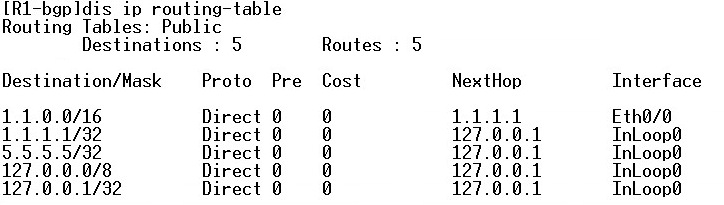
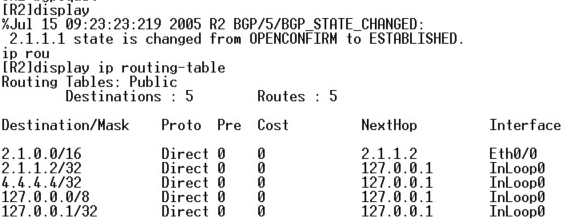
**实验3 BGP协议实验**

**注意：为了提高效率，上传文件到服务器可以用在实验报告上粘贴相关内容的截图代替。**

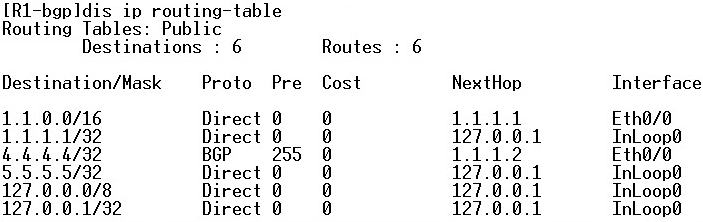
1．查看R1和R2的路由表，注入路由信息前，是否有对方loopback的路由信息？注入路由信息后，是否有对方loopback的路由信息？为什么？

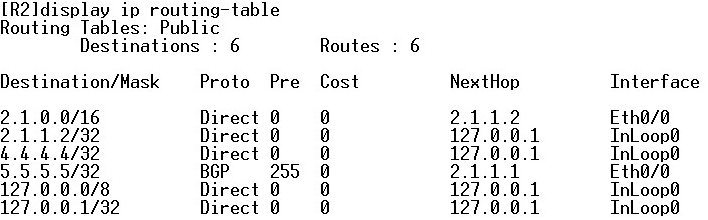
**答：没有对方loopback路由信息。注入路由信息后，有对方的loopback路由信息。因为bgp没有加入4.4.4.4和5.5.5.5的网段，并没有将这个网段通过bgp发送AS外的路由。**

**注入后，通过bgp，R1和R2都能知晓这两个网段，建立bgp连接进行了更新。**



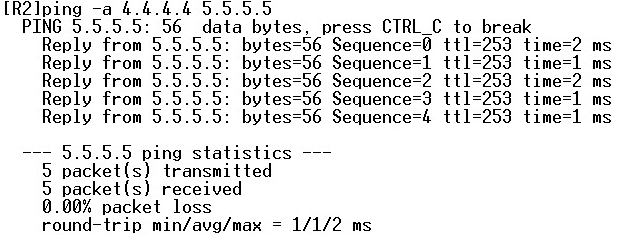
**注入路由信息后：**



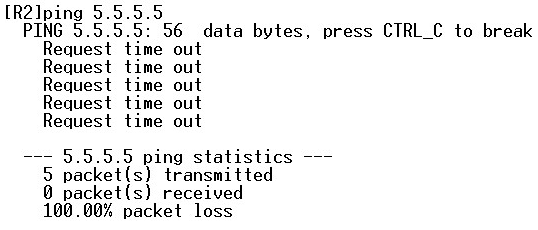


2．[R2]ping –a 4.4.4.4 5.5.5.5 能否ping通？ 如果不用ping命令的－a参数是否能ping通？为什么？

**答：可以。ping –a正常作用是将地址解析成主机名，而书上这里是用4.4.4.4作为源地址去ping 5.5.5.5地址，而上面的实验已经在BGP里面注入loopback网段的路由信息，4.4.4.4与5.5.5.5之间路由信息正确，报文知道怎么到达所以可以ping通。**



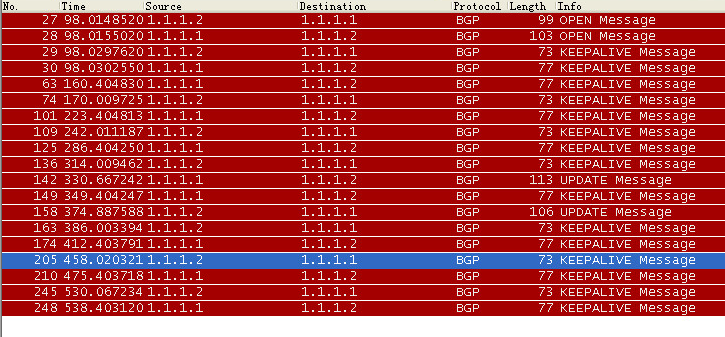
**假如不加入-a的指令，则不可以ping通。因为路由表R2中没有返回R1地址1.1.1.1的路由信息。**



3．把所截报文命名为BGP1-学号，并上传到服务器。根据截获的BGP报文的顺序和结构，填写下表。

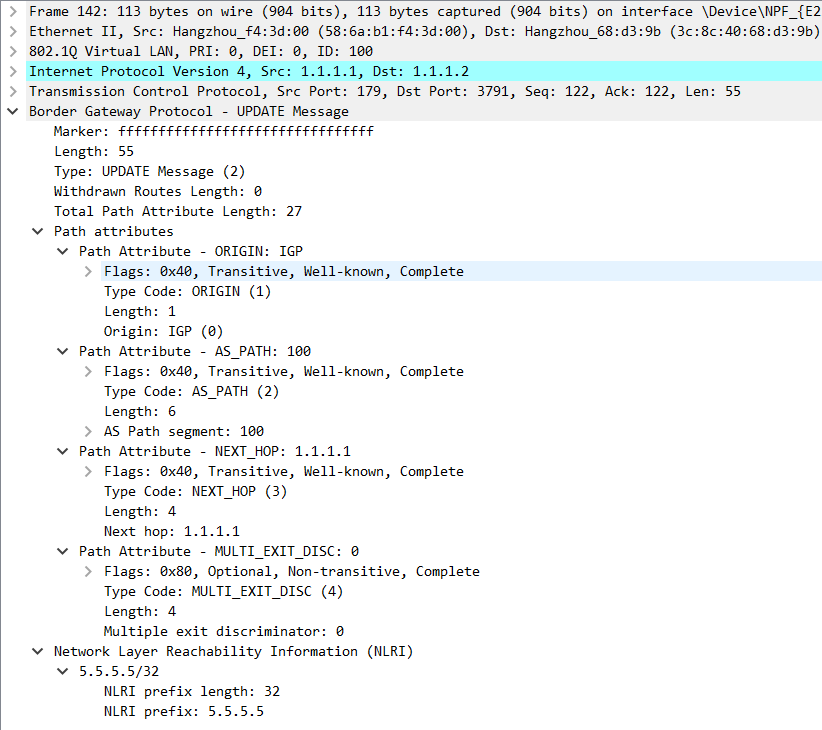
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 报文序号 | 报文种类 | 源地址及端口号 | 目的地址及端口号 | 报文的作用 |
| 27 | open | 1.1.1.2 3791 | 1.1.1.1 179 | 对等体交换各自版本、自治系统号、保持时间、BGP标识符等信息协商 |
| 28 | open | 1.1.1.1 179 | 1.1.1.2 3791 | 对等体交换各自版本、自治系统号、保持时间、BGP标识符等信息协商 |
| 29 | keepalive | 1.1.1.2 3791 | 1.1.1.1 179 | 对等体周期发送，确保连接保持有效 |
| 30 | keepalive | 1.1.1.1 179 | 1.1.1.2 3791 | 对等体周期发送，确保连接保持有效 |
| 142 | update | 1.1.1.1 179 | 1.1.1.2 3791 | 路由更新信息，包括撤销、可达路由信息及其路径属性 |
| 158 | update | 1.1.1.2 3791 | 1.1.1.1 179 | 路由更新信息，包括撤销、可达路由信息及其路径属性 |
| 245 | keepalive | 1.1.1.2 3791 | 1.1.1.1 179 | 对等体交换各自版本、自治系统号、保持时间、BGP标识符等信息协商 |
| 248 | keepalive | 1.1.1.1 179 | 1.1.1.2 3791 | 对等体交换各自版本、自治系统号、保持时间、BGP标识符等信息协商 |

4. 思考题：在实验截获的报文中是否有NOTIFICATION报文？为什么？



**答：没有，因为BGP只有检测到差错（连接中断、协商出错、报文差错等）才会发送NOTIFICATION报文，关闭同对等体的连接，两个bgp路由已经建立通信，没有出错。**

5. 写出一个Update报文的完整结构，并指出报文中路由信息所携带的路由属性。



**答：报文中路由信息所携带的路由属性有：ORIGIN:IGP、AS\_PATH:100、NEXT\_HOP:1.1.1.1、MULTI\_EXIT\_DISC:0。**

**路由属性：第一条IGP表示BGP由network引入**

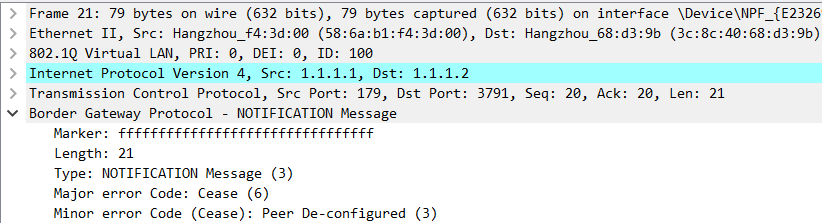
**第二条AS\_PATH表示一条顺序从后往前的先后所经过的AS序号，最后经过AS放在列表前面，这里为AS100**

**第三条为通过EBGP获得的路由再发布给IBGP邻居时，IBGP邻居收到的路由下一跳不改变，此时可以修改下一跳地址指向自己**

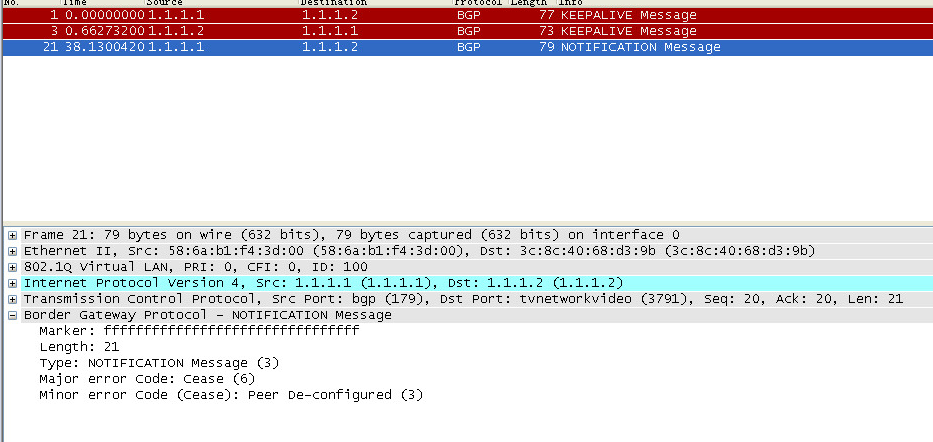
**第四条被典型用在AS间的链路上，以区分到达相同的邻居AS的多个出口点**

**最后一个Network Layer表示5.5.5.5网络可达。**

6. 在2.6节步骤4，观察截获的BGP的NOTIFICATION报文，将字段值填入实验报告中。



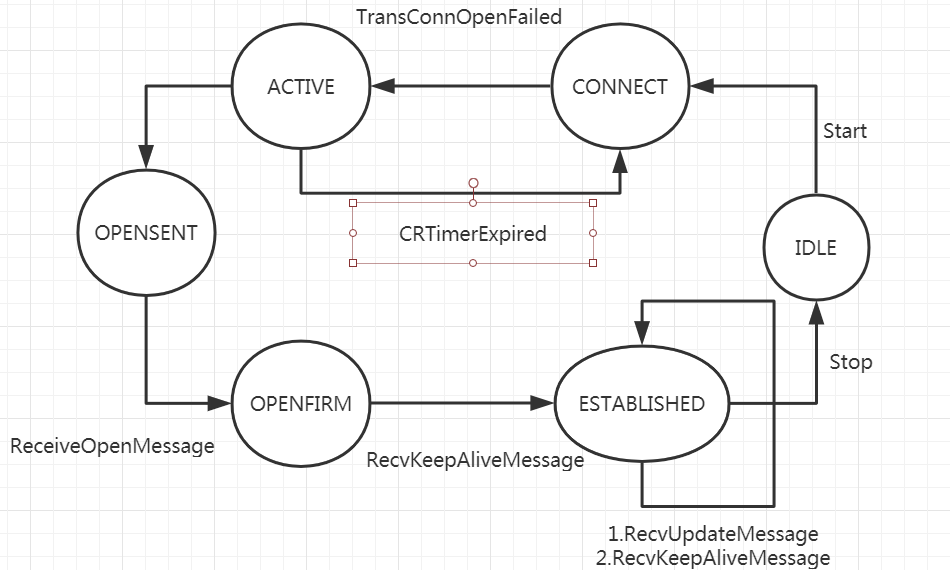
7. 思考题：此时，观察到R1发来NOTIFICATION 报文，检测到是什么错误？



**答：Cease错误，错误类型为退出（错误类型6）。**

**8．第3.4节的实验步骤：**

**步骤一** 继续上一节的实验，从debug信息中分析BGP协议的状态机，画出具体的状态转换图。

****

**步骤二** 将R1与S1之间的网线断开，在S1上观察邻居R1的状态变为\_\_\_IDLE\_\_\_

**步骤三** 将R1的e1接口与S1的eth0/1相连，不用给R1的e1接口配置ip地址，在S1上观察邻居R1的状态变为\_\_ACTIVE\_\_

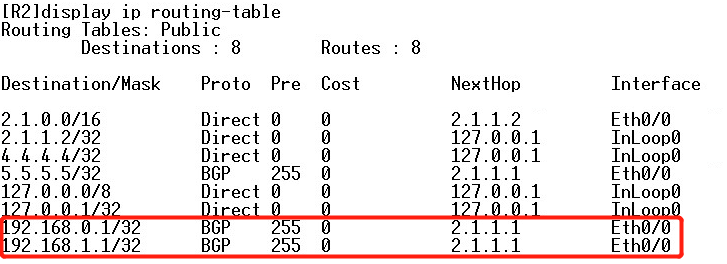
**步骤四** 将R1与S1之间的连线复原，在S1上观察邻居R1的状态变为\_\_ESTABLISHED\_\_

**9．第4.5节实验步骤：**

**步骤一** 在上一节的基础上,在R1上添加两个loopback，(192.168.0.1/24和192.168.1.1/24)分别将他们引入BGP路由（如图-7）。观察R2的路由表。

R2获得两条新路由为**192.168.0.1/32 BGP 255 0 2.1.1.1 Eth0/0**

**192.168.1.1/32 BGP 255 0 2.1.1.1 Eth0/0**

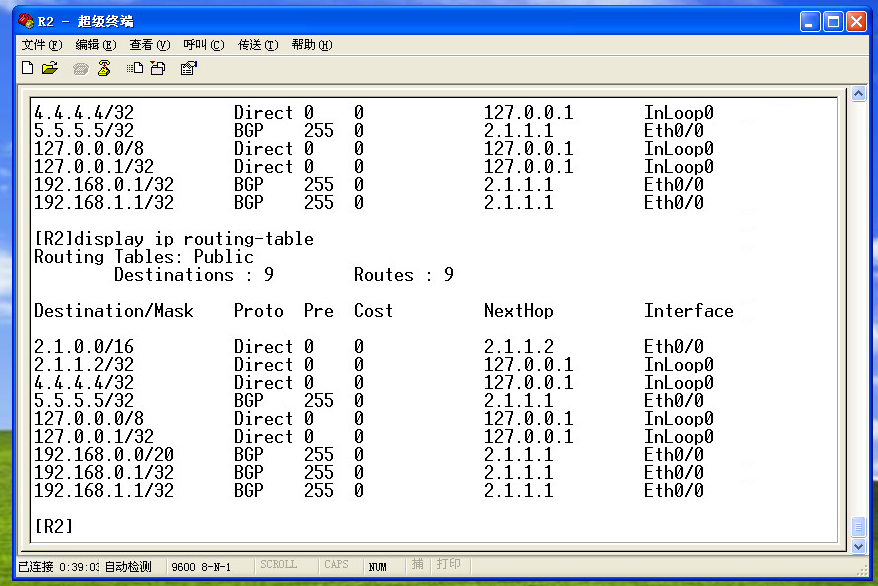


**步骤二** 在R1上配置路由聚合，然后再观察路由表与配置路由聚合之前的路由表有何不同之处。

(1) 同时通告聚合路由和具体路由，请描述R2上路由表的变化。

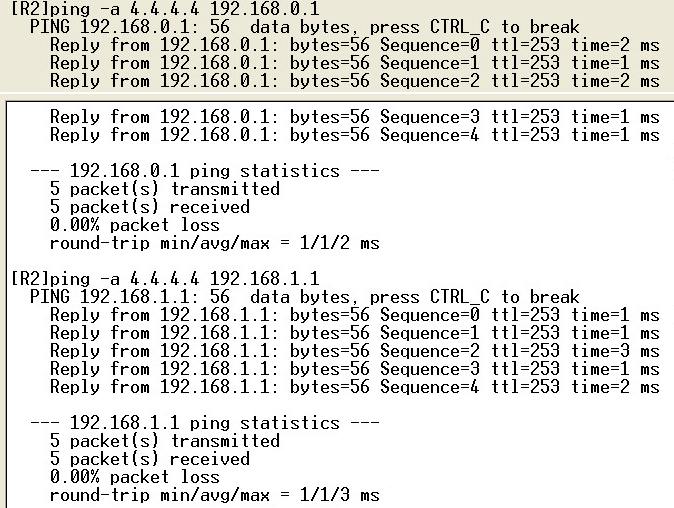
**答：R2路由表在原来基础上增加：**

**192.168.0.0/20 BGP 255 0 2.1.1.1 Eth0/0**



(2) 只通告聚合路由，请描述R2上路由表的变化。用R2 ping 192.168.0.1或192.168.1.1，是否能ping通？

**答：可以ping通，R2上路由表只存在聚合路由。**



**步骤三** 在路由聚合完成后取消参与聚合的某个Loopback接口，观察各路由表分别有什么变化？体会路由聚合都有什么作用？

**答：没有变化，这就体现了路由聚合的作用，避免无用的路由信息在网络中进行充斥，通过路由聚合有效的缩减了需要更新的路由信息。**

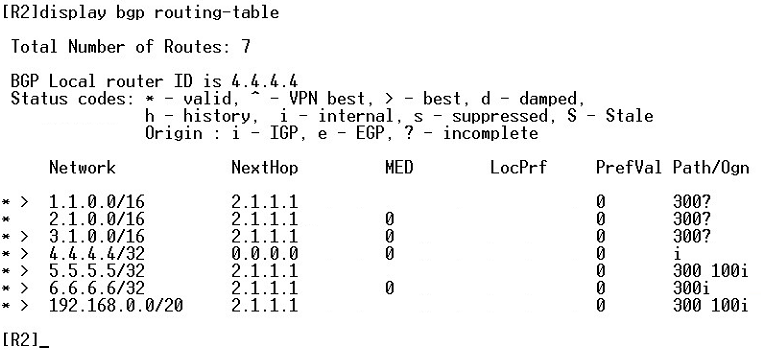
**5.5 BGP的基本路由属性分析**

10．将各路由的ORIGIN和AS-PATH属性值填入下表：

|  |  |  |
| --- | --- | --- |
| Destination/Mask | Origin | Path |
| **1.1.0.0/16** | **incomplete** | **300** |
| **2.1.0.0/16** | **Incomplete** | **300** |
| **3.1.0.0/16** | **Incomplete** | **300** |
| **4.4.4.4/32** | **IGP** | **无** |
| **5.5.5.5/32** | **IGP** | **300 100** |
| **6.6.6.6/32** | **IGP** | **300** |
| **192.168.0.0/20** | **IGP** | **300 100** |
|  |  |  |
|  |  |  |

分析上表中ORIGIN属性和PATH属性的含义。

**答：ORIGIN属性定义路径信息的来源，标记一条路由是怎么样成为BGP路由的，如IGP、EGP、Incomplete。PATH属性标识路由经过的AS序列，即列出在到达所通告的网络之前所经过的AS清单，BGP发言人经自己的AS前置到接受的所有AS路径头部。**

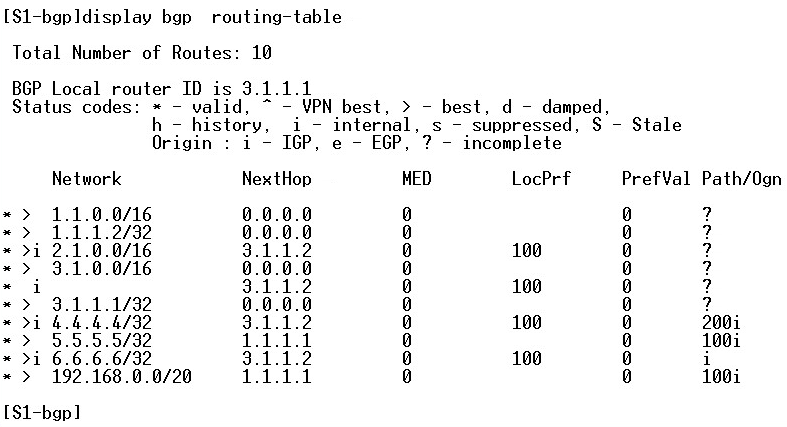


**对于R2来言，4.4.4.4-6.6.6.6都是通过设置network可追溯BGP的网段，属于IGP属性，而其它网段都是通过重分发加入的，并不能追溯，所以属于Incomplete。而path就是这个追溯的路径，例如192.168.0.0/20网段，就是通过AS300-AS100可追溯到。**

11．先观察S1上的到5.0.0.0和4.0.0.0网段的路由的下一跳分别为**\_1.1.1.1\_**和 **\_3.1.1.2\_**

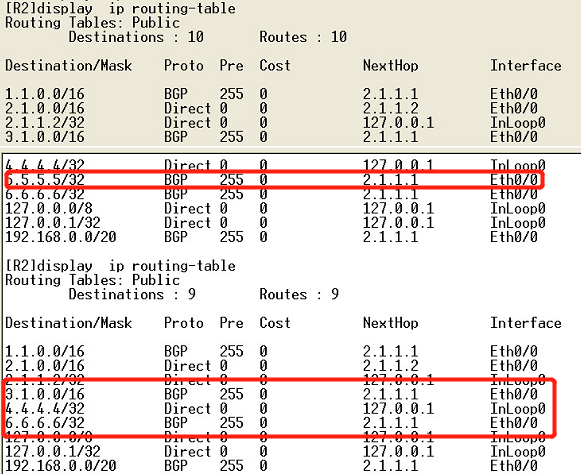
并分析原因.

**答：5.0.0.0网段下一跳为1.1.1.1是因为5.5.5.5地址在EBGP R1上面，所以下一跳地址为R1的接口地址1.1.1.1，而4.0.0.0网段下一跳3.1.1.2是因为IBGP邻居之间设置了强制下一跳，如果邻居之间不是直接连接路由就会被丢弃，或出现路由黑洞。**

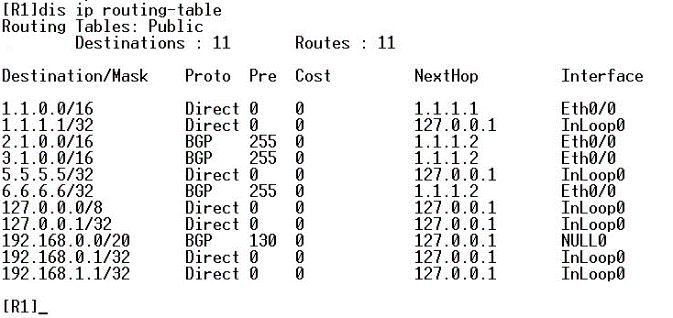


**6.4 BGP的路由策略**

12．观察R2的路由表，是否有5.0.0.0网段的路由？ **答：没有。**

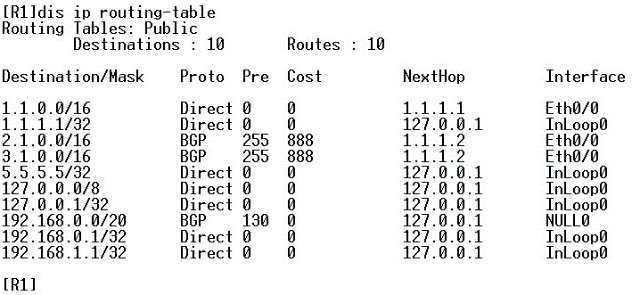


观察R1的路由表的变化，是否还有4.0.0.0网段的路由？**答：没有。**



13．观察R1的BGP路由表信息，是否还有6.0.0.0网段的路由：**答：没有。**

S1通告给R1的路由的med值为：\_888\_ 。



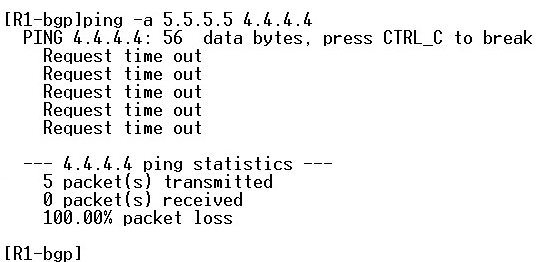
**7.5 BGP的同步机制**

14．BGP不同步引起的问题及一些解决方法

（1）查看R1和R2的路由表，是否有对方loopback的路由信息？为什么？在R1和R2上以本身的loopback为源地址ping对方的loopback地址。能否ping通？为什么？

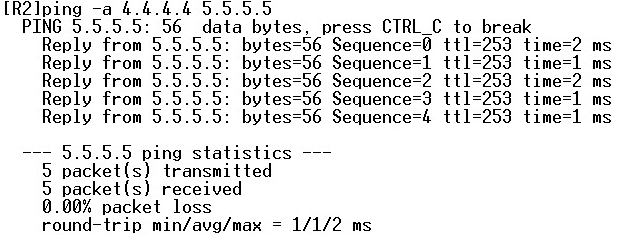
**答：没有对方的loopback路由信息，因为尽管在R1和R2的BGP中分别注入5.5.5.5/32和4.4.4.4/32网段的路由信息，但是中间的S2没有配置BGP，路由信息没法传递。因为BGP路由没有同步，R2需要通过S1知道AS100中5.5.5.5的情况，但是由于S2没有进行设置，所以S1想要强制下一条至R2会在S2中找不到发送的下一跳。同理R2想要传送报文给IBGP S1需要经过S2，但S2的路由表没有得到更新，所以无法建立两者的BGP交换。**

**Ping不通，报文走到S2这里不知道往哪里转发所以丢弃。**

****

（2）在S2上分别配置到5.5.5.5/8网段、下一跳为3.1.1.1和4.4.4.4/8网段、下一跳为2.1.1.2的静态路由信息，在R1上以5.5.5.5为源地址ping 4.4.4.4，能否ping通？为什么？

**答：可以ping通，因为AS300内部配置了OSPF协议，能够发现路由信息，而中间没有配置BGP的交换机S2在配置两条静态路由信息之后，路由信息表完整，报文转发路径连通，于是可以ping通。**

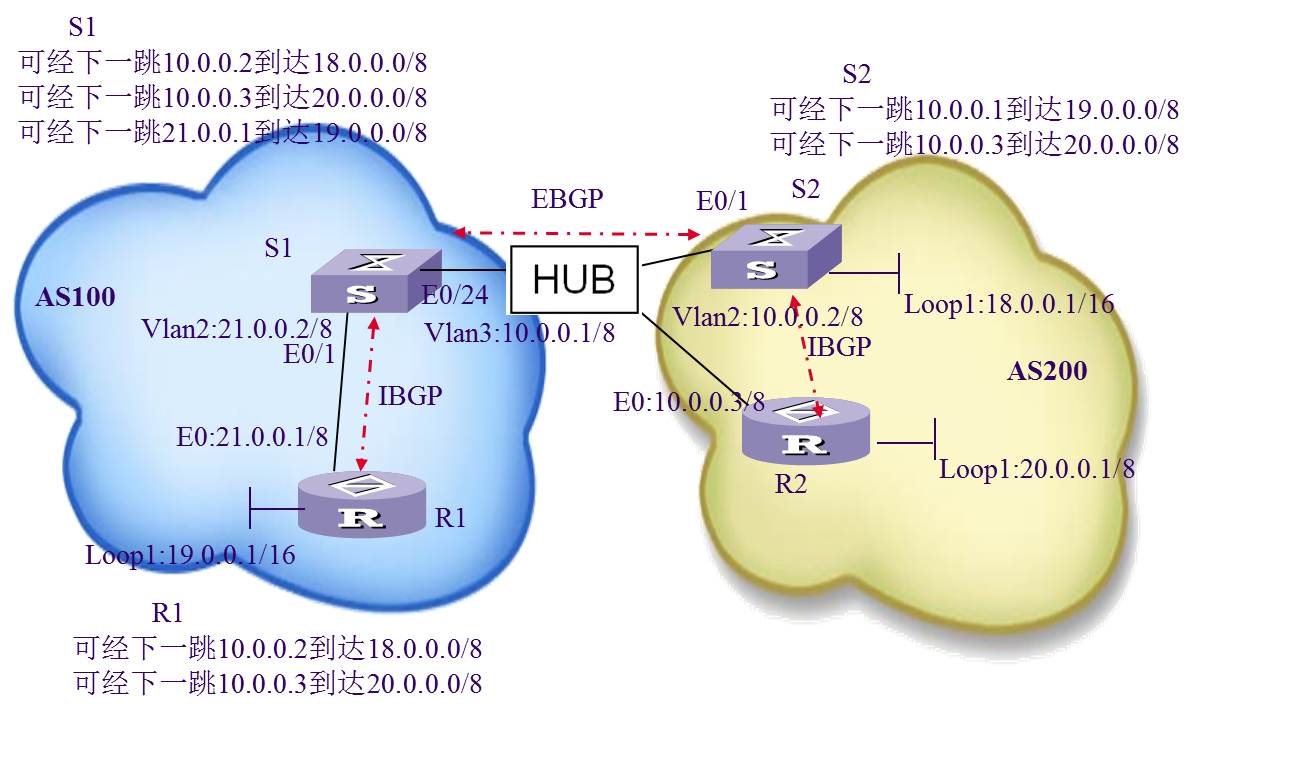


（3）试分析BGP不同步引起的问题。

**答：BGP不同步主要造成的是IGP不同步的问题，在一个AS中，由于IBGP之间由于没有互联且没有完善的设置，从而造成IBGP之间的信息没有得到更新，最终导致的是AS内路由表与外部关系的不完整。**

**15. 设计实验1**

按照图-12的路由要求进行配置，并分析BGP的next-hop属性。



BGP路由协议配置组网图

S1

Vlan 2

Port e1/0/1

Inter vlan 2

Ip add 21.0.0.2 8

Vlan 3

Port e1/0/24

Inter vlan 3

Ip add 10.0.0.1 8

Bgp 100

Peer 21.0.0.1 as 100

Peer 10.0.0.2 as 200

~~Peer 21.0.0.1 next-hop-local~~

Import-route direct

R1

Inter vlan e0/0

Ip add 21.0.0.1 8

Inter loop 1

Ip add 19.0.0.1 32

Bgp 100

Peer 21.0.0.2 as 100

Network 19.0.0.1 32

S2

Vlan 2

Port e1/0/1

Port e1/0/2

Inter vlan 2

Ip add 10.0.0.2 8

Inter loop 1

Ip add 18.0.0.1 32

Bgp 200

Peer 10.0.0.1 as 100

Peer 10.0.0.3 as 200

Network 18.0.0.1 32

~~Peer 10.0.0.3 next-hop-local~~

R2

Inter e0/0

Ip add 10.0.0.3 8

Inter loop 1

Ip add 20.0.0.1 32

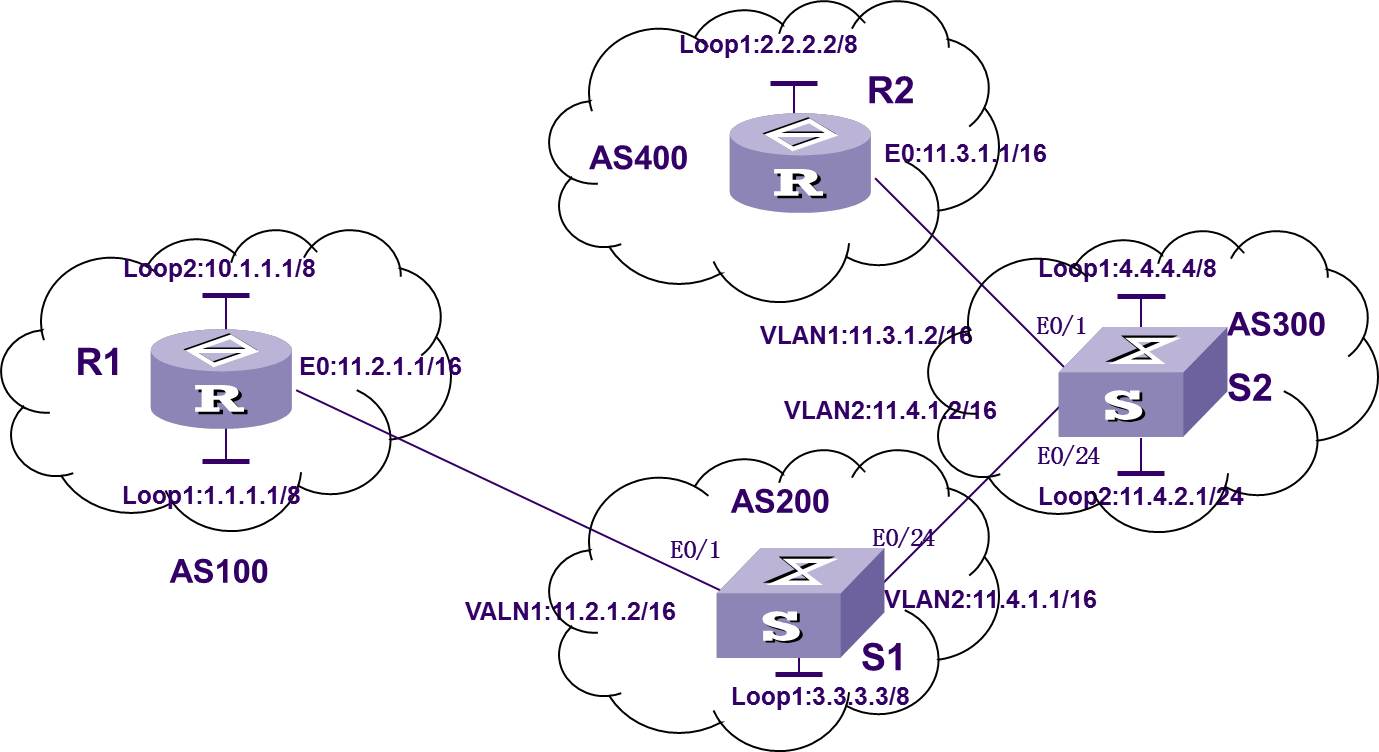
Bgp 200

Peer 10.0.0.2 as 200

Network 20.0.0.1 32

**16. 设计实验2**

按照图-13组网，设计并配置BGP，使得S1不向R1通告AS300内的任何路由。



基于AS-path的路由策略设计组网图

R2

Inter loop 1

Ip add 2.2.2.2 32

Inter e0/0

Ip add 11.3.1.1 16

Bgp 400

Peer 11.3.1.2 as 300

Network 2.2.2.2 32

R1

Inter e0/0

Ip add 11.2.1.1 16

Inter loop 1

Ip add 1.1.1.1 32

Inter loop 2

Ip add 10.1.1.1 32

Bgp 100

Peer 11.2.1.2 as 200

Network 10.1.1.1 32

Network 1.1.1.1 32

S1

Inter e1/0/1

Ip add 11.2.1.2 16

Inter e1/0/24

Ip add 11.4.1.1 16

Inter loop 1

Ip add 3.3.3.3 32

Bgp 200

Peer 11.2.1.1 as 100

Peer 11.4.2.1 as 300

Network 3.3.3.3 32

Quit

Ip as-path 1 deny \b300$

Ip as-path 1 permit ^ $

Bgp 200

Peer 11.2.1.1 as-path-acl 1 export

S2

Vlan 2

Port e1/0/24

Ip add 11.4.1.2 16

Inter vlan 1

ip add 11.3.1.2 16

inter loop 1

ip add 4.4.4.4 32

inter loop 2

ip add 11.4.2.1 24

bgp 300

peer 11.4.1.1 as 200

peer 11.3.1.1 as 400

network 11.4.2.1 32

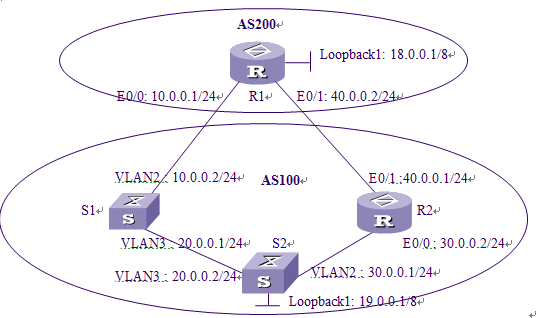
**17. 设计实验3**

要求：（1）配置Local-preference属性，实现S2到18.0.0.0/8的路由，优先选择S2->R2->R1。

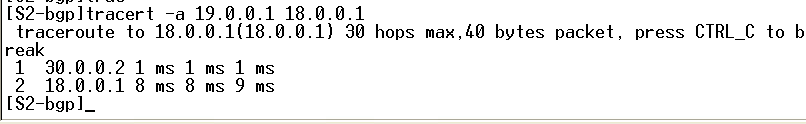
（2）配置Med属性，实现R1到19.0.0.0/8的路由优先选择R1->S1 -> S2。

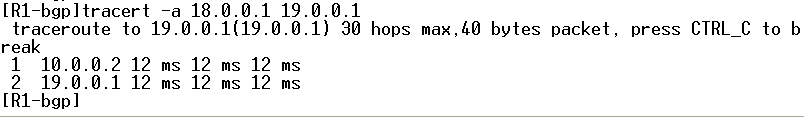
提示：med默认值为0，local-preference默认值为100。

配置命令：[S1-bgp]default med 10； [R1-bgp]default local-preference 10



应用Local-preference属性和Med属性实现路径控制组网图





S1

vlan 2

port e1/0/1

inter vlan 2

ip addr 10.0.0.2 24

vlan 3

port e1/0/24

inter vlan 3

ip addr 20.0.0.1 24

bgp 100

peer 10.0.0.1 as-number 200

peer 20.0.0.2 as-number 100

peer 20.0.0.2 next-hop-local

default local-preference 10

S2

vlan 2

port e1/0/24

inter vlan 2

ip addr 30.0.0.1 24

vlan 3

port e1/0/1

inter vlan 3

ip addr 20.0.0.2 24

inter loop1

ip addr 19.0.0.1 32

bgp 100

peer 30.0.0.2 as-number 100

peer 20.0.0.1 as-number 100

network 19.0.0.1 32

R1

Inter e0/0

Ip addr 10.0.0.1 24

Inter s1/0

Ip addr 40.0.0.2 24  
inter loop1

Ip addr 18.0.0.1 32

bgp 200

peer 10.0.0.2 as-number 100

peer 40.0.0.1 as-number 100

network 18.0.0.1 32

R2

Inter e0/0

Ip addr 30.0.0.2 24

Inter s1/0

Ip addr 40.0.0.1 24  
bgp 100

peer 40.0.0.2 as-number 200

peer 30.0.0.1 as-number 100

peer 30.0.0.1 next-hop-local

default med 10