

---

## BGP

通常可以将路由协议分为 IGP(内部网管协议)和 EGP（外部网关协议），EGP 主要用于 ISP 之间路由信息。目前使用最为广泛的 EGP 是 BGP 版本 4，它是第一个支持 CIDR 和路由汇总的 BGP 版本。RFC1772 对 BGP 有详细的定义。

### 1 BGP 概述

#### 1.1 BGP 特征

**BGP** 被称为路径向量路由协议，它的任务在自治系统之间交换路由信息，同时确保没有路由环路，其特征如下；

- ① 用属性(Attribute)描述路径，而不是用度量值；
- ② 使用 **TCP** (端口 **179**) 作为传输协议，继承了 **TCP** 的可靠性和面向连接的特性；
- ③ 通过 **Keepaliv** 信息来检验 **TCP** 的连接；
- ④ 具有丰富的属性特征，方便实现基于策略的路由；
- ⑤ 拥有自己的 **BGP** 表；
- ⑥ 支持 **VLSM** 和 **CIDR**；
- ⑦ 适合在大型网路中使用。

在详细讨论 **BGP** 之前，首先应用掌握如下 **BGP** 术语。

- ① 对等体 (**peer**)：当两台 **BGP** 路由器之间建立了一条基于 **TCP** 的连接后，就称它们为邻居或对等体；
- ② **AS**：是一组处于统一管理控制和策略下的路由器或主机、**AS** 号由 **Internet** 注册机构分配，范围为 **1~65 535**，其中 **64 512~65535** 是私有使用的；
- ③ **iBGP**；当 **BGP** 在一个 **AS** 内运行时，被称为内部 **BGP** (**iBGP**)；
- ④ **eBGP**；当 **BGP** 运行在 **AS** 之间时，被称为外部 **BGP**(**eBGP**)。

---

⑤ **NLRI**(网络层可达到信息): **BGP** 通过 **NLRI** 支持 **CIDR**, **NLRI** 是 **BGP** 更新报文的一部分, 用于列出可到达的目的的集合;

⑥ 同步;在 **BGP** 能通告路由之前, 该路由必须存在于当前的 **IP** 路由表中, 也就是说, **BGP** 和 **IGP** 必须在网络能被通告前同步, **Cisco** 允许通过命令“no synchronization”来关闭同步;

⑦ **iBGP** 水平分割; 通过 **iBGP** 学到的路由不能通告给其他的 **iBGP** 邻居..

## 1.2BGP 属性

**BGP** 具有丰富的属性, 为网络管理员进行路由控制带来很大的方便, **BGP** 路由属性分为以下 4 类.

### (1) 公认必遵 (Well—Known Mandatory)

**BGP** 更新报文必须包含的, 且必须被所有 **BGP** 厂商实现所能识别的属性包括 **ORIGIN**, **AS\_PATH** 和 **Next\_Hop**.

① **ORIGIN**(起源);这个属性说明起源路由是怎样放到 CD 表中的. 有 3 个可能的源——**IGP**, **EGP** 以及 **INCOMPLETE** 路由器在多个路由选者的处理中使用这个信息. 路由器选择具有最低 **ORIGIN** 类型的路径. **ORIGIN**; 类型从低处到高的顺序为;**IGP**<**EGP**<**INCOMPLETE**.

② **AS\_PATH** (**AS** 路径);指出包含在 **UPDATE** 报文中的路由信息所经过的自治系统的序列.

③ **NEXT\_HOP** (下一跳);路由器所获得的 **BGP** 路由的下一跳. 对 **EBGP** 会话来说, 下一跳就是通过该路由的邻居路由器的源地址. 对于 **IBGP** 会话, 有良种情况, 一是起源 **AS** 内部的路由的下一跳就是通告该路由器的源地址;二是由 **EBGP** 注入 **AS** 的路由, 它的下一跳会不变的带入 **IBGP** 中.

### (2) 公认自决 (Well-Known Discretionary)

该属性指必须被所有 **BGP** 实现所识别, 但是在 **BGP** 更新报文中可以发送, 也可以不发送的属性, 包括 **LOCAL\_PREF** 和 **ATOMIC\_AGGREGATE**。

---

①**LOCAL\_PREF (本地优先级)**：本地优先级属性是用于告诉自治系统内的路由器在有多条路径的时候，怎么离开自治系统。本地优先级越高，路由优先级越高。该属性仅仅在 IBGP 邻居之间传递。

②**ATOMIC\_AGGREGATE (原子聚合)**：原子聚合属性指出已被丢失了的信息。当路由聚合时将会导致信息的丢失，因为聚合来自具有不同属性的不同源。如果一个路由器发送了导致信息丢失的聚合，路由器被要求将原子聚合属性附加到该路由上。

(3) 可选过渡 (Optional Transitive)

可选过渡属性并不要求所有的 BGP 实现都支持。如果该属性不能被 BGP 进程识别，它就会去看过渡标志。如果过渡标志被设置，BGP 进程会接受这个属性并将它不加改变的传送，包括 AGGREGATOR 和 COMMUNITY。

①**AGGREGATOR (聚合者)**：此属性标明了实施路由聚合的 BGP 路由器 ID 和聚合路由的路由器的 AS 号。

②**COMMUNITY (团体)**：此属性指共享一个公共属性的一组路由器。

(4) 可选非过渡 (Optional Nontransitive)

可选非过渡属性并不要求所有的 BGP 实现都支持。如果这些属性被发送到不能对其实别的路由器，这些属性将会被丢弃，不能传送给 BGP 邻居，包括 MED，ORIGINATOR\_ID 和 CLUSTER\_LIST。

①**MED (多出口区分)**：该属性通知 AS 以外的路由器采用哪一条路径到达 AS，它也被认为是路由的外部度量，低 MED 值表示高的优先级。MED 属性在自治系统间交换，但 MED 属性不能传递到第三方 AS。

②**ORIGINATOR\_ID (起源 ID)**：路由反射器会附加到这个属性上，它携带本 AS 源路由器的路由器 ID，用以防止环路。

③**CLUSTER\_LIST (簇列表)**：此属性显示了采用的反射路径。

### 1.3 BGP 路由判定

BGP 使用了描述路由特性的很多属性，这些属性和每一个路由一起在 BGP 更新报文中被发送，路由器使用这些属性去悬着到目的地的最佳路由。理解 BGP 路由判定的过程很重要的，下面按优先顺序给出了路由器在 BGP 路径选择中的判定过程：

- 
- ①如果下一跳不可达，则不考虑该路由；
  - ②优先选取工具有最大权重（Weight）值得路径，权重是 Cisco 专有属性；
  - ③如果权重值相同，优先选取具有最高本地优先级的路由；
  - ④如果本地优先级相同，优先选区源自于本路由器（即县一跳为 “0.0.0.0”）上 BGP 的路由；
  - ⑤如果本地优先级相同，并且没有源自于本路由器的路由，优先选取具有最短 AS 路径的路由；
  - ⑥如果有相同的 AS 路径长度，优先选取具有最低起源代码（IGP<EGP<INCOMPLETE）的路由；
  - ⑦如果起源代码相同，优先选取具有最低 MED 值得路径；
  - ⑧如果 MED 都相同，在 EBGp 路由和联盟 EBGp 路由中，首选 EBGp 路由，在联盟 EBGp 路由和 IBGP 路由中，首选联盟 EBGp 路由；
  - ⑨如果前面所有属性都相同，优先选取离 IGP 邻居最近的路径；
  - ⑩如果内部路径也相同，优先选取具有最低 BGP 路由器 ID 的路径。

## 2 实验 1：iBGP 和 eBGP 基本配置

### 1. 实验目的

通过本实验可以掌握：

- ①启动 BGP 路由进程；
- ②BGP 进程中通告网络；
- ③IBGP 邻居配置；
- ④EBGP 邻居配置；
- ⑤BGP 路由更新源配置；
- ⑥next-hop-self 配置；
- ⑦BGP 路由汇总配置；

---

⑧BGP 路由调试;

## 2. 拓扑结构

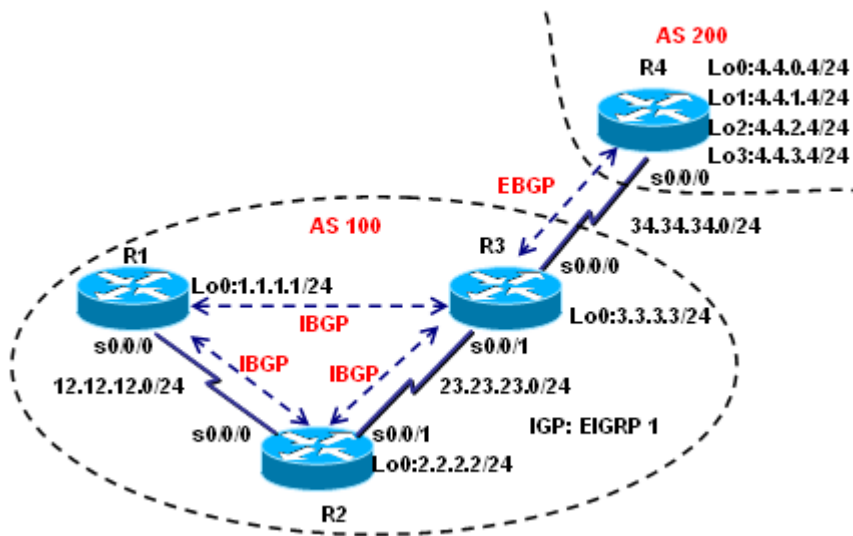


图 24-1

## 3. 实验步骤

因为本实验中 IBGP 的路由器 (R1, R2 和 R3) 形成全互联 (Full Mesh) 的邻居关系, 所以路由器 R1, R2 和 R3 均关闭同步。IBGP 路由器之间运行的 IGP 是 EIGRP, 为了提供 BGP 建立邻居关系的 TCP 连接和 BGP 下一跳可达。

### (1) 步骤 1: 配置路由器 R1

```
R1(config)#router eigrp 1
```

```
R1(config-router)#network 1.1.1.0 255.255.255.0
```

```
R1(config-router)#network 12.12.12.0 255.255.255.0
```

```
R1(config-router)#no auto-summary
```

```
R1(config)#router bgp 100 //启动 BGP 进程
```

```
R1(config-router)#no synchronization //关闭同步
```

```
R1(config-router)#bgp router-id 1.1.1.1 //配置 BGP 路由器 ID
```

```
R1(config-router)#neighbor 2.2.2.2 remote-as 100 //指定邻居路由器及所在的 AS
```

```
R1(config-router)#neighbor 2.2.2.2 update-source Loopback0 //制定更新源
```

---

```
R1(config-router)#neighbor 3.3.3.3 remote-as 100
```

```
R1(config-router)#neighbor 3.3.3.3 update-source Loopback0
```

```
R1(config-router)#network 1.1.1.0 mask 255.255.255.0 //通告网络
```

```
R1(config-router)#no auto-summary //关闭自动汇总
```

(2) 步骤 2: 配置路由器 R2

```
R2(config)#router eigrp 1
```

```
R2(config-router)#network 2.2.2.0 255.255.255.0
```

```
R2(config-router)#network 12.12.12.0 255.255.255.0
```

```
R2(config-router)#network 23.23.23.0 255.255.255.0
```

```
R2(config-router)#no auto-summary
```

```
R2(config)#router bgp 100
```

```
R2(config-router)#bgp router-id 2.2.2.2
```

```
R2(config-router)#neighbor 1.1.1.1 remote-as 100
```

```
R2(config-router)#neighbor 1.1.1.1 update-source Loopback0
```

```
R2(config-router)#neighbor 3.3.3.3 remote-as 100
```

```
R2(config-router)#neighbor 3.3.3.3 update-source Loopback0
```

```
R2(config-router)#no auto-summary
```

(3) 步骤 3: 配置路由器 R3

```
R3(config)#router eigrp 1
```

```
R3(config-router)#network 3.3.3.0 255.255.255.0
```

```
R3(config-router)#network 23.23.23.0 255.255.255.0
```

```
R3(config-router)#no auto-summary
```

---

```
R3(config)#router bgp 100
```

```
R3(config-router)#no synchronization
```

```
R3(config-router)#bgp router-id 3.3.3.3
```

```
R3(config-router)#neighbor 1.1.1.1 remote-as 100
```

```
R3(config-router)#neighbor 1.1.1.1 update-source Loopback0
```

```
R3(config-router)#neighbor 1.1.1.1 next-hop-self
```

//配置下一跳自我，即对从EBGP邻居传入的路由，在通告给IBGP邻居的同时，强迫路由通告自己是发送BGP更新的下一跳，而不是EBGP邻居

```
R3(config-router)#neighbor 2.2.2.2 remote-as 100
```

```
R3(config-router)#neighbor 2.2.2.2 update-source Loopback0
```

```
R3(config-router)#neighbor 2.2.2.2 next-hop-self
```

```
R3(config-router)#neighbor 34.34.34.4 remote-as 200
```

```
R3(config-router)#no auto-summary
```

(4) 步骤4：配置路由器R4

```
R4(config)#router bgp 200
```

```
R4(config-router)#no synchronization
```

```
R4(config-router)#bgp router-id 4.4.4.4
```

```
R4(config-router)#neighbor 34.34.34.3 remote-as 100
```

```
R4(config-router)#no auto-summary
```

```
R4(config-router)# network 4.4.0.0 mask 255.255.255.0
```

```
R4(config-router)# network 4.4.1.0 mask 255.255.255.0
```

```
R4(config-router)# network 4.4.2.0 mask 255.255.255.0
```

```
R4(config-router)# network 4.4.3.0 mask 255.255.255.0
```

---

```
R4(config-router)# network 4.4.0.0 mask 255.255.252.0
```

```
//用 network 做路由汇总通报
```

```
R4(config)#ip route 4.4.0.0 255.255.252.0 null0
```

```
//在 IGP 表中构造该汇总路由，否则不能用 network 通告
```

### 【技术要点】

①一台路由器只能启动一个 BGP 进程；

②命令” neighbor” 后边跟的是邻居路由器 BGP 路由更新源的地址。

③BGP 中的” network” 命令与 IGP 中存在的路由条目（可以是直连，静态路由或动态路由）在 BGP 中通告。同时” network” 命令参数” mask” 来通告单独的子网。如果 BGP 的自动汇总功能没有关闭，如果 BGP 的自动汇总能关闭，则通告必须严格匹配掩码长度。

④命令” neighbor” 后边跟” update-source” 参数，是由来指定更新源的，如果网络中有多条路径，那么用环回接口建立 TCP 连接，并作为 BGP 路由的更新源，会增加 BGP 的稳定性。

⑤命令” neighbor” 后边跟” next-hop-self” 参数是为了解决下一跳可达的问题，因为当路由通过 EBGp 注入到 AS 时，从 EBGp 获得的下一跳会被不变的在 IBGP 中传递，” next-hop-self” 参数使得路由器会把自己作为发送 BGP 更新的下一跳来通报给 IBGP 邻居。

⑥BGP 的下一跳是指 BGP 路由表中路由条目的下一跳，也就是相应” network” 命令所指的地址。

## 4. 实验调试

(1) show tcp brief

该命令用来查看 TCP 连接信息摘要。

```
R3#show tcp brief
```

| TCP      | Local Address | Foreign Address | (state) |
|----------|---------------|-----------------|---------|
| 64752BAC | 3.3.3.3.11002 | 1.1.1.1.179     | ESTAB   |
| 64753B5C | 3.3.3.3.11000 | 2.2.2.2.179     | ESTAB   |

---

```
6472708          34.34.34.3.11001  34.34.34.4.179      ESTAB
```

以上输出标明路由器 R3 和路由器 R1, R2 和 R4 的 179 端口建立了 TCP 连接。建立 TCP 连接的双方使用 BGP 路由更新源的地址。只要两台路由器之间建立了一条 TCP 连接, 就可以形成 BGP 邻居关系。

(2) show ip bgp neighbors

该命令用来查看邻居的 TCP 和 BGP 连接的详细信息。

```
R3#show ip bgp neighbors 34.34.34.4
```

```
BGPneighbor is34.34.34.4    remote AS 200.extenal link
```

```
BGP version 4.remote router ID 4.4.4.4
```

```
BGP stae=Established, up for 00;50;29
```

```
Last read 00;00;21, hold time is 180, keepalive interval is 60 seconds
```

```
Neighbor capabilities;
```

```
Route refresh;advertised and received (old & new)
```

```
Address family IPv4 Unicast;advertised and received
```

```
.....
```

以上输出表明路由器有一个外部 BGP 邻居路由器 R4 (34.34.34.4) 在 AS 200, 此邻居的路由器 ID 号是 4.4.4.4。命令” show ip bgp neighbors” 显示出的信息最重要的一部分是” BGP stae= “那一行, 此行给出了 BGP 连接的状态。” Established” 状态表示 BGP 对等体间的会话是打开的并正在运行。如果显示的是其他状态, 如 Idle, Connect, Active, OpenSent 或 OpenConfirm, 那就。存在问题

(3) show ip bgp summary

该命令用来查看 BGP 连接的摘要信息。

```
R3#show ip bgp summary
```

```
BGP router identifier 3.3.3.3, local AS number 100
```

```
//路由器 ID 及本地 AS
```

---

```
BGP table version is 11, main routing table version 11
```

```
//BGP 表的内部版本号 (BGP 表变化时号码会逐此加 1) 和注入到主路由表的最后版本号
```

```
5 network entries using 505 bytes of memory
```

```
//网络条目和使用的 memory
```

```
5 path entries using 505 bytes of memorymemory
```

```
//路径条目和使用的 memory
```

```
2 BGP path attribute entries using 120 bytes of memory
```

```
1 BGP AS-PATH entries using 24 bytes of memory
```

```
0 BGP route-map cache entries using 0 bytes of memory
```

```
0 BGP filter-list cache entries using 0 bytes of memory
```

```
BGP using 889 total bytes of memory
```

```
BGP activity 5/0 prefixes, 6/1paths, scan interval 60 secs
```

```
Neighbor      V AS   MsgRcvd  MsgSent  TblVer  InQ  OutQ  Up/Down   State/pfxrcd
1.1.1.1       4 100    80       81       11    0    0    00:38:29  1
2.2.2.2       4 100    74       77       11    0    0    01:12:46  0
34.34.34.4    4 200    71       74       11    0    0    01:07:47  4
```

以上输出的邻居表的各个字段的含义如下所述.

①Neighbor;BGP 邻居的 ID;

②V;BGP 的版本为 4;

③AS;邻居所在的 AS 号码;

---

④MsgRcvd;接受的信息;

⑤MsgSent;发送的信息;

⑥TblVer;BGP 表的内部版本号;

⑦Up/Down;邻居关系建立的时间;

⑧State/PfxRcd;BGP 连接的状态或者通告的路由前缀。

(4) show ip bgp

该命令用来查看 BGP 表的信息。

```
R3#show ip bgp
```

```
BGP table version is 11, local router ID is 3.3.3.3
```

```
//BGP 表的内部版本号和本路由器的 BGP 路由器 ID
```

```
Status codes: s suppressed, d damped, h history, * valid, >best, i-internal
```

```
                r RIP-failure, S Stale
```

```
Origin codes: i-IGP, e-EGP, ?-incomplete
```

| Network        | Next Hop   | Metric | LocPrf | Weight | Path |
|----------------|------------|--------|--------|--------|------|
| r>i 1.1.1.0/24 | 1.1.1.1    | 0      | 100    | 0      | i    |
| *> 4.4.0.0/24  | 34.34.34.4 | 0      |        | 0      | 200i |
| *> 4.4.0.0/22  | 34.34.34.4 | 0      |        | 0      | 200i |
| *> 4.4.1.0/24  | 34.34.34.4 | 0      |        | 0      | 200i |
| *> 4.4.2.0/24  | 34.34.34.4 | 0      |        | 0      | 200i |
| *> 4.4.3.0/24  | 34.34.34.4 | 0      |        | 0      | 200i |

在以上输出中，路由条目表项的状态代码（Status Codes）的含义解释如下所述。

①s;表示路由条目被抑制;

- 
- ②d;表示路由条目由于被惩罚而受到抑制，从而阻止了不稳定路由的发布；
  - ③h;表示该路由正在被惩罚，但还未到抑制阈值而使它被抑制；
  - ④\*;表示该路由条目有效；
  - ⑤>;表示该路由条目最优，可以被传递，达到最优的重要前提是下一跳可达；
  - ⑥i;表示该路由条目是从 IBGP 邻居学到的；
  - ⑦r;表示将 BGP 表中的路由条目放入到 IP 路由表中失败。

下面具体地解释 BGP 路由条目：

“r>i1.1.1.0/24 1.1.1.1 0 100 0 i” 的含义。

①r;因为路由器 R3 通过 EIGRP 学到” 1.1.1.0/24” 路由条目，其管理距离为 90，而通过 IBGP 学到” 1.1.1.0/24” 路由条目的管理距离是 200，而且关闭了同步，BGP 表中的路由条目放入到 IP 路由表中失败，所以出现代码” r” ；

- ②>;表示该路由条目最优，可以被传递；
- ③i;表示该路由条目是从 IBGP 邻居学到的；
- ④1.1.1.1: 表示该 BGP 路由的下一跳；
- ⑤0(标题栏对应 Metric): 表示该路由外部度量值即 MED 值为 0；
- ⑥100; 表示该路由本地优先级为 100；
- ⑦0(标题栏对应 Weight): 表示该路由的权重值为 0，如果是本地生产的，默认权重值是 32 768；如果是从邻居学来的，默认权重值为 0；
- ⑧由于该路由是通过相同 AS 的 IBGP 邻居传地来，所以 PATH 字段为空；
- ⑨i;表示该路由条目源为 IGP 它是路由器 R1 用” network” 命令通告的。

(5) show ip route

```
R1#show ip route
```

```
Codes:C-connected, S-static, R-RIP, M-mobile, B-BGP
```

```
D-EIGRP, EX-EIGRP external, O-OSPF, IA-OSPFinter area
```

---

N1-OSPF NSSAexternal type 1, N2-OSPFexternal type 2

E1-OSPFexternal type 1, E2-OSPFexternal type 2

i-IS-IS, L1-IS-IS, L2-IS-IS level-2, ia-IS-IS inter area

\*-candidate default, U-per-user static route, o-ODR

P-periodic downloaded static route

Gateway of last resort is not set

1.0.0.0/24 is subnetted, 1 subnets

C 1.1.1.0 is directly connected, Loopback0

2.0.0.0/24 is subnetted, 1 subnets

D 2.2.2.0[90/2297856]via 12.12.12.2, 03:03:44, Serial0/0/0

3.0.0.0/24 is subnetted, 1 subnets

D 3.3.3.0[90/2809856]via 12.12.12.2, 03:03:44, Serial0/0/0

4.0.0.0/8 is subnetted, 1 subnets

B 4.4.0.0/24[200/0] via 3.3.3.3, 03:02:52

B 4.4.0.0/22[200/0] via 3.3.3.3, 01:37:48

B 4.4.1.0/24[200/0] via 3.3.3.3, 03:02:52

B 4.4.2.0/24[200/0] via 3.3.3.3, 03:02:52

B 4.4.3.0/24[200/0] via 3.3.3.3, 03:02:52

23.0.0.0/24 is subnetted, 1 subnets

D 23.23.23.0[90/2681856]via 12.12.12.2, 03:03:45, Serial0/0/0

12.0.0.0/24 is subnetted, 1 subnets

---

C 12.12.12.0 is directly connected, Serial0/0/0

R3#show ip route

Codes;C-connected, S-static, R-RIP, M-mobile, B-BGP

D-EIGRP, EX-EIGRP external, O-OSPF, IA-OSPFinter area

N1-OSPF NSSAexternal type 1, N2-OSPFexternal type 2

E1-OSPFexternal type 1, E2-OSPFexternal type 2

i-IS-IS, L1-IS-IS, L2-IS-IS level-2, ia-IS-IS inter area

\*-candidate default, U-per-user static route, o-ODR

P-periodic downloaded static route

Gateway of last resort is not set

34.0.0.0/24 is sudetted, 1 sudnets

C 34.34.34.0 is directly connected, Serial0/0/0

1.0.0.0/24 is sudetted, 1 sudnets

D 1.1.1.0[90/2809856]via 12.12.12.2, 02;17;48, Serial0/0/0

3.0.0.0/24 is sudetted, 1 sudnets

D 2.2.2.0[90/2297856]via 23.23.23.2, 02;51;36, Serial0/0/0

3.0.0.0/24 is sudetted, 1 sudnets

B 4.4.0.0/24[20/0] via 34.34.34.4, 02;45;33

B 4.4.0.0/22[20/0] via 34.34.34.4, 00;51;20

B 4.4.1.0/24[20/0] via 34.34.34.4, 02;45;33

B 4.4.2.0/24[20/0] via 34.34.34.4, 02;45;33

---

```
B    4.4.3.0/24[20/0] via 34.34.34.4, 02;45;34
```

```
23.0.0.0/24 is sudetted, 1 sudnets
```

```
C    23.23.23.0 is directly connected, Serial0/0/0
```

```
12.0.0.0/24 is sudetted, 1 sudnets
```

```
D    12.12.12.0[90/2681856]via 23.23.23.2, 02;17;53, Serial0/0/1
```

以上输出表明 IBGP 的管理距离是 200，EBGP 的管理距离是 20.

(6)ping

在路由器 R1 上 ping4.4.0.4，结果是不通的，原因很简单，就是路由器 R1 和 R2 路由表中没有 34.34.34.0 的路由，此时如果执行扩展 ping，就是通的：

```
R1#ping
```

```
Protocol[ip];
```

```
Target IP address;4.4.0.4
```

```
Repeat count[5];2
```

```
Datagram size[100];
```

```
Timeout in seconds[2];
```

```
Extended commands[n];y
```

```
Source address or interface;.1.1.1.1
```

```
Type of service[0];
```

```
Set DF bit in IP header?[on];
```

```
Validate reply data?[on];
```

```
Data pattern [0xABCD];
```

```
Looes, Strict, Record, Timestamp, Verbose[none];
```

```
Sweep range of sizes [n];
```

---

```
Type escape sequence to abort.
```

```
Sending 2, 100-byte ICMP Echos to 4.4.0.4, timeout is 2 seconds;
```

```
Pasket sent with a source address of 1.1.1.1
```

```
!!
```

如果一定要标准 ping，无非就是让路由器 R1 和 R2 学到” 34.34.34.0” 的路由，方法很多，比如在路由器 R3 上重分布直连。

(7) 在 R1 上打开 BGP 同步，然后查看 BGP 表

```
R1(config)#router bgp 100
```

```
R1(config-router)#synchronization //打开同步
```

```
R1#clear ip bgp* //重置 BGP 连接
```

```
R1#show ip bgp
```

```
BGP table version is 1, local routerID is 1.1.1.1
```

```
Status codes: s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
          r RIB-failure, S Stale
```

```
Origin codes: i-IGP, e-EGP, ?-incomplete
```

| Network      | Next Hop | Metric | LocPrf | Weight | Path |
|--------------|----------|--------|--------|--------|------|
| *i4.4.0.0/24 | 3.3.3.3  | 0      | 100    | 0      | 200i |
| *i4.4.0.0/22 | 3.3.3.3  | 0      | 100    | 0      | 200i |
| *i4.4.1.0/24 | 3.3.3.3  | 0      | 100    | 0      | 200i |
| *i4.4.2.0/24 | 3.3.3.3  | 0      | 100    | 0      | 200i |
| *i4.4.3.0/24 | 3.3.3.3  | 0      | 100    | 0      | 200i |

以上输出表明 BGP 路由不是被优化的，因为 IGP 的路由表中并没有这些路由条目。

---

(8) R1, R2, R3 之间的邻居关系删除路由器 R1 和 R3 之间的邻居关系, 保持路由器 R1 和 R2 建立邻居关系, 建立路由器 R2 和 R3 建立邻居关系, 操作如下:

```
R1(config)#router bgp 100
```

```
R1(config-router)#no synchronization
```

```
R1(config-router)#no neighbor 3.3.3.3
```

```
R3(config)#router bgp 100
```

```
R3(config-router)#no neighbor 1.1.1.1
```

在路由器 R1 和 R2 上查看 BGP 表

```
R1#show ip bgp
```

```
BGP table version is 2, local routerID is 1.1.1.1
```

```
Status codes: s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
                r RIB-failure, S Stale
```

```
Origin codes: i-IGP, e-EGP, ?-incomplete
```

| Network      | Next Hop | Metric | LocPrf | Weight | Path |
|--------------|----------|--------|--------|--------|------|
| *>1.1.1.0/24 | 0.0.0.0  | 0      | 100    | 32768  | i    |

```
R2#show ip bgp
```

```
BGP table version is 8, local routerID is 2.2.2.2
```

```
Status codes: s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
                r RIB-failure, S Stale
```

```
Origin codes: i-IGP, e-EGP, ?-incomplete
```

| Network | Next Hop | Metric | LocPrf | Weight | Path |
|---------|----------|--------|--------|--------|------|
|---------|----------|--------|--------|--------|------|

---

|              |         |   |     |   |      |
|--------------|---------|---|-----|---|------|
| r>1.1.1.0/24 | 1.1.1.1 | 0 | 100 | 0 | i    |
| *i4.4.0.0/24 | 3.3.3.3 | 0 | 100 | 0 | 200i |
| *i4.4.0.0/22 | 3.3.3.3 | 0 | 100 | 0 | 200i |
| *i4.4.1.0/24 | 3.3.3.3 | 0 | 100 | 0 | 200i |
| *i4.4.2.0/24 | 3.3.3.3 | 0 | 100 | 0 | 200i |
| *i4.4.3.0/24 | 3.3.3.3 | 0 | 100 | 0 | 200i |

以上输出表明路由器 R2 并没有将路由器 R3 通告的路由通告给路由器 R1，这进一步验证了 IBGP 水平分割的基本原理；通过 IBGP 学到的路由不能通告给相同 AS 内的其他的 IBGP 邻居。通常的解决办法由两个：IBGP 形成全互连邻居关系或使用路由反射器。

### 3 实验 2 BGP 地址聚合

#### 1. 实验目的

通过本实验可以掌握：

- ①启动 BGP 路由进程
- ②BGP 中通告网络
- ③EBGP 邻居配置
- ④BGP 地址聚合配置和调试
- ⑤地址聚合参数” as-set” 含义
- ⑥地址聚合参数” summary-only” 含义
- ⑦地址聚合参数” suppress-map” 含义

#### 2. 拓扑结构

实验拓扑图如图 24-2 所示。

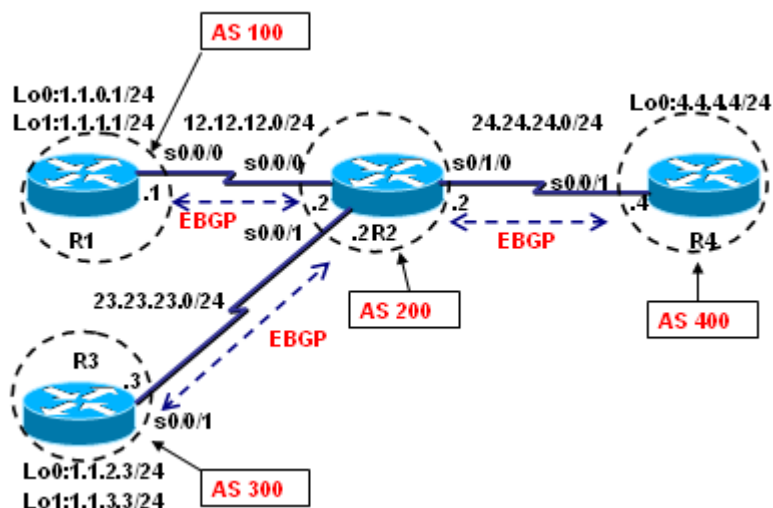


图 24-2

### 3. 实验拓扑

本实验在路由器 R2 上将路由器 R1 和路由器 R3 通告的环回接口的路由进行地址聚合，并通告给路由器 R4。在路由器 R1、R3、R4、配置静态路由实现网络互通。

#### (1) 步骤 1：配置路由器 R1

```
R1(config)#router bgp 100

R1(config-router)#no synchronization

R1(config-router)#no auto-summary

R1(config-router)#bgp router-id 1.1.1.1

R1(config-router)#neighbor 12.12.12.2 remote-as 200

R1(config-router)#network 12.12.12.2 mask 255.255.255.0

R1(config-router)#network 1.1.0.0 mask 255.255.255.0

R1(config-router)#network 1.1.1.0 mask 255.255.255.0

R1(config)#ip route 24.24.24.0 255.255.255.0 12.12.12.2

R1(config)#ip route 23.23.23.0 255.255.255.0 12.12.12.2
```

#### (2) 步骤 2：配置路由器 R2

---

```
R2(config)#router bgp 200
```

```
R2(config-router)#no synchronization
```

```
R2(config-router)#no auto-summary
```

```
R2(config-router)#bgp router-id 2.2.2.2
```

```
R2(config-router)#neighbor 12.12.12.1 remote-as 100
```

```
R2(config-router)#neighbor 23.23.23.3 remote-as 300
```

```
R2(config-router)#neighbor 24.24.24.4 remote-as 400
```

```
R2(config-router)#aggregate-address 1.1.0.0 255.255.252.0//配置地址聚合
```

(3) 步骤3: 配置路由器 R3

```
R3(config)#router bgp 300
```

```
R3(config-router)#no synchronization
```

```
R3(config-router)#no auto-summary
```

```
R3(config-router)#bgp router-id 3.3.3.3
```

```
R3(config-router)#neighbor 23.23.23.2 remote-as 200
```

```
R3(config-router)# network 1.1.2.0 mask 255.255.255.0
```

```
R3(config-router)# network 1.1.3.0 mask 255.255.255.0
```

```
R3(config)#ip route 12.12.12.0 255.255.255.0 23.23.23.2
```

```
R3(config)#ip route 24.24.24.0 255.255.255.0 23.23.23.2
```

(4) 步骤4: 配置路由器 R4

```
R4(config)#router bgp 400
```

```
R4(config-router)#no synchronization
```

```
R4(config-router)#no auto-summary
```

```
R4(config-router)#bgp router-id 4.4.4.4
```

---

```
R4(config-router)#neighbor 24.24.24.2 remote-as 200
```

```
R4(config-router)#network 4.4.4.0 mask 255.255.255.0
```

```
R4(config)#ip route 12.12.12.0 255.255.255.0 24.24.24.2
```

```
R4(config)#ip route 23.23.23.0 255.255.255.0 24.24.24.2
```

#### 4. 实验调试

(1) 在路由器 R1、R4 上查看 BGP 表

```
R1(config)#router bgp 100
```

```
R1(config-router)#synchronization
```

```
R1#clear ip bgp
```

```
R1#show ip bgp
```

```
BGP table version is 21, local routerID is 1.1.1.1
```

```
Status codes: s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
          r RIB-failure, S Stale
```

```
Origin codes: i-IGP, e-EGP, ?-incomplete
```

| Network      | Next Hop   | Metric | LocPrf | Weight | Path     |
|--------------|------------|--------|--------|--------|----------|
| *>1.1.0.0/24 | 0.0.0.0    | 0      |        | 32768  | i        |
| *>1.1.0.0/22 | 12.12.12.2 | 0      |        | 0      | 200i     |
| *>1.1.1.0/24 | 0.0.0.0    | 0      |        | 32768  | i        |
| *>1.1.2.0/24 | 12.12.12.2 | 0      |        | 0      | 200 300i |
| *>1.1.3.0/24 | 12.12.12.2 | 0      |        | 0      | 200 300i |
| *>4.4.4.0/24 | 12.12.12.2 | 0      |        | 0      | 200 400i |

```
R4(config)#router bgp 400
```

```
R4(config-router)#synchronization
```

```
R4#clear ip bgp
```

```
R4#show ip bgp
```

```
BGP table version is 45, local routerID is 4.4.4.4
```

```
Status codes: s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
          r RIB-failure, S Stale
```

```
Origin codes: i-IGP, e-EGP, ?-incomplete
```

| Network      | Next Hop | Metric | LocPrf | Weight | Path |
|--------------|----------|--------|--------|--------|------|
| *>1.1.0.0/24 | 2        |        | 0      | 200    | 100i |
| *>1.1.0.0/22 | 2        | 0      | 0      | 200    | i    |
| *>1.1.1.0/24 | 2        |        | 0      | 200    | 100i |
| *>1.1.2.0/24 | 2        |        | 0      | 200    | 300i |
| *>1.1.3.0/24 | 2        |        | 0      | 200    | 300i |
| *>4.4.4.0/24 |          | 0      |        | 32768  | i    |

以上输出表明：

①在路由器 R1、R4 接收到” 1.1.0.0/22” 聚合路由，通过 AS-PATH 属性可以看出，执行地址聚合的路由器 R2 成为新路由的创造者，原来 AS-PATH 属性丢失；

②路由器 R4 同时也接收到 4 条明细路由在显示的 AS-PATH 序列中，路由的始发 AS 在列表的末端（右侧），每个接收到该路由，并把它传递给其他 AS 的 BGP 对等体会把它自己的 AS 追加在列表的开头（左侧）；

③BGP 路由器下一跳为”0.0.0.0”，表示该 BGP 路由起源本地，Weight 值为”32 768”；

④因为所有 BGP 路由条目的代码为”\*>”，所以所有 BGP 路由条目都为最优。

（2）” as-set” 参数

“as-set”参数可以使 BGP 聚合路由不丢失原来的 AS-PATH 属性，从而避免路由环路，在路由器 R2 上操作如下：

```
R2(config-router)#aggregate-address 1.1.0.0 255.255.252.0 as-set
```

在路由器 R1、R4 上查看 BGP 表：

```
R1#show ip bgp
```

```
BGP table version is 22, local routerID is 1.1.1.1
```

```
Status codes: s suppressed, d damped, h history, *valid, .best, i-internal,
               r RIB-failure, S Stale
```

```
Origin codes: i-IGP, e-EGP, ?-incomplete
```

| Network      | Next Hop   | Metric | LocPrf | Weight | Path     |
|--------------|------------|--------|--------|--------|----------|
| *>1.1.0.0/24 | 0.0.0.0    | 0      |        | 32768  | i        |
| *>1.1.1.0/24 | 0.0.0.0    | 0      |        | 32768  | i        |
| *>1.1.2.0/24 | 12.12.12.2 |        |        | 0      | 200 300i |
| *>1.1.3.0/24 | 12.12.12.2 |        |        | 0      | 200 300i |
| *>4.4.4.0/24 | 12.12.12.2 |        |        | 0      | 200 400i |

```
R4#show ip bgp
```

```
BGP table version is 56, local routerID is 4.4.4.4
```

```
Status codes: s suppressed, d damped, h history, *valid, .best, i-internal,
               r RIB-failure, S Stale
```

```
Origin codes: i-IGP, e-EGP, ?-incomplete
```

| Network | Next Hop | Metric | LocPrf | Weight | Path |
|---------|----------|--------|--------|--------|------|

---

|              |         |   |   |       |              |
|--------------|---------|---|---|-------|--------------|
| *>1.1.0.0/24 | 2       |   | 0 | 200   | 100i         |
| *>1.1.0.0/22 | 2       | 0 | 0 | 200   | {100, 300} i |
| *>1.1.1.0/24 | 2       |   | 0 | 200   | 100i         |
| *>1.1.2.0/24 | 2       |   | 0 | 200   | 300i         |
| *>1.1.3.0/24 | 2       |   | 0 | 200   | 300i         |
| *>4.4.4.0/24 | 0.0.0.0 | 0 |   | 32768 | i            |

以上输出表明：

①在路由器 R4 上接受到的汇总路由” 1.1.0.0/22” 中，AS-PATH 包含了被聚合路由中所有的 AS 号码的集合” {100, 300}” ；

②聚合路由正是由于携带了所有的 AS，所以在路由器 R1 的 BGP 表中没有出现，当然在路由器 R3 的 BGP 表中也不会出现。

#### 【技术要点】

BGP 使用 AS-PATH 属性作为路由更新的一部分来确保没有路由环路。因为在 BGP 对等体之间传递的每条路由都携带它所经过的 AS 号码序列表，如果该路由被通告给它始发的 AS，该 AS 路由器将在 AS 序列表中看到自己的 AS，它将不接受该路由。以下的输出充分的说明了这一点。

```
R2#show ip bgp neighbor 12.12.12.1 advertised-routes
```

```
BGP table version is 8, local routerID is 2.2.2.2
```

```
Status cods; s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
          r RIB-failure, S Stale
```

```
Origin codes; i-IGP, e-EGP, ?-incomplete
```

| Network | Next Hop | Metric | LocPrf | Weight | Path |
|---------|----------|--------|--------|--------|------|
|---------|----------|--------|--------|--------|------|

|              |            |     |       |            |
|--------------|------------|-----|-------|------------|
| *>1.1.0.0/22 | 0.0.0.0    | 100 | 32768 | {100,300}i |
| *>1.1.2.0/24 | 23.23.23.3 | 0   | 0     | 300i       |
| *>1.1.3.0/24 | 23.23.23.3 | 0   | 0     | 300i       |
| *>4.4.4.0/24 | 4          | 0   | 0     | 400i       |

以上输出表明路由器 R2 仍然向邻居 12.12.12.1 发送聚合路由” 1.1.0.0/22”。

```
R1#show ip bgp neighbor 12.12.12.2 received-routes
```

```
BGP table version is 22, local routerID is 1.1.1.1
```

```
Status codes: s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
                r RIB-failure, S Stale
```

```
Origin codes: i-IGP, e-EGP, ?-incomplete
```

| Network      | Next Hop   | Metric | LocPrf | Weight | Path     |
|--------------|------------|--------|--------|--------|----------|
| *>1.1.2.0/24 | 12.12.12.2 |        |        | 0      | 200 300i |
| *>1.1.3.0/24 | 12.12.12.2 |        |        | 0      | 200 300i |
| *>4.4.4.0/24 | 12.12.12.2 |        |        | 0      | 200 400i |

以上输出表明路由器 R1 没有接收聚合路由” 1.1.0.0/22”，因为它发现聚合路由条目中的 S-PATH 属性列表” {100, 300}”中包含自己的 AS 号码 100，所以不接收。同理，路由器 R3 也不会接收该聚合路由条目。

### 【提示】

要执行” show ip bgp neighbor 12.12.12.2 received-routes”命令，必须完成下面这条命令：

```
R1(config-router)#neighbor 12.12.12.2 soft-reconfiguration inbound
```

(3) ” summary-only” 参数

---

如果在路由器 R4 上只想看到汇总路由，没有明细路由，” summary-only” 参数可以实现。在路由器 R2 上的配置如下；

```
R2(config-router)#aggregate-address 1.1.0.0 255.255.252.0 as-set summary-only
```

在路由器 R2、R4 上查看 BGP 表；

```
R2#show ip bgp
```

```
BGP table version is 14, local routerID is 2.2.2.2
```

```
Status codes: s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
                r RIB-failure, S Stale
```

```
Origin codes: i-IGP, e-EGP, ?-incomplete
```

| Network      | Next Hop   | Metric | LocPrf | Weight | Path       |
|--------------|------------|--------|--------|--------|------------|
| s>1.1.0.0/24 | 12.12.12.1 | 0      |        | 0      | 100i       |
| *>1.1.0.0/22 | 0.0.0.0    |        | 100    | 32768  | {100,300}i |
| s>1.1.1.0/24 | 12.12.12.1 | 0      |        | 0      | 100i       |
| s>1.1.2.0/24 | 23.23.23.3 | 0      |        | 0      | 300i       |
| s>1.1.3.0/24 | 23.23.23.3 | 0      |        | 0      | 300i       |
| *>4.4.4.0/24 | 4 0        |        | 0      | 400i   |            |

```
R4#show ip bgp
```

```
BGP table version is 62, local routerID is 4.4.4.4
```

```
Status codes: s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
                r RIB-failure, S Stale
```

```
Origin codes: i-IGP, e-EGP, ?-incomplete
```

| Network      | Next Hop | Metric | LocPrf | Weight           | Path |
|--------------|----------|--------|--------|------------------|------|
| *>1.1.0.0/22 | 2 0      |        | 0      | 200 {100, 300} i |      |
| *>4.4.4.0/24 | 0.0.0.0  | 0      |        | 32768            | i    |

以上输出表明

①路由器 R2 上所有被聚合的明细路由被标记为” s”，表示被抑制，不被发送；

②路由器 R4 只接收到一条聚合路由” 1.1.0.0/22” 如果不加” as-set” 参数，则路由器 R1、R3 也会接收到该聚合路由。

#### (4) 抑制部分明细路由

如果有特殊要求，在聚合后只抑制部分明细路由条目，参数” summary-map” 可以完成。本实验要求路由器 R2 地址聚合后，路由器 R1 的两条明细路由被抑制，而路由器 R3 的明细路由要求传送给路由器 R4，路由器 R2 配置步骤如下：

```
R2(config)#ip prefix-list 1 permit 1.1.0.0/24 //匹配路由条目，以便进行控制
```

```
R2(config)# ip prefix-list 1 permit 1.1.1.0/24
```

```
R2(config)#route-map sup permit 10
```

```
R2(config-router-map)#match ip address prefix-list 1
```

```
R2(config)#router bgp 200
```

```
R2(config-router)#aggregate-address 1.1.0.0 255.255.252.0 as-set suppress-map
sup
```

分别在 4 台路由器上查看 BGP 表

```
R1#show ip bgp
```

```
BGP table version is 28, locl routerID is 1.1.1.1
```

```
Status cods; s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
r RIB-failure, S Stale
```

```
Origin codes;i-IGP, e-EGP, ?-incomplete
```

|  |  |  |  |  |  |
|--|--|--|--|--|--|
|  |  |  |  |  |  |
|--|--|--|--|--|--|

| Network      | Next Hop   | Metric | LocPrf | Weight | Path     |
|--------------|------------|--------|--------|--------|----------|
| *>1.1.0.0/24 | 0.0.0.0    | 0      |        | 32768  | i        |
| *>1.1.1.0/24 | 0.0.0.0    | 0      |        | 32768  | i        |
| *>1.1.2.0/24 | 12.12.12.2 |        |        | 0      | 200 300i |
| *>1.1.3.0/24 | 12.12.12.2 |        |        | 0      | 200 300i |
| *>4.4.4.0/24 | 12.12.12.2 |        |        | 0      | 200 400i |

R2#show ip bgp neighbor 12.12.12.1 advertised-routes

BGP table version is 20, locl routerID is 2.2.2.2

Status cods; s suppressed, d damped, h history, \*valid, .best, i-internal,

r RIB-failure, S Stale

Origin codes;i-IGP, e-EGP, ?-incomplete

|  |  |  |  |  |  |
|--|--|--|--|--|--|
|  |  |  |  |  |  |
|--|--|--|--|--|--|

| Network      | Next Hop   | Metric | LocPrf | Weight | Path        |
|--------------|------------|--------|--------|--------|-------------|
| s>1.1.0.0/24 | 12.12.12.1 | 0      |        | 0      | 100i        |
| *>1.1.0.0/22 | 0.0.0.0    |        | 100    | 32768  | {100,300} i |
| s>1.1.1.0/24 | 12.12.12.1 | 0      |        | 0      | 100i        |
| s>1.1.2.0/24 | 23.23.23.3 | 0      |        | 0      | 300i        |
| s>1.1.3.0/24 | 23.23.23.3 | 0      |        | 0      | 300i        |
| *>4.4.4.0/24 | 4 0        |        | 0      | 400i   |             |

R3#show ip bgp

BGP table version is 58, locl routerID is 3.3.3.3

Status cods; s suppressed, d damped, h history, \*valid, .best, i-internal,

---

```
      r RIB-failure, S Stale
```

```
Origin codes: i-IGP, e-EGP, ?-incomplete
```

| Network      | Next Hop   | Metric | LocPrf | Weight | Path     |
|--------------|------------|--------|--------|--------|----------|
| *>1.1.2.0/24 | 0.0.0.0    | 0      |        | 32768  | i        |
| *>1.1.3.0/24 | 0.0.0.0    | 0      |        | 32768  | i        |
| *>4.4.4.0/24 | 23.23.23.2 | 0      |        | 0      | 200 400i |

```
R4#show ip bgp
```

```
BGP table version is 64, local routerID is 4.4.4.4
```

```
Status codes: s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
      r RIB-failure, S Stale
```

```
Origin codes: i-IGP, e-EGP, ?-incomplete
```

| Network      | Next Hop | Metric | LocPrf | Weight         | Path |
|--------------|----------|--------|--------|----------------|------|
| *>1.1.0.0/22 | 2 0      |        | 0      | 200 {100, 300} | i    |
| *>1.1.2.0/24 | 2        | 0      |        | 200 300        | i    |
| *>1.1.3.0/24 | 2        |        | 0      | 200 300        | i    |
| *>4.4.4.0/24 | 0.0.0.0  | 0      |        | 32768          | i    |

以上输出表明

①由于在路由器 R2 上将路由器 R1 的明细路由” 1.1.0.0/24” 和” 1.1.1.0/24” 抑制，所以路由器 R3 和 R4 不能接收到；

②由于在路由器 R2 上没有将明细路由” 1.1.2.0/24” 和” 1.1.3.0/24” 抑制，没有配置” summary-only” 参数，所以 4 台路由器全部接收到” 1.1.2.0/24” 和” 1.1.3.0/24” 的路由条目；

③由于配置了” as-set” 参数，所以只有 R4 接收到汇总路由” 1.1.0.0/22”。

## 4 用 BGP 属性控制选路

BGP 具有丰富的属性，但本节只研究 Origin, AS-PATH, LOCAL\_PREF, Weight 和 MED 属性。本节的实验是一个有机的整体，根据 BGP 路由判定的顺序（优先级别从低到高）设计实验，每个分解实验都是以较高的优先级别影响前面分解实验的 BGP 路由选路。实验的拓扑如图 24-3 所示。通过修改 Origin, AS-PATH, LOCAL\_PREF, 和 Weight 属性来控制在 AS 100 内路由器 R1, R2 和 R3 对路由器 R4 上通告的 4.4.4.0/24 路由选路。最后通过在路由器 R2 和 R3 上发布环路接口来控制从路由器 R4 进入 AS 100 的选路。在本实验中，IBGP 的路由器（R1、R2 和 R3）形成全互连（Full Mesh）的邻居关系。IGP 运行 EIGRP。在完成每个分解实验后，最好用” clear ip bgp” 清除下一 BGP 路由表，然后在查看结果。

### 4.1 实验 3 用 BGP Origin 属性控制选路

#### 1. 实验目的

通过本实验可以掌握：

- ①BGP 路由传递的条件；
- ②Origin 代码的优先级；
- ③用 origin 属性选路的原则。

#### 2. 拓扑结构

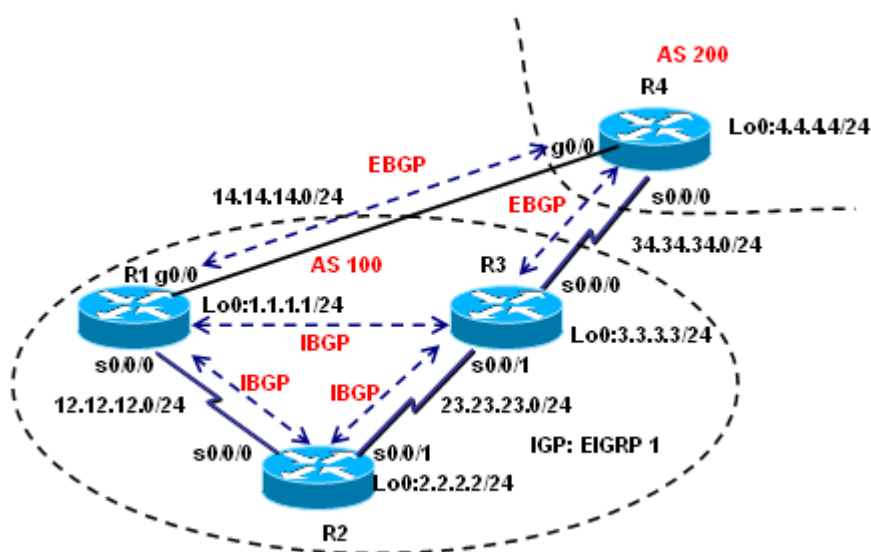


图 24-3

---

### 3. 实验步骤

本实验是在路由器 R4 上配置”4. 4. 4. 0/24”的起源代码属性为 EGP, 并通过 EBGp 邻居”14. 14. 14. 1”传入 AS 100 内, 然后观察路由器 R1、R2 和 R3 对路由器 R4 上通告”4. 4. 4. 0/24”路由的选路。

#### (1) 步骤 1: 配置路由器 R1

```
R1(config)#router eigrp 1

R1(config-router)#network 1.1.1.0 255.255.255.0

R1(config-router)#network 12.12.12.0 255.255.255.0

R1(config-router)#no auto-summary

R1(config)#router bgp 100

R1(config-router)#no synchronization

R1(config-router)#no auto-summary

R1(config-router)#bgp router-id 1.1.1.1

R1(config-router)#neighbor 2.2.2.2 remote-as 100

R1(config-router)#neighbor 2.2.2.2 update-source Loopback0

R1(config-router)#neighbor 2.2.2.2 next-hop-self

R1(config-router)#neighbor 3.3.3.3 remote-as 100

R1(config-router)#neighbor 3.3.3.3 update-source Loopback0

R1(config-router)#neighbor 3.3.3.3 next-hop-self

R1(config-router)#neighbor 14.14.14.4 remote-as 200
```

#### (2) 步骤 2: 配置路由器 R2

```
R2(config)#router eigrp 1

R2(config-router)#network 2.2.2.0 255.255.255.0

R2(config-router)#network 12.12.12.0 255.255.255.0
```

---

```
R2(config-router)#network 23.23.23.0 255.255.255.0
```

```
R2(config-router)#no auto-summary
```

```
R2(config)#router bgp 100
```

```
R2(config-router)#no synchronization
```

```
R2(config-router)#no auto-summary
```

```
R2(config-router)#bgp router-id 2.2.2.2
```

```
R2(config-router)#neighbor 1.1.1.1 remote-as 100
```

```
R2(config-router)#neighbor 1.1.1.1 update-source Loopback0
```

```
R2(config-router)#neighbor 3.3.3.3 remote-as 100
```

```
R2(config-router)#neighbor 3.3.3.3 update-source Loopback0
```

(3) 步骤3: 配置路由器 R3

```
R3(config)#router eigrp 1
```

```
R3(config-router)#network 3.3.3.0 255.255.255.0
```

```
R3(config-router)#network 23.23.23.0 255.255.255.0
```

```
R3(config-router)#no auto-summary
```

```
R3(config)#router bgp 100
```

```
R3(config-router)#no synchronization
```

```
R3(config-router)#no auto-summary
```

```
R3(config-router)#bgp router-id 3.3.3.3
```

```
R3(config-router)#neighbor 1.1.1.1 remote-as 100
```

```
R3(config-router)#neighbor 1.1.1.1 update-source Loopback0
```

```
R3(config-router)#neighbor 1.1.1.1 next-hop-self
```

```
R3(config-router)#neighbor 2.2.2.2 remote-as 100
```

---

```
R3(config-router)#neighbor 2.2.2.2 update-source Loopback0
```

```
R3(config-router)#neighbor 2.2.2.2 next-hop-self
```

```
R3(config-router)#neighbor 34.34.34.4 remote-as 200
```

(4) 步骤 4: 配置路由器 R4

```
R4(config)#ip prefix-list 1 permit 4.4.4.0/24
```

```
R4(config)#route-map egp permit 10
```

```
R4(config-route-map)#match ip address prefix-list 1
```

```
R4(config-route-map)#set origin egp 900 //设置起源代码
```

//特别注意: 这里的 egp 不是 ebgp, 是 bgp 没有以前的 egp(外部网关协议), origin 选路时 igp<egp<incomplete

```
R4(config)#router bgp 200
```

```
R4(config-router)#no synchronization
```

```
R4(config-router)#no auto-summary
```

```
R4(config-router)#bgp router-id 4.4.4.4
```

```
R4(config-router)#network 4.4.4.0 mask 255.255.255.0
```

```
R4(config-router)#neighbor 14.14.14.1 remote-as 100
```

```
R4(config-router)#neighbor 14.14.14.1 router-map egp out
```

//在出方向为去往邻居 14.14.14.1 的路由设置策略

```
R4(config-router)#neighbor 34.34.34.3 remote-as 100
```

#### 4. 实验调试

在路由器 R1、R2 和 R3 上查看 BGP 表。

```
R1#show ip bgp
```

```
BGP table version is 6, local routerID is 1.1.1.1
```

---

Status cods; s suppressed, d damped, h history, \*valid, .best, i-internal,

r RIB-failure, S Stale

Origin codes;i-IGP, e-EGP, ?-incomplete

| Network       | Next Hop   | Metric | LocPrf | Weight | Path |
|---------------|------------|--------|--------|--------|------|
| *>i4.4.4.0/24 | 3.3.3.3    | 0      | 100    | 0      | 200i |
| *             | 14.14.14.4 | 0      |        | 0      | 200e |

R2#show ip bgp

BGP table version is 5, locl routerID is 2.2.2.2

Status cods; s suppressed, d damped, h history, \*valid, .best, i-internal,

r RIB-failure, S Stale

Origin codes;i-IGP, e-EGP, ?-incomplete

| Network       | Next Hop | Metric | LocPrf | Weight | Path |
|---------------|----------|--------|--------|--------|------|
| *>i4.4.4.0/24 | 3.3.3.3  | 0      | 100    | 0      | 200i |

R3#show ip bgp

BGP table version is 4, locl routerID is 3.3.3.3

Status cods; s suppressed, d damped, h history, \*valid, .best, i-internal,

r RIB-failure, S Stale

Origin codes;i-IGP, e-EGP, ?-incomplete

| Network      | Next Hop   | Metric | LocPrf | Weight | Path |
|--------------|------------|--------|--------|--------|------|
| *>4.4.4.0/24 | 34.34.34.4 | 0      |        | 0      | 200i |

以上输出表明路由器 R1 学到两条关于” 4.4.4.0/24” 的路由，但是由于起源代码” i” 优先于” e”，所以从路由器 R3 学到的路由被优化，而从邻居路由器 R4 学到的路由不能被优化（路由代码为” \*”，没有” > “），不能继续通告给路由器 R2 和 R3，所以路由器 R2 和 R3 只有一条关于” 4.4.4.0/24” 的路由。

## 4.2 实验 4 用 BGP AS-PATH 属性控制选路

### 1. 实验目的

通过本实验可以掌握；

- ①AS-PATH 控制路由环路的原理；
- ②配置 AS-PATH 属性；
- ③用 AS-PATH 属性选路原则。

### 2. 拓扑结构

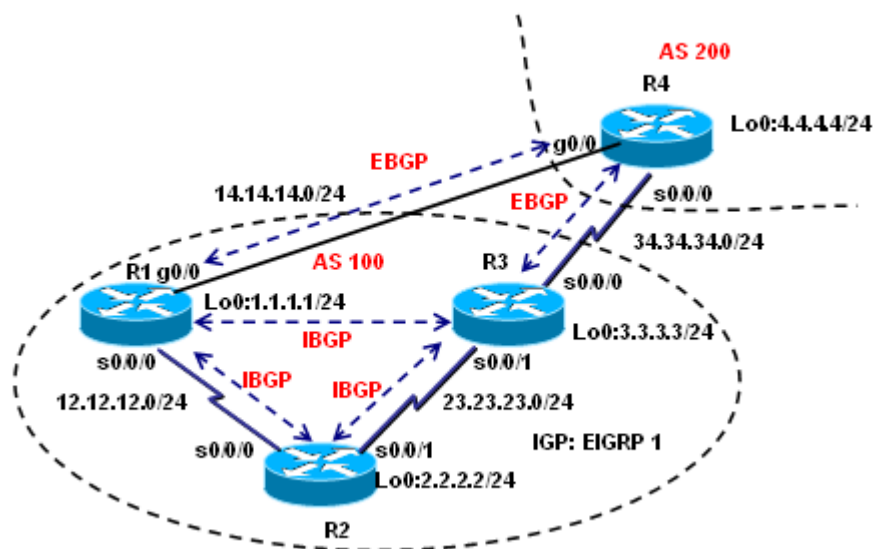


图 24-3

### 3. 实验拓扑

路由器 R1、R2 和 R3 上的配置和前面 4.1 实验 3 相同，路由器 R4 改动配置如下：

```
R4(config)#ip prefix-list 1 permit 4.4.4.0/24
```

```
R4(config)#router-map aspath permit 10
```

---

```
R4(config-route-map)#match ip address prefix-list 1
```

```
R4(config-route-map)#set as-path prepend 600 700 //为匹配的路由条目追加 AS
```

```
R4(config)#router bgp 200
```

```
R4(config-router)#neighbor 34.34.34.3 router-map aspath out
```

#### 4. 实验调试

在路由器 R1、R2 和 R3 上查看 BGP 表：

```
R1#show ip bgp
```

```
BGP table version is 11, local routerID is 1.1.1.1
```

```
Status codes: s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
                r RIB-failure, S Stale
```

```
Origin codes: i-IGP, e-EGP, ?-incomplete
```

| Network      | Next Hop   | Metric | LocPrf | Weight | Path |
|--------------|------------|--------|--------|--------|------|
| *>4.4.4.0/24 | 14.14.14.4 | 0      |        | 0      | 200e |

```
R2#show ip bgp
```

```
BGP table version is 9, local routerID is 2.2.2.2
```

```
Status codes: s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
                r RIB-failure, S Stale
```

```
Origin codes: i-IGP, e-EGP, ?-incomplete
```

| Network      | Next Hop | Metric | LocPrf | Weight | Path |
|--------------|----------|--------|--------|--------|------|
| *>4.4.4.0/24 | 1.1.1.1  | 0      | 100    | 0      | 200e |

```
R3#show ip bgp
```

```
BGP table version is 9, local routerID is 3.3.3.3
```

```
Status codes: s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
          r RIB-failure, S Stale
```

```
Origin codes: i-IGP, e-EGP, ?-incomplete
```

| Network      | Next Hop   | Metric | LocPrf | Weight | Path    |
|--------------|------------|--------|--------|--------|---------|
| *>4.4.4.0/24 | 1.1.1.1    | 0      | 100    | 0      | 200e    |
| *            | 34.34.34.4 | 0      |        | 0      | 200 600 |

```
700i
```

以上输出表明路由器 R3 学到两条关于“4.4.4.0/24”的路由，但是由于下一跳为“1.1.1.1”的路由的 AS-PATH 比下一跳为“34.34.34.4”的路由的 AS-PATH 短，所以优选下一跳为“1.1.1.1”的路由，而不能被优化（路由代码为“\*”），不能继续通告给路由器 R1 和 R2，所以路由器 R1 和 R2 支有一条关于“4.4.4.0/24”的路由。同时也说明 BGP 在路由判定时，AS-PATH 属性是优于 ORIGIN 属性的。

### 4.3 实验 5 用 BGP LOCAL\_PREF 属性控制选路

#### 1. 实验目的

通过本实验可以掌握：

- ①配置 LOCAL\_PREF 属性；
- ②用 LOCAL\_PREF 属性选路原则。

#### 2. 拓扑结构

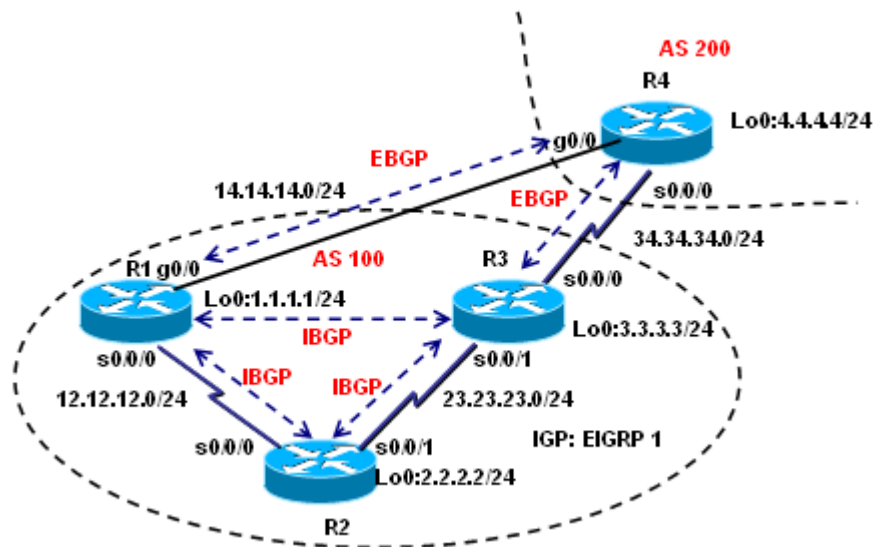


图 24-3

### 3. 实验拓扑

路由器 R1、R2 和 R4 上的配置和前面 4.2 实验 4 相同，路由器 R3 改动配置如下：

```
R3(config)#ip prefix-list 1 permit 4.4.4.0/24
```

```
R3(config)#router-map local permit 10
```

```
R3(config-route-map)#match ip address prefix-list 1
```

```
R3(config-route-map)#set local-preference 2000 //修改 LOCAL_PREF 值
```

```
R3(config)#router bgp 100
```

```
R3(config-router)#neighbor 34.34.34.4 router-map local in
```

//对从邻居 34.34.34.4 进入的路由条目设置策略

### 4. 实验调试

在路由器 R1、R2 和 R3 上查看 BGP 表。

```
R1#show ip bgp
```

```
BGP table version is 2, local routerID is 1.1.1.1
```

```
Status codes: s suppressed, d damped, h history, *valid, .best, i-internal,
```

---

```

      r RIB-failure, S Stale

Origin codes;i-IGP, e-EGP, ?-incomplete

Network          Next Hop        Metric      LocPrf   Weight      Path
*>4.4.4.0/24      14.14.14.4    0                0          200e
*>i              3.3.3.3       0            2000      0          200 600
700i
```

```

R2#show ip bgp

BGP table version is 2, locl routerID is 2.2.2.2

Status cods; s suppressed, d damped, h history, *valid, .best, i-internal,

      r RIB-failure, S Stale

Origin codes;i-IGP, e-EGP, ?-incomplete

Network          Next Hop        Metric      LocPrf   Weight      Path
*>4.4.4.0/24      3.3.3.3       0            2000      0          200 600
700i
```

```

R3#show ip bgp

BGP table version is 2, locl routerID is 3.3.3.3

Status cods; s suppressed, d damped, h history, *valid, .best, i-internal,

      r RIB-failure, S Stale

Origin codes;i-IGP, e-EGP, ?-incomplete

Network          Next Hop        Metric      LocPrf   Weight      Path
```

---

|              |            |   |      |   |         |
|--------------|------------|---|------|---|---------|
| *>4.4.4.0/24 | 34.34.34.4 | 0 | 2000 | 0 | 200 600 |
|--------------|------------|---|------|---|---------|

700i

以上输出表明路由器 R1 学到两条关于” 4.4.4.0/24” 的路由，但是由于下一跳为” 3.3.3.3” 的路由本地优先级的值比下一跳为” 14.14.14.4” 的路由的本地优先级的值高，所以优选下一跳为” 3.3.3.3” 的路由，而下一跳为” 14.14.14.4” 的路由不能被优化，不能继续通告给路由器 R2 和 R3，所以路由器 R2 和 R3 只有一条关于” 4.4.4.0/24” 的路由。同时也说明 BGP 在路由判定时，**本地优先级属性是优于其他路由**

#### 【提示】

①默认情况下，**本地优先级的值为 100**；

②本地优先级属性只在 AS 内部传递，不会通告 EBGP 邻居；

③本地优先级属性值越高，路由的优先程度越高；

④命令” `bgp default local-preference`” 也可以修改本地优先级属性，只是说用 `route-map` 设置本地优先级灵活性更大。

### 4.4 实验 6 用 BGP Weight 属性控制选路

#### 1. 实验目的

通过本实验可以掌握：

①配置 Weight 属性；

②用 Weight 属性选路原则。

#### 2. 拓扑结构

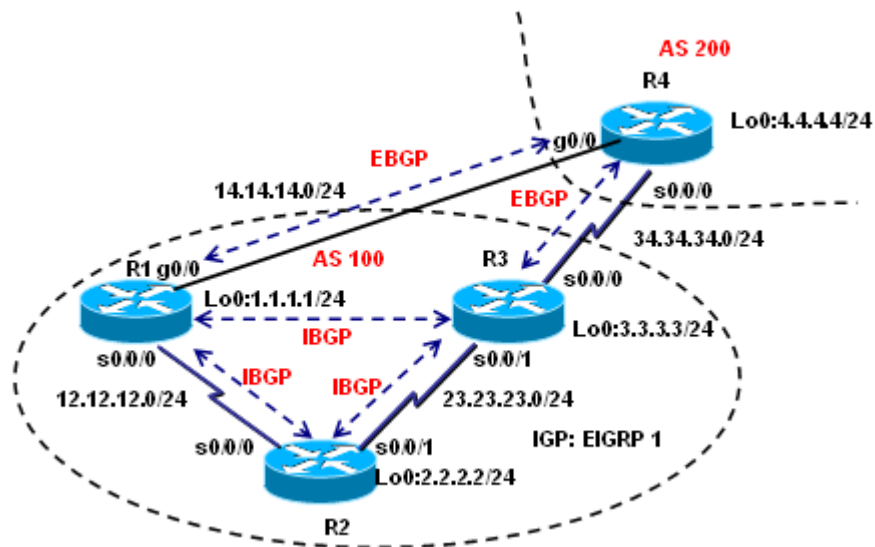


图 24-3

### 3. 实验拓扑

路由器 R2、R3 和 R4 上的配置和前面 4.3 实验 5 相同，路由器 R1 改动配置如下：

```
R1(config)#router bgp 100

R1(config-router)#neighbor 2.2.2.2 weight 200

//为从 2.2.2.2 学到的路由设置权重

R1(config-router)#neighbor 3.3.3.3 weight 200

//为从 3.3.3.3 学到的路由设置权重

R1(config-router)#neighbor 14.14.14.4 weight 500

//为从 14.14.14.4 学到的路由设置权重
```

### 4. 实验调试

在路由器 R1、R2 和 R3 上查看 BGP 表。

```
R1#show ip bgp

BGP table version is 3, locl routerID is 1.1.1.1

Status codes; s suppressed, d damped, h history, *valid, .best, i-internal,
```

|                                         |            |        |        |        |         |
|-----------------------------------------|------------|--------|--------|--------|---------|
| r RIB-failure, S Stale                  |            |        |        |        |         |
| Origin codes;i-IGP, e-EGP, ?-incomplete |            |        |        |        |         |
|                                         |            |        |        |        |         |
| Network                                 | Next Hop   | Metric | LocPrf | Weight | Path    |
| *>4.4.4.0/24                            | 14.14.14.4 | 0      |        | 500    | 200e    |
| *>i                                     | 3.3.3.3    | 0      | 2000   | 200    | 200 600 |

700i

R2#show ip bgp

BGP table version is 2, locl routerID is 2.2.2.2

Status cods; s suppressed, d damped, h history, \*valid, .best, i-internal,

| r RIB-failure, S Stale | | | | | |
| Origin codes;i-IGP, e-EGP, ?-incomplete | | | | | |
|  | | | | | |
| Network | Next Hop | Metric | LocPrf | Weight | Path |
| \*i4.4.4.0/24 | 1.1.1.1 | 0 | 100 | 0 | 200e |
| \*>i | 3.3.3.3 | 0 | 2000 | 0 | 200 600 |

700i

R3#show ip bgp

BGP table version is 2, locl routerID is 3.3.3.3

Status cods; s suppressed, d damped, h history, \*valid, .best, i-internal,

| r RIB-failure, S Stale | | | | | |
| Origin codes;i-IGP, e-EGP, ?-incomplete | | | | | |
|  | | | | | |
| Network | Next Hop | Metric | LocPrf | Weight | Path |

|              |            |   |      |   |         |
|--------------|------------|---|------|---|---------|
| *i4.4.4.0/24 | 1.1.1.1    | 0 | 100  | 0 | 200e    |
| *>           | 34.34.34.4 | 0 | 2000 | 0 | 200 600 |

700i

以上输出表明路由器 R1 学到两条关于” 4.4.4.0/24” 的路由，但是由于下一跳为” 14.14.14.4” 的路由 Weight 值比下一跳为” 3.3.3.3” 的路由 Weight 值高，所以优选下一跳为” 14.14.14.4” 的路由。因为 Weight 属性只影响本地路由选路，所以对于路由器 R2 和 R3 仍是通过本地优先级选路。路由器 R1 的选路说明 BGP 在路由判定时，Weight 属性是优于本地优先级属性的。

#### 4.5 实验 7 用 MED 属性控制选路

##### 1. 实验目的

通过本实验可以掌握：

- ①配置 MED 属性；
- ②用 MED 属性选路原则。

##### 2. 拓扑结构

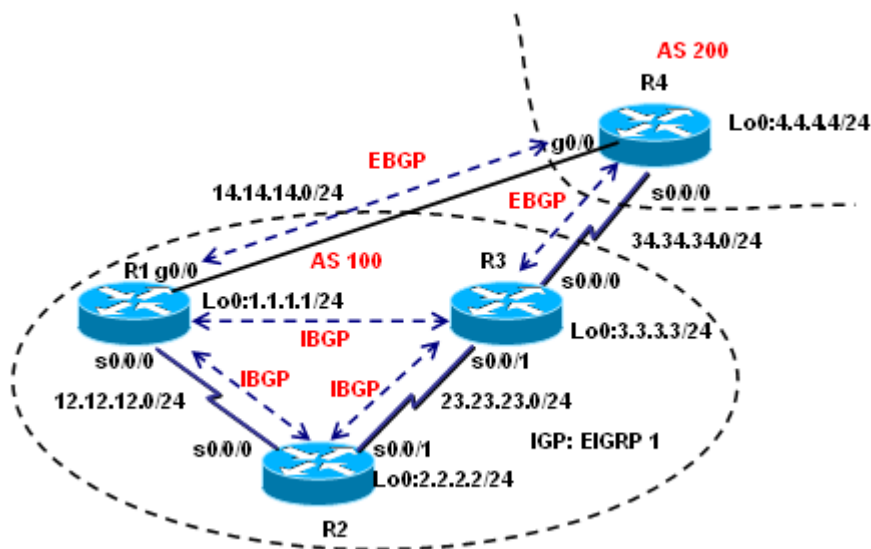


图 24-3

实验拓扑图如图 24-3 所示。本实验需要在路由器 R2 添加一个环回地址 Lo1;20.1.1.2/24，并在 BGP 中发布；在路由器 R3 添加一个环回地址 Lo1;30.1.1.3/24，并

---

在 BGP 中发布。通过设置 MED 属性，使得在路由器 R4 上访问 30.1.1.3 时，走 R4-R3-R4 路径；在 R4 上访问 20.1.1.2 时，走 R4-R1-R2-R4 的路径。

### 3. 实验步骤

路由器 R4 上的配置和 4.4 实验 6 相同，路由器 R1、R2、R3 改动配置如下。

#### (1) 步骤 1：配置路由器 R1

```
R1(config)#ip prefix-list 20 permit 20.1.1.0/24
R1(config)#ip prefix-list 30 permit 30.1.1.0/24
R1(config)#route-map med permit 20
R1(config-route-map)#match ip address prefix-list 20
R1(config-route-map)#set metric 50
R1(config)#route-map med permit 30
R1(config-route-map)#match ip address prefix-list 30
R1(config-route-map)#set metric 100
R1(config)#route bgp 100
R1(config-router)#neighbor 14.14.14.4 router-map med out
```

#### (2) 步骤 2：配置路由器 R2

```
R2(config)#router bgp 100
R2(config-router)#network 20.1.1.0 mask 255.255.255.0
```

#### (3) 步骤 3：配置路由器 R3

```
R3(config)#ip prefix-list 20 permit 20.1.1.0/24
R3(config)#ip prefix-list 30 permit 30.1.1.0/24
R3(config)#route-map med permit 20
R3(config-route-map)#match ip address prefix-list 20
```

```
R3(config-route-map)#set metric 100

R3(config)#route-map med permit 30

R3(config-route-map)#match ip address prefix-list 30

R3(config-route-map)#set metric 50

R3(config)#router bgp 100

R3(config-router)#network 30.1.1.0 mask 255.255.255.0

R3(config-router)#neighbor 34.34.34.4 route-map med out
```

#### 4. 实验调试

在路由器 R4 上查看 BGP 表。

```
R4#show ip bgp

BGP table version is 5, local routerID is 4.4.4.4

Status codes: s suppressed, d damped, h history, *valid, .best, i-internal,
               r RIB-failure, S Stale

Origin codes: i-IGP, e-EGP, ?-incomplete


```

| Network       | Next Hop   | Metric | LocPrf | Weight | Path |
|---------------|------------|--------|--------|--------|------|
| *>4.4.4.0/24  | 0.0.0.0    | 0      |        | 32768  | i    |
| * 20.1.1.0/24 | 34.34.34.4 | 100    |        | 0      | 100i |
| *>            | 14.14.14.4 | 50     |        | 0      | 100i |
| *>30.1.1.0/24 | 34.34.34.4 | 50     |        | 0      | 100i |
| *             | 14.14.14.4 | 100    |        | 0      | 100i |

以上输出表明路由器 R4 学到的 BGP 路由是携带了 MED 的值,而且优选 MED 值低的路径。

**【提示】**

---

①MED 只用来向 EBGp 邻居发送;

②MED 用来影响外部 AS 选路

③进入到一个 AS 中的 MED 属性是不会从这个 AS 中再传递出去;

④MED 的值越低, 路由的优选程度越高。

(2) 用扩展 ping 跟踪路径

R4#ping

Protocol[ip];

Target IP address;20.1.1.2

Repeat count[5];1

Datagram size[100];

Timeout in seconds[2];

Extended commands[n];y

Source address or interface;.4.4.4.4

Type of service[0];

Set DF bit in IP header?[on];

Validate reply data?[on];

Data pattern [0xABCD];

Loose, Strict, Record, Timestamp, Verbose[none];R

Number of hops[9];6

Loose, Strict, Record, Timestamp, Verbose[RV];

Sweep range of sizes [n];

Type escape sequence to abort.

Sending 2, 100-byte ICMP Echos to 20.1.1.2, timeout is 2 seconds;

---

Pasket sent with a source address of 4.4.4.4

Pasket has IP options; Total option bytes=27, padded length=28

Record route;<\*>

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

Reply to request 0(148ms). Received packet has optins

Total option bytes=27, padded length=28

Record route;

(14.14.14.4)

(12.12.12.1)

(23.23.23.2)

(34.34.34.3)

(4.4.4.4)

<\*>

End of list

以上输出表明，在路由器 R4 上访问 20.1.1.2 时，走 R4→R1→R2→R4 的路经

R4#ping

Protocol[ip];

Target IP address;30.1.1.3

---

Repeat count[5];1

Datagram size[100];

Timeout in seconds[2];

Extended commands[n];y

Source address or interface;.4.4.4.4

Type of service[0];

Set DF bit in IP header?[on];

Validate reply data?[on];

Data pattern [0xABCD];

Looes, Strict, Record, Timestamp, Verbose[none];R

Number of hops[9];4

Looes, Strict, Record, Timestamp, Verbose[RV];

Sweep range of sizes [n];

Type escape sequence to abort.

Sending 2, 100-byte ICMP Echos to 30.1.1.3, timeout is 2 seconds;

Pasket sent with a source address of 4.4.4.4

Pasket has IP options; Total option bytes=19, padded length=20

Record route;<\*>

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

(0.0.0.0)

Reply to request 0(248ms). Received packet has optins

---

Total option bytes=20, padded length=20

Record route;

(34.34.34.4)

(30.1.1.3)

(34.34.34.3)

(4.4.4.4)

<\*>

End of list

以上输出表明，在路由器 R4 上访问 30.1.1.3 时，走 R4→R3→R4 路径

## 5 实验 8 路由反射器（RR）配置

### 1. 实验目的

通过本实验可以掌握：

- ①RR 的反射原理和反射规则；
- ②RR 的配置
- ③ORIGINATOR\_ID(起源 ID)属性；
- ④CLUSTER\_LIST(簇列表)属性。

### 2. 拓扑结构

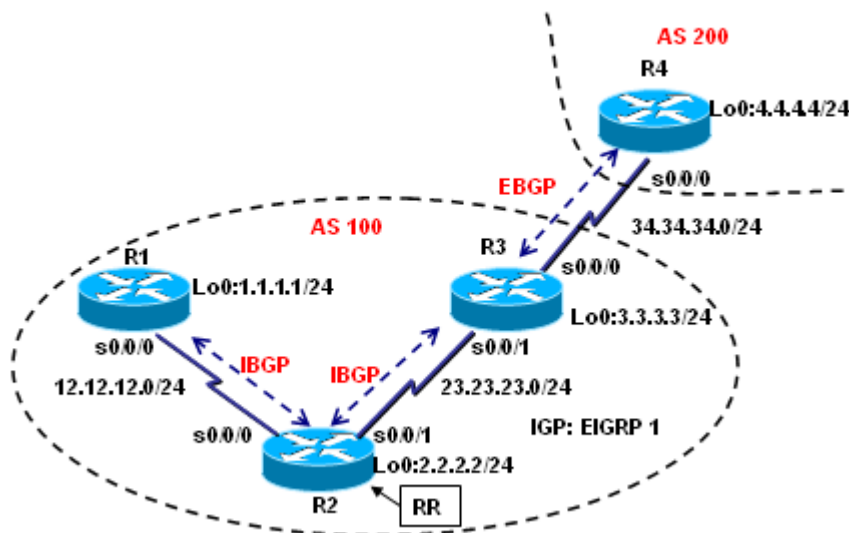


图 24-4

本实验中，路由器 R2 作为路由反射器，路由器 R1 和 R3 作为它的客户端。

### 3. 实验步骤

#### (1) 步骤 1：配置路由器 R1

```
R1(config)#router eigrp 1
```

```
R1(config-router)#network 1.1.1.0 255.255.255.0
```

```
R1(config-router)#network 12.12.12.0 255.255.255.0
```

```
R1(config-router)#no auto-summary
```

```
R1(config)#router bgp 100
```

```
R1(config-router)#no synchronization
```

```
R1(config-router)#no auto-summary
```

```
R1(config-router)#bgp router-id 1.1.1.1
```

```
R1(config-router)#neighbor 2.2.2.2 remote-as 100
```

```
R1(config-router)#neighbor 2.2.2.2 update-source Loopback0
```

```
R1(config-router)#network 1.1.1.0 mask 255.255.255.0
```

---

(2) 步骤 2: 配置路由器 R2

```
R2(config)#router eigrp 1
```

```
R2(config-router)#network 2.2.2.0 255.255.255.0
```

```
R2(config-router)#network 12.12.12.0 255.255.255.0
```

```
R2(config-router)#network 23.23.23.0 255.255.255.0
```

```
R2(config-router)#no auto-summary
```

```
R2(config)#router bgp 100
```

```
R2(config-router)#no synchronization
```

```
R2(config-router)#no auto-summary
```

```
R2(config-router)#bgp router-id 2.2.2.2
```

```
R2(config-router)#neighbor 1.1.1.1 remote-as 100
```

```
R2(config-router)#neighbor 1.1.1.1 update-source Loopback0
```

```
R2(config-router)#neighbor 1.1.1.1 route-reflector-client//配置 RR 客户端
```

```
R2(config-router)#neighbor 3.3.3.3 remote-as 100
```

```
R2(config-router)#neighbor 3.3.3.3 update-source Loopback0
```

```
R2(config-router)#neighbor 3.3.3.3 route-reflector-client
```

(3) 步骤 3: 配置路由器 R3

```
R3(config)#router eigrp 1
```

```
R3(config-router)#network 3.3.3.0 255.255.255.0
```

```
R3(config-router)#network 23.23.23.0 255.255.255.0
```

```
R3(config-router)#no auto-summary
```

```
R3(config)#router bgp 100
```

```
R3(config-router)#no synchronization
```

---

```
R3(config-router)#no auto-summary
```

```
R3(config-router)#bgp router-id 3.3.3.3
```

```
R3(config-router)#neighbor 2.2.2.2 remote-as 100
```

```
R3(config-router)#neighbor 2.2.2.2 update-source Loopback0
```

```
R3(config-router)#neighbor 2.2.2.2 next-hop-self
```

```
R3(config-router)#neighbor 34.34.34.4 remote-as 200
```

(4) 步骤 4: 配置路由器 R4

```
R4(config)#router bgp 200
```

```
R4(config-router)#no synchronization
```

```
R4(config-router)#no auto-summary
```

```
R4(config-router)#bgp router-id 4.4.4.4
```

```
R4(config-router)#neighbor 34.34.34.3 remote-as 100
```

```
R4(config-router)#network 4.4.4.0 mask 255.255.255.0
```

### 【说明】

当一个 AS 包含多个 IBGP 对等体时，路由反射非常有用，因为 IBGP 客户只需要和路由反射建立邻居关系，从而减低了 IBGP 的连接数量。路由反射器和它的客户合称为一个簇，路由反射器是克服 IBGP 水平分割的重要手段。

### 【技术要点】

RR 的反射规则如下：

- ①如果路由是从非客户的 IBGP 邻居学来的，则 RR 只将它反射给客户；
- ②如果路由是从客户学来的，RR 会将它反射给所有的非客户和客户（除了发起该路由的客户）；
- ③如果路由是从 EBGp 邻居学来的，RR 会将它反射给所有的非客户和客户。

## 4. 实验调试

---

(1) show ip bgp neighbors

```
R2#show ip bgp neighbors 1.1.1.1
```

```
BGPneighbor is 1.1.1.1 remote AS 100.extenal link
```

```
BGP version 4.remote router ID 1.1.1.1
```

```
BGP stae=Established, up for 00;31;06
```

```
Last read 00;00;07, hold time is 180, keepalive interval is 60 seconds
```

```
.....
```

```
For address family;IPv4 Unicast
```

```
BGP table version 4, neighbor version 4
```

```
Index 1, Offset 0, Mask 0x2
```

```
Route-Reflector Client
```

```
.....
```

以上输出表明邻居 1.1.1.1 是路由反射的客户端。

(2) show ip bgp 4.4.4.0

```
R2#show ip bgp 4.4.4.0
```

```
BGP routing table entry for 4.4.4.0/24, version4
```

```
Paths(1 available, best #1, table Default-IP-Routing-Table)
```

```
Adverised to non peer-group peer;
```

```
1.1.1.1
```

```
200, (Receivd from a RR-client)
```

```
3.3.3.3(metric 2297856)from 3.3.3.3(3.3.3.3)
```

```
Origin IGP, metric 0, localpref 100, valid, internal, best
```

---

以上输出表明 BGP 路由条目 ”4. 4. 4. 0/24” 是从 RR 的客户端接收到的, 客户端是 3. 3. 3. 3, 并且将它反射给 1. 1. 1. 1。注意上面输出 ” (3. 3. 3. 3) ” 指的是路由器 ID。

```
R1#show ip bgp 4.4.4.0
```

```
BGP routing table entry for 4.4.4.0/24, version 3
```

```
Paths(1 available, best #1, table Default-IP-Routing-Table)
```

```
Not advertised to any peer
```

```
200
```

```
3.3.3.3(metric 28098056) from 2.2.2.2 (2.2.2.2)
```

```
Origin IGP, metric 0, localpref 100, valid, internal, best
```

```
Originator: 3.3.3.3, Cluster list: 2.2.2.2
```

以上输出表明在 AS 100 内 BGP 路由条目 ” 4. 4. 4. 0/24 ” 的创造者是 3. 3. 3. 3, 簇 ID 是 2. 2. 2. 2。

### 【术语】

①ORIGINATOR\_ID; 路由器反射器生成, 是本 AS 内路由创造者的路由器 ID;

②CLUSTER\_ID(ID); 一个 AS 内的每一个簇必须用一个唯一的 4 字节的簇 ID 来标识, 如果簇内只有一个 RR, 那么簇 ID 就是 RR 的路由器 ID, 当 RR 接收到一个更新消息的时候, 它检查 CLUSTER\_LIST, 如果发现在列表中有自己的簇 ID, 就知道出现了路由环路。

## 6 实验 9 BGP 联邦配置

### 1. 实验目的

通过本实验可以掌握:

①BGP 联邦的含义;

②BGP 联邦的配置。

### 2. 拓扑结构

实验拓扑图如图 24-5 所示。

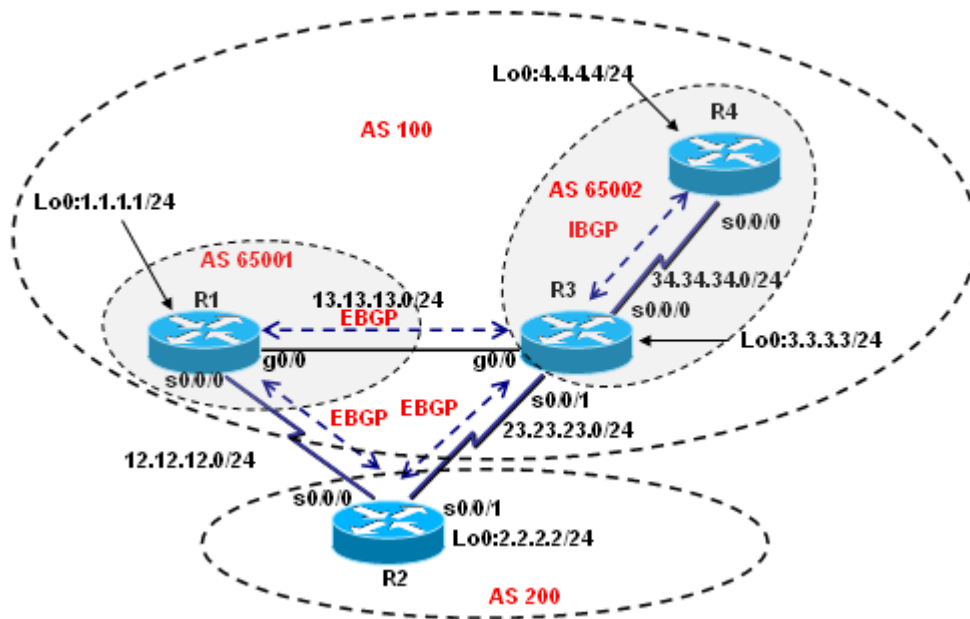


图 24-5

### 3. 实验拓扑

本实验联邦的成员为 AS 65001 和 AS65002，联邦对外 AS 为 100。

#### (1) 步骤 1：配置路由器 R1

```
R1(config)#router bgp 65001

R1(config-router)#no synchronization

R1(config-router)#no auto-summary

R1(config-router)#bgp router-id 1.1.1.1

R1(config-router)#bgp confederation identifier 100

R1(config-router)#bgp confederation peers 65002

R1(config-router)#network 1.1.1.0 mask 255.255.255.0

R1(config-router)#neighbor 12.12.12.2 remote-as 200

R1(config-router)#neighbor 13.13.13.3 remote-as 65002

R1(config-router)#neighbor 13.13.13.3 next-hop-self
```

---

(2) 步骤 2: 配置路由器 R2

```
R2(config)#router bgp 200
```

```
R2(config-router)#no synchronization
```

```
R2(config-router)#no auto-summary
```

```
R2(config-router)#bgp router-id 2.2.2.2
```

```
R2(config-router)#network 2.2.2.0 mask 255.255.255.0
```

```
R2(config-router)#neighbor 12.12.12.1 remote-as 100
```

```
R2(config-router)#neighbor 23.23.23.3 remote-as 100
```

(3) 步骤 3: 配置路由器 R3

```
R3(config)#router eigrp 1
```

```
R3(config-router)#network 3.3.3.0 0.0.0.255
```

```
R3(config-router)#network 34.34.34.0 0.0.0.255
```

```
R3(config-router)#no auto-summary
```

```
R3(config)#router bgp 65002
```

```
R3(config-router)#no synchronization
```

```
R3(config-router)#no auto-summary
```

```
R3(config-router)#bgp router-id 3.3.3.3
```

```
R3(config-router)#bgp confederation identifier 100
```

```
R3(config-router)#bgp confederation peers 65001
```

```
R3(config-router)#network 3.3.3.0 mask 255.255.255.0
```

```
R3(config-router)#neighbor 4.4.4.4 remote-as 65002
```

```
R3(config-router)#neighbor 4.4.4.4 update-source Loopback0
```

```
R3(config-router)#neighbor 4.4.4.4 next-hop-self
```

---

```
R3(config-router)#neighbor 13.13.13.1 remote-as 65001
```

```
R3(config-router)#neighbor 13.13.13.1 next-hop-self
```

```
R3(config-router)#neighbor 23.23.23.2 remote-as 200
```

(4) 步骤 4: 配置路由器 R4

```
R4(config)#router eigrp 1
```

```
R4(config-router)#network 4.4.4.0 0.0.0.255
```

```
R4(config-router)#network 34.34.34.0 0.0.0.255
```

```
R4(config-router)#no auto-summary
```

```
R4(config)#router bgp 65002
```

```
R4(config-router)#no synchronization
```

```
R4(config-router)#no auto-summary
```

```
R4(config-router)#bgp router-id 4.4.4.4
```

```
R4(config-router)#network 4.4.4.0 mask 255.255.255.0
```

```
R4(config-router)#neighbor 3.3.3.3 remote-as 65002
```

```
R4(config-router)#neighbor 3.3.3.3 update-source Loopback0
```

#### 【技术要点】

BGP 联邦用于将 AS 分割成多个子 AS，是控制大型 IBGP 对等的另一条途径，而子 AS 被称为成员自治系统。每个联邦都有被分配一个联邦 ID，对联邦外部来讲，这个联邦 ID 是代表整个联邦的 AS 号码。外部看不到联邦内部结构，联邦看起来就是一个 AS，成员自治系统信息被隐藏起来。

#### 4. 实验调试

(1) 在路由器 R2 上查看 BGP 表。

```
R2#show ip bgp
```

```
BGP table version is 5, local routerID is 2.2.2.2
```

---

```
Status cods; s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
    r RIB-failure, S Stale
```

```
Origin codes; i-IGP, e-EGP, ?-incomplete
```

| Network      | Next Hop   | Metric | LocPrf | Weight | Path |
|--------------|------------|--------|--------|--------|------|
| * 1.1.1.0/24 | 23.23.23.3 |        |        | 0      | 100i |
| *>           | 12.12.12.1 | 0      |        | 0      | 100i |
| *>2.2.2.0/24 | 0.0.0.0    | 0      |        | 32768  | i    |
| * 3.3.3.0/24 | 12.12.12.1 |        |        | 0      | 100i |
| *>           | 23.23.23.3 | 0      |        | 0      | 100i |
| * 4.4.4.0/24 | 12.12.12.1 |        |        | 0      | 100i |
| *>           | 23.23.23.3 |        |        | 0      | 100i |

以上输出表明学到的” 1.1.1.0/24” , ” 3.3.3.0/24” 和” 4.4.4.0/24” 网络都有两条路径, 而且都是来自 AS 100。由此看出 BGP 联邦内所有成员的信息对外都隐藏。

(2) 在路由器 R3 上查看 BGP 表。

```
R3#show ip bgp
```

```
BGP table version is 6, local routerID is 3.3.3.3
```

```
Status cods; s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
    r RIB-failure, S Stale
```

```
Origin codes; i-IGP, e-EGP, ?-incomplete
```

| Network      | Next Hop   | Metric | LocPrf | Weight | Path     |
|--------------|------------|--------|--------|--------|----------|
| *>1.1.1.0/24 | 13.13.13.1 | 0      | 100    | 0      | (65001)i |

---

|               |            |   |     |       |      |
|---------------|------------|---|-----|-------|------|
| * 2.2.2.0/24  | 13.13.13.1 | 0 | 100 | 0     |      |
| (65001)200i   |            |   |     |       |      |
| *>            | 23.23.23.3 | 0 |     | 0     | 200i |
| *>3.3.3.0/24  | 0.0.0.0    | 0 |     | 32768 | i    |
| r>i4.4.4.0/24 | 4.4.4.4    | 0 | 100 | 0     | i    |

以上输出表明，联邦内的 AS-PATH 用 ” ( ) ” 表示。

#### 【技术要点】

在联邦范围内，将成员 AS 加入到 AS-PATH 中，并且用括号扩起来，但是并不将它们公布到联邦的范围以外。AS-PATH 中联邦的 AS 号用于避免出现路由环路。

## 7 实验 10 BGP 团体配置

### 1. 实验目的

通过本实验可以掌握：

- ①BGP 团体属性配置；
- ②BGP 团体属性 local-AS；
- ③BGP 团体属性 no-export；；
- ④BGP 团体属性 no-advertise。；

### 2. 拓扑结构

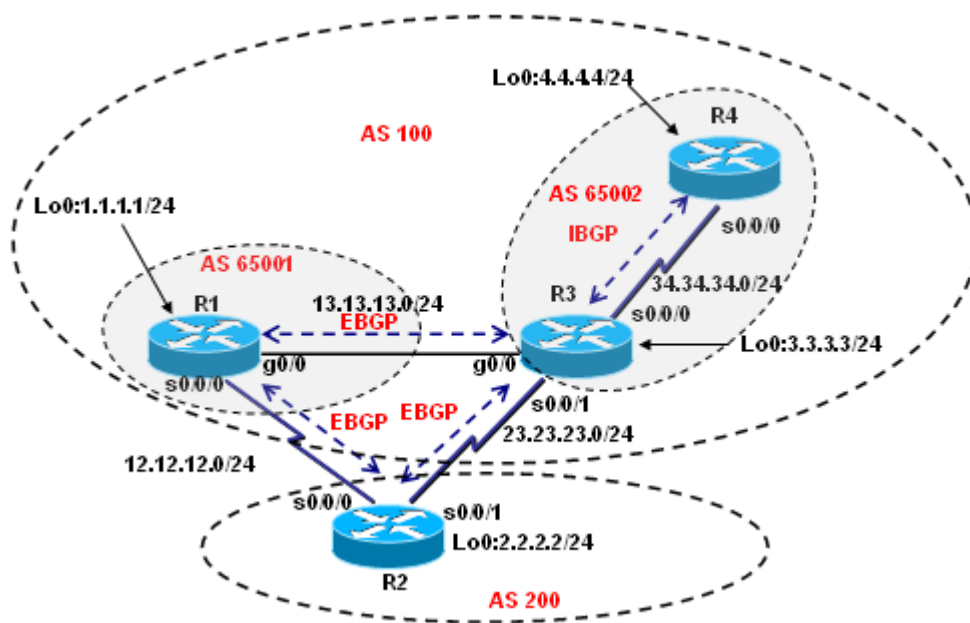


图 24-5

### 3. 实验拓扑与试验测试

保留实验 9 的所有配置，因本实验完全在实验 9 的配置的基础上完成。通过让路由器 R4 上的” 4. 4. 4. 0” 携带不同的团体属性，来验证团体的各个属性的传递特征。对团体的讨论，我们仅仅讨论熟知的属性” local-AS”，” no-export” 和” no-advertise”。本实验只给出在实验 9 基础上增加的配置。

(1) 步骤 1：在路由器 R4 配置团体属性 local-AS

```
R4(config)#ip prefix-list 1 permit 4.4.4.0/24 //定义前缀列表
```

```
R4(config)#route-map Local_AS permit 10 //定义 route-map
```

```
R4(config-route-map)#match ip address prefix-list 1 //匹配前缀列表
```

```
R4(config-route-map)#set community local-AS //设置团体属性
```

```
R4(config-route-map)#router bgp 65002
```

```
R4(config-router)#neighbor 3.3.3.3 send-community //开启发送团体属性的能力
```

```
R4(config-router)#neighbor 3.3.3.3 route-map Local-AS out
```

```
//在出方向向邻居发送团体属性
```

(2) 团体属性 local-AS 测试，分别在路由器 R3、R2 和 R1 上查看 BGP 表

```
R3#show ip bgp
```

```
BGP table version is 6, locl routerID is 3.3.3.3
```

```
Status cods; s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
      r RIB-failure, S Stale
```

```
Origin codes; i-IGP, e-EGP, ?-incomplete
```

| Network      | Next Hop   | Metric | LocPrf | Weight | Path      |
|--------------|------------|--------|--------|--------|-----------|
| *>1.1.1.0/24 | 13.13.13.1 | 0      | 100    | 0      | (65001) i |

|              |            |   |     |   |  |
|--------------|------------|---|-----|---|--|
| * 2.2.2.0/24 | 13.13.13.1 | 0 | 100 | 0 |  |
|--------------|------------|---|-----|---|--|

(65001) 200i

|    |            |   |  |   |      |
|----|------------|---|--|---|------|
| *> | 23.23.23.3 | 0 |  | 0 | 200i |
|----|------------|---|--|---|------|

|              |         |   |  |       |   |
|--------------|---------|---|--|-------|---|
| *>3.3.3.0/24 | 0.0.0.0 | 0 |  | 32768 | i |
|--------------|---------|---|--|-------|---|

|               |         |   |     |   |   |
|---------------|---------|---|-----|---|---|
| r>i4.4.4.0/24 | 4.4.4.4 | 0 | 100 | 0 | i |
|---------------|---------|---|-----|---|---|

```
R2#show ip bgp
```

```
BGP table version is 4, locl routerID is 2.2.2.2
```

```
Status cods; s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
      r RIB-failure, S Stale
```

```
Origin codes; i-IGP, e-EGP, ?-incomplete
```

| Network | Next Hop | Metric | LocPrf | Weight | Path |
|---------|----------|--------|--------|--------|------|
|---------|----------|--------|--------|--------|------|

|              |            |  |  |   |      |
|--------------|------------|--|--|---|------|
| * 1.1.1.0/24 | 23.23.23.3 |  |  | 0 | 100i |
|--------------|------------|--|--|---|------|

|    |            |   |  |   |      |
|----|------------|---|--|---|------|
| *> | 12.12.12.1 | 0 |  | 0 | 100i |
|----|------------|---|--|---|------|

```
*>2.2.2.0/24      0.0.0.0      0      32768      i
```

```
*>3.3.3.0/24      23.23.23.3   0      0      100i
```

```
R1#show ip bgp
```

```
BGP table version is 4, local routerID is 1.1.1.1
```

```
Status cods; s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
      r RIB-failure, S Stale
```

```
Origin codes; i-IGP, e-EGP, ?-incomplete
```

| Network | Next Hop | Metric | LocPrf | Weight | Path |
|---------|----------|--------|--------|--------|------|
|---------|----------|--------|--------|--------|------|

|              |         |   |  |       |   |
|--------------|---------|---|--|-------|---|
| *>1.1.1.0/24 | 0.0.0.0 | 0 |  | 32768 | i |
|--------------|---------|---|--|-------|---|

|              |            |   |     |   |  |
|--------------|------------|---|-----|---|--|
| * 2.2.2.0/24 | 13.13.13.3 | 0 | 100 | 0 |  |
|--------------|------------|---|-----|---|--|

```
(65002)200i
```

|    |            |   |  |   |      |
|----|------------|---|--|---|------|
| *> | 12.12.12.2 | 0 |  | 0 | 200i |
|----|------------|---|--|---|------|

|              |            |   |     |   |          |
|--------------|------------|---|-----|---|----------|
| *>3.3.3.0/24 | 13.13.13.3 | 0 | 100 | 0 | (65002)i |
|--------------|------------|---|-----|---|----------|

以上输出表明携带团体” local-AS” 属性的条目” 4.4.4.0/24” 之传递给路由器 R3, 因为路由器 R3 和 R4 都在 AS 65002 内, 并没有传递给路由器 R2 和 R1。由此可见, ”local-AS” 团体属性只能在本 AS 内传递。

(3) 步骤 2: 在路由器 R4 上配置团体属性 no-export

```
R4(config)#route-map NO-EXPORT permit 10
```

```
R4(config-route-map)#match ip address prefix-list 1
```

```
R4(config-route-map)#set community no-export //设置团体属性
```

```
R4(config-route-map)#router bgp 65002
```

```
R4(config-router)#neighbor 3.3.3.3 send-community
```

```
R4(config-router)#neighbor 3.3.3.3 route-map NO-EXPORT out
```

---

//在出方向向邻居发送团体属性

(4) 团体属性 no-export 测试，分别在路由器 R3、R2 和 R1 上查看 BGP 表

R3#show ip bgp

BGP table version is 6, local routerID is 3.3.3.3

Status codes: s suppressed, d damped, h history, \*valid, .best, i-internal,

r RIB-failure, S Stale

Origin codes: i-IGP, e-EGP, ?-incomplete

| Network       | Next Hop   | Metric | LocPrf | Weight | Path     |
|---------------|------------|--------|--------|--------|----------|
| *>1.1.1.0/24  | 13.13.13.1 | 0      | 100    | 0      | (65001)i |
| *>2.2.2.0/24  | 23.23.23.2 | 0      |        | 0      | 200i     |
| *>3.3.3.0/24  | 0.0.0.0    | 0      |        | 32768  | i        |
| r>i4.4.4.0/24 | 4.4.4.4    | 0      | 100    | 0      | i        |

R2#show ip bgp

BGP table version is 4, local routerID is 2.2.2.2

Status codes: s suppressed, d damped, h history, \*valid, .best, i-internal,

r RIB-failure, S Stale

Origin codes: i-IGP, e-EGP, ?-incomplete

| Network      | Next Hop   | Metric | LocPrf | Weight | Path |
|--------------|------------|--------|--------|--------|------|
| * 1.1.1.0/24 | 23.23.23.3 |        |        | 0      | 100i |
| *>           | 12.12.12.1 | 0      |        | 0      | 100i |
| *>2.2.2.0/24 | 0.0.0.0    | 0      |        | 32768  | i    |

```
* 3.3.3.0/24          12.12.12.1    0              0 100        i
```

```
*>                   23.23.23.3    0              0 100        i
```

```
R1#show ip bgp
```

```
BGP table version is 5, local routerID is 1.1.1.1
```

```
Status codes: s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
                r RIB-failure, S Stale
```

```
Origin codes: i-IGP, e-EGP, ?-incomplete
```

| Network | Next Hop | Metric | LocPrf | Weight | Path |
|---------|----------|--------|--------|--------|------|
|---------|----------|--------|--------|--------|------|

|              |         |   |  |       |   |
|--------------|---------|---|--|-------|---|
| *>1.1.1.0/24 | 0.0.0.0 | 0 |  | 32768 | i |
|--------------|---------|---|--|-------|---|

|              |            |   |     |   |  |
|--------------|------------|---|-----|---|--|
| * 2.2.2.0/24 | 13.13.13.3 | 0 | 100 | 0 |  |
|--------------|------------|---|-----|---|--|

```
(65002)200i
```

|    |            |   |  |   |      |
|----|------------|---|--|---|------|
| *> | 12.12.12.2 | 0 |  | 0 | 200i |
|----|------------|---|--|---|------|

|              |            |   |     |   |          |
|--------------|------------|---|-----|---|----------|
| *>3.3.3.0/24 | 13.13.13.3 | 0 | 100 | 0 | (65002)i |
|--------------|------------|---|-----|---|----------|

|              |            |   |     |   |          |
|--------------|------------|---|-----|---|----------|
| *>4.4.4.0/24 | 13.13.13.3 | 0 | 100 | 0 | (65002)i |
|--------------|------------|---|-----|---|----------|

以上输出表明携带团体”no-export”属性的条目”4.4.4.0/24”之传递给路由器 R3 和 R1, 因为路由器 R1、R3 和 R4 都在 AS 100 内, 并没有传递给路由器 R2。由此可见”no-export”团体属性能在联邦的大 AS 内传递, 如果没有联邦, 只能在本 AS 内传递。

#### 【提示】

通过命令”show ip bgp community no-export”来查看 BGP 表哪些条目携带相应的属性, 如下所述。

```
R1#show ip bgp community no-export
```

```
BGP table version is 5, local routerID is 1.1.1.1
```

---

```
Status cods; s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
      r RIB-failure, S Stale
```

```
Origin codes;i-IGP, e-EGP, ?-incomplete
```

```

Network      Next Hop      Metric      LocPrf      Weight      Path
*>4.4.4.0/24  13.13.13.3    0           100         0           (65002)i
```

```
R3#show ip bgp community no-export
```

```
BGP table version is 6, locl routerID is 3.3.3.3
```

```
Status cods; s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
      r RIB-failure, S Stale
```

```
Origin codes;i-IGP, e-EGP, ?-incomplete
```

```

Network      Next Hop      Metric      LocPrf      Weight      Path
r>i4.4.4.0/24  4.4.4.4       0           100         0           i
```

以上输出表明路由器 R1 和 R3 的 BGP 表的” 4.4.4.0/24” 携带了” no-advertise” 团体属性。

(5) 步骤 3: 在路由器 R4 上配置团体属性 no-advertise

```
R4(config)#route-map NO_ADV permit 10
```

```
R4(config-route-map)#match ip address prefix-list 1
```

```
R4(config-route-map)#set community no-advertise
```

```
R4(config-route-map)#router bgp 65002
```

```
R4(config-router)#neighbor 3.3.3.3 send-community
```

```
R4(config-router)#neighbor 3.3.3.3 route-map NO-ADV out
```

(6) 团体属性 no-advertise 测试，分别在路由器 R3、R2 和 R1 上查看 BGP 表

```
R3#show ip bgp
```

```
BGP table version is 6, locl routerID is 3.3.3.3
```

```
Status cods; s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
      r RIB-failure, S Stale
```

```
Origin codes; i-IGP, e-EGP, ?-incomplete
```

| Network      | Next Hop   | Metric | LocPrf | Weight | Path      |
|--------------|------------|--------|--------|--------|-----------|
| *>1.1.1.0/24 | 13.13.13.1 | 0      | 100    | 0      | (65001) i |
| * 2.2.2.0/24 | 13.13.13.1 | 0      | 100    | 0      |           |
| (65001) 200i |            |        |        |        |           |

|    |            |   |  |   |      |
|----|------------|---|--|---|------|
| *> | 23.23.23.3 | 0 |  | 0 | 200i |
|----|------------|---|--|---|------|

|              |         |   |  |       |   |
|--------------|---------|---|--|-------|---|
| *>3.3.3.0/24 | 0.0.0.0 | 0 |  | 32768 | i |
|--------------|---------|---|--|-------|---|

|               |         |   |     |   |   |
|---------------|---------|---|-----|---|---|
| r>i4.4.4.0/24 | 4.4.4.4 | 0 | 100 | 0 | i |
|---------------|---------|---|-----|---|---|

```
R2#show ip bgp
```

```
BGP table version is 4, locl routerID is 2.2.2.2
```

```
Status cods; s suppressed, d damped, h history, *valid, .best, i-internal,
```

```
      r RIB-failure, S Stale
```

```
Origin codes; i-IGP, e-EGP, ?-incomplete
```

| Network      | Next Hop   | Metric | LocPrf | Weight | Path |
|--------------|------------|--------|--------|--------|------|
| * 1.1.1.0/24 | 23.23.23.3 |        |        | 0      | 100i |
| *>           | 12.12.12.1 | 0      |        | 0      | 100i |

|              |         |   |       |   |
|--------------|---------|---|-------|---|
| *>2.2.2.0/24 | 0.0.0.0 | 0 | 32768 | i |
|--------------|---------|---|-------|---|

|              |            |  |   |      |
|--------------|------------|--|---|------|
| * 3.3.3.0/24 | 12.12.12.1 |  | 0 | 100i |
|--------------|------------|--|---|------|

|    |            |   |   |      |
|----|------------|---|---|------|
| *> | 23.23.23.3 | 0 | 0 | 100i |
|----|------------|---|---|------|

R1#show ip bgp

BGP table version is 4, local routerID is 1.1.1.1

Status codes: s suppressed, d damped, h history, \*valid, .best, i-internal,

r RIB-failure, S Stale

Origin codes: i-IGP, e-EGP, ?-incomplete

| Network | Next Hop | Metric | LocPrf | Weight | Path |
|---------|----------|--------|--------|--------|------|
|---------|----------|--------|--------|--------|------|

|              |         |   |  |       |   |
|--------------|---------|---|--|-------|---|
| *>1.1.1.0/24 | 0.0.0.0 | 0 |  | 32768 | i |
|--------------|---------|---|--|-------|---|

|              |            |   |     |   |  |
|--------------|------------|---|-----|---|--|
| * 2.2.2.0/24 | 13.13.13.3 | 0 | 100 | 0 |  |
|--------------|------------|---|-----|---|--|

(65002) 200i

|    |            |   |  |   |      |
|----|------------|---|--|---|------|
| *> | 12.12.12.2 | 0 |  | 0 | 200i |
|----|------------|---|--|---|------|

|              |            |   |     |   |           |
|--------------|------------|---|-----|---|-----------|
| *>3.3.3.0/24 | 13.13.13.3 | 0 | 100 | 0 | (65002) i |
|--------------|------------|---|-----|---|-----------|

以上输出表明携带团体”no-advertise”属性的条目”4.4.4.0/24”之传递给路由器R3，并没有传递给路由器R2和R1。由此可见，携带”no-advertise”团体属性的条目被接收到后，将不通知给任何BGP对等体。

#### 【提示】

可以为一条BGP路由设置多个团体属性。