

值来代表。该值越高,可信度评价就越低。

边界路由器:

将一个自治系统与另一个自治系统连接起来的路由器,它常常负责路由再发布。

被动接口:

被配置成只接收路由更新而不发送路由更新的路由器接口。

策略路由:

根据用户所定义的策略将数据包转发到特定接口的路由方案。这种策略可以指定由某个特定网络发出的数据流应该被转发到某个指定接口,而其他数据流应该被转发到另一个接口。

路由过滤器:

使用“distribute-list”命令的一种配置,它用于从进入的或外出的路由更新中去掉一条或多条路由。

路由再发布:

将由一种路由选择协议所发现的路由信息发布到另一种路由信息中去的过程。它有时也被简称为“再发布”。

种子度量值:

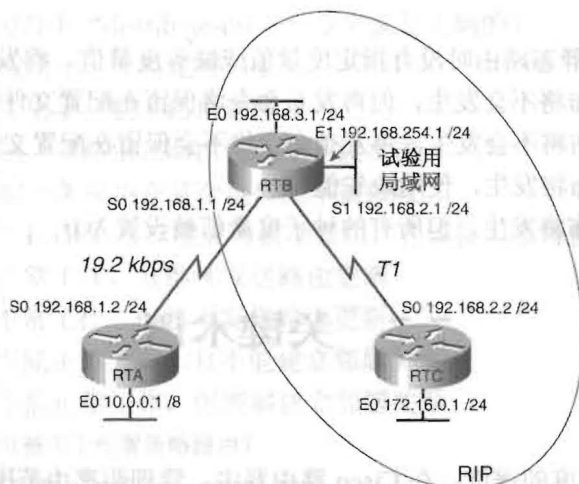
在路由再发布中,它是一条被输入进来的路由的初始度量值。

7.8 实验室练习

下面的3个实验室练习复习了本章所介绍过的重要概念。在实验7-1中,我们将配置一个路由过滤器、被动接口和路由再发布。在实验7-2中,我们将通过创建一个路由映射图来实施策略路由。最后,实验7-3提供了一个使用RIP和OSPF的复杂的路由再发布场景。

这些实验练习中还含有围绕着关键概念的一些问题,对这些问题的答案请参见附录B的“实验室练习问题答案”。

7.8.1 实验7-1: 配置发布控制列表和被动接口



1. 目标

在这个实验中，我们将组合配置一些高级路由特性以优化路由。这些特性包括发布列表 (distribute list)、被动接口、缺省路由和路由再发布。

2. 背景

XYZ 公司使用 RIP 作为其动态路由选择协议。不幸的是，RIP 的缺省行为不能产生想要的结果。

XYZ 公司使用一条非常慢的 19.2 kbit/s 链路来连接它的分支办事处路由器 RTA。为了减少带宽耗费，XYZ 公司想要我们在该链路上避免使用动态路由。

公司还想让我们在路由器 RTC 的以太网接口上过滤 RIP 更新，因为这些更新现在没有任何意义。

最后，路由器 RTB 连接到一个被保留用于研究和开发的局域网 (192.168.254.0/24)。XYZ 公司想让 192.168.3.0/24 网段上的用户具有对该实验网络的访问权，但却不想让公司的其余部分访问该局域网。XYZ 公司要求在路由器 RTB 的两个以太网接口上发送 RIP 路由更新以与这些局域网上的 UNIX 主机交换路由信息。

3. 步骤

在开始这个实验前，建议在删除各路由器的初始配置后再重新启动路由器。这样可以防止由残留配置所造成的问题。在我们准备好设备后，进行步骤 1。

步骤 1：根据前面的图示组建和配置网络，但暂时先不要配置 RIP。

用 “ping” 命令来核验我们的工作，并测试串行接口之间的连通性（注意，在做进一步配置之前，路由器 RTA 应该 ping 不到路由器 RTC）。

步骤 2：在路由器 RTC 上，配置 RIP 让它通告两个直连网络，如下所示：

```
RTC(config)#router rip
RTC(config-router)#network 192.168.2.0
RTC(config-router)#network 172.16.0.0
```

因为在以太网上不存在 RIP 路由器，所以该接口不需要发送 RIP 路由更新。然而，如果我们不在 RIP 配置中包含网络 172.16.0.0，路由器 RTC 就不会将它通告给路由器 RTB。但我们可以配置路由器 RTC 让它不在接口 E0 上发送 RIP 路由更新。使用下面的命令将路由器 RTC 的接口 E0 对 RIP 设为被动模式：

```
RTC(config-router)#passive-interface e0
```

这样一来，RIP 路由更新就不再通过接口 E0 被发送出去。

步骤 3：在路由器 RTB 上配置 RIP，只在网络 192.168.2.0/24 上启用 RIP，以让路由器 RTB 能与路由器 RTC 交换路由信息：

```
RTB(config)#router rip
RTB(config-router)#network 192.168.2.0
```

在路由器 RTB 上输入了该 RIP 配置后，用 “show ip route” 命令来检查路由器 RTC 的路由表。注意，路由器 RTC 还没有通过 RIP 学到任何路由：

```
RTC#show ip route
<output omitted>
172.16.0.0/24 is subnetted, 1 subnets
```

```
C 172.16.0.0 is directly connected, FastEthernet0/0
```

```
C 192.168.2.0/24 is directly connected, Serial0/0
```

问题 1. 路由器 RTC 为什么还没有学到网络 192.168.3.0/24 和 192.168.1.0/24?

步骤 4: 在检查了 XYZ 公司的需求后, 我们决定在路由器 RTB 的接口 E0 和 E1 上启用 RIP, 以让这些局域网上的 UNIX 主机能收到路由信息:

```
RTB(config-router)#network 192.168.3.0
```

```
RTB(config-router)#network 192.168.254.0
```

现在, RIP 在按照公司的要求在这些网络上发送路由更新。再次检查路由器 RTC 的路由表:

```
RTC#show ip route
```

```
<output omitted>
```

```
172.16.0.0/24 is subnetted, 1 subnets
```

```
C 172.16.0.0 is directly connected, FastEthernet0/0
```

```
R 192.168.254.0/24 [120/1] via 192.168.2.1, 00:00:07, Serial0/0
```

```
C 192.168.2.0/24 is directly connected, Serial0/0
```

```
R 192.168.3.0/24 [120/1] via 192.168.2.1, 00:00:27, Serial0/0
```

路由器 RTC 现在学到网络 192.168.3.0/24 (这是想要的) 和 192.168.254.0/24 (这是不想要的) 的路由。请记住, XYZ 公司想要让网络 192.168.254.0/24 对公司的其余部分不可见。

步骤 5: 要让路由器 RTB 不发送有关网络 192.168.254.0/24 的路由更新 (且不对那个网络关闭 RIP), 我们可以用 “distribute-list” 命令将它从外出路由更新中去除。发布列表允许我们过滤进入的或外出的路由更新的内容:

```
RTB(config)#access-list 1 deny 192.168.254.0
```

```
RTB(config)#access-list 1 permit any
```

```
RTB(config)#router rip
```

```
RTB(config-router)#distribute-list 1 out
```

在路由器 RTB 上用 “show ip protocols” 命令验证该过滤器已经被应用上了。

问题 2. 根据该命令的输出结果, 该对外出方向的路由更新过滤列表被应用在哪个接口上?

在路由器 RTB 上配置了该发布列表之后, 返回到路由器 RTC, 并用 “clear ip route *” 命令重清其路由表。至少等待 5 秒钟, 然后再用 “show ip route” 命令检查路由器 RTC 的路由表。

问题 3. 到网络 192.168.254.0/24 去的路由在路由器 RTC 的路由表中吗?

问题 4. 到网络 192.168.3.0/24 去的路由在路由器 RTC 的路由表中吗?

步骤 6: 路由器 RTC 的路由表已基本完整了, 但它还不含有到网络 192.168.1.0/24 的路由, 该网络是路由器 RTB 的一个直连网络。我们可以在路由器 RTB 的 RIP 路由进程配置中输入相应的 “network” 命令, 以让它通告该网络。当然, XYZ 公司不想让 RIP 路由更新在速率为 19.2 kbit/s 的低速链路上被发送, 所以我们将不得不把路由器 RTB 的 S0 接口置为被动模式。但我们还有另外一种解决办法: 配置路由器 RTB 将直连路由再发布到 RIP 中。在路由器 RTB 上输入下面的命令:

```
RTB(config
```

```
RTB(conf
```

```
在发出了
```

```
络 192.168.1.0
```

```
通过在路由器
```

```
RTB#show
```

```
Routing
```

```
Known
```

```
Redist
```

```
Advert
```

```
Routing
```

```
* dire
```

```
Route
```

```
从该命令
```

```
检查路由
```

```
192.168.3.0/24
```

```
步骤 7: 在
```

```
:
```

```
:
```

```
路由器 R
```

```
可以配置一条
```

```
RTA(conf
```

```
校验路由
```

```
命令。让路由
```

```
控制台, ping
```

```
RTC 的调试输
```

```
下一步,
```

```
入 “ping [ent
```

```
接口 E0 (10.
```

```
RTA#pin
```

```
Protoco
```

```
Target
```

```
Repeat
```

```
Datagra
```

```
Timeout
```

```
Extende
```

```
Source
```

```
Type of
```

```
Set DF
```

```
Validat
```

```
RTB(config)#router rip
```

```
RTB(config-router)#redistribute connected
```

在发出了该命令之后，路由器 RTB 会将所有直连路由输入到 RIP 路由进程中。因此，网络 192.168.1.0/24 被再发布到 RIP 中，并被作为 RIP 路由更新的一部分发送给路由器 RTC。通过在路由器 RTB 上发出下面的命令来验证我们的配置：

```
RTB#show ip route 192.168.1.1
```

```
Routing entry for 192.168.1.0/24
```

```
Known via "connected ", distance 0, metric 0 (connected, via interface)
```

```
Redistributing via rip
```

```
Advertised by rip
```

```
Routing Descriptor Blocks:
```

```
* directly connected, via Serial0/0
```

```
Route metric is 0, traffic share count is 1
```

从该命令的输出我们可以确定，该直连路由被再发布到 RIP 中并被 RIP 通告出去。

检查路由器 RTC 的路由表。路由器 RTC 现在应该有到网络 192.168.1.0/24 和 192.168.3.0/24 的 RIP 路由。

步骤 7：在路由器 RTB 和 RTC 之间的路由基本完成之后，再将注意力转向路由器 RTA。

XYZ 公司要求我们在 RTA 的广域网链路上不要实施动态路由，所以我们必须使用一条静态路由。

路由器 RTA 是一个末节网络，因为它到外部世界只有一个出口点。在这种情况下，我们可以配置一条静态缺省路由，它用于所有的非本地数据流：

```
RTA(config)#ip route 0.0.0.0 0.0.0.0 192.168.1.1
```

核验路由器 RTA 在使用缺省路由。首先，从路由器 RTC 的控制台输入“debug ip packet”命令。让路由器 RTC 的控制台会话保持打开状态，同时回到路由器 RTA。从路由器 RTA 的控制台，ping 路由器 RTC 的接口 E0 (172.16.0.1)。这些 ping 应该是成功的。注意，路由器 RTC 的调试输出报告说，这些 ping 已经被收到并做了回应。

下一步，用扩展的 ping 命令来 ping 路由器 RTC（扩展 ping 命令是通过在特权模式下输入“ping [enter]”命令激活的）。在扩展 ping 命令中，将 ping 的源地址指定为路由器 RTA 的接口 E0 (10.0.0.1)，如下所示：

```
RTA#ping
```

```
Protocol [ip ]: ip
```

```
Target IP address: 172.16.0.1
```

```
Repeat count [5 ]: 5
```

```
Datagram size [100 ]: 100
```

```
Timeout in seconds [2 ]: 2
```

```
Extended commands [n ]: y
```

```
Source address or interface: 10.0.0.1
```

```
Type of service [0 ]: 0
```

```
Set DF bit in IP header?[no ]: no
```

```
Validate reply data?[no ]: no
```


Data pattern [0xABCD]: 0xABCD

Loose, Strict, Record, Timestamp, Verbose [none]: none

Sweep range of sizes [n]: n

问题5. 这些ping 成功了吗?

检查路由器 RTC 上“debug ip packet”命令的输出结果。我们应该能看到有 ping 包 (ICMP 回声请求包) 到达。

问题6. 路由器 RTC 为什么不作响应?

检查路由器 RTC 的路由表。

问题7. 路由器 RTC 有到网络 10.0.0.0/8 的路由吗?

步骤8: 要让路由器 RTB 和 RTC 能路由到网络 10.0.0.0/8, 我们必须配置一条静态路由。我们决定在路由器 RTB 上配置该静态路由, 然后让路由器 RTB 将该路由动态传播给其他路由器 (RTC) (这可以让我们不必在每台路由器都输入一条静态路由)。在路由器 RTB 上输入下列命令:

```
RTB(config)#ip route 10.0.0.0 255.0.0.0 192.168.1.2
```

该命令配置了一条去往网络 10.0.0.0/8 的静态路由, 它使用路由器 RTA 的接口 S0 作为下一跳。

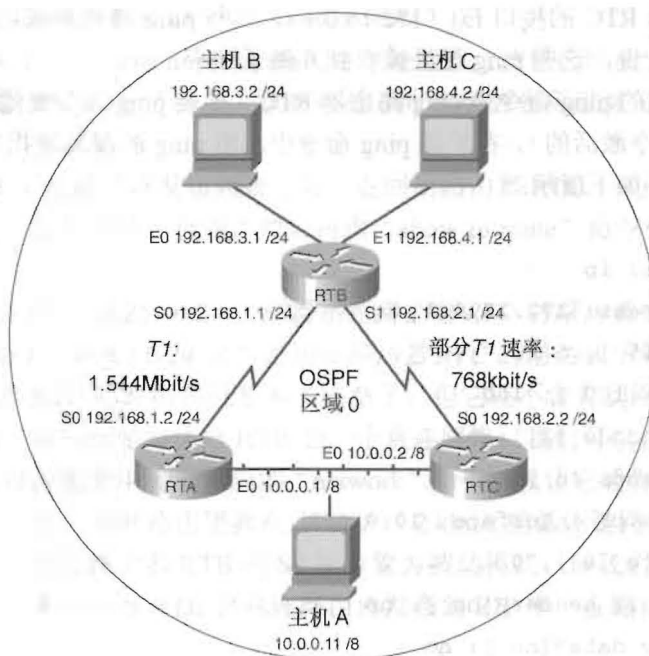
要让路由器 RTB 对路由器 RTC 动态发送有关该路由信息的更新, 我们应在路由器 RTB 上配置 RIP 将静态路由再发布进来。可以使用下面的命令。

```
RTB(config)#router rip
```

```
RTB(config-router)#redistribute static
```

最后, 检查路由器 RTC 的路由表。它现在应该是完整的了。通过从路由器 RTC ping 路由器 RTA 的接口 E0 来验证连通性。

7.8.2 实验 7-2: 配置路由映射图 (Route Map)



1. 目标

在这个实

2. 背景

XYZ 公司
Mbit/s), 另一
用 T1 链路,
定, 从局域网
另有新的通知

3. 步骤

在开始这
止由残留配置

步骤 1:

用“ping
间的连通性。

步骤 2:

问题 8.

配置路由

RTB (con

RTB (con

RTC (con

RTC (con

在设置

问题 9.

步骤 3:

RTB (co

下一步

访问控制列

RTB (co

RTB (co

RTB (co

该策略

在接口 E0

RTB (c

该命令

1. 目标

在这个实验中，我们将通过配置一个路由映射图来应用一个路由策略。

2. 背景

XYZ 公司的维护着到网络 10.0.0.0/8 的两条广域网链路。一条链路的速率是 T1 (1.544 Mbit/s)，另一条是 T1 链路的一部分，其速率为 768 kbit/s。在 OSPF 下，路由器 RTB 优先选用 T1 链路，因为它的带宽高（路由成本低）。然而，XYZ 公司的网络运维中心（NOC）决定，从局域网 192.168.3.0 去往网络 10.0.0.0/8 的所有数据流应取道低速的部分 T1 链路，除非另有新的通知。我们必须通过在路由器 RTB 上配置一个路由映射图来实施该策略。

3. 步骤

在开始这个实验前，建议在删除各路由器的初始配置后再重新启动路由器。这样可以防止由残留配置所造成的问题。在我们准备好设备后，进行步骤 1。

步骤 1：根据前面的图示组建和配置网络。为 OSPF 区域 0 配置所有的接口。按图所示用 IP 地址和缺省网关配置主机 A 和主机 B。

用“ping”命令和“show ip route”命令来核验我们的工作并测试所有接口和两台主机之间的连通性。

步骤 2：

问题 8. 检查路由器 RTB 的路由表。它有多少条到网络 10.0.0.0/8 的路由？

配置路由器 RTB 的接口 S1 和路由器 RTC 的接口 S0，让它们精确反映广域网链路的带宽：

```
RTB(config)#interface s1
```

```
RTB(config-if)#bandwidth 768
```

```
RTC(config)#interface s0
```

```
RTC(config-if)#bandwidth 768
```

在设置了带宽之后，检查路由器 RTB 的路由表。应该只有一条到网络 10.0.0.0/8 的路由。

问题 9. OSPF 使用哪个接口来去往网络 10.0.0.0/8？

步骤 3：配置一个路由映射图来迫使路由器 RTB 使用接口 S1 路由从网络 192.168.3.0/24 去往网络 10.0.0.0/8 的数据流。在配置路由映射图之前，我们必须创建一个访问控制列表，以匹配需要被策略路由的数据流。因为我们想要影响源自网络 192.168.3.0/24 的数据流，所以我们将路由器 RTB 上创建下列访问控制列表：

```
RTB(config)#access-list 1 permit 192.168.3.0 0.0.0.255
```

下一步，创建一个路由映射图，给它起名为“slow4u”。该路由映射图引用我们刚创建的访问控制列表 1。可以使用下面的命令：

```
RTB(config)#route-map slow4u permit 10
```

```
RTB(config-route-map)#match ip address 1
```

```
RTB(config-route-map)#set interface s1
```

该策略将被应用于路由器 RTB 的接口 E0，因为该接口将接收要被策略路由的数据流。

在接口 E0 上，输入下列命令：

```
RTB(config-if)#ip policy route-map slow4u
```

该命令将名为“slow4u”的策略应用到接口 E0 上。

在完成了该配置之后,用“show ip route”命令来验证接口 S0 是路由器去往网络 10.0.0.0/8 所用的外出接口。

```
RTB#show ip route
```

```
<output omitted>
```

```
C 192.168.4.0/24 is directly connected, FastEthernet0/1
```

```
O 10.0.0.0/8 [110/65] via 192.168.1.2, 00:09:52, Serial10/0
```

```
C 192.168.1.0/24 is directly connected, Serial10/0
```

```
C 192.168.2.0/24 is directly connected, Serial10/1
```

```
C 192.168.3.0/24 is directly connected, FastEthernet0/0
```

步骤 4: 验证我们的策略是否生效了。首先,在路由器 RTB 的控制台上发出“debug ip policy”命令。让该窗口保持打开状态。

从主机 B,使用一个路由追踪程序(例如“tracert”)来跟踪到目的地 10.0.0.1 的路由。

```
HostB:\cisco>tracert 10.0.0.1
```

在该路由跟踪过程中,我们应该能看到“debug ip policy”命令的输出指示出数据包被策略路由了。

分析主机 B 路由跟踪的结果。

问题 10. 该路由跟踪经过 192.168.2.2 了吗?

问题 11. 数据包是从路由器 RTB 的哪个接口外出的?

从主机 C,使用一个路由追踪程序(例如“tracert”)来跟踪到目的地 10.0.0.2 的路由。

```
HostC:\cisco>tracert 10.0.0.2
```

分析主机 C 路由跟踪的结果。

问题 12. 该路由跟踪经过 192.168.1.2 了吗?

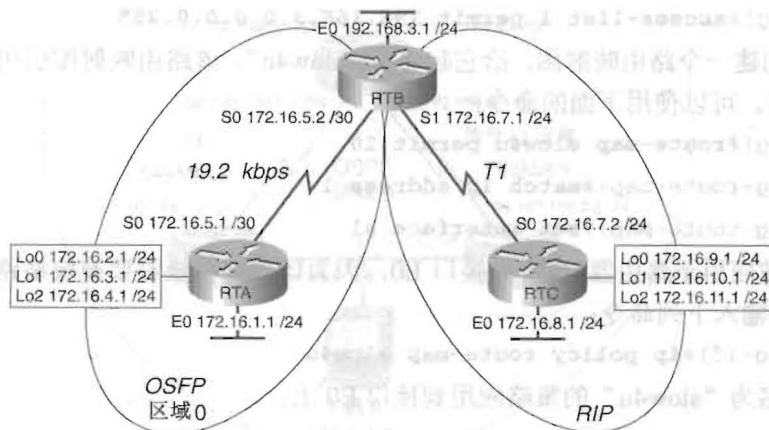
问题 13. 数据包是从路由器 RTB 的哪个接口外出的?

最后,在路由器 RTB 上发出“show route-map”命令。

```
RTB#show route-map
```

问题 14. 有多少数据包被执行了策略路由?

7.8.3 实验 7-3: 再发布 RIP 和 OSPF



1. 目标

在这个实验中，我们将配置 RIPv1 和 OSPF 之间的双向路由再发布。

2. 背景

XYZ 公司为其广域网的一部分使用 RIPv1，因为这部分网络必须支持老式路由器和主机。OSPF 被用作网络其余部分的路由选择协议。公司要求我们在 ASBR 路由器 RTB 上为这两种协议配置双向路由再发布。在 RIP 路由域中的路由器应该了解 OSPF 网络，OSPF 路由域中的路由器应该了解 RIP 网络。

3. 步骤

在开始这个实验前，建议在删除各路由器的初始配置后再重新启动路由器。这样可以防止由残留配置所造成的问题。在我们准备好设备后，进行步骤 1。

步骤 1：根据前面的图示组建和配置网络。按图所示为路由器的环回接口配置 IP 地址。

这些环回接口将用于模拟在每个路由域中其他网络的存在。在路由器 RTA 上为所有直连网络配置 OSPF，在路由器 RTC 上为所有直连网络配置 RIP。暂时先不要在路由器 RTB 上配置路由功能。

用 “show running-config” 命令来验证我们的工作。

步骤 2：在路由器 RTB 的串行接口上配置路由功能。使用下面的命令来配置路由器 RTB 的路由进程：

```
RTB(config)#router rip
RTB(config-router)#network 172.16.0.0
RTB(config-router)#passive-interface s0
RTB(config-router)#passive-interface e0
RTB(config-router)#router ospf 1
RTB(config-router)#network 172.16.5.0 0.0.0.3 area 0
```

问题 15. 为什么在 RIP 中对接口 S0 和 E0 应用 “passive-interface” 命令语句？

问题 16. 为什么在 OSPF 配置中不需要配置任何被动接口？

在配置了路由功能之后，检查路由器 RTB 的路由表。路由器 RTB 应该通过 OSPF 学到了路由器 RTA 的直连网络，并通过 RIP 学到了路由器 RTC 的直连网络。否则可能就需要进行检查并排除故障了。

现在检查一下路由器 RTA 的路由表。目前应该不存在动态学到的路由。最后，检查路由器 RTC 的路由表。

问题 17. 路由器 RTC 通过 RIP 学到任何路由了吗？为什么？

步骤 3：在路由器 RTB 上配置路由再发布，以将 OSPF 路由注入到 RIP 路由进程中。可以使用下面的命令在路由器 RTB 上配置路由再发布：

```
RTB(config)#router rip
RTB(config-router)#redistribute ospf 1
RTB(config-router)#default-metric 2
```

问题 18. “default-metric” 命令是干什么用的？我们需要使用它吗？

检查 RIP 路由域中路由器 RTC 的路由表。

问题 19. 路由器 RTC 从 OSPF 那边学到任何网络了吗?

问题 20. 这些路由的度量值是多少?

步骤 4: 在路由器 RTB 上配置路由再发布, 以将 RIP 路由注入到 OSPF 路由进程中。可

以使用下面的命令在路由器 RTB 上配置路由再发布:

```
RTB(config)#router ospf 1
```

```
RTB(config-router)#redistribute rip
```

```
RTB(config-router)#default-metric 10
```

检查 OSPF 路由域中路由器 RTA 的路由表。路由器 RTA 还没有从 RIP 那边学到任何网络。

问题 21. 路由器 RTA 为什么还没有从 RIP 那边学到任何网络?

步骤 5: 路由器 RTA 的路由表应该没有什么新路由。这是因为 OSPF 缺省情况下不会再

发布子网路由, 除非在“redistribute”命令中明确使用“subnets”关键字。回

到路由器 RTB 上, 输入下列配置:

```
RTB(config)#router ospf 1
```

```
RTB(config-router)#redistribute rip subnets
```

在输入了这些命令之后, 再次检查路由器 RTA 的路由表。

问题 22. 路由器 RTA 从 RIP 路由域中学到任何路由了吗?

问题 23. 这些路由的度量值是多少?

在路由表中, 我们将看到被再发布进来的路由被标缀有字符“E2”。

问题 24. 这些字符的含意是什么?

步骤 6:

问题 25. 仔细看一下路由器 RTA 的路由表。有到网络 192.168.3.0/24 的路由吗?

问题 26. 路由器 RTC 有到网络 192.168.3.0/24 的路由吗?

问题 27. 路由器 RTA 的路由表中为什么没有这条路由?

我们可以通过将路由器 RTB 配置为向 OSPF 中再发布直连路由来补全路由器 RTA 的路由表:

```
RTB(config)#router ospf 1
```

```
RTB(config-router)#redistribute connected subnets
```

问题 28. 被再发布的直连路由的种子度量值是多少?

还应配置路由器 RTB 的 RIP 路由进程来再发布直连路由:

```
RTB(config)#router ospf 1
```

```
RTB(config-router)#redistribute connected subnets
```

最后再检查一下路由表。路由器 RTA 和 RTB 都应该具有所有的路由。路由器 RTC 将有 10 条路由, 因为网络 172.16.0.0 被划分了变长子网, 而 RIPv1 不支持 VLSM。因此, 路由器 RTC 没有到 172.16.5.0/30 的路由。