

# Configure Protocol Redistribution for Routers

## Contents

---

### [Introduction](#)

### [Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Conventions](#)

### [Background Information](#)

### [Metrics](#)

### [Administrative Distance](#)

### [Redistribution Configuration Syntax and Examples](#)

[EIGRP](#)

[OSPF](#)

[RIP](#)

### [Redistribute Static Routes Except Gateway of Last resort in RIP With Route Map](#)

[IS-IS](#)

### [Connected Routes](#)

---

## Introduction

This document describes how you can redistribute a routing protocol, connected or static routes, into another dynamic routing protocol.

## Prerequisites

### Requirements

There are no specific requirements for this document.

### Components Used

The information in this document is based on these software and hardware versions:

- Cisco IOS® Software Release 12.2(10b)
- Cisco 2500 Series Routers

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

### Conventions

Refer to [Cisco Technical Tips Conventions](#) for more information on document conventions.

## Background Information

When you must redistribute a single routing protocol you can consider distribution through multiple protocol routing. Multiple protocol routing is use when a company mergers, multiple departments managed by multiple network administrators, and multi-vendor environments. Part of network design is when you run different routing protocols. In any case, when you have a multiple protocol environment it makes redistribution a necessity.

Differences in routing protocol characteristics, such as metrics, administrative distance, classful and classless capabilities can effect redistribution. Consideration must be given to these differences for redistribution to succeed.

## Metrics

When you redistribute one protocol into another, remember that the metrics of each protocol play an important role in redistribution. Each protocol uses different metrics. For example, the Routing Information Protocol (RIP) metric is based on hop count, and Enhanced Interior Gateway Routing Protocol (EIGRP) use a composite metric-based on bandwidth, delay, reliability, load, and maximum transmission unit (MTU), where bandwidth and delay are the only parameters used by default. When routes are redistributed, you must define a metric for a protocol that the route that receives can understand. There are two methods to define metrics when routes are redistributed.



*OSPF and RIP Topology*

1. You can define the metric for that specific redistribution only:

```
router rip
 redistribute static metric 1
 redistribute ospf 1 metric 1
```

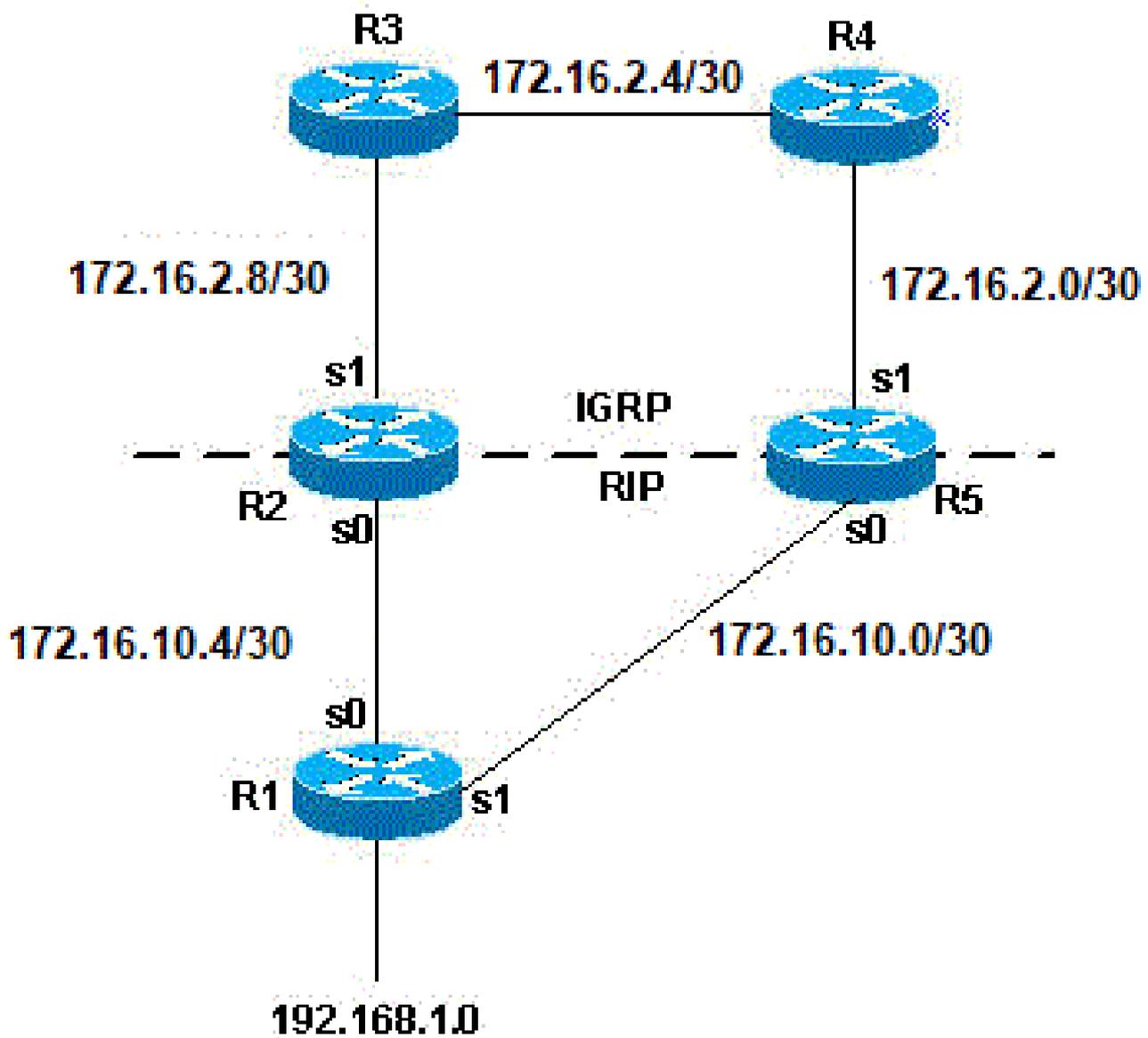
2. You can use the same metric as a default for all redistribution (with the **default-metric** command which saves you work because it eliminates the need to define the metric separately for each redistribution):

```
router rip
 redistribute static
 redistribute ospf 1
 default-metric 1
```

## Administrative Distance

If a router runs more than one routing protocol and learns a route to the same destination with both routing protocols, then which route must be selected as the best route? Each protocol uses its own metric type to determine the best route. You cannot compare routes with different metric types. Administrative distances take care of this problem. Administrative distances are assigned to route sources so that the route from the most preferred source can be chosen as the best path. Refer to [Route Selection in Cisco Routers](#) for more information about administrative distances and route selection.

Administrative distances help with route selection among different routing protocols, but they can cause problems for redistribution. These problems can be in the form of routing loops, convergence problems, or inefficient routing. See this next image displays a topology and description of a possible problem.



*Topology of a Possible Problem*

In the previous topology example, if R1 runs RIP, and R2 and R5 both run RIP and EIGRP and redistribute RIP into EIGRP, then there is a potential problem. For example, R2 and R5 both learn about network 192.168.1.0 from R1 through RIP. This knowledge is redistributed into EIGRP. R2 learns about network 192.168.1.0 through R3, and R5 learns about it from R4 through EIGRP. EIGRP has a lower administrative distance than RIP (90 versus 120); therefore, the EIGRP route is used in the routing table. Now there is a potential routing loop. Even if split horizon, or any other feature meant to help prevent routing loops is used, there is still a convergence problem.

If R2 and R5 also redistribute EIGRP into RIP (this is mutual redistribution) and the network, 192.168.1.0, is not directly connected to R1 (R1 learns from another router upstream from it), then there is a potential problem that R1 can learn the network from R2 or R5 with a better metric than from the original source.

---

 **Note:** The mechanics of route redistribution is proprietary on Cisco routers. The rules for redistribution on a Cisco router dictate that the redistributed route be present in the routing table. It is not sufficient that the route be present in the routing topology or database. Routes with a lower Administrative Distance (AD) are always installed in the routing table. For example, if a static route is redistributed into EIGRP on R5, and then EIGRP subsequently redistributed into RIP on the same router (R5), the static route is not redistributed into RIP because it never got entered into the EIGRP routing table. This is due to the fact that static routes have an AD of 1 and EIGRP routes have an AD of 90 and the static route is installed in the routing table. In order to redistribute the static route into EIGRP on R5, you need to use the **redistribute static** command under the **router rip** command.

---

The default behavior for RIP and EIGRP is to advertise directly connected routes when a **network** statement under the routing protocol includes the connected interface subnet. There are two methods to get a connected route:

- An interface is configured with an IP address and mask, this corresponding subnet is considered a connected route.
- A static route is configured with only an outgoing interface, and not an IP next-hop, this is also considered a connected route.

<#root>

```
Router#configure terminal
Router(config)#
ip route 10.0.77.0 255.255.255.0 ethernet 0/0
Router(config)#end
Router#
show ip route static

10.0.0.0/24 is subnetted, 1 subnets

S 10.0.77.0 is directly connected, Ethernet0/0
```

A **network** command configured under EIGRP or RIP that includes (or "covers") either of these types of connected routes includes that subnet for advertisement.

For example, if an interface has address 10.0.23.1 and mask 255.255.255.0, the subnet 10.0.23.0/24 is a connected route and can be advertised by these routing protocols when a **network** statement is configured:

```
router rip | eigrp #
network 10.0.0.0
```

This static route, 10.0.77.0/24, is also advertised by these routing protocols, because it is a connected route

and it is "covered" by the **network** statement.

See the Avoid Problems Due to Redistribution section of this document for more information.

## Redistribution Configuration Syntax and Examples

### EIGRP

This output shows an EIGRP router that redistributes static, Open Shortest Path First (OSPF), RIP, and Intermediate System-to-Intermediate System (IS-IS) routes.

```
router eigrp 1
network 10.10.108.0
redistribute static
redistribute ospf 1
redistribute rip
redistribute isis
default-metric 10000 100 255 1 1500
```

EIGRP need five metrics when they redistribute other protocols: bandwidth, delay, reliability, load, and MTU, respectively.

Metric	Value
bandwidth	In units of kilobits per second; 10000 for Ethernet
delay	In units of tens of microseconds; for Ethernet it is $100 \times 10$ microseconds = 1 ms
reliability	255 for 100 percent reliability
load	Effective load on the link expressed as a number from 0 to 255 (255 is a 100 percent load)
MTU	Minimum MTU of the path; usually equals that for the Ethernet interface, which is 1500 bytes

Multiple EIGRP processes can run on the same router, with redistribution between them. For example, EIGRP1 and EIGRP2 can run on the same router. However, you do not need to run two processes of the same protocol on the same router, and this can consume the router memory and CPU. The redistribution of EIGRP into another EIGRP process does not require any metric conversion, so there is no need to define metrics or use the **default-metric** command with the redistribution.

A redistributed static route takes precedence over the summary route because the static route has an administrative distance of 1 while the EIGRP summary route has an administrative distance of 5. This happens when a static route is redistributed with the use of the `redistribute static` command under the EIGRP process and the EIGRP process has a default route.

## OSPF

This output shows an OSPF router that redistributes static, RIP, EIGRP, and IS-IS routes.

```
router ospf 1
network 10.10.108.0 0.0.255.255 area 0
redistribute static metric 200 subnets
redistribute rip metric 200 subnets
redistribute eigrp 1 metric 100 subnets
redistribute isis metric 10 subnets
```

The OSPF metric is a cost value based on  $10^8$ / bandwidth of the link in bits/sec. For example, the OSPF cost of Ethernet is 10:  $10^8/10^7 = 10$

---

 **Note:** If a metric is not specified, OSPF puts a default value of 20 when it redistributes routes from all protocols except Border Gateway Protocol (BGP) routes, which gets a metric of 1.

---

When there is a major net that is sub-netted, you need to use the keyword sub-netted to redistribute protocols into OSPF. Without this keyword, OSPF only redistributes major nets that are not sub-netted.

It is possible to run more than one OSPF process on the same router. This is rarely needed and it consumes the router memory and CPU.

You do not need to define metric or use the **default-metric** command when you redistribute one OSPF process into another.

## RIP

---

 **Note:** The principles in this document apply to RIP versions I and II.

---

This output shows a RIP router that redistributes static, EIGRP, OSPF, and IS-IS routes:

```
router rip
network 10.10.108.0
redistribute static
redistribute eigrp 1
redistribute ospf 1
redistribute isis
default-metric 1
```

The RIP metric is composed of hop count, and the maximum valid metric is 15. Anything greater than 15 is considered infinite; you can use 16 to describe an infinite metric in RIP. When you redistribute a protocol into RIP, Cisco recommends that you use a low metric, such as 1. A high metric, such as 10, limits RIP even further. If you define a metric of 10 for redistributed routes, these routes are only advertised to routers up to 5 hops away, at which point the metric (hop count) exceeds 15. If you define a metric of 1, you enable a route to travel the maximum number of hops in a RIP domain. But this can increase the possibility of routing loops if there are multiple redistribution points and if a router learns about the network with a better

metric from the redistribution point than from the original source. Therefore, you have to make sure that the metric is neither too high, which prevents the route from advertisement to all the routers, or too low, which leads to routing loops when there are multiple redistribution points.

## Redistribute Static Routes Except Gateway of Last resort in RIP With Route Map

This configuration is an example of how to redistribute static routes, except gateway of last gateway resort, in RIP through a route map.

This is the initial configuration for this example:

```
router rip
version 2
network 10.0.0.0
default-information originate
no auto-summary
!
ip forward-protocol nd
ip route 0.0.0.0 0.0.0.0 10.32.32.3
ip route 10.32.42.211 255.255.255.255 192.168.0.102
ip route 10.98.0.0 255.255.255.0 10.32.32.1
ip route 10.99.0.0 255.255.255.0 10.32.32.1
ip route 10.99.99.0 255.255.255.252 10.32.32.5
ip route 10.129.103.128 255.255.255.240 10.32.31.1
ip route 172.16.231.0 255.255.255.0 10.32.32.5
ip route 172.16.28.0 255.255.252.0 10.32.32.5
ip route 192.168.248.0 255.255.255.0 10.32.32.5
ip route 192.168.0.43 255.255.255.0 10.32.32.5
ip route 192.168.0.103 255.255.255.0 10.32.32.5
```

To create this configuration:

1. Create an access-list in order to match all networks that need to be redistributed:

```
Router#show access-lists 10
Standard IP access list 10
 10 permit 10.32.42.211
 20 permit 10.98.0.0, wildcard bits 0.0.0.255
 30 permit 10.99.0.0, wildcard bits 0.0.0.255
 40 permit 10.129.103.128, wildcard bits 0.0.0.15
 50 permit 172.16.231.0, wildcard bits 0.0.0.255<
 60 permit 172.16.28.0, wildcard bits 0.0.3.255
 70 permit 192.168.248.0, wildcard bits 0.0.0.255
 80 permit 192.168.0.43, wildcard bits 0.0.0.255
 90 permit 192.168.0.103, wildcard bits 0.0.0.255
```

2. Call this access-list in a route map.

```
route-map TEST
  match ip address 10
```

3. Redistribute in RIP with the route map at and remove the **default information originate** command from the RIP process.

```
router RIP
  version 2
  network 10.0.0.0
  redistribute static route-map TEST
  no auto-summary
```

## IS-IS

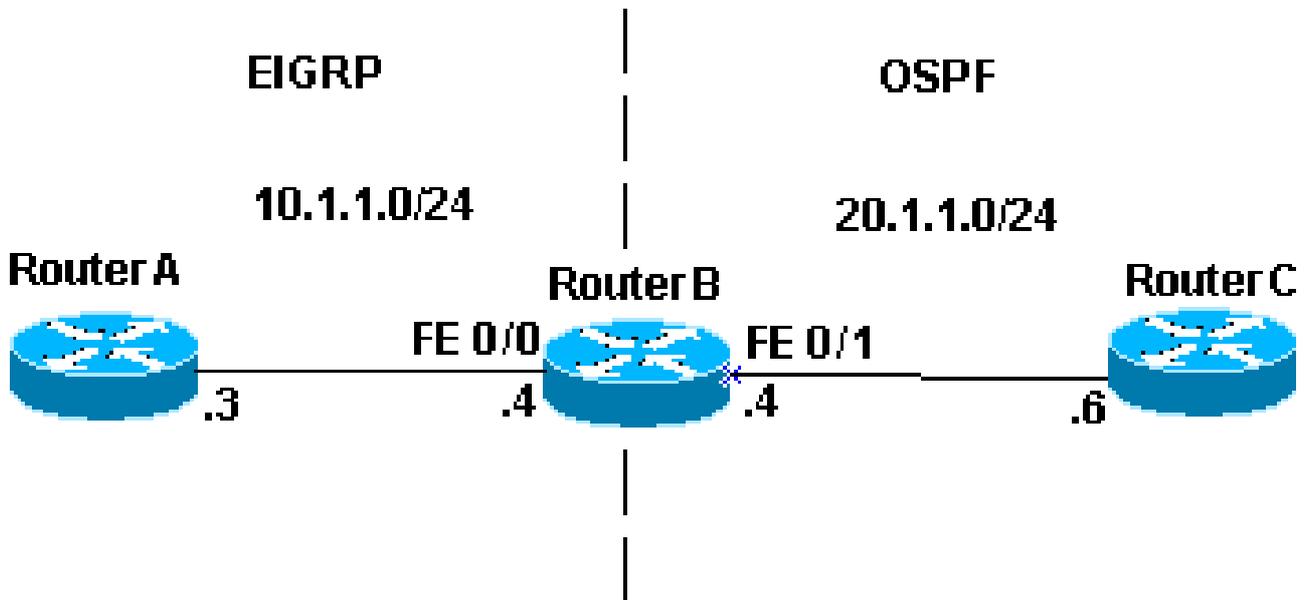
This output shows an IS-IS router that redistributes static, RIP, EIGRP, and OSPF routes.

```
router isis
  network 49.1234.1111.1111.1111.00
  redistribute static
  redistribute rip metric 20
  redistribute eigrp 1 metric 20
  redistribute ospf 1 metric 20
```

The IS-IS metric must be between 1 and 63. There is no default metric option in IS-IS. You must define a metric for each protocol, as shown in the previous example. If no metric is specified for the routes that are redistributed into IS-IS, a metric value of 0 is used by default.

## Connected Routes

**Direct redistribution** of connected networks into routing protocols is not a common practice and is not shown in any of the examples in this document for this reason. However, it is important to notice that it can be done, both directly and indirectly. In order to directly redistribute connected routes, use the **redistribute connected** router configuration command. You must also define a metric in this case. You can also indirectly redistribute connected routes into routing protocols as shown in this example:



### *Redistribute Connected Routers*

In the image example, Router B has two Fast Ethernet interfaces. FastEthernet 0/0 is in network 10.1.1.0/24 and FastEthernet 0/1 is in network 10.1.1.0/24. Router B runs EIGRP with Router A, and OSPF with Router C. Router B is mutually redistributed between the EIGRP and OSPF processes. This is the configuration information for Router B:

```
interface FastEthernet0/0
 ip address 10.1.1.4 255.255.255.0

interface FastEthernet0/1
 ip address 10.1.10.4 255.255.255.0

router eigrp 7
 redistribute ospf 7 metric 10000 100 255 1 1500
 network 10.1.1.0 0.0.0.255
 auto-summary
 no eigrp log-neighbor-changes
!
router ospf 7
 log-adjacency-changes
 redistribute eigrp 7 subnets
 network 10.1.1.0 0.0.0.255 area 0
```

The routing table for Router B displays:

```
<#root>

routerB#
show ip route
```

Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP  
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area  
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP  
i - IS-IS, L1 - IS-IS level-1, 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

Gateway of last resort is not set

```
10.0.0.0/24 is subnetted, 1 subnets
C    10.1.1.0 is directly connected, FastEthernet0/1
10.0.0.0/24 is subnetted, 1 subnets
C    10.1.1.0 is directly connected, FastEthernet0/0
```

From the previous configuration and routing table there are three pertinent things to notice:

- The networks in question are in the Router B routing table as directly connected networks.
- Network 10.1.1.0/24 is part of the EIGRP process and network 10.1.1.0/24 is part of the OSPF process.
- Router B mutually redistributes between EIGRP and OSPF.

Router A and C routing tables:

<#root>

routerA#

show ip route

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
U - per-user static route, o - ODR
```

Gateway of last resort is not set

```
10.0.0.0/24 is subnetted, 1 subnets
C    10.1.1.0 is directly connected, FastEthernet0
10.0.0.0/24 is subnetted, 1 subnets
D EX 10.1.1.0 [170/284160] via 10.1.1.4, 00:07:26, FastEthernet0
```

routerC#

show ip route

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, 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

```
10.0.0.0/24 is subnetted, 1 subnets
C    10.1.1.0 is directly connected, FastEthernet1 O E2
10.1.1.0 [110/20] via 10.1.10.4, 00:07:32, FastEthernet1
```

Router A has learned about network 10.1.1.0/24 via EIGRP, which is shown as an external route because it was redistributed from OSPF into EIGRP. Router C has learned about network 10.1.1.0/24 via OSPF as an external route because it was redistributed from EIGRP into OSPF. Although Router B does not redistribute connected networks, it does advertise the network 10.1.1.0/24, which is part of the EIGRP process redistributed into OSPF. Similarly, Router B advertises network 10.1.1.0/24, which is part of the OSPF process redistributed into EIGRP.

Refer to [Redistributing Connected Networks into OSPF](#) for more information about connected routes redistributed into OSPF.

---

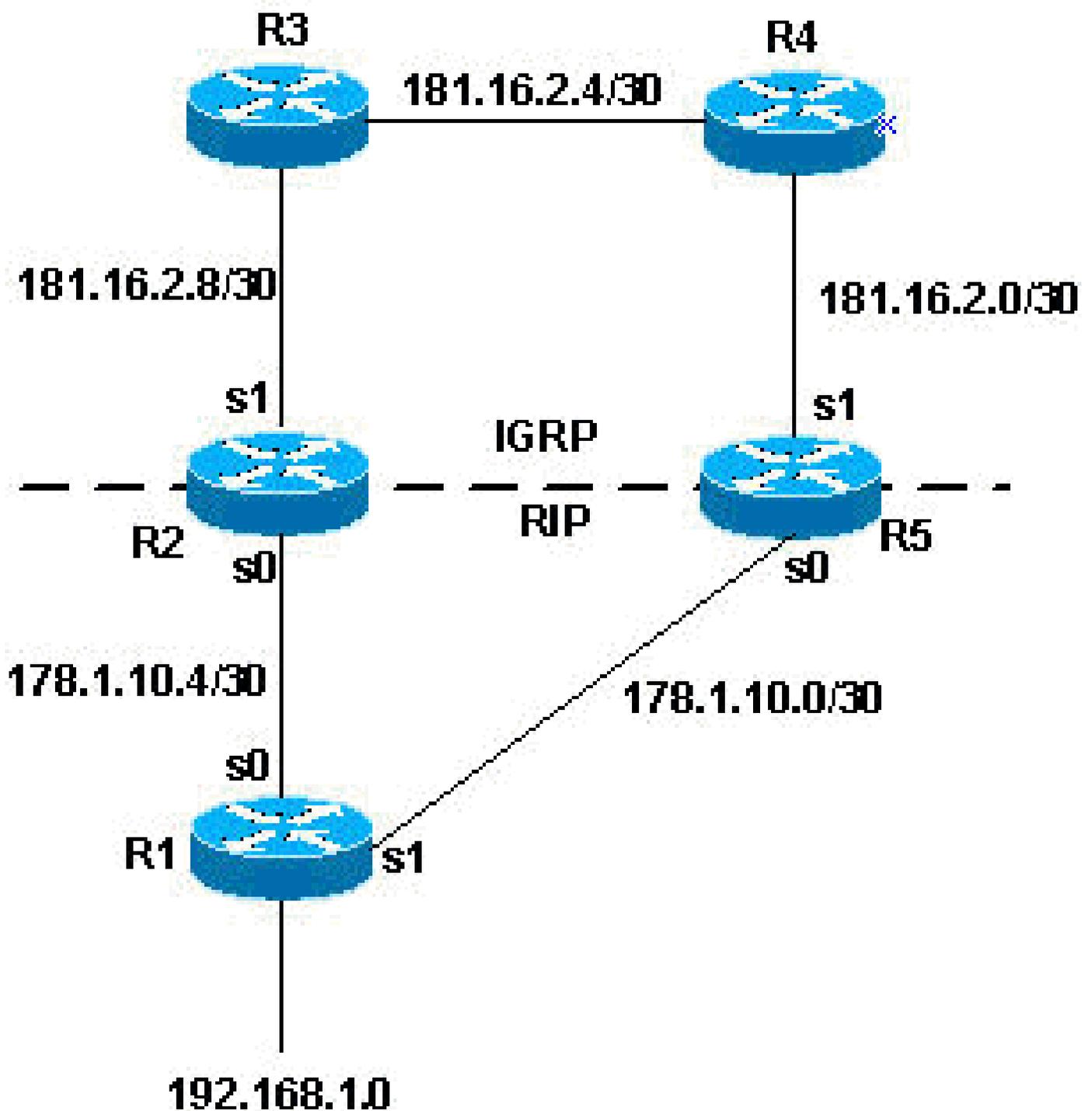
 **Note:** By default, only EIGRP-learned information is a candidate for redistribution into Interior Gateway Protocol (IGP) when the **redistribute bgp** command is issued. The Interior BGP (iBGP) routes are not redistributed into IGP until the **bgp redistribute-internal** command is configured under the **router bgp** command. But precautions must be taken in order to avoid loops within the Autonomous System when iBGP routes are redistributed into IGP.

---

## Avoid Problems Due to Redistribution

The Administrative Distance section describes how redistribution can potentially cause problems such as the next topology of optimal routing, routing loops, or slow convergence. You can avoid these problems if you never announce the information originally received from routing process X back into routing process X.

### Example 1



#### *R2 and R5 Mutual Redistribution*

In this topology example, R2 and R5 are in mutual redistribution. RIP is redistributed into EIGRP and EIGRP is redistributed to RIP, as this next configuration shows.

#### **R2**

<#root>

```
router eigrp 7
network 172.16.0.181

redistribute rip metric 1 1 1 1 1
```

```
router rip
network 172.16.0.0

redistribute eigrp 7 metric 2
```

## R5

```
<#root>
```

```
router eigrp 7
network 172.16.0.181

redistribute rip metric 1 1 1 1 1
```

```
router rip
network 172.16.0.0

redistribute eigrp 7 metric 2
```

With the previous configuration example, you have the potential for any of the problems previously described. In order to avoid them, you can filter routing updates:

## R2

```
<#root>
```

```
router eigrp 7
network 172.16.0.181

redistribute rip metric 1 1 1 1 1

distribute-list 1 in s1
```

```
router rip
network 172.16.0.0

redistribute eigrp 7 metric 2
```

```
access-list 1 deny 192.168.1.0
access-list 1 permit any
```

## R5

```
<#root>
```

```
router eigrp 7
network 172.16.0.181

redistribute rip metric 1 1 1 1 1

distribute-list 1 in s1
```

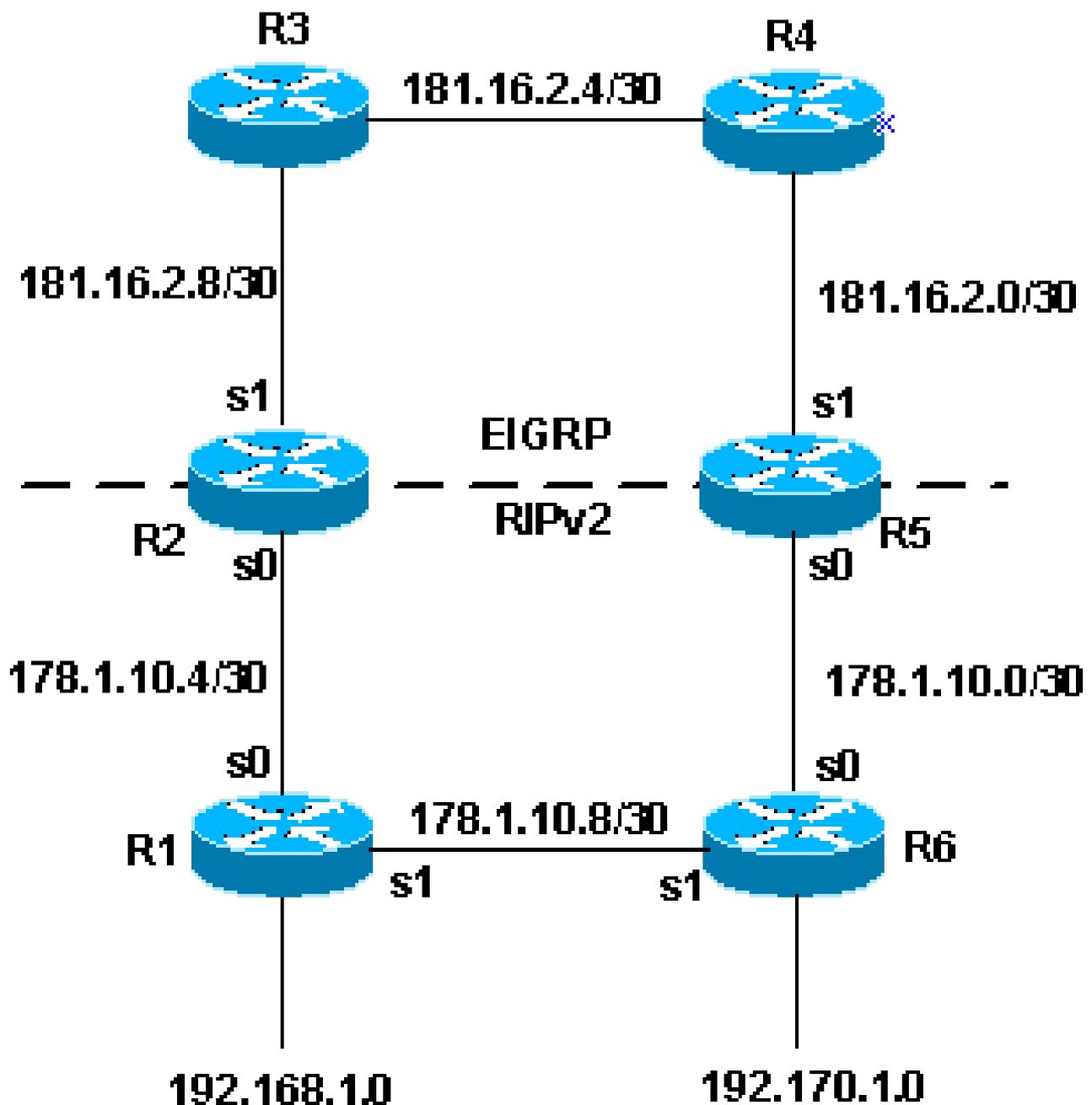
```
router rip
```

```
network 172.16.0.0
redistribute eigrp 7 metric 2
access-list 1 deny 192.168.1.0
access-list 1 permit any
```

The distribute lists added to the configurations, as shown in the previous example, filter any EIGRP updates that come into the serial 1 interface of the routers. If the routes in the updates are permitted by access list 1, the router accepts them in the update; otherwise, it does not. In this example, the routers are told that they must not learn network 192.168.1.0 through the EIGRP updates they receive on their serial 1 interface. Therefore, the only knowledge these routers have for network 192.168.1.0 is through RIP from R1.

Also keep in mind that in this case it is not necessary to use the same filter strategy for the RIP process because RIP has a higher administrative distance than EIGRP. If routes that originate in the EIGRP domain were fed back to R2 and R5 through RIP, the EIGRP routes still take precedence.

## **Example 2**



### *IGRP Take Precedence*

The topology in the previous example, shows another method to avoid redistributed problems. This method is preferable. This method uses route-maps to set tags for various routes. Routing processes can then redistribute based on the tags. Notice that redistribution based on tags does not work with RIP version 1.

One of the problems you can run into in the previous topology is:

- R1 advertises network 192.168.1.0 to R2. R2 then redistributes to EIGRP. R5 learns the network via EIGRP and redistributes it to RIPv2. Based on the metric that R5 sets for the RIPv2 route, R6 can prefer the less desirable route through R5 instead of through R1 to reach the network.

The next configuration example shows how to prevent this with setting tags and then to redistribute based on the tags.

## R2

```
router eigrp 7
 network 172.16.0.181
 redistribute rip route-map rip_to_eigrp metric 1 1 1 1 1

!--- Redistributes RIP routes that are
!--- permitted by the route-map rip_to_eigrp

router rip
 version 2
 network 172.16.0.0
 redistribute eigrp 7 route-map eigrp_to_rip metric 2

!--- Redistributes EIGRP routes and set the tags
!--- according to the eigrp_to_rip route-map route-map rip_to_eigrp deny 10 match tag 88

route-map rip_to_eigrp deny 10
 match tag 88

!--- Route-map statement to deny any routes that have a tag of "88"
!--- from being redistributed into EIGRP
!--- Notice the routes tagged with "88" must be the EIGRP
!--- routes that are redistributed into RIPv2

route-map rip_to_eigrp permit 20
 set tag 77

!--- Route-map statement to set the tag
!--- on RIPv2 routes redistributed into EIGRP to "77"

route-map eigrp_to_rip deny 10
 match tag 77

!--- Route-map statement to deny any routes that have a
!--- tag of "77" from being redistributed into RIPv2
!--- Notice the routes tagged with "77" must be the RIPv2
!--- routes that are redistributed into EIGRP

route-map eigrp_to_rip permit 20 s
 set tag 88

!--- Route-map statement to set the tag on EIGRP
!--- routes redistributed into RIPv2 to "88"
```

## R5

```
router eigrp 7
 network 172.16.0.181
 redistribute rip route-map rip_to_eigrp metric 1 1 1 1 1

!--- Redistributes RIPv2 routes that are permitted
!--- by the route-map rip_to_eigrp

router rip
```

```

version 2
network 172.16.0.0
redistribute eigrp 7 route-map eigrp_to_rip metric 2

!--- Redistributes EIGRP routes and sets the tags
!--- according to the eigrp_to_rip route-map

route-map rip_to_eigrp deny 10
  match tag 88

!--- Route-map statement to deny any routes that have a tag
!--- of "88" from being redistributed into EIGRP
!--- Notice the routes tagged with "88" must be the EIGRP routes
!--- that are redistributed into RIPv2

route-map rip_to_eigrp permit 20
  set tag 77

!--- Route-map statement to set the tag on rip routes
!--- redistributed into EIGRP to "77"

route-map eigrp_to_rip deny 10
  match tag 77

!--- Route-map statement to deny any routes that have a tag
!--- of "77" from being redistributed into RIPv2

!--- Notice the routes tagged with "77" must be the RIPv2 routes

!--- that are redistributed into EIGRP

route-map eigrp_to_rip permit 20
  set tag 88

!--- Route-map statement to set the tag on EIGRP routes
!--- redistributed into RIPv2 to "88"

```

With the previous example configuration completed, you can look at some specific routes in the routing table to see that the tags have been set. The output from the **show ip route** command for specific routes on R3 and R1 is:

```
<#root>
```

```
R3#
```

```
show ip route 172.16.10.8
```

```
Routing entry for 172.16.10.8/30
```

```
  Known via "eigrp 7", distance 170, metric 2560512256
```

```
Tag 77, type external
```

```
  Redistributing via eigrp 7
```

```
  Last update from 172.16.2.10 on Serial0, 00:07:22 ago
```

```
  Routing Descriptor Blocks:
```

```
    * 172.16.2.10, from 172.16.2.10, 00:07:22 ago, via Serial0
```

```
Route metric is 2560512256, traffic share count is 1
Total delay is 20010 microseconds, minimum bandwidth is 1 Kbit
Reliability 1/255, minimum MTU 1 bytes
Loading 1/255, Hops 1
```

R1#

```
show ip route 172.16.2.4
```

```
Routing entry for 172.16.0.181/16
  Known via "rip", distance 120, metric 2
```

Tag 88

```
Redistributing via rip
Last update from 172.16.10.50 on Serial0, 00:00:15 ago
Routing Descriptor Blocks:
* 172.16.10.50, from 172.16.10.50, 00:00:15 ago, via Serial0
  Route metric is 2, traffic share count is 1
```

EIGRP uses five different variables to calculate the metric. However, redistributed routes do not have these parameters, and this causes irregularities in the route setting. The best practice is to set a **default-metric** when you redistribute routes. By setting the default metric, the performance of EIGRP can be improved. For EIGRP, the default values are entered with this command:

<#root>

```
Router(config-router)#
```

```
default-metric 10000 100 255 100 1500
```

### Example 3

Redistribution can also take place among different processes of the same routing protocol. The next configuration is an example of a redistribution policy used to redistribute two EIGRP process that run on the same router or on multiple routers:

```
router eigrp 3
 redistribute eigrp 5 route-map to_eigrp_3
 default-metric 10000 100 255 1 1500

!--- Redistributes EIGRP 5 into EIGRP 3, setting the tags
!--- according to the route map "to_eigrp_3"

router eigrp 5
 redistribute eigrp 3 route-map to_eigrp_5
 default-metric 10000 100 255 1 1500

!--- Redistributes EIGRP 3 into EIGRP 5
!--- Routes with tag 33 can not be redistributed
!--- due to route map "to_eigrp_5"
```

*!--- Though the default-metric command is not required  
!--- when redistributing between different EIGRP processes,  
!--- you can use it optionally as shown in the previous example to advertise  
!--- the routes with specific values for calculating the metric.*

```
route-map to_eigrp_3 deny 10  
  match tag 55
```

*!--- Route-map statement used to deny any routes that have a tag  
!--- of "55" from being redistributed into EIGRP 3  
!--- Notice the routes tagged with "55" must be the EIGRP 3 routes  
!--- that are redistributed into EIGRP 5*

```
route-map to_eigrp_3 permit 20  
  set tag 33
```

*!--- Route-map statement used to set the tag on routes  
!--- redistributed from EIGRP 5 to EIGRP 3 to "33"*

```
route-map to_eigrp_5 deny 10  
  match tag 33
```

*!--- Route-map statement used to deny any routes that have a tag  
!--- of "33" from being redistributed into EIGRP 5  
!--- Notice the routes tagged with "33" must be the EIGRP 5 routes  
!--- that are redistributed into EIGRP 3*

```
route-map to_eigrp_5 permit 20  
  set tag 55
```

*!--- Route-map statement used to set the tag on routes  
!--- redistributed from EIGRP 3 to EIGRP 5 to "55"*

This document provides several strategies to filter routes. However, there can be other valid strategies you can use.

## **Example 4**

In example 4, you have two routers, one is a high-end router that runs the BGP protocol, and the other one is low-end router that runs the RIP protocol. When you redistribute BGP routes into RIP, you can lose some packets.

The redistribution of BGP into RIP protocol is generally not recommended and protocols like iBGP, OSPF, and EIGRP are scalable and have wide options available.

In case you encounter this scenario, which is the redistribution between BGP to RIP, and lose some packets, it possible that you have to configure this command on the RIP process:

```
<#root>
```

```
Router(Config)#
```

```
router rip
```

```
Router(Config-router)#
```

```
input-queue 1024
```

---

 **Note:** Consider the use of the input-queue command if you have a high-end router that sends at high speed to a low-speed router that cannot receive at the high speeds. The configuration of this command helps prevent the loss of information from the routing table.

---

## Example 5



### *Redistribute Static Route*

This example illustrates how to redistribute a static route into RIP routing protocol. As per the topology example, we have three routers (R1, R2, and R3). R1 and R2 have RIP configured on interface Fast Ethernet 0/0. R1 has a static route to reach the Lo 0 interface (ip address 10.10.10.10/32) of Router R3. This static route is redistributed in the RIP routing protocol. Router R3 is configured with a default route R3# ip route 0.0.0.0 0.0.0.0 FastEthernet 0/0.

```
<#root>
```

```
R1(config)#
```

```
ip route 10.10.10.10 255.255.255.255 10.13.13.3
```

```
R1(config)#
```

```
router rip
```

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

```
redistribute static metric 10
```

On Router R2, route 10.10.10.10 is displayed via the show ip route command:

```
<#root>
```

```
R2#
```

```
show ip route
```

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP  
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area  
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2  
E1 - OSPF external type 1, E2 - OSPF external type 2  
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, 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

```
C    192.168.12.12/24 is directly connected, FastEthernet0/0
    10.0.0.3/32 is subnetted, 1 subnets
R      10.10.10.10 [120/10] via 192.168.12.1, 00:00:07, FastEthernet0/0
```

## How to Redistribute Single Static Route

To redistribute a single static route, use route-map to select the static route that needs to be redistributed.

```
<#root>
```

```
Router(config)#
access-list 1 permit <network no> <mask>
Router(config)#
route-map <route-map name> permit 10
Router(config-route-map)#
match ip address access list number
Router(config)#
router eigrp <As number>
Router(config-router)#
redistribute static route-map <map-name> metric <value>
```

## Related Information

- [RIP and OSPF Redistribution](#)
- [Understand and Use the Enhanced Interior Gateway Routing Protocol](#)
- [Redistributing Between Classful and Classless Protocols: EIGRP or OSPF into RIP or IGRP](#)
- [BGP Case Studies](#)
- [IP Routing Support Page](#)
- [Technical Support & Documentation - Cisco Systems](#)