**计算机网络技术实践**

**实验报告**



**实验名称： 实验四—VLAN组网配置实验**

**姓 名： 陈朴炎 实 验 日 期：2023.11.19**

**学 号：2021211138 实验报告日期：2023.12.19**

**报告退发： （ 订正、 重做 ）**

目录

[一、 环境 3](#_Toc153915869)

[1.1 运行环境 3](#_Toc153915870)

[1.2 网络拓扑 3](#_Toc153915871)

[1.2.1 网络拓扑图 3](#_Toc153915872)

[1.2.2 网络拓扑说明 4](#_Toc153915873)

[二、实验目的 4](#_Toc153915874)

[三、实验内容及步骤 4](#_Toc153915875)

[3.1 配置PC 4](#_Toc153915876)

[3.2 配置交换机及前两部分实验 5](#_Toc153915877)

[3.2.1 配置交换机 5](#_Toc153915878)

[3.2.2 实验四第一部分 9](#_Toc153915879)

[3.2.3 实验四第二部分 10](#_Toc153915880)

[3.3 配置路由器 11](#_Toc153915881)

[3.3.1 方法1 和access口相连 11](#_Toc153915882)

[3.3.2 方法2 和trunk口相连 13](#_Toc153915883)

[3.3.3 配置路由器之间的动态路由 15](#_Toc153915884)

[四、实验结果 18](#_Toc153915885)

[4.1 第一部分 18](#_Toc153915886)

[4.2 第二部分 20](#_Toc153915887)

[4.3 第三部分 不同VLAN互联 22](#_Toc153915888)

[4.3.1 方法1 22](#_Toc153915889)

[4.3.2 方法2 24](#_Toc153915890)

[4.3.3 跨路由器互联 26](#_Toc153915891)

[五、实验中的问题及心得 27](#_Toc153915892)

[六、实验思考 30](#_Toc153915893)

[6.1 同个局域网配置不同IP网段 30](#_Toc153915894)

[6.2 分析数据包传输流程 31](#_Toc153915895)

[6.3 物理网络、VLAN、IP网段的关系 33](#_Toc153915896)

# 一、 环境

## 1.1 运行环境

本次实验在Windows 11 上完成，使用GNS3虚拟仿真平台。

GNS3版本：22.44.1

GNS3的服务器运行在虚拟机上，使用GNS3 VM.ova作为映像。

运行虚拟机的程序为VM Ware workstation pro 17.x

VM Version：0.15.0

Qemu version 4.2.1

Ubuntu version focal

KVM support availabe: True

IP：192.168.177.127 port：80

## 1.2 网络拓扑

### 1.2.1 网络拓扑图

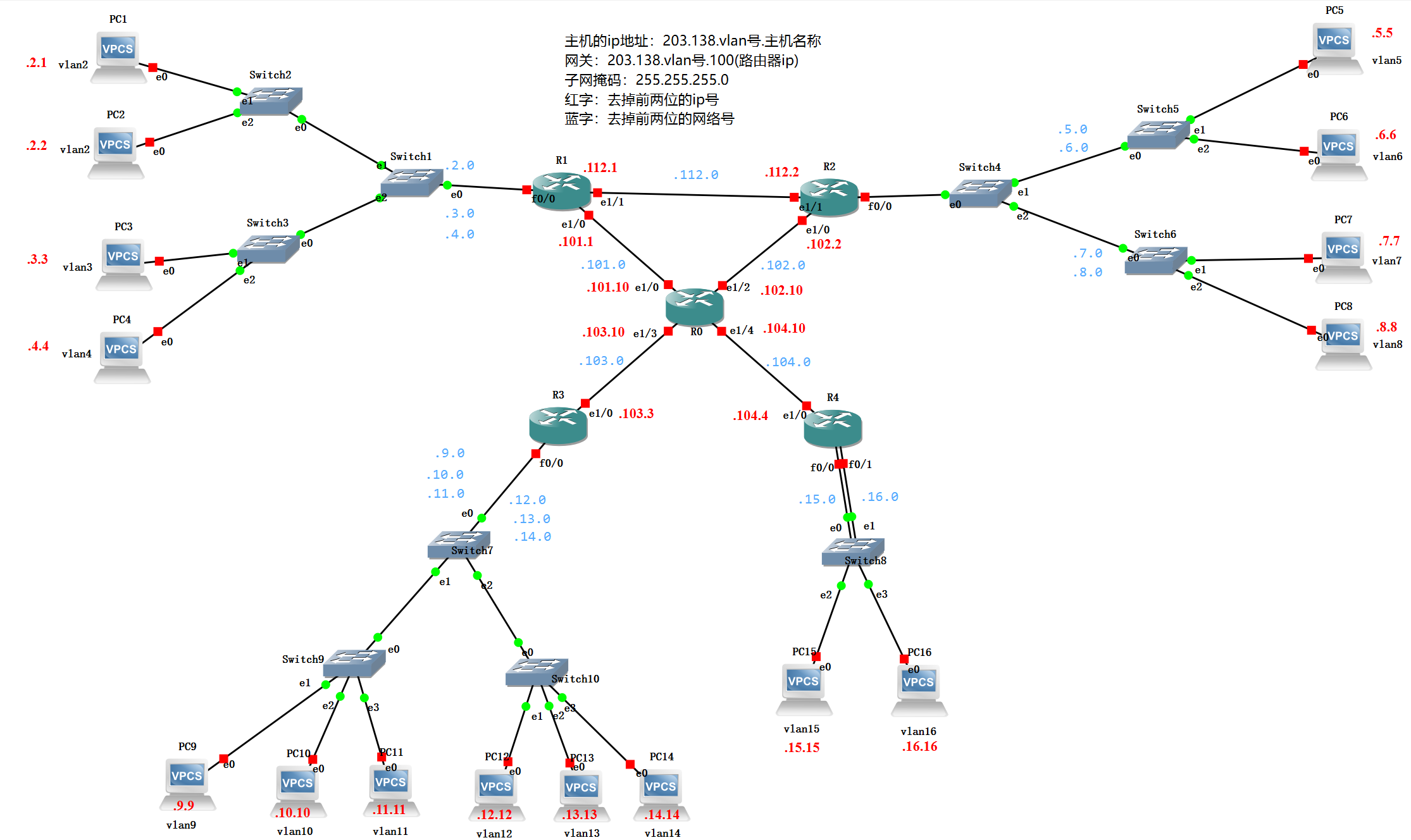


图1-1 网络拓扑图

### 1.2.2 网络拓扑说明

网络拓扑中有5个路由器，10个交换机，16个PC机，15个VLAN。

整个网络拓扑的子网掩码都是255.255.255.0

网络号的前两位都为203.138，在网络拓扑图中，用蓝色的数字标明。

网络拓扑图中，红色的为各个主机、端口的IP号后两位。

其中，主机的IP号为203.138.vlan号.主机号。vlan号标注在了PC的旁边，比如：PC2的vlan号为2，PC1的vlan号也为2。PC1的主机号就为1，PC2的主机号就为2以此类推。

路由器互联部分的网络号为203.138.“1RnRm”.0，比如说，R1和R2相连的那部分网络号就为203.138.112.0，第三位的后两位为相连的路由器的标号。

路由器和PC连的端口为各个子网的网关，IP号的最后一位为100。

# 二、实验目的

能够清楚的知道不同VLAN配置的区别和原理

能够说明一个数据包从一台主机到另一台主机的发送传输流程

能够设计网络拓扑，配置各台机器，并令网络联通

# 三、实验内容及步骤

## 3.1 配置PC

双击PC，对PC输入如下命令，配置PC的ip：

ip 203.138.vlan号.PC号/24 203.138.vlan号.100

比如PC1，就输入 ip 203.138.2.1/24 203.138.2.100

203.138.2.1为PC的IP号，/24为子网掩码的位数，203.138.2.100为这个网段的网关。如下图所示

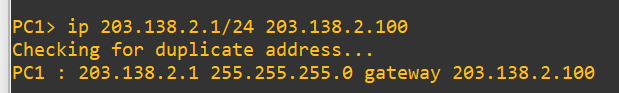


图3-1 配置PC的ip图

同理，配置其他的16个PC机，过程都相同，不再重复罗列。

配置完后，使用save命令保存PC的配置，如下图所示：

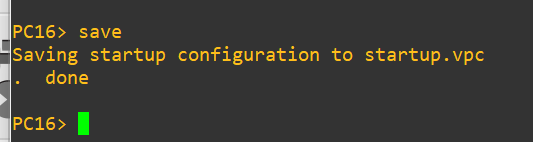


图3-2 保存PC配置

## 3.2 配置交换机及前两部分实验

### 3.2.1 配置交换机

双击交换机，出现如下配置界面

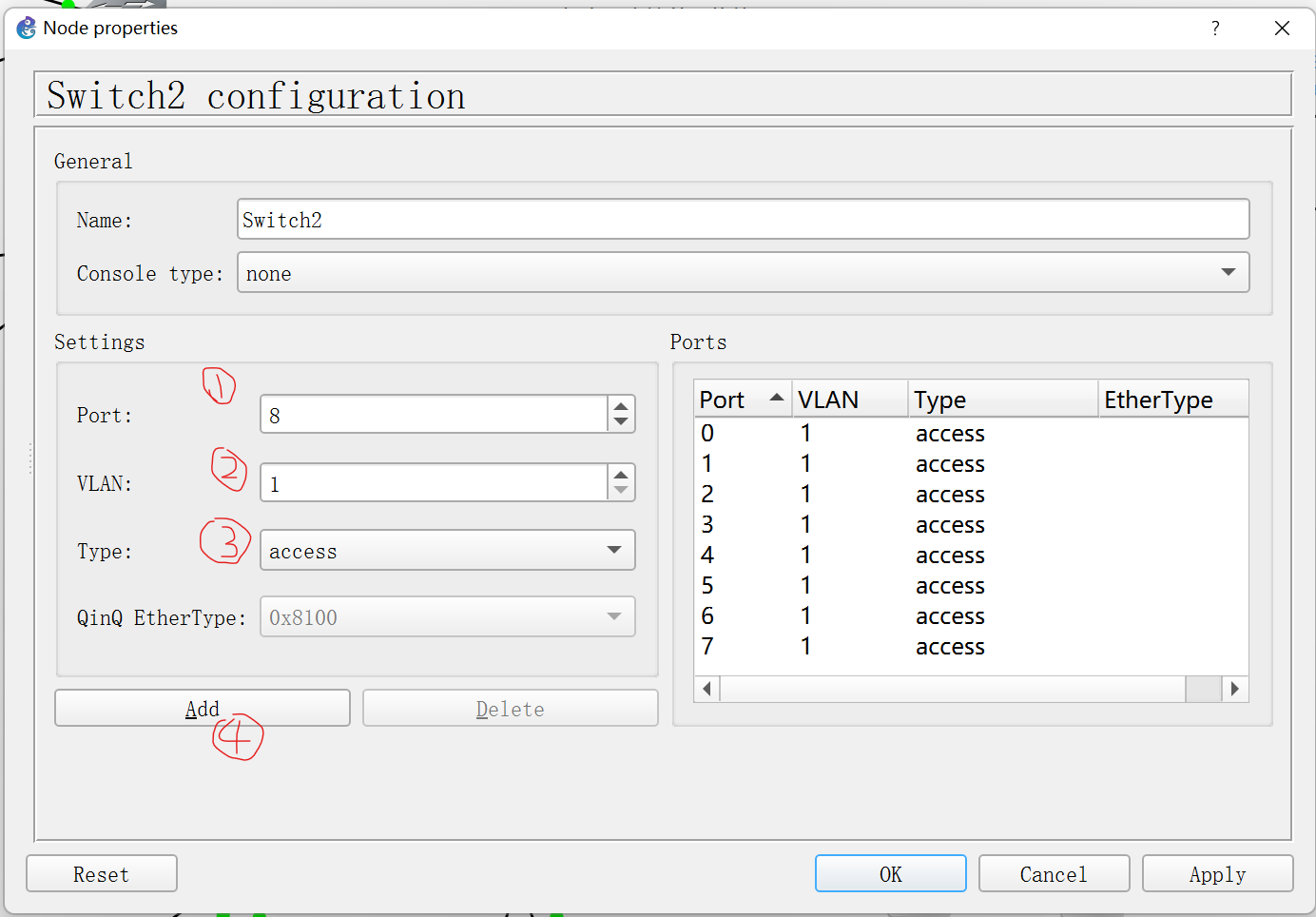


图3-3 交换机配置界面

首先选择端口号，以Switch2为例，Switch2的e1端口连PC1，e2端口连PC2，e0端口连Switch1。在这里，我把交换机和PC连接的端口类型都设置为access口，而交换机跟交换机/路由器连的端口类型都设置为trunk口（除了Switch8和R4）。配置交换机的步骤如下：

① 选择端口

② 更改VLAN号

③ 选择端口的类型

④ 点击Add

⑤ 点击Apply 🡪 OK

前四步做完之后，Switch2的配置结果如下，Type为dot1q表示该端口类型为trunk口。我把0号端口设置为trunk，1、2号端口设置为access口，并且1、2端口的VLAN号为2。trunk口的vlan号最好设置成1，不然会出现问题（后面会讨论）。

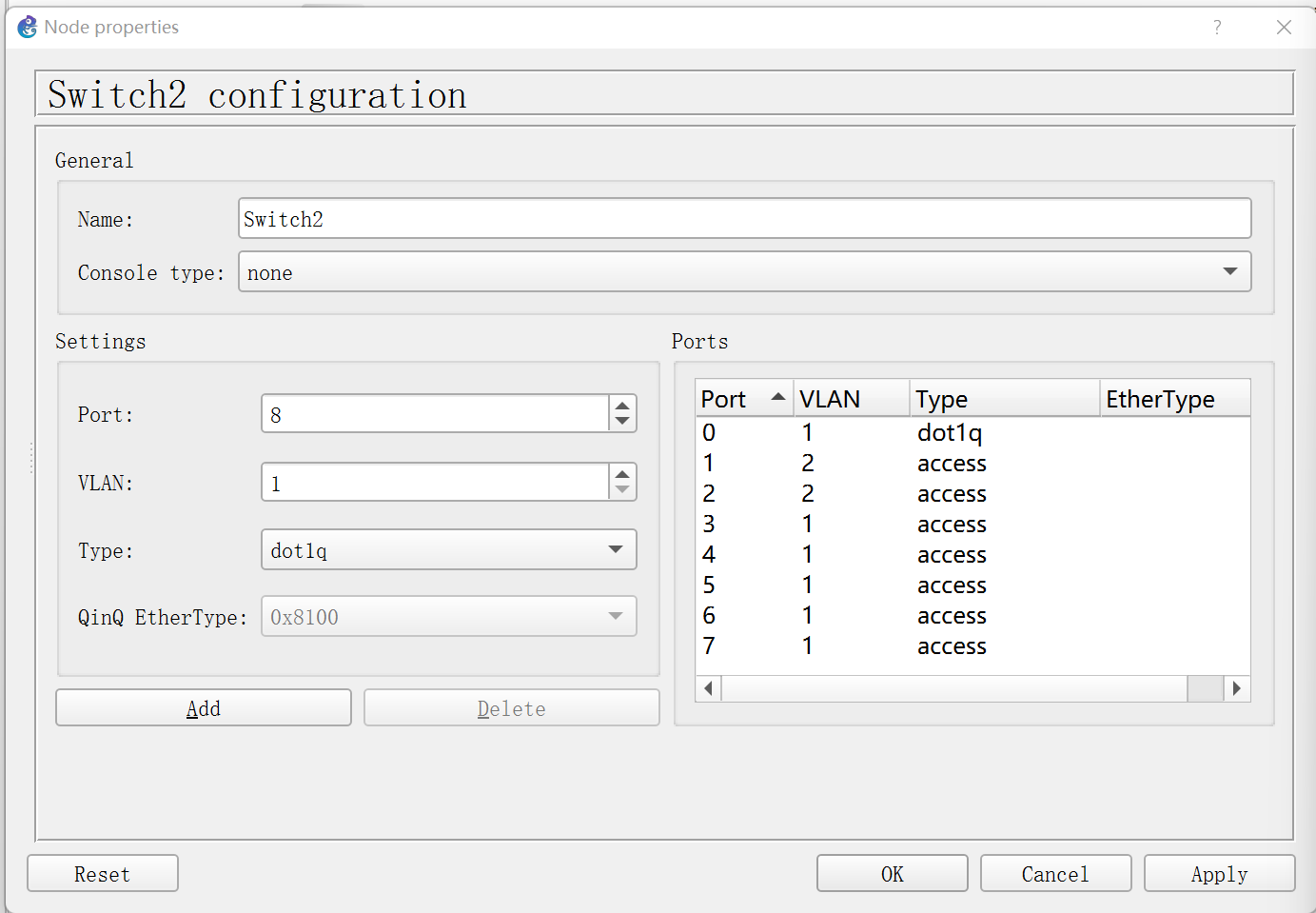


图3-4 Switch2配置图

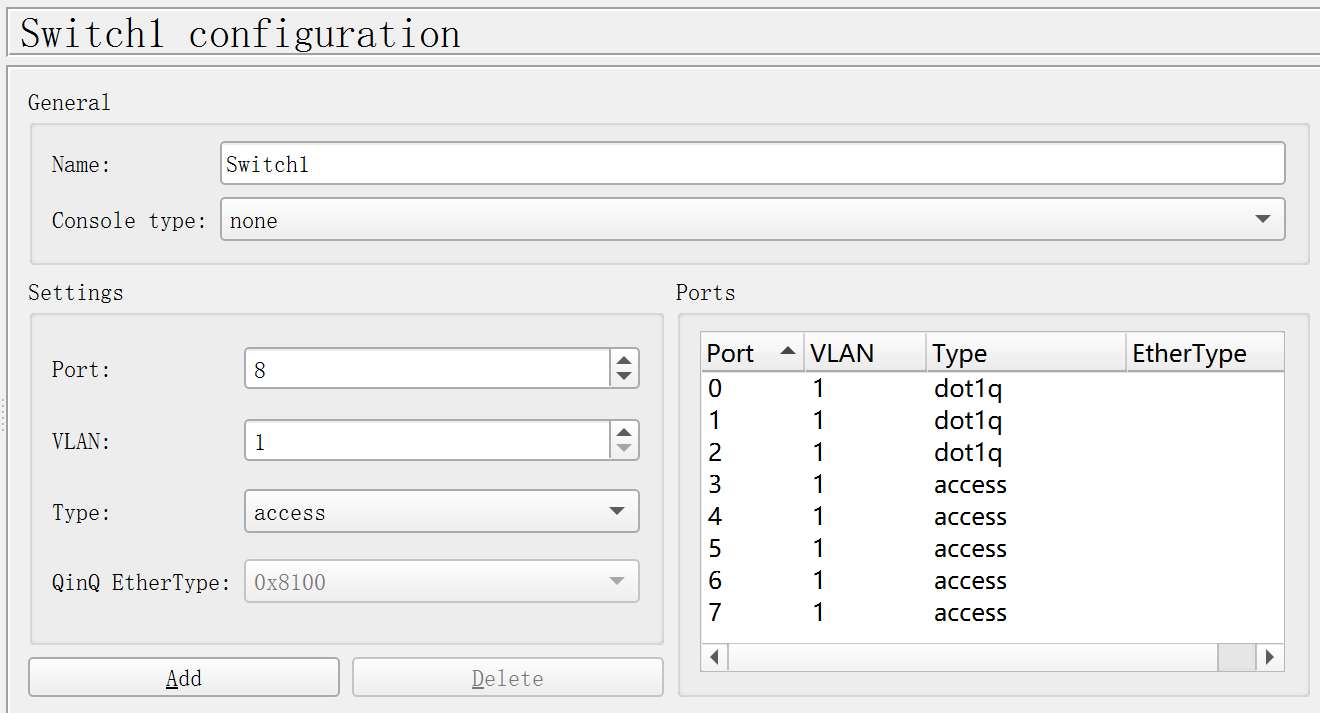


图3-5 Switch1 配置图

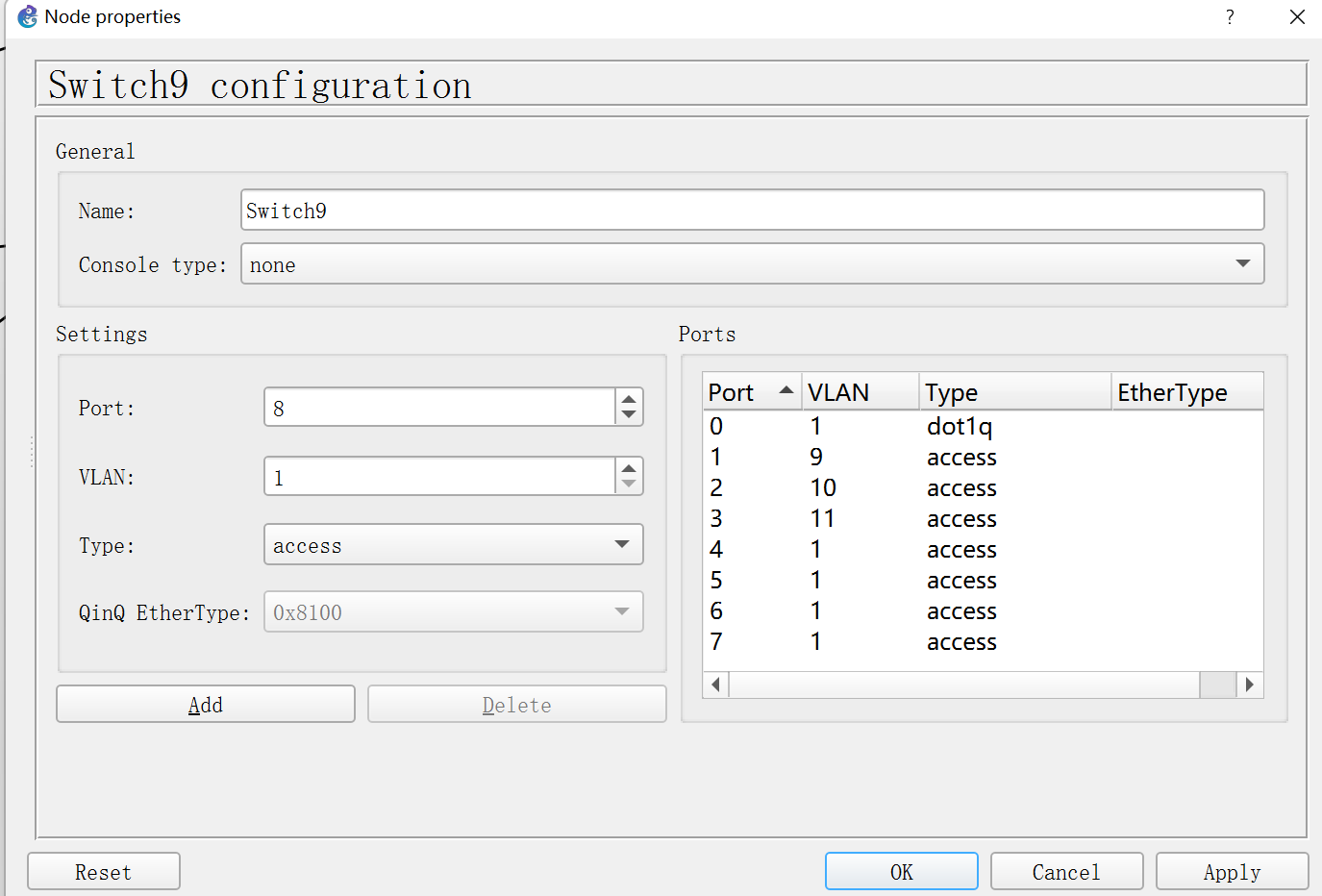


图3-6 Switch9配置图

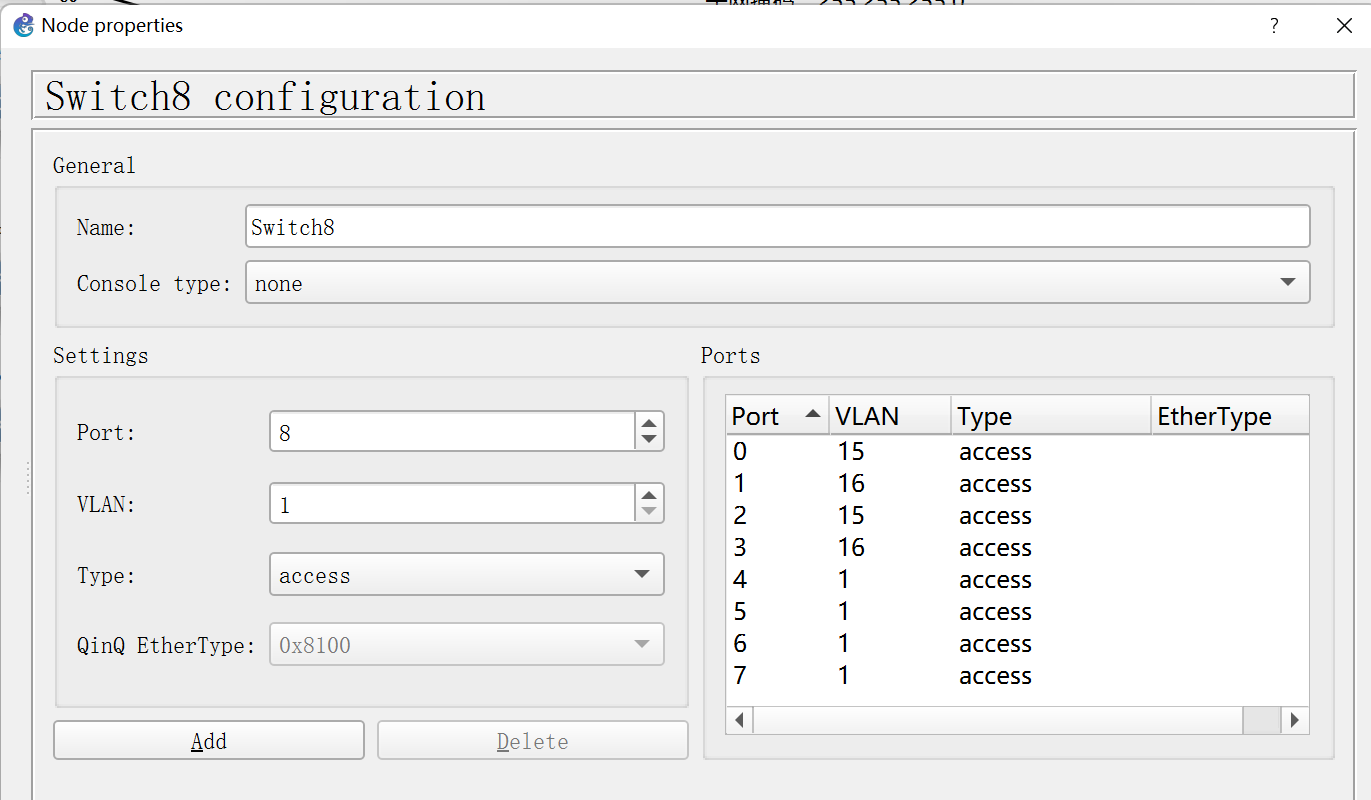


图3-7 Switch8配置图

以上是几个在我的网络拓扑图中重要/典型的交换机配置，其他交换机配置和它们类似。

### 3.2.2 实验四第一部分

现在做实验4的第一部分：

启动Switch1、2，PC1、2。PC1的IP配置[同3.1.1节所示](#_3.1_配置PC)，PC2的ip配置为203.138.2.2。将Switch2的端口e1设置为vlan2，e2设置为vlan3，如下图所示：

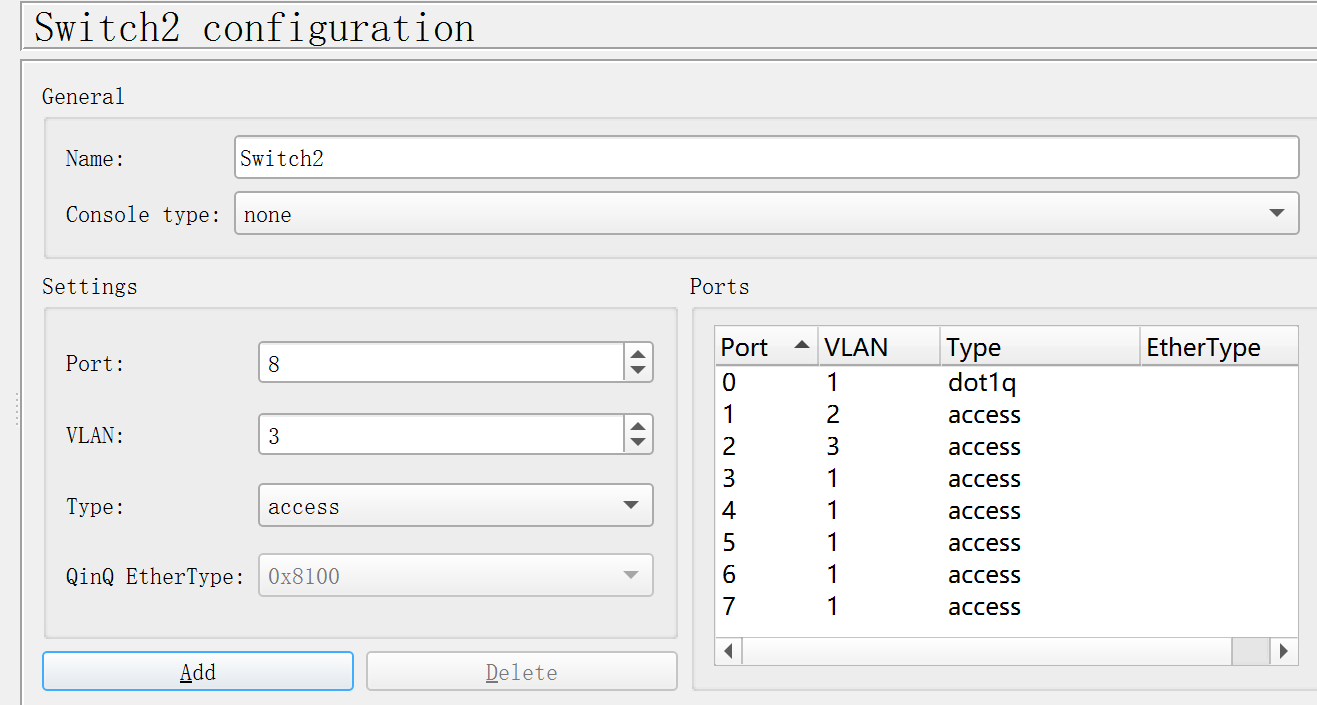


图3-8 第一部分Switch2配置1

查看PC1和PC2能否ping通：

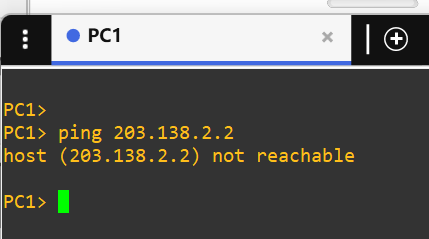


图3-9 PC1 ping PC2

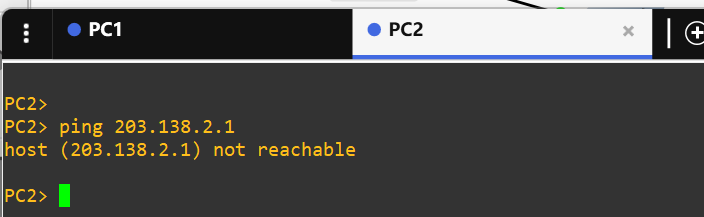


图3-10 PC2 ping PC1

不能ping通。

将SWitch2的e2端口配置成vlan2，查看能否ping通：

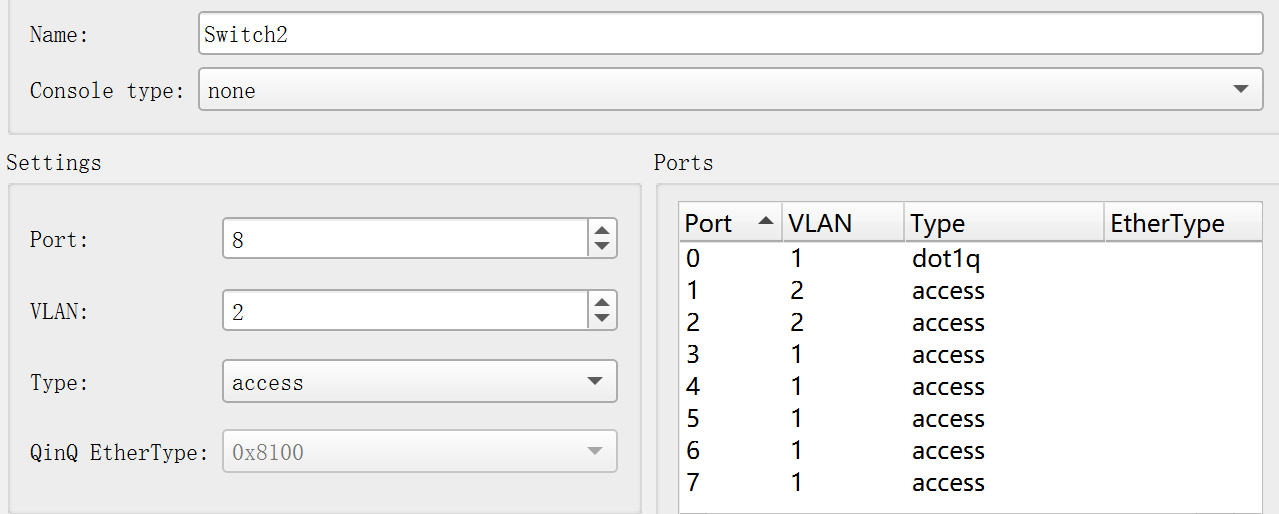


图3-11 修改Switch2配置

重新查看PC1和PC2能否ping通。实验结果在[4.1第一部分](#_4.1_第一部分)。

### 3.2.3 实验四第二部分

用拓扑图的左上角部分PC1、PC3、SWitch1、Switch2、Switch3进行实验。

将Switch3配置信息更改，e1端口配置成vlan2，如下：

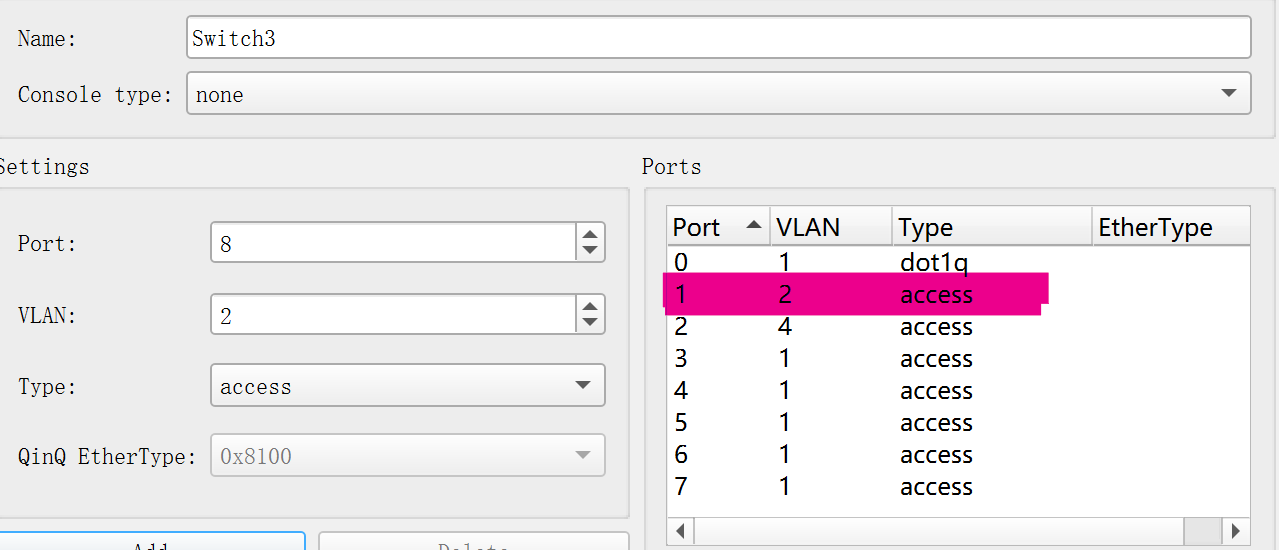


图3-12 修改Switch3端口标签

修改PC3的ip号为203.138.2.3，配置如下：



图3-13 PC3的ip配置图

而SWitch1、2、3相连的部分都为trunk口。现在尝试PC1 、PC3 互相ping，实验结果及分析在[4.2 第二部分](#_4.2_第二部分)。

## 3.3 配置路由器

路由器的端口有两种配置方法，一种是一个端口一个vlan号，另一种是一个端口划分成多个子端口，一个子端口一个vlan号。下面我将详细说明两种配置方法的实验步骤。

### 3.3.1 方法1 和access口相连

由于交换机的access口只允许同一个vlan标签的数据包传输，因此需要为路由器连接的每一个vlan配置一个端口。

配置一个端口一个vlan号的路由器，步骤如下。

在拓扑图中，我使用R4作为方法1的路由器。R4的f0/0为PC15的网关，R4的f0/1为PC16的网关。

首先双击R4路由器，进入配置界面。

输入如下配置信息：

|  |
| --- |
| R4# conf t  R4(config)# in f0/0  R4(config-if)# ip add 203.138.15.100 255.255.255.0  R4(config-if)# no shutdown  R4(config-if)# exit |

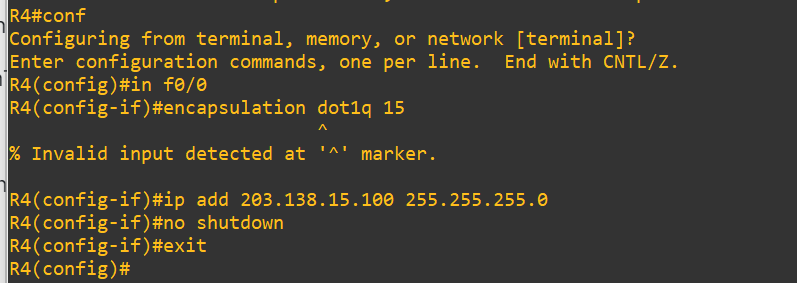


图3-14 配置R4 f0/0端口示意图

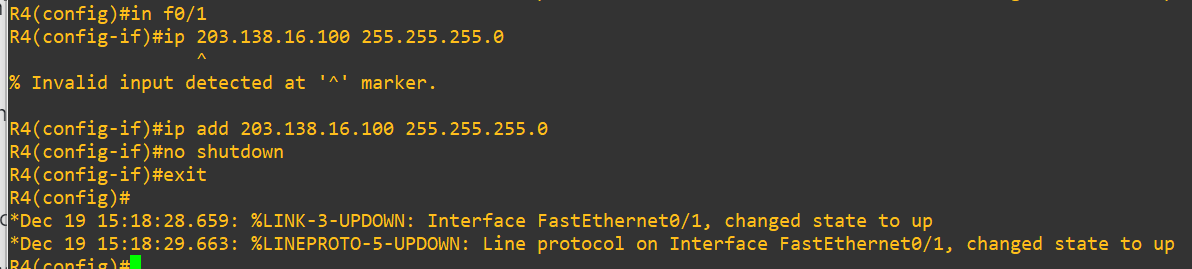


图3-15 配置R4 f0/1端口示意图

查看R4此时的路由表，如下：

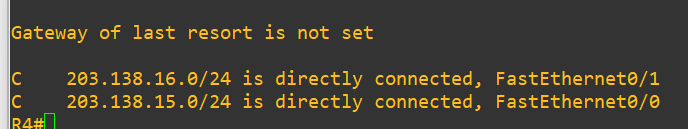


图3-16 R4路由表

配置好后，尝试PC15和PC16互通，实验结果在[4.3.1 方法1](#_4.3.1_方法1)。

### 3.3.2 方法2 和trunk口相连

当路由器的一个端口和交换机的trunk口相连的时候，路由器需要为这个端口划分多个子端口，来让它的一个端口作为不同虚拟局域网的网关。配置过程如下所示，这里我选择R1为例，R1的f0/0连接了vlan2、vlan3、vlan4，因此需要为R1划分三个子端口：f0/0.2, f0/0.3, f0/0.4，并为每一个子端口配置ip地址和协议。步骤如下：

|  |
| --- |
| **R1#conf t**  **Enter configuration commands, one per line. End with CNTL/Z.**  **R1(config)#in f0/0.2**  **R1(config-subif)#enc**  **R1(config-subif)#encapsulation dot1q 2**  **R1(config-subif)#ip add 203.138.2.100 255.255.255.0**  R1(config-subif)#exit |

conf t表示进入配置环境，in f0/0.2 表示进入到接口f0/0的.2子接口配置环境。encapsulation dot1q 2表示为该子接口封装上dot1q协议，其中标签为2。最后加上该子接口的ip地址和子网掩码。no shutdown用来开启该接口。

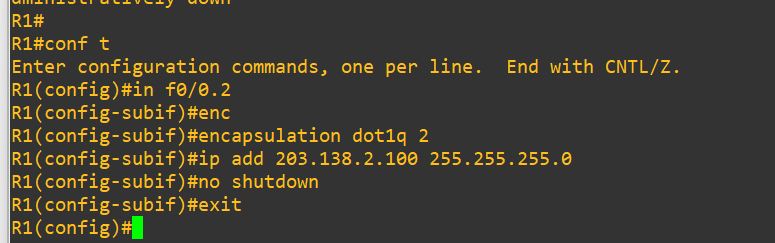


图3-17 R1的f0/0.2配置过程

同理，配置R1的f0/0.3、f0/0.4，过程如下：

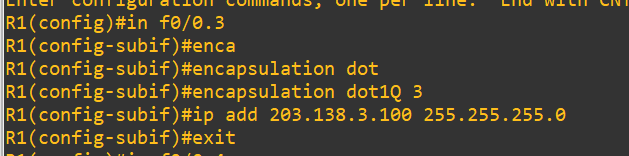


图4-18 R1的f0/0.3配置过程

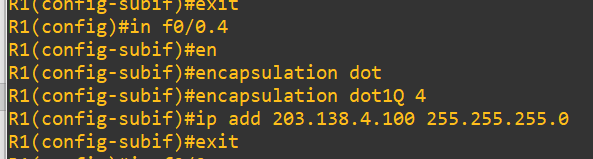


图4-19 R1的f0/0.4配置过程

最后，要进入子接口对应的物理接口，把该端口打开，如下：

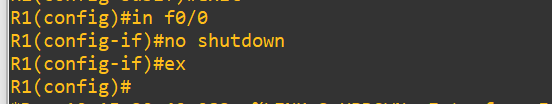


图4-20 打开R1的f0/0端口

检查配置是否成功：用PC3去ping PC1、PC4

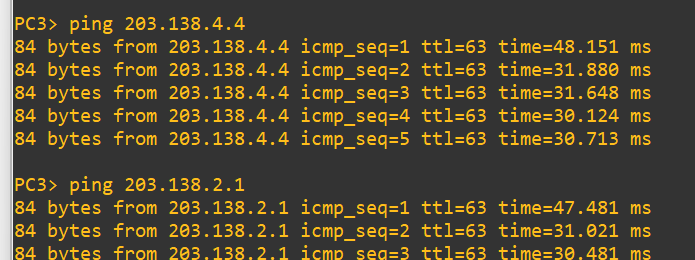


图4-21 PC3 ping PC4、PC1测试结果

结果显示，路由器f0/0接口配置成功。查看R1的路由表，如下：

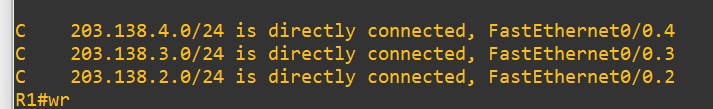


图4-22 R1路由表1

按R1的方法配置R2、R3，过程重复，不再罗列。

这里测试使用PC2 和 PC4，查看它们之间能否互联。实验结果请看[4.3.2 不同vlan互联——方法2实验结果](#_4.3.2_方法2)

### 3.3.3 配置路由器之间的动态路由

路由器之间需要动态路由协议，这样才能让不同路由器连的不同vlan的PC机互联。这里我选择使用ospf协议，配置过程如下：

同样的，以R1为例

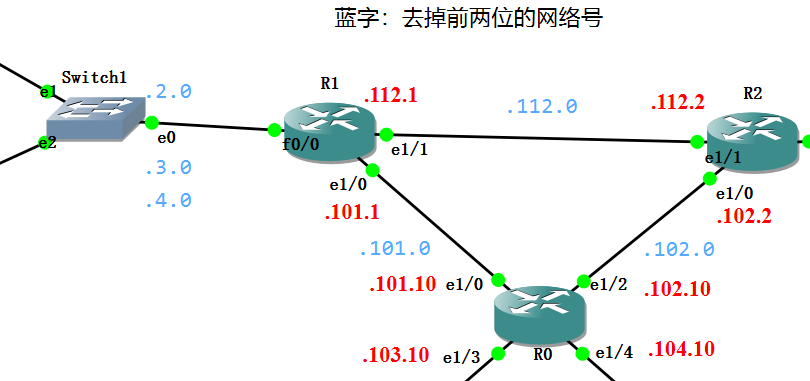


图4-23 R1的连接情况

R1的e1/1端口要配OSPF协议，e1/0端口也要配OSPF协议。由于我的路由器卡槽插的是PA-8E，所以不用配置clockrate和封装ppp协议。配置步骤如下：

|  |
| --- |
| R1#conf t  R1(config)#in e1/1  **R1(config-if)#ip ospf hel**  **R1(config-if)#ip ospf hello-interval 5**  **R1(config-if)#ip ospf dead**  **R1(config-if)#ip ospf dead-interval 20**  **R1(config-if)#no shutdown**  R1(config-if)# |

这些命令的解释如下：

conf t 表示进入到该路由器的配置界面。in e1/1表示进入到e1/1接口的配置环境中。ip ospf hello-interval 5表示该接口每隔5秒都会像隔壁邻居发送一个hello包。ip ospf dead-interval 20表示若是20秒收不到hello包就代表宕机。最后no shutdown 启动该接口。

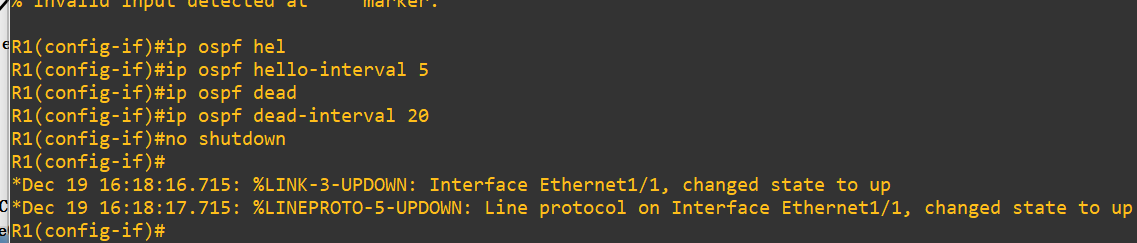


图4-24 R1 e1/1端口配置ospf协议示意图

同理，还要配置R1 e1/0端口的ospf协议，

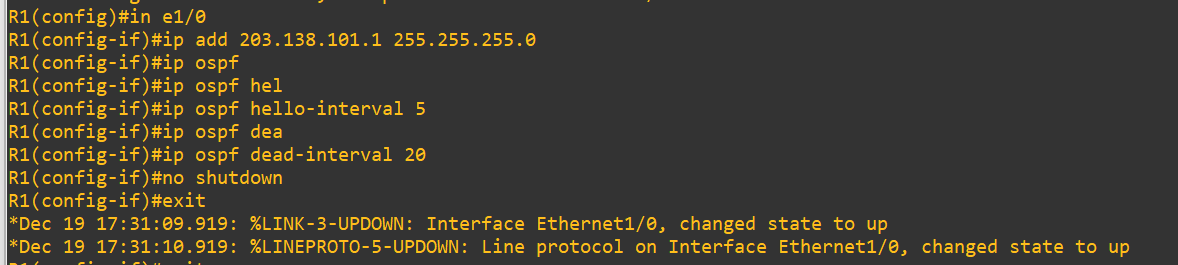


图4-25 R1 e1/0端口配置ospf协议示意图

配完端口，还需要配置ospf 的路由邻居，配置命令如下：

|  |
| --- |
| router ospf 10  network 203.138.2.0 255.255.255.0 area 0 |

router ospf 10表示开启10号进程来运行ospf路由协议

network 命令后面跟着的网络是和该路由直连的网络，255.255.255.0为子网掩码。

area 0表示该网络被划分在区域0

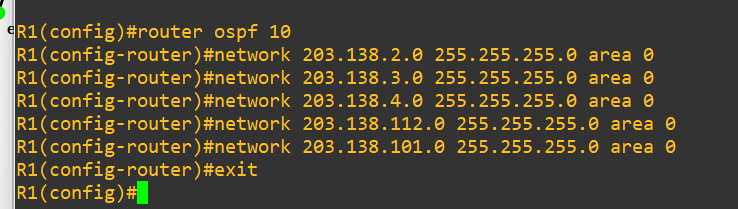


图4-26 R1 配置ospf路由协议

这样，R1就配置完成了。用同样的方式配置R0、R2、R3、R4，过程重复，不再一一罗列。

配置完成所有的路由器的动态路由协议后，查看中枢路由器R0的路由表，如下图所示：

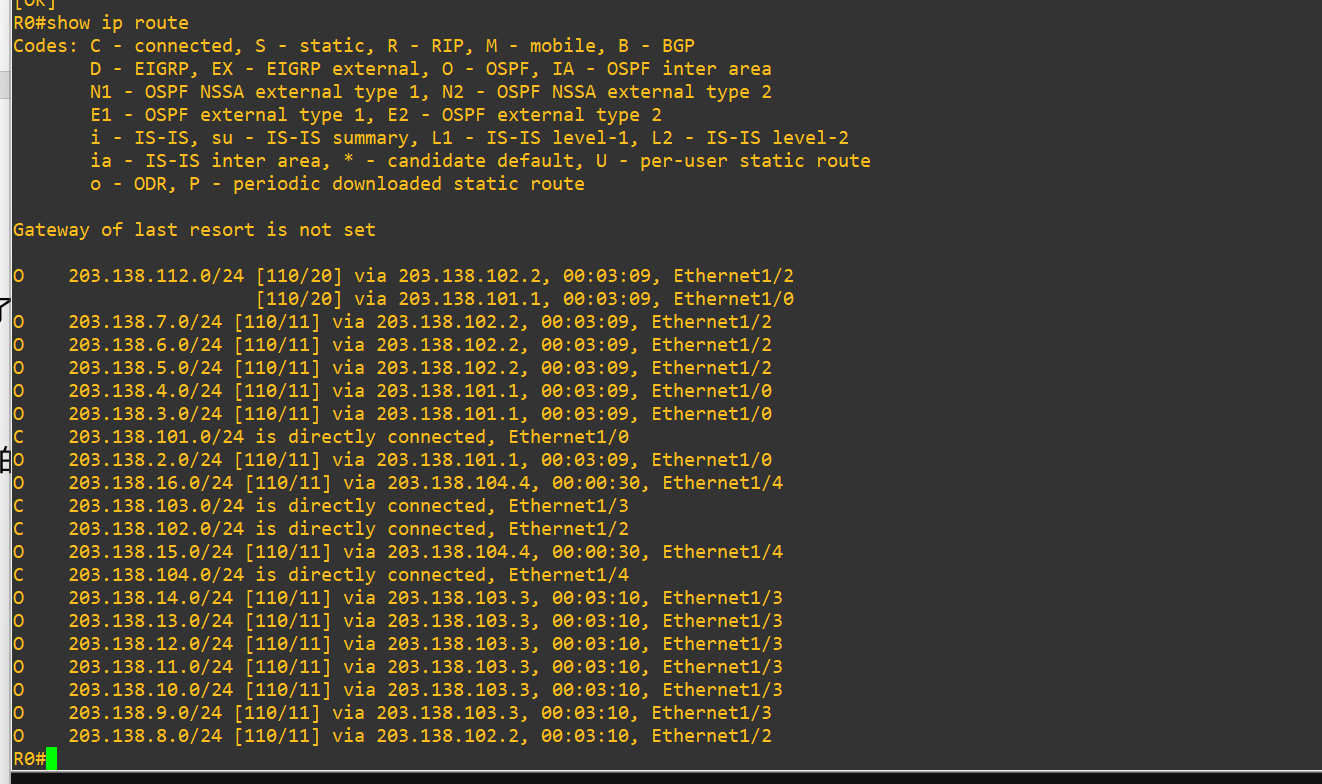


图4-30 R0的路由表

路由表中有21个表项，表示有21个网段，以C为标记的是直连的网络，以O为标记的是ospf协议动态学习得来的网络。从R0的路由表可以看出，R0可以到达网络拓扑图中的任意一个网络。

配置所有路由器后，尝试用PC1 ping 通PC16 和 PC8。结果见[4.3.3 跨路由器互联](#_4.3.3_跨路由器互联互联)。

# 四、实验结果

## 4.1 第一部分

第一次：Switch2的e1端口vlan号为2，e2端口的vlan号为3，PC1和PC2的互相ping的结果如下：

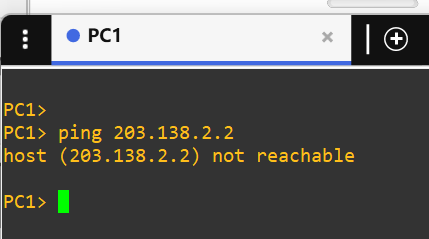


图4-1 第一次PC1 ping PC2



图4-2 第一次PC2 ping PC1

PC1和PC2不能互相ping通。

第二次：Swtich2的e1端口和e2端口vlan号都为2，结果如下：

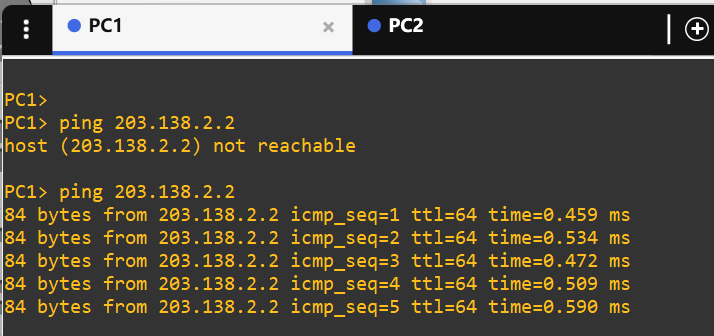


图4-3 第二次 PC1 ping PC2

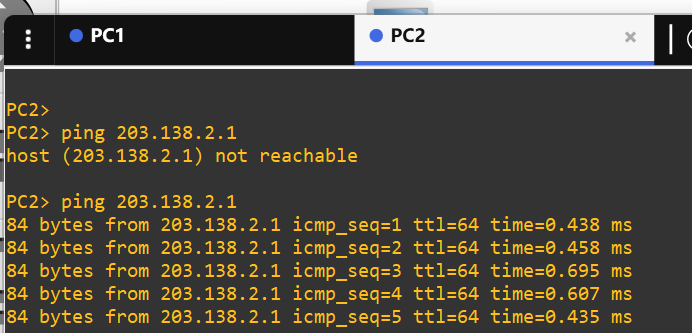


图4-4 第二次PC2 ping PC1

实验结果：第一次，由于PC1和PC2连接的端口是同一个交换机下不同的vlan标签，导致PC1和PC2不能互相ping通；第二次，由于PC1和PC2连接的端口是同一个交换机下的同一个vlan标签，因此PC1和PC2可以互相ping通。

结果分析：

在第一次互相ping的时候，以PC1 ping PC2为例：交换机收到PC1的报文，此时交换机的e1端口给报文打上vlan2的标签。交换机会查看自己的各个端口有没有可以转发的端口，发现只有一个trunk口可以转发，其他的端口标签号都不是vlan2，就将PC1的报文发送给e0的trunk口，而不进入e2的vlan3 access口。

在第二次互相ping的时候，以PC1 ping PC2为例：交换机的e1的access口收到PC1的报文，此时报文没有标签，交换机为它打上标签，然后查找自己的各个端口有没有满足vlan2的标签的端口，发现有access端口e2，有trunk端口e0，交换机将PC1的ping包发送给这两个端口。通过access口转发的数据包会将标签去掉再转发。

Access端口的收发报文逻辑如下：

Access端口收报文：收到一个报文,判断是否有VLAN信息：如果没有则打上端口的PVID，并进行交换转发,如果有则直接丢弃（缺省） Access端口发报文：将报文的VLAN信息剥离，直接发送出去

## 4.2 第二部分

尝试用PC1 ping PC3，结果如下：

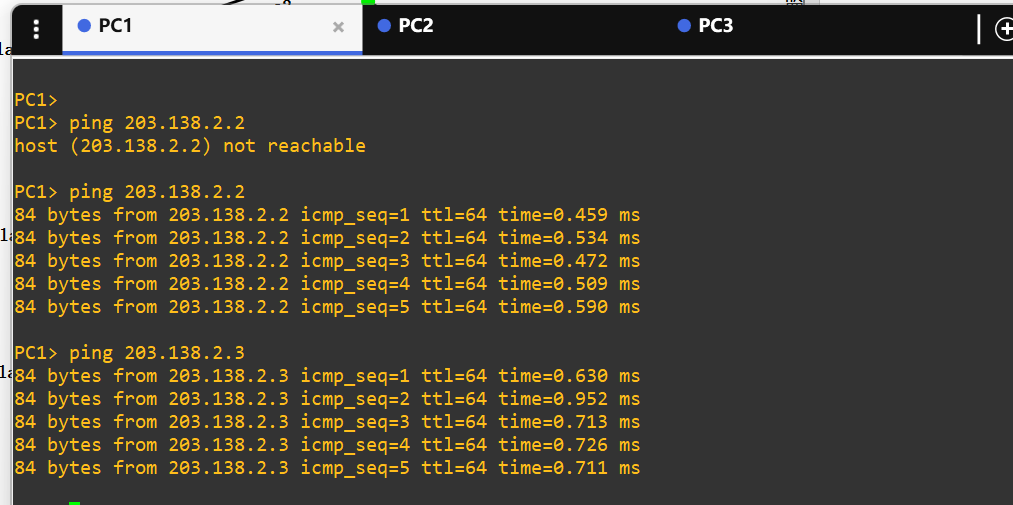


图4-5 PC1 ping PC3 结果图

尝试用PC3 ping PC1，结果如下：

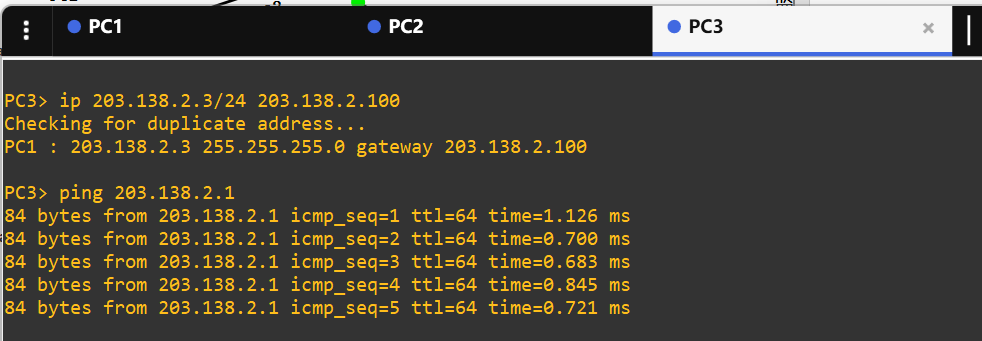


图4-6 PC3 ping PC1 结果图

从结果显示，PC1 和 PC3 可以互相ping通。现在来分析PC1 ping PC3数据包的传输过程以及交换机的工作流程：

PC1发送数据包给交换机SWitch2的e1端口，e1端口是access口，vlan号为2。e1端口检查PC1的数据包上有没有vlan标签——没有，打上vlan1的标签，并进行转发。Switch2里允许vlan2通过的有e2和e0端口，这两个端口转发该数据包。e0端口为trunk口，标签为vlan1，e0端口在发送这个数据包时，首先将数据包的标签和自己的标签进行比较——不一样，直接转发。因此数据包从Switch2转发给Switch1。Switch1的e1端口收到这个数据包。e1端口为trunk口，trunk口接收数据包时会先判断有没有vlan信息，发现有标签信息——判断vlan2能进入，并给e2端口转发，e2端口直接转发该数据包给Switch3。Switch3收到这个数据包后检查数据包上的标签：vlan2标签，并将该数据包从vlan2的access口e0转发。

其中，Trunk口的收发报文的工作流程可以总结如下：

在Trunk端口上发送报文时，先会将要发送报文的vlan标记与Trunk端口的PVID进行比较，如果PVID相等，则从报文中去掉VLAN标记再发送；如果与PVID不相等，则直接发送。这样一来，如果将交换机级联端口都设置为Trunk，并允许所有vlan通过后，默认情况下除了vlan1外的所有来自其他vlan中的报文将直接发送（因为这些vlan不是trunk端口的默认vlan），而作为trunk端口默认vlan的vlan1，则需要通过去掉报文中的vlan信息后再发送。

在Trunk端口收到一个报文时，会首先判断是否有vlan信息：如果没有vlan标记，则打上该trunk端口的pvid，视同该帧是来自pvid所对应的vlan转发到PVID所对应的vlan接口上；如果有vlan标记，判断该Trunk端口是否允许该VLAN的报文进入，如果允许则直接转发，否则丢弃。

## 4.3 第三部分 不同VLAN互联

### 4.3.1 方法1

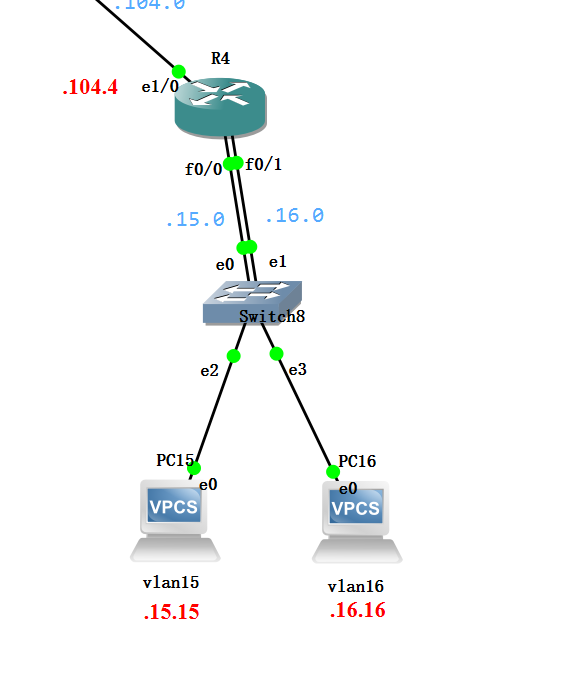


图4-7 方法1

使用方法1配置路由器，如上图所示，尝试ping通PC15 和 PC16，实验结果如下所示：

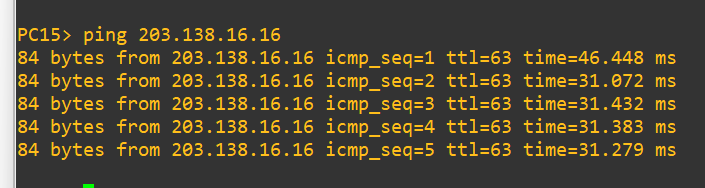


图4-8 方法1 PC15 ping PC16 实验结果

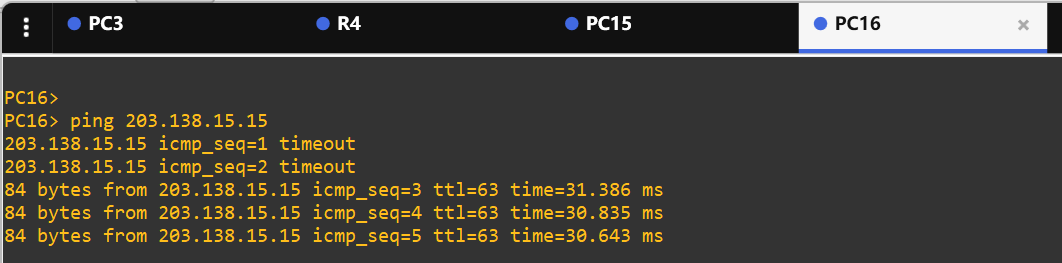


图4-9 方法1 PC16 ping PC15 实验结果

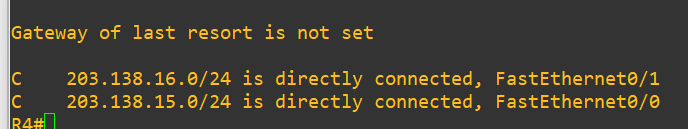


图4-10 R4路由表

PC15 和 PC16 成功ping通。

结果分析：

以PC15 发送数据包给 PC16 为例，数据包的收发过程如下：

PC15发送数据包给Switch8的e2端口，e2端口打上vlan15标签，并通过e0端口转发，e0端口是剥离了vlan15的标签后发送。路由器f0/0收到数据包后，通过子网掩码和目的地按位与并查询路由表，判断要用哪个端口转发——f0/1端口。数据包由f0/1端口发送后Switch8的e1端口收到报文后，access口e1打上vlan16的标签，并发送给e3，e3去标签转发给PC16。

### 4.3.2 方法2

这一小部分实验通过R1来完成，R1底下的网络拓扑如图所示：

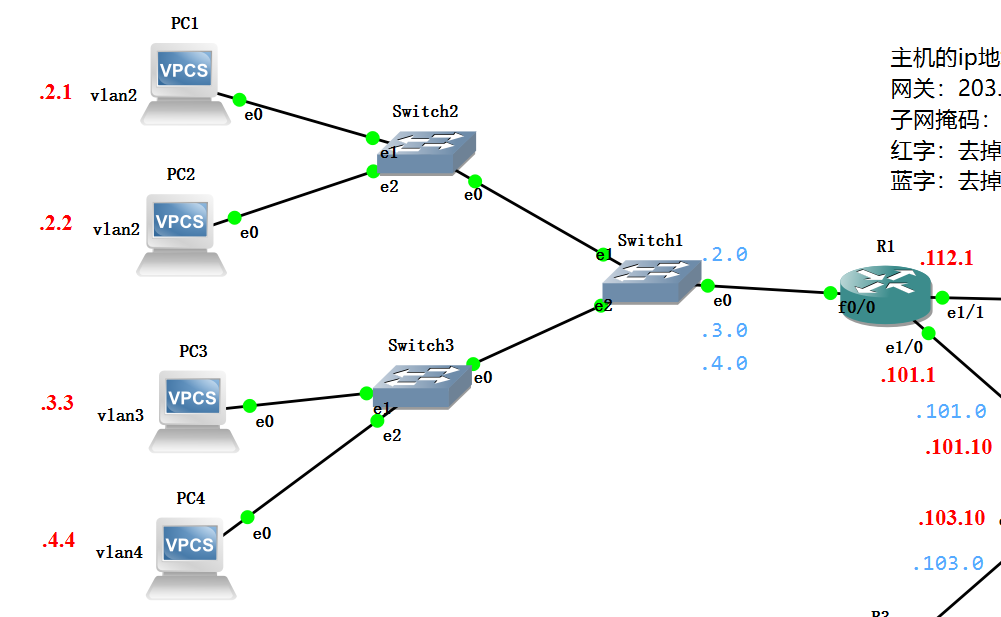


图4-11 R1的网络结构图

需要测试的PC2 和 PC4，他们隶属于不同的vlan，且路由器和交换机通过trunk口，dot1q协议连接，路由器的物理接口被划分成多个子接口。

实验结果如下所示：

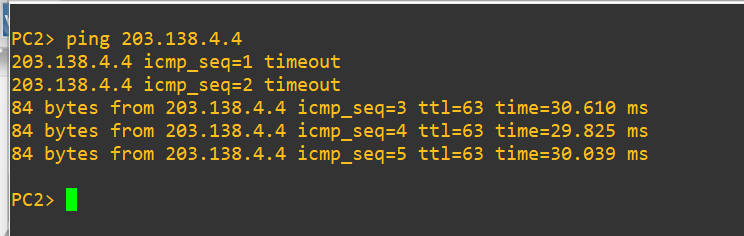


图4-12 PC2 ping PC4实验结果

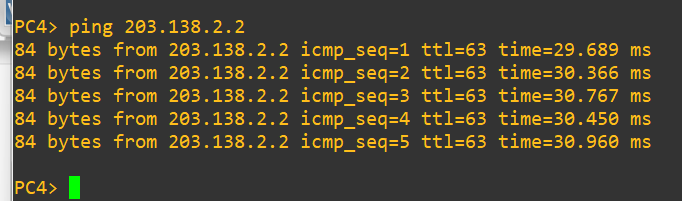


图4-13 PC4 ping PC2实验结果

可以看到，PC2 和PC4都互相ping通了，不同的vlan通过一个路由器实现了互联互通。

结果分析：

以PC2 ping PC4 为例，成功ping通的数据包的传输流程如下：

PC2 发送数据包给Switch2，Switch2的e2接口接收数据包，e2为access口，access接收到无标签包会打上标签，并给trunk口e0转发。e0转发时由于标签2和标签1不一样，因此直接发送。Switch1收到标签2的报文，直接从e0端口转发给R1。R1收到后首先查看是否有vlan标签——有，那就看f0/0哪个子接口能够处理该帧——f0/0.2 封装的是dot1q 2，可以处理。因此将这个数据帧的标签从帧里剥离。然后通过路由表，查看要从哪个接口转发。在路由器转发之前，路由器会将该帧重新封装，由于要发往203.138.4.0网络，因此打上标签4，再发给Switch1。Switch1的e0收到标签为4的数据帧直接转发。Switch3的e0口收到后交给e2转发，e2口剥离标签后发送给PC4。

PC4回复时先发送回应报文给Switch3的e2口，e2口为access口，收到无标签的报文会打上vlan 4的标签，然后交给能处理vlan4的端口处理该报文。e0端口可以处理vlan4，因为它是trunk口，e0转发时先查看该报文标签和自己的一不一样——一个4一个1，不一样，直接转发给Switch1，Switch1都是trunk口，而且都是vlan 1的标签，和vlan4不一样，因此将直接转发给路由器R1。R1收到报文后，首先会解析它的vlan标签，发现是标签4，于是看自己哪个逻辑子端口可以处理这个标签——f0/0.4可以处理。路由器通过报文解析出目的地址后，查路由表转发该报文。在转发前，会给该报文打上目的地vlan2的标签。类似的，该报文经过几个trunk口后到达Switch1的e2端口，e2为trunk口，会剥离标签再转发给PC2。

要注意的是，vlan标签并不会进入到路由器中。

### 4.3.3 跨路由器互联

配置好所有的路由器后，尝试使用PC1 去ping PC16、PC8，结果如下：

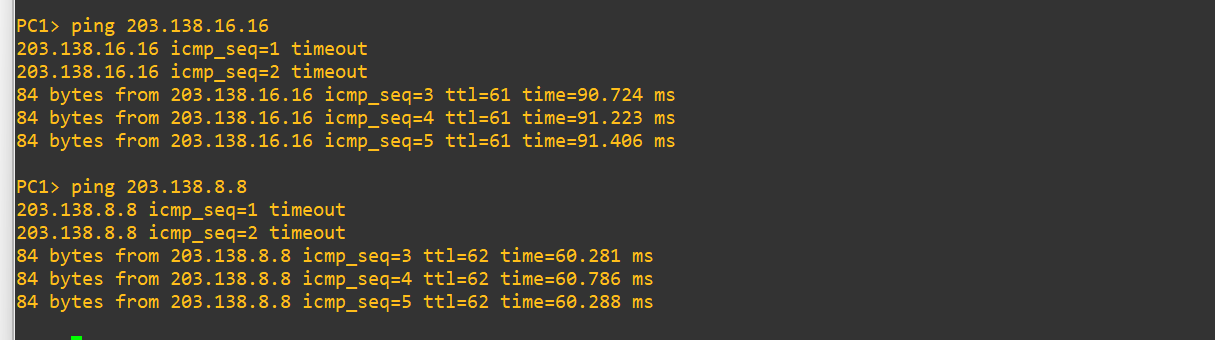


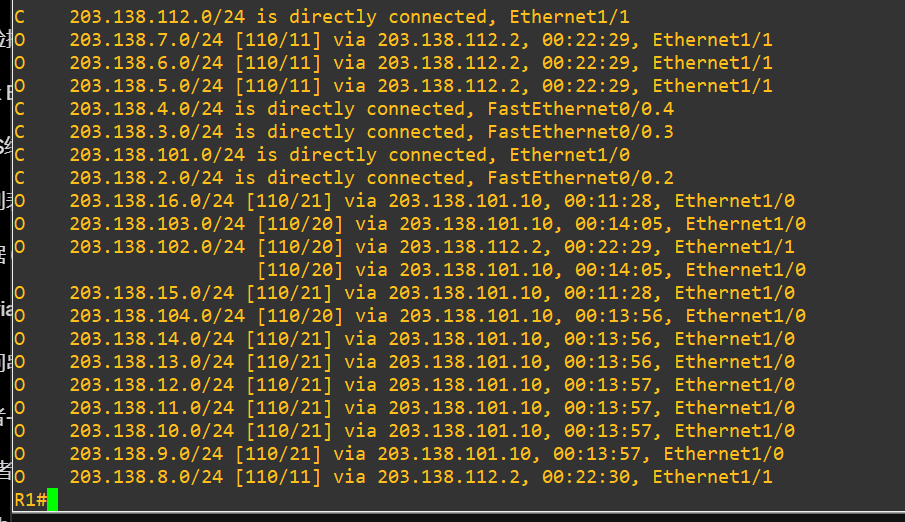
图4-14 PC1 pint PC16、PC8实验结果

如上图所示，PC1成功ping通了PC16、PC8。

结果分析：

以成功的PC1 ping PC16 为例。数据包从PC1到R1的传输过程在上文中已经复述多次，不再赘述。这里主要描述一下路由器处理带vlan标签的数据报文的过程。

路由器的物理接口收到一个带有 dot1Q VLAN 标签的数据包。这个标签包含 VLAN 信息，指示数据包属于哪个 VLAN。之后R1根据 dot1q 协议提取出报头的 VLAN 标签，确定数据包所属的 VLAN。R1根据解析出的 VLAN ID 将数据包分配给相应的子接口。每个子接口通常与一个特定的 VLAN 相关联，将数据包引导到正确的逻辑接口f0/0.2上。由于R1使用 OSPF 协议维护一个路由表，其中包含了网络的路由信息。当数据包到达正确的子接口后，路由器会查找路由表，确定应该将数据包发送到哪个下一跳或目标。路由表如下图所示：

图4-15 R1的路由表

R1根据路由表的查找结果，决定将数据包转发到相应的下一跳或目标——203.138.101.10 e1/0。转发之前，R1将数据包重新封装，由于在路由器之间传输不需要vlan标签，因此将其发送到相应的物理接口。

数据包经过R1、R0到达R4后，同样，R4会查找路由表，并重装这个数据帧，打上16的标签并转发。

# 五、实验中的问题及心得

（1）一开始我使用GNS3作为我的仿真工具。但是在配GNS3的时候卡了半天。首先是GNS3的虚拟机老是装不成功，我又决定不装虚拟机，直接在本机上跑。但是老是出现wait for localhost …… 的弹窗，让我烦不胜烦。我删掉重装了好几次GNS3，最后沉下心来成功安装了虚拟机，并且设置好了环境，也导入了所需要的路由器，这才得以安心实验。

（2）在GNS3软件上，我被一个小bug折磨了好久。一开始，我想知道GNS3的简单工作流程，我设计了一个简单的拓扑，两台PC机和一台路由器，我设置好了IP和网关，但是两台PC怎么样也ping不通。我上网找也没找到有相关的问题，我还以为是我安装软件忘记配置什么东西了。

我想着抓包分析一下包在哪里被丢掉了，但是在抓包时我又犯难了。打开Wireshark，几乎所有的网口我都尝试了一遍，但是都没有找到我想看到的包。机缘巧合之下，我在GNS3界面中右键点击了一条相连的线，看到了start capture的选项，这回才成功抓包了。

对于一开始设计的简单网络，我看了从PC1到PC2的数据包和PC2到PC2的数据包，发现PC2去ping PC1的时候根本没发数据包出来。我检查了好几次PC2的配置，觉得都没问题，有理由怀疑PC2坏了。我又拉出了一个PC3，放到该网络里，这回PC1 ping PC3就成功了，PC3 ping PC1也成功了，但是PC2还是ping不通，那就说明是PC2坏了，应该是这个软件的小bug。

（3）搞好GNS3后，我配置好路由器的路由协议后，路由器死活都没有动态路由信息，我而且每个路由器都没有。我测试了在同一个路由器下不通vlan号之间的互联，是成功的，但是跨路由器的时候就失败了。当时请教了很多人，有人认为是我路由器中间又用交换机连起来导致的，但是我把拓扑结构修改后还是不行。最后电脑重启，重启程序，解决了，服了。

（4）还有一个问题是由于我没有搞懂trunk口转发的过程，导致在相同的局域网内，相同的vlan内互相ping不通，如下：

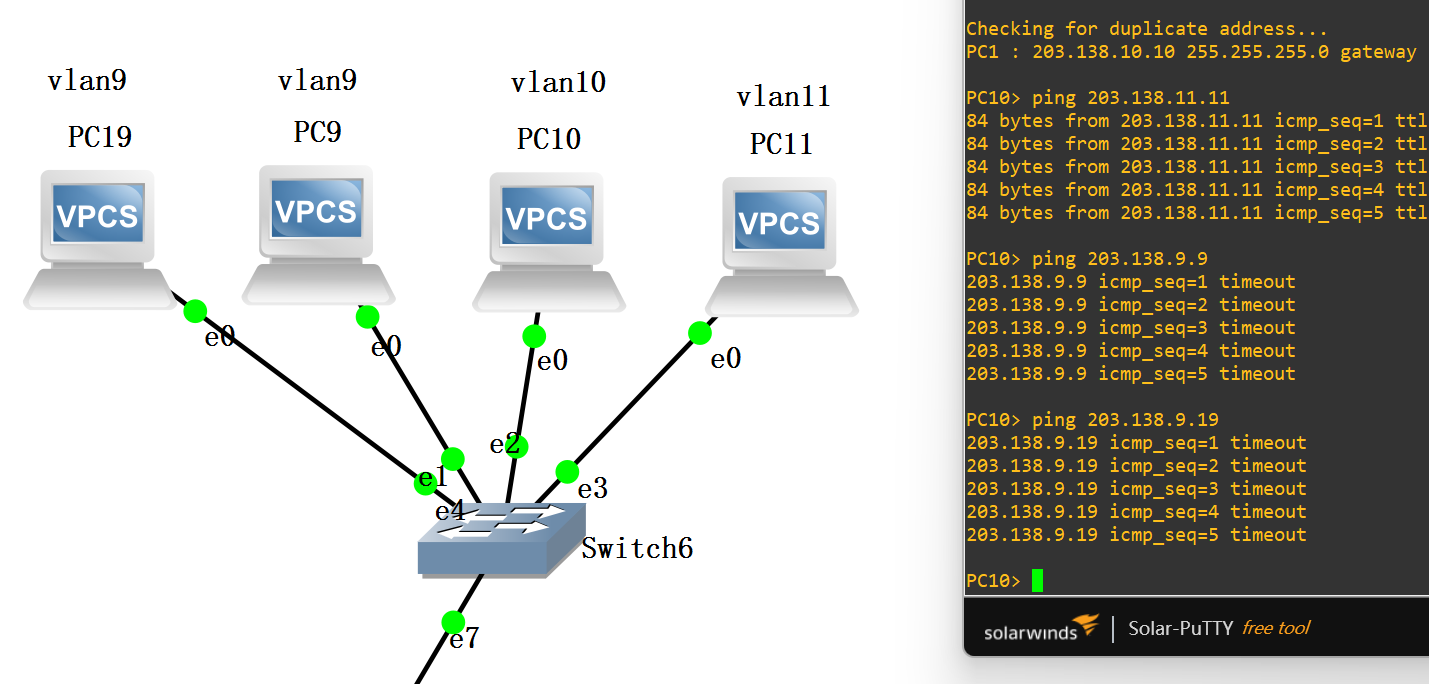


图5-1 相同局域网内联不通示意图

我想用同一个vlan号的PC10去ping PC9，但是ping不通。原因如下：我把Swithc6的e7端口vlan标签号设置成vlan9的标签。IP包从PC10到路由器，从路由器到PC9之间，从路由器发回的包经过e7 trunk口，由于e7的标签为9，和报文里的vlan标签相同，因此trunk口会去标签再转发。这样trunk口收到的报文就不知道发往哪一个端口了（因为我的其他端口都是access口），就导致PC9接收不到PC10的请求。

（5）这次实验使我大致掌握了GNS3的使用

（6）我现在清楚的知道了数据包在一个实际vlan网络的传输流程，也知道了交换机和路由器处理带vlan标签的数据包的方法及过程。trunk口和access口不同，且路由器也有两种连接vlan的方式，这些我都掌握了。

# 六、实验思考

## 6.1 同个局域网配置不同IP网段

题目：如何在同一个局域网中，配置两个IP网段？（要求这两个网段的设备可以互相ping通，采用两种以上的配置方法）

解答如下：

第一种：

首先我们要知道两点：

1. Ping的时候没有子网掩码，不知道目标的确切网络号。

2. 路由表在查表时不是根据目标的子网掩码来找目标的网络，而是通过某一表项的子网掩码来推算网络号。

根据这两点可以获得我们的一个答案，请看下例。

有如下两个ip地址和子网掩码：

PC1: 1.1.0.2 255.0.0.0

PC2: 1.1.1.2 255.255.0.0

上例中PC1 ping PC2用自己的子网掩码推算PC2的网络是不是一样：用255.0.0.0 跟ip2来与运算：发现一样，因此任务PC2和自己在同一个网络里，就直接发送数据包过去。发过去了PC2发回来的时候也是，用ip1和PC2的子网掩码相与，发现网络号一样，便传回去了。

第二种：

在PC机上加一条接口转发的路由信息，就能ping通。

PC2 2.1.1.1 255.0.0.0

PC3 3.1.1.1 255.0.0.0

PC2 ip route 3.0.0.0 255.0.0.0 f0/0

PC3 ip route 2.0.0.0 255.0.0.0 f0/0

这里假设PC2 ping PC3。PC2初始时不知道ping PC3的下一跳地址要往哪里发送，因此会发送一个arp广播包。arp解析的ip地址就是对方的ip。由于在同一个网络里，arp广播使得PC3 得到 PC2的mac，告诉PC2可以往我这里。PC2成功发送数据包到PC3。可是当PC3要发回去时，不知道PC2，又一次发送arp广播包，PC2通过广播包获取PC3的请求，并告诉PC3往我这里发送。这样PC3的回应报文也能成功被PC2接收。

## 6.2 分析数据包传输流程

题目：选择自己拓扑中两个不同VLAN中的PC机，中间要经过trunk链路连接的路由器，阐述互相ping时的完整传输流程。（包括交换机和路由器的简单处理过程，并且要指出数据包中VLAN标签的变化过程）

解答：

这一小部分实验通过R1来完成，R1底下的网络拓扑如图所示：

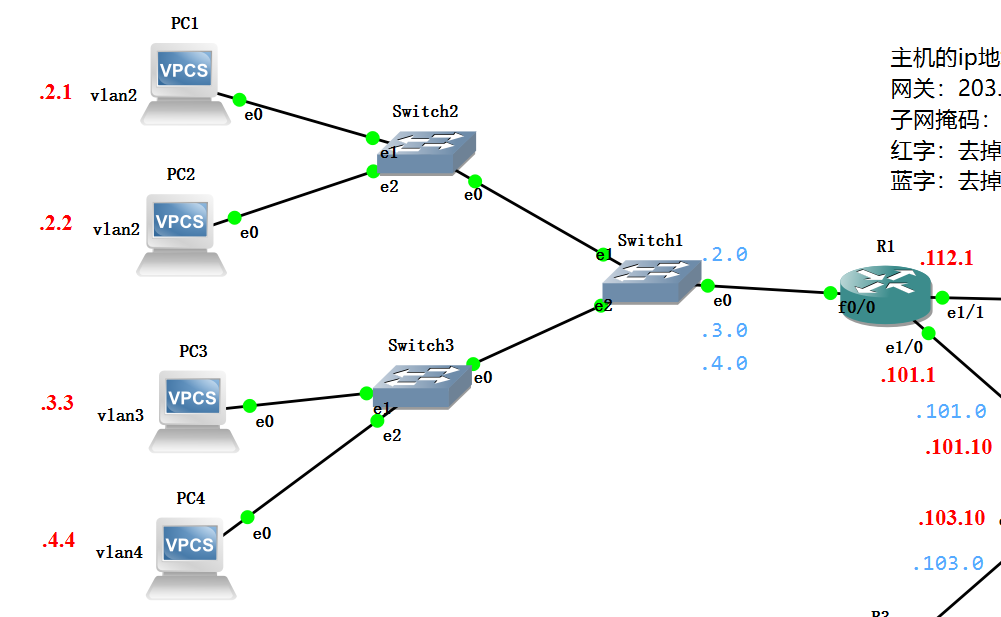


图6-1 R1的网络结构图

需要测试的PC2 和 PC4，他们隶属于不同的vlan，且路由器和交换机通过trunk口，dot1q协议连接，路由器的物理接口被划分成多个子接口。

实验结果如下所示：

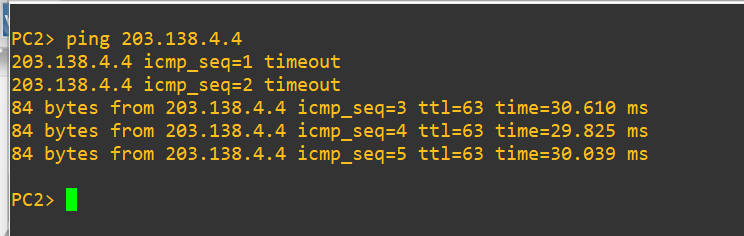


图6-2 PC2 ping PC4实验结果

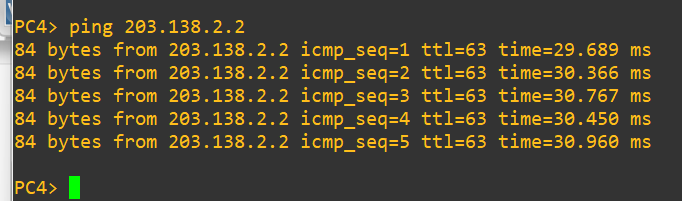


图6-3 PC4 ping PC2实验结果

可以看到，PC2 和PC4都互相ping通了，不同的vlan通过一个路由器实现了互联互通。

结果分析：

以PC2 ping PC4 为例，成功ping通的数据包的传输流程如下：

PC2 发送数据包给Switch2，Switch2的e2接口接收数据包，e2为access口，access接收到无标签包会打上标签2，并给trunk口e0转发。e0转发时由于标签2和标签1不一样，因此直接发送。Switch1收到标签2的报文，直接从e0端口转发给R1。R1收到后首先查看是否有vlan标签——有，那就看f0/0哪个子接口能够处理该帧——f0/0.2 封装的是dot1q 2，可以处理。因此将这个数据帧的标签从帧里剥离。然后通过路由表，查看要从哪个接口转发。在路由器转发之前，路由器会将该帧重新封装，由于要发往203.138.4.0网络，因此打上标签4，再发给Switch1。Switch1的e0收到标签为4的数据帧直接转发。Switch3的e0口收到后交给e2 access口转发，e2口剥离标签后发送给PC4。

PC4回复时先发送回应报文给Switch3的e2口，e2口为access口，收到无标签的报文会打上vlan 4的标签，然后交给能处理vlan4的端口处理该报文。e0端口可以处理vlan4，因为它是trunk口，e0转发时先查看该报文标签和自己的一不一样——一个4一个1，不一样，直接转发给Switch1，Switch1都是trunk口，而且都是vlan 1的标签，和vlan4不一样，因此将直接转发给路由器R1。R1收到报文后，首先会解析它的vlan标签，发现是标签4，于是看自己哪个逻辑子端口可以处理这个标签——f0/0.4可以处理。路由器通过报文解析出目的地址后，查路由表转发该报文。在转发前，会给该报文打上目的地vlan2的标签。类似的，该报文经过几个trunk口后到达Switch1的e2端口，e2为trunk口，会剥离标签再转发给PC2。

要注意的是，vlan标签并不会进入到路由器中。

## 6.3 物理网络、VLAN、IP网段的关系

题目：请阐述物理网络、VLAN及IP网段的关系，说明路由器是如何把不同物理网络连通的。

解答：

Vlan是在同一个物理网络再进行划分

不同的物理网路可以划分vlan1、2、3、4，不同物理网路相同vlan标签的虚拟网络是独立的

每个 VLAN 可以有自己的IP网段，即一个 VLAN 对应一个 IP 网段。

路由器通常被用于连接不同的物理网络或不同的 VLAN，以实现跨网络的通信。

路由器通常有多个物理接口，每个接口连接到一个不同的物理网络。这些物理接口可以是以太网口、无线接口等。如果在路由器上使用 VLAN 划分，路由器的接口可以配置为多个子接口，每个子接口关联一个 VLAN。这样，路由器可以在同一个物理接口上连接多个逻辑上独立的 VLAN。对于每个子接口，路由器会分配一个IP地址，该IP地址属于子接口所关联的 VLAN 的 IP 网段。这样，路由器就能够在不同的 IP 网段之间进行路由。路由器维护一个路由表，用于决定将数据包从一个接口转发到另一个接口的方式。当路由器收到一个数据包时，它会根据目标IP地址查找路由表，确定应该将数据包转发到哪个接口。路由器会检查目标IP地址所属的网络，然后将数据包转发到适当的接口，实现不同物理网络之间的通信。