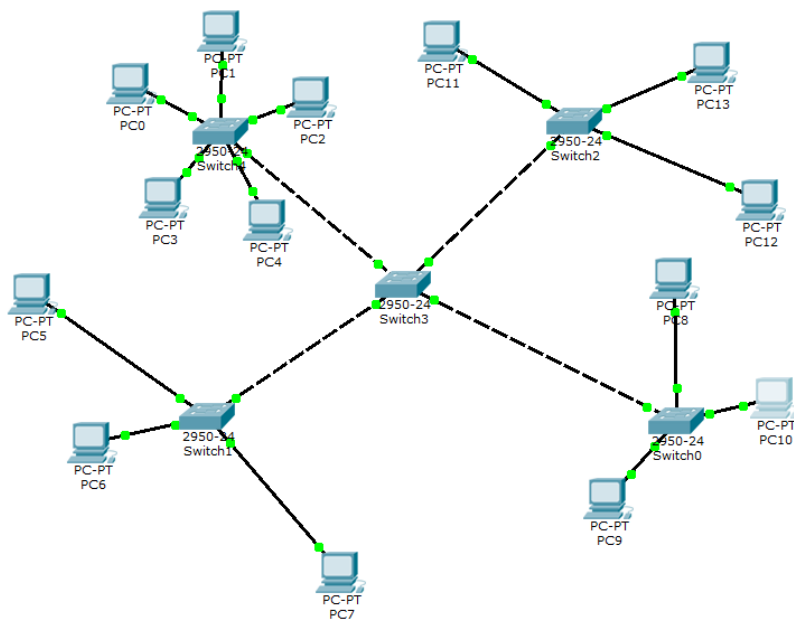
```

发现同为 VLAN2 是可以互相 ping 通的。



观察数据包的流动,发现数据包根本就未发送向不属于同一 VL 下的主机。因而另一 VLAN 下的主机实际上根本无法收到数据包。

- 在第 1-4 局域网上,把属于 VLAN 1 的 PC 的 IP 地址都改成 10.1.0.0/16 子网内的地址,把属于 VLAN 2 的 PC 的 IP 地址都改成 10.2.0.0/16 子网内的地址。使用 Ping 命令检查同属于 VLAN 1,但在不同局域网的 PC 之间的联通性(应该通)。使用 Ping 命令检查同属于 VLAN 2,但在不同局域网的 PC 之间的联通性(应该不通)。



首先，把属于 VLAN 1 的 PC 的 IP 地址都改成 10.1.0.0/16 子网内的地址，把属于 VLAN 2 的 PC 的 IP 地址都改成 10.2.0.0/16 子网内的地址。

在 PC10 尝试 ping PC2 （vlan1），结果如下：

```
Packet Tracer PC Command Line 1.0
PC>ping 10.1.0.3

Pinging 10.1.0.3 with 32 bytes of data:

Reply from 10.1.0.3: bytes=32 time=0ms TTL=128
Reply from 10.1.0.3: bytes=32 time=0ms TTL=128
Reply from 10.1.0.3: bytes=32 time=0ms TTL=128
Reply from 10.1.0.3: bytes=32 time=0ms TTL=128

Ping statistics for 10.1.0.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

发现可以 ping 通。

尝试将 pc5 至为 vlan2，之后用 PC0 去 ping：

```
PC>ping 10.2.0.3

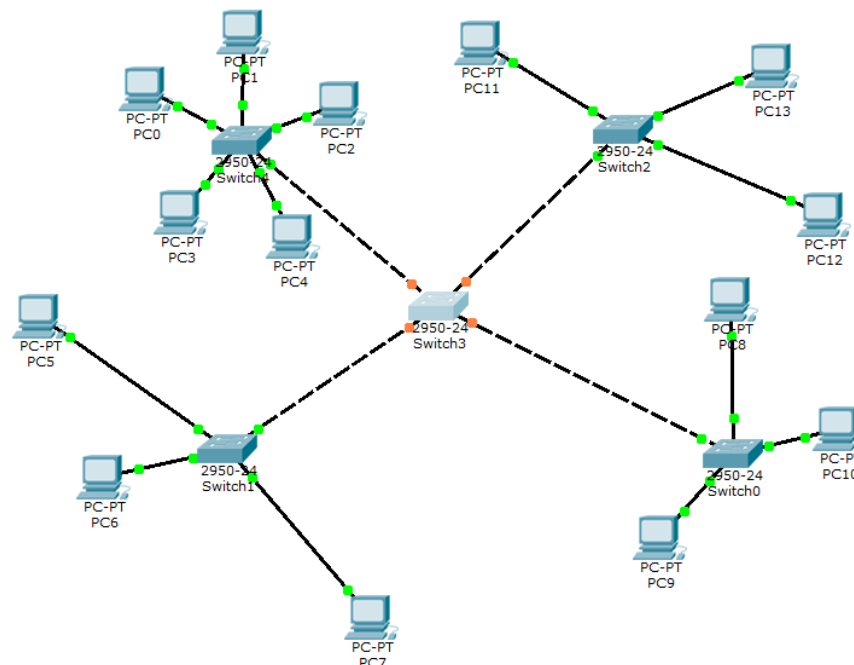
Pinging 10.2.0.3 with 32 bytes of data:

Request timed out.
```

可以看出是无法 ping 通的。

- 在第 5 个交换机（互联交换机）上新增 VLAN 2，将该交换机上连接 4 个局域网的

端口（包括 4 个交换机的端口）都修改为属于 VLAN 2。使用 Ping 命令检查同属于 VLAN 2, 但在不同局域网的 PC 之间的联通性（应该通）。此时，再次使用 Ping 命令检查同属于 VLAN 1, 但在不同局域网的 PC 之间的联通性（应该不通）。



先使用 PC10 尝试 ping 通同属于 VLAN1 下的 PC2，结果如下：

```
C:\>ping 10.1.0.3

Pinging 10.1.0.3 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 10.1.0.3:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

可以看出，无法 ping 通。

在 PC0 尝试 ping 同属于 VLAN2 下的 PC5：

```
C:\>ping 10.2.0.1

Pinging 10.2.0.1 with 32 bytes of data:

Reply from 10.2.0.1: bytes=32 time=41ms TTL=128
Reply from 10.2.0.1: bytes=32 time<1ms TTL=128
Reply from 10.2.0.1: bytes=32 time=1ms TTL=128
Reply from 10.2.0.1: bytes=32 time<1ms TTL=128
```

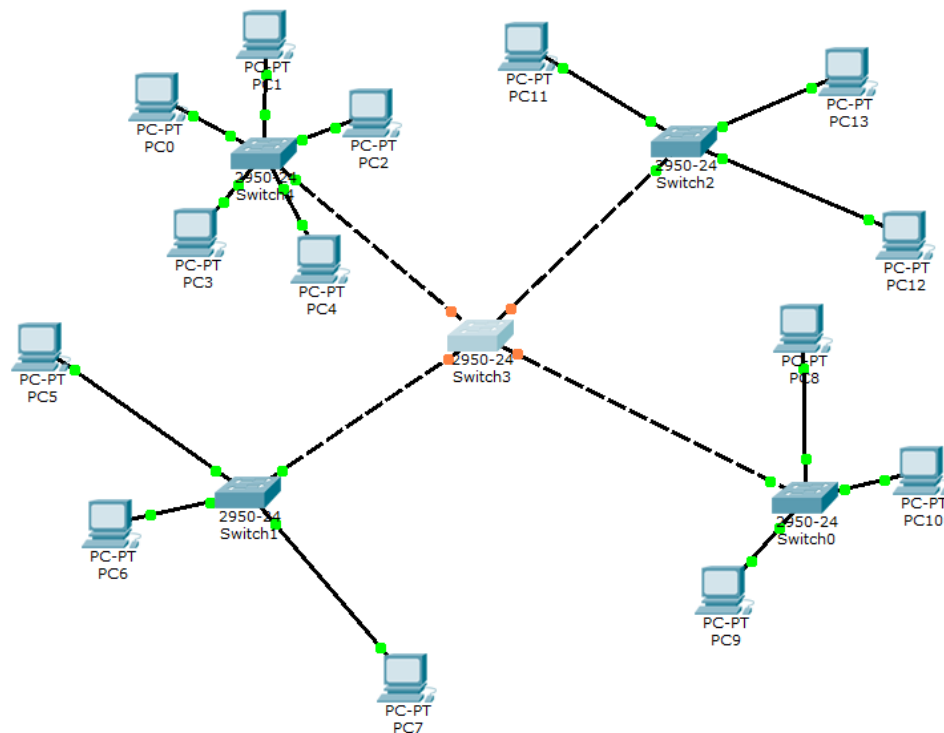
可以看出。可以 ping 通。

- 在第 5 个交换机（互联交换机）上将连接 4 个局域网的端口都修改为 VLAN Trunk

模式。使用 Ping 命令检查同属于 VLAN 1，但在不同局域网的 PC 之间的连通性。

使用 Ping 命令检查同属于 VLAN 2，但在不同局域网的 PC 之间的连通性。

XU



如图，将端口模式修改为 VLAN Trunk。

和之前一样，我们尝试使用 PC10 去 ping PC2，结果如图所示：

```
PC>ping 10.1.0.3

Pinging 10.1.0.3 with 32 bytes of data:

Reply from 10.1.0.3: bytes=32 time=1ms TTL=128
Reply from 10.1.0.3: bytes=32 time=1ms TTL=128
Reply from 10.1.0.3: bytes=32 time=0ms TTL=128
Reply from 10.1.0.3: bytes=32 time=0ms TTL=128

Ping statistics for 10.1.0.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

发现是可以 ping 通的。

在 PC0（VLAN2）上尝试 ping PC5， 结果如下：


```

PC>ping 10.2.0.3

Pinging 10.2.0.3 with 32 bytes of data:

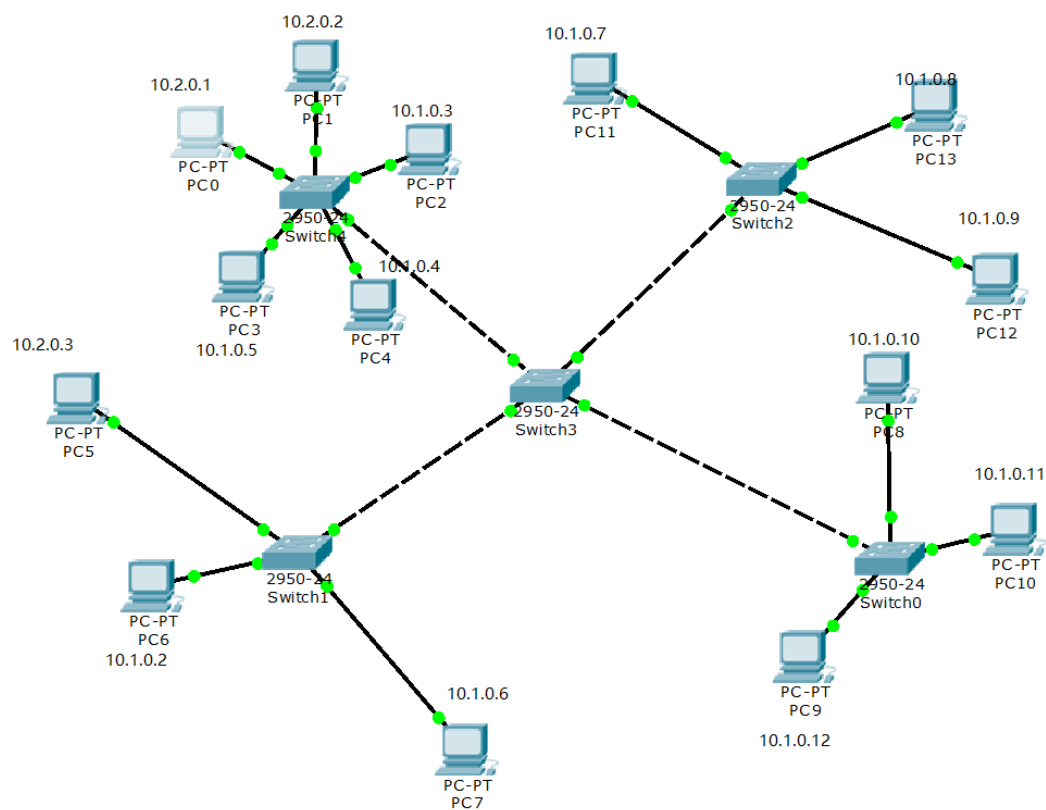
Reply from 10.2.0.3: bytes=32 time=0ms TTL=128
Reply from 10.2.0.3: bytes=32 time=0ms TTL=128
Reply from 10.2.0.3: bytes=32 time=0ms TTL=128
Reply from 10.2.0.3: bytes=32 time=0ms TTL=128

Ping statistics for 10.2.0.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

```

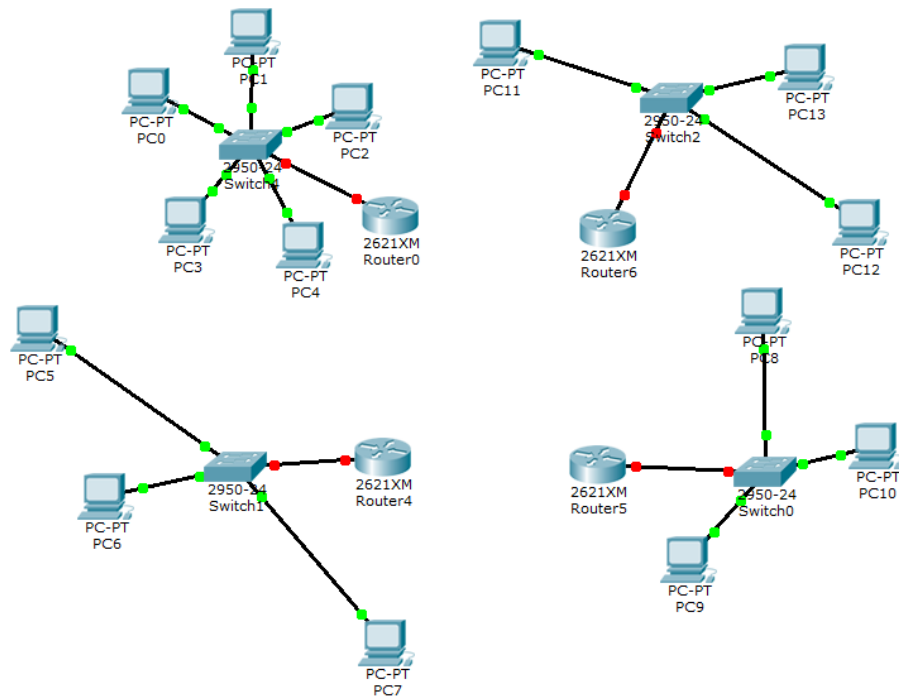
也是可以 ping 通的。

- 请把本部分的最后完整拓扑图记录在此。并标注每台 PC 的 IP 地址、子网掩码和属于的 VLAN。（将实验拓扑数据保存为 **part2.pkt** 文件，随实验报告一起打包上交）



-----Part 3. 路由-----

- 将第 5 个交换机删除，为每个局域网增加一个路由器（选择有 2 个以太网端口的，比如 2621 或通用型），并用网线将本局域网的路由器与交换机连接起来（记录下拓扑图）。



- 下面的 2 个步骤在第 1 个局域网中进行
 - 步骤 1：在路由器上与交换机连接的物理端口上创建 2 个逻辑子接口（命令格式：`interface 物理接口.子接口`，如 `interface e0/0.1`），2 个子接口的 IP 地址分别配置为 10.1.0.0/16 和 10.2.0.0/16 子网内的地址，让 2 个子接口分别属于 VLAN 1 和 VLAN 2。（本步骤截取实际使用的配置命令）

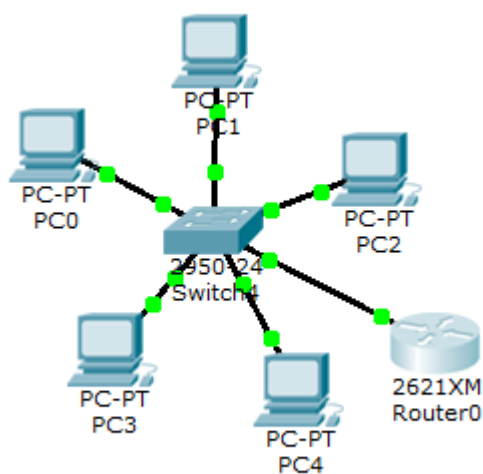
```

Router>enable
Router#configure t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int f0/0.1
Router(config-subif)#en
Router(config-subif)#encapsulation dot1Q 1
Router(config-subif)#ip add 10.1.0.10 255.255.0.0
Router(config-subif)#int f0/0.2
Router(config-subif)#en
Router(config-subif)#encapsulation dot1Q 2
Router(config-subif)#ip add 10.2.0.20 255.255.0.0
Router(config-subif)#no shutdown
Router(config-subif)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console

```

此外还有重要一步：将 port 的状态置为 on

FastEthernet0/0	
Port Status	<input checked="" type="checkbox"/> On
Bandwidth	<input checked="" type="checkbox"/> Auto
<input type="radio"/> 10 Mbps <input checked="" type="radio"/> 100 Mbps	
Duplex	<input checked="" type="checkbox"/> Auto
<input checked="" type="radio"/> Full Duplex <input type="radio"/> Half Duplex	
MAC Address	000A.4142.6201
IP Address	
Subnet Mask	
Tx Ring Limit	10



此时发现路由器和子网已经成功联通了。

- 步骤 2: 给 2 个 VLAN 内的 PC 机配置默认路由器/网关地址，分别设置为路由器上属于同一个 VLAN 的子接口 IP 地址。使用 Ping 检查属于不同 VLAN 的 PC 之间的联通性（应该通）。

PC0

IP Configuration

IP Configuration

DHCP

Static

IP Address

10.2.0.1

Subnet Mask

255.255.0.0

Default Gateway

10.2.0.20

DNS Server

0.0.0.0

IPv6 Configuration

DHCP

Auto Config

Static

IPv6 Address

/

Link Local Address

FE80::2D0:FFFF:FE52:4A25

IPv6 Gateway

IPv6 DNS Server

首先为 PC0 配置属于 10.2 网段的默认网关；

PC2

IP Configuration

IP Configuration

☐ DHCP ☒ Static

IP Address	10.1.0.3
Subnet Mask	255.255.0.0
Default Gateway	10.1.0.10
DNS Server	0.0.0.0

IPv6 Configuration

☐ DHCP ☐ Auto Config ☒ Static

IPv6 Address		/	
Link Local Address	FE80::20A:F3FF:FEEC:1EC4		
IPv6 Gateway			
IPv6 DNS Server			

然后为 PC2 配置属于 10.2 网段的默认网关；

然后尝试用 PC0 去 ping PC2：

```
C:\>ping 12.1.0.5

Pinging 12.1.0.5 with 32 bytes of data:

Reply from 12.1.0.5: bytes=32 time=11ms TTL=127
Reply from 12.1.0.5: bytes=32 time=15ms TTL=127
Reply from 12.1.0.5: bytes=32 time=12ms TTL=127
Reply from 12.1.0.5: bytes=32 time=37ms TTL=127

Ping statistics for 12.1.0.5:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 11ms, Maximum = 37ms, Average = 18ms

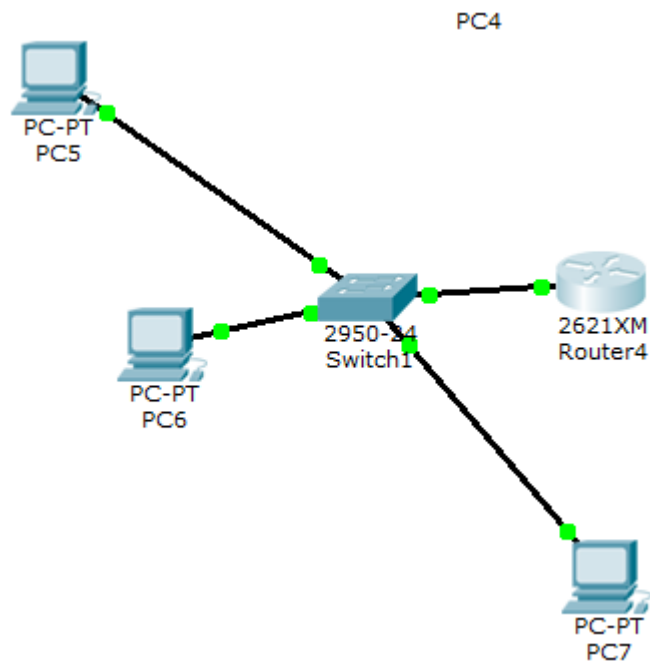
C:\>
```

发现可以 ping 通。

- 修改第 2 个局域网中 PC 的 IP 地址，把 10.1.0.0/16、10.2.0.0/16 子网内的 IP 地址分别改成 12.1.0.0/16、12.2.0.0/16 子网内的 IP 地址。然后按照第 1 个局域网的 2 个步骤，给路由器分配 IP 地址，给 PC 配置默认路由器/网关地址。用 Ping 检查不同 VLAN 的 PC 之间联通性。

如图。修改 IP 地址如下：

```
Router>enable
Router#conf
Router#configure t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int f0/0.1
Router(config-subif)#enca
Router(config-subif)#encapsulation dot1Q 1
Router(config-subif)#ip add 12.1.0.10 255.255.0.0
Router(config-subif)#int f0/0.2
Router(config-subif)#encs
Router(config-subif)#enca
Router(config-subif)#encapsulation dot1Q 2
Router(config-subif)#ip add 12.2.0.10 255.255.0.0
Router(config-subif)#no shutdown
Router(config-subif)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console
```



<input type="radio"/> DHCP <input checked="" type="radio"/> Static	
IP Address	12.2.0.1
Subnet Mask	255.255.0.0
Default Gateway	12.2.0.10
DNS Server	0.0.0.0
IPv6 Configuration	
<input type="radio"/> DHCP <input type="radio"/> Auto Config <input checked="" type="radio"/> Static	
IPv6 Address	
Link Local Address	FE80::209:7CFF:FE49:863B
IPv6 Gateway	
IPv6 DNS Server	

<input type="radio"/> DHCP <input checked="" type="radio"/> Static	
IP Address	12.1.0.4
Subnet Mask	255.255.0.0
Default Gateway	12.1.0.10
DNS Server	0.0.0.0

```
Pinging 12.1.0.5 with 32 bytes of data:

Reply from 12.1.0.5: bytes=32 time=11ms TTL=127
Reply from 12.1.0.5: bytes=32 time=15ms TTL=127
Reply from 12.1.0.5: bytes=32 time=12ms TTL=127
Reply from 12.1.0.5: bytes=32 time=37ms TTL=127
```

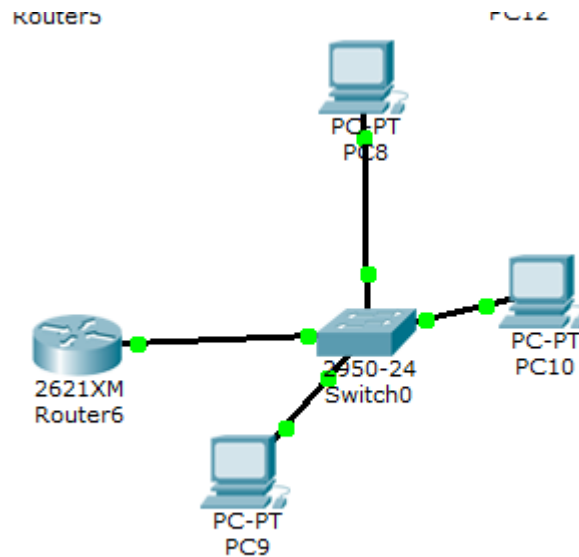
发现 PC5 和 PC7 之间的连通性还是很好的。

- 修改第 3 个局域网中 PC 的 IP 地址，把 10.1.0.0/16、10.2.0.0/16 子网内的 IP 地址分别改成 13.1.0.0/16、13.2.0.0/16 子网内的 IP 地址。然后按照第 1 个局域网的 2 个步骤，给路由器分配 IP 地址，给 PC 配置默认路由器/网关地址。用 Ping 检查不同 VLAN 的 PC 之间联通性。

```

Router>enable
Router#configure t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int f0/0.1
Router(config-subif)#ens
Router(config-subif)#enc
Router(config-subif)#encapsulation dot
Router(config-subif)#encapsulation dot1Q 1
Router(config-subif)#ip add 13.1.0.10 255.255.0.0
Router(config-subif)#int f0/0.2
Router(config-subif)#enc
Router(config-subif)#encapsulation dot
Router(config-subif)#encapsulation dot1Q 2
Router(config-subif)#ip add 13.2.0.10 255.255.0.0
Router(config-subif)#no shutdown
Router(config-subif)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console

```



<input type="radio"/> DHCP	<input checked="" type="radio"/> Static
IP Address	13.1.0.7
Subnet Mask	255.255.0.0
Default Gateway	13.1.0.10
DNS Server	0.0.0.0

测试:


```

C:\>ping 13.1.0.6

Pinging 13.1.0.6 with 32 bytes of data:

Reply from 13.1.0.6: bytes=32 time=11ms TTL=127
Reply from 13.1.0.6: bytes=32 time=11ms TTL=127
Reply from 13.1.0.6: bytes=32 time=11ms TTL=127
Reply from 13.1.0.6: bytes=32 time=11ms TTL=127

Ping statistics for 13.1.0.6:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 11ms, Maximum = 11ms, Average = 11ms

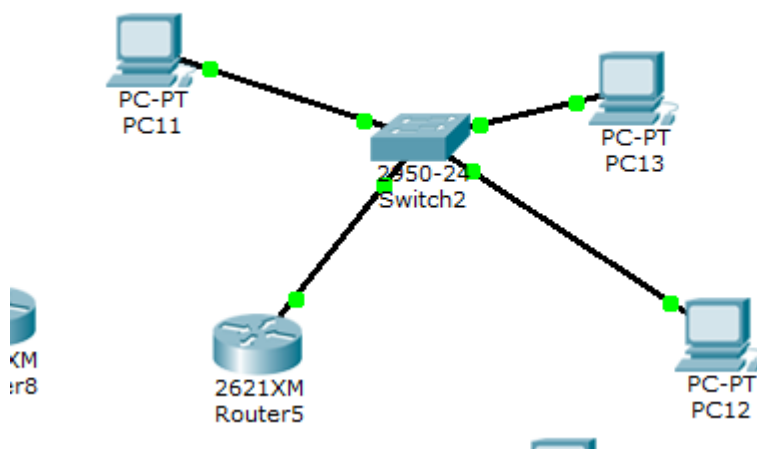
```

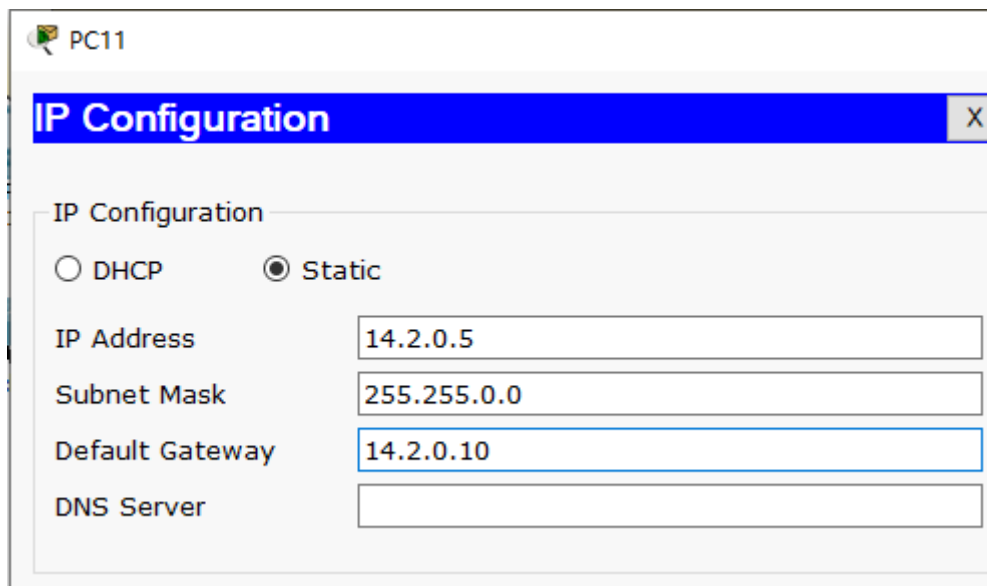
- 修改第 4 个局域网中 PC 的 IP 地址，把 10.1.0.0/16、10.2.0.0/16 子网内的 IP 地址分别改成 14.1.0.0/16、14.2.0.0/16 子网内的 IP 地址。然后按照第 1 个局域网的 2 个步骤，给路由器分配 IP 地址，给 PC 配置默认路由器/网关地址。用 Ping 检查不同 VLAN 的 PC 之间联通性。

```

Router>enable
Router#conf
Router#configure t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int f0/0.1
Router(config-subif)#enc
Router(config-subif)#encapsulation do
Router(config-subif)#encapsulation dot1Q 1
Router(config-subif)#ip add 14.1.0.10 255.255.0.0
Router(config-subif)#int f0/0.2
Router(config-subif)#enc
Router(config-subif)#encapsulation do
Router(config-subif)#encapsulation dot1Q 2
Router(config-subif)#ip add 14.2.0.10 255.255.0.0
Router(config-subif)#no shutdown
Router(config-subif)#exit
Router(config)#

```





测试:

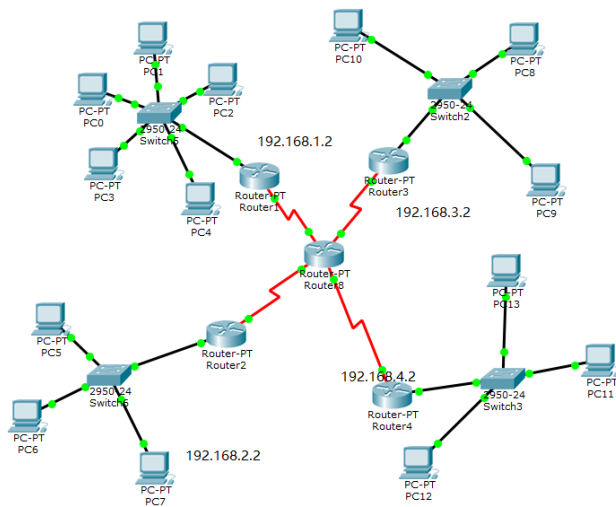
```
C:\>ping 14.1.0.8

Pinging 14.1.0.8 with 32 bytes of data:

Reply from 14.1.0.8: bytes=32 time=14ms TTL=127
Reply from 14.1.0.8: bytes=32 time=11ms TTL=127
Reply from 14.1.0.8: bytes=32 time=11ms TTL=127
Reply from 14.1.0.8: bytes=32 time=8ms TTL=127

Ping statistics for 14.1.0.8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 8ms, Maximum = 14ms, Average = 11ms
```

- 使用第 5 台路由器（选择通用型）分别连接 4 个局域网的路由器，并给路由器互联的 4 对端口分别配置以下子网内的 IP 地址：192.168.1.0/24、192.168.2.0/24、192.168.3.0/24、192.168.4.0/24。（记录下拓扑图，并标记 4 对端口的 IP 地址和子网掩码）



- 启用各路由器上的动态路由协议 RIP（命令：router rip），将本路由器上的各端口所在子网加入到路由信息交换（命令：network 子网，如 network 10.1.0.0）。等待一段时间后，使用 Ping 命令测试下各 PC 之间的连通性。截图显示各路由器上的路由表信息（标记哪些路由是直连的，哪些是通过路由协议动态获取的）。

首先激活 RIP，首先对 PC0 和 PC11 的连通性进行测试

```
Pinging 14.2.0.5 with 32 bytes of data:
Reply from 14.2.0.5: bytes=32 time=75ms TTL=125
Reply from 14.2.0.5: bytes=32 time=12ms TTL=125
Reply from 14.2.0.5: bytes=32 time=15ms TTL=125
Reply from 14.2.0.5: bytes=32 time=14ms TTL=125
```

对 PC0 和 PC10 的连通性进行测试

```
Pinging 13.1.0.6 with 32 bytes of data:
Reply from 13.1.0.6: bytes=32 time=56ms TTL=125
Reply from 13.1.0.6: bytes=32 time=16ms TTL=125
Reply from 13.1.0.6: bytes=32 time=13ms TTL=125
Reply from 13.1.0.6: bytes=32 time=16ms TTL=125
```

对 PC0 和 PC5 的连通性进行测试

```
Pinging 12.2.0.1 with 32 bytes of data:
Reply from 12.2.0.1: bytes=32 time=14ms TTL=125
Reply from 12.2.0.1: bytes=32 time=15ms TTL=125
Reply from 12.2.0.1: bytes=32 time=13ms TTL=125
Reply from 12.2.0.1: bytes=32 time=13ms TTL=125
```

此时³⁴ 考察第一个路由器的路由表,可以注意到 10.1.0.0、10.2.0.0 和 192.168.0.0 是直接连接, 其他为通过动态方式获取的:

考察第二个路由器的路由表,可以看到 12.1.0.0、12.2.0.0 和 192.168.2.0 为直连, 其他为动态获取:

考察第三个路由器的路由表,可以看到 13.1.0.0、13.2.0.0 和 192.168.3.0 为直连, 其他为动态获取。

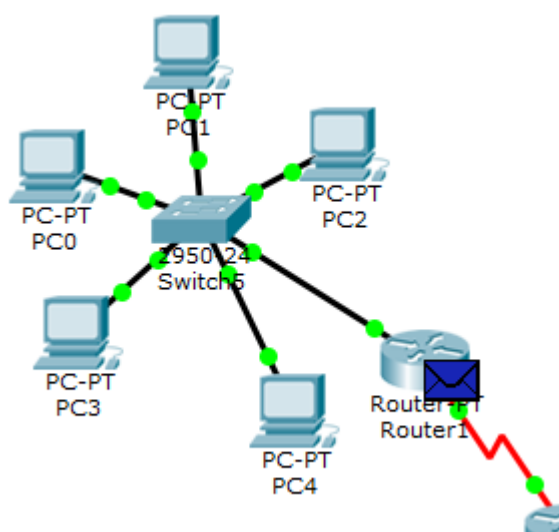
考察第四个路由器的路由表,可以看到 14.1.0.0、14.2.0.0 和 192.168.4.0 为直连, 其他为动态获取。

- 关闭各路由器上的动态路由协议 RIP (命令: `no router rip`)。等待一段时间后, 显示各路由器上的路由表信息 (动态获取的路由信息是否消失了?)。再次使用 Ping 命令测试下各 PC 之间的联通性。哪些通? 哪些不通? 针对不通的数据包, 跟踪下在哪个环节不再转发了 (是不是在该路由器上缺少对应的路由信息?)。

动态路由协议的关闭会直接导致动态的路由信息丢失, 只剩下直连的信息。

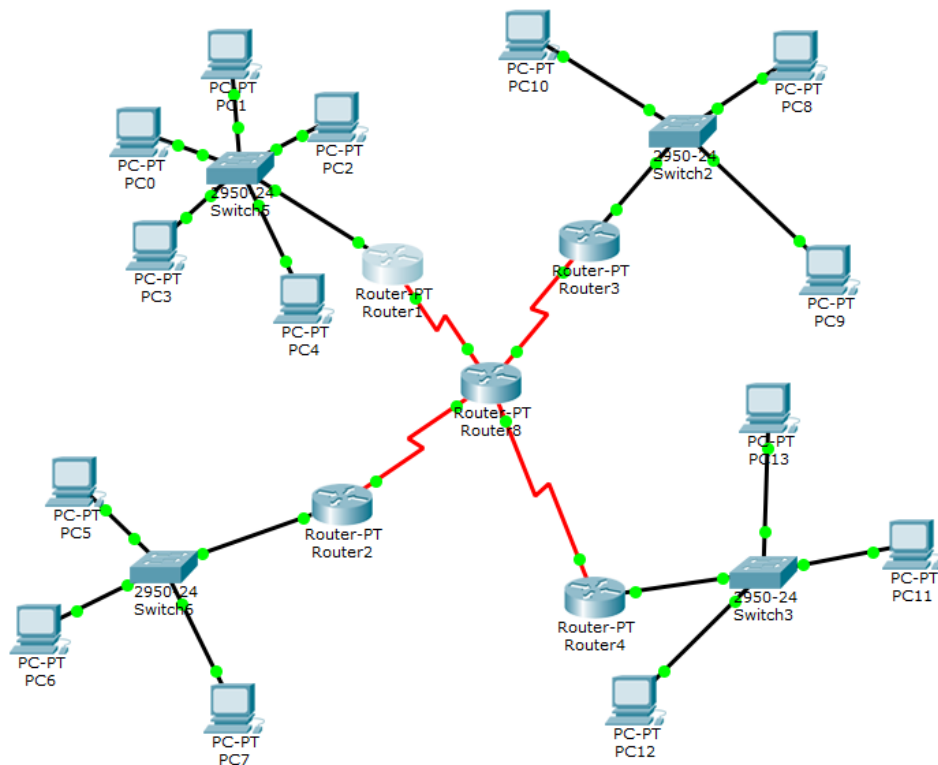
这时, 如 PC0 与 PC1 同属于 VLAN2, PC0 能 ping 通 PC1:

此外, PC0 与 PC3 位于同一局域网内, 即便 PC0 属于 VLAN2, PC3 属于 VLAN1, PC0 还是能 ping 通 PC3 的。但是, 如果 P0 向不通的局域网中的不通 VLAN 下的 PC 发出 ping 指令, 都无法 ping 通。这是由于 RIP 服务关闭导致依赖动态路由的连接断开。



- 请把本部分的最后完整拓扑图记录在此。并标注每台 PC 的 IP 地址、子网掩码和属于的 VLAN, 以及每个路由器的端口分配的 IP 地址、子网掩码。(将实验拓扑数

据保存为 part3.pkt 文件，随实验报告一起打包上交)



六、 实验结果与分析

- 在同一个局域网内的 2 台 PC 机，IP 地址分别为 10.0.0.1/8 和 10.1.0.1/8，都属于 VLAN1，一开始可以互相 Ping 通，为什么把子网掩码长度从 8 位变成 16 位，就不通了？

这是因为子网掩码变化对子网的网络地址解析错误，从而找不到主机。

- 仅使用二层交换机的情况下，同一个局域网内，属于不同 VLAN 的 PC 之间为何不能 Ping 通呢？

VLAN 有隔离广播流量的作用，它能够对二层网络进行分段，每个分段一个广播域。

如果不在一个 VLAN 中，则本机向交换机发出的 ARP 广播包中请求的网关的

MAC 地址。但二层交换机中并没有网关，所以连接无法建立。

- 交换机的端口设置为 VLAN Trunk 模式后，在通过该端口转发数据包时，交换机会插入什么信息，使得对方交换机能够将数据包转发到正确的 VLAN？

当交换机端口设置为 VLANTrunk 模式后，该端口收到数据包后，会判断其有没有 VLAN 信息，若没有，则打上端口的 PVID，并进入交换转发；如果有，则判断该 trunk 端口是否允许该 VLAN 的数据进入。如果可以进，则转发消息，否则丢弃。要发送数据时，先比较端口的 PVID 和将要发送的数据的 VLAN 信息。

如果两者相等，则剥离 VLAN 信息并发送；如果不相等则直接发送。

- 为了让不同局域网的 PC 之间能够互相 Ping 通，在设置静态路由时，所有路由器之间互联的子网是否一定要全部加入到第 1-4 个局域网路由器的路由表中？

不需要。一方面不现实，一方面，如果相应的信息没有得到很好的记录，可以将数据包发送到已知路由中，再利用该路由中更充足的信息来导航数据包。

七、 讨论、心得

在本次实验中，我通过学习 packet tracer 的使用，复习了课堂上学习的网络层相关知识，熟悉了路由器等网络设备的作用机理，受益良多。

本次实验是本学期我遇到的最为困难的实验。主要挑战有以下几点：

1. 题量巨大，涉及大量截图、修图等冗繁操作，严重拖慢了实验进度；直到最后我也有许多题目做得不甚完美；
2. 软件版本不同导致 bug；
3. 学习成本较大，周期长；

本次实验极大地锻炼了我的能力，但也不得不指出，实验的量实在是太大了。我们专业大部分同学，即使最终完成了实验，也大多花费了超过两天、乃至三天的时间。而我由于平常做实验比较缓慢，后面一部分实验由于各种 bug 没能很好的完成，最后一部分不得不利用其他同学搭好的拓扑进行测试（我想我应当承认这一点），这令我感到无比后悔与遗憾。