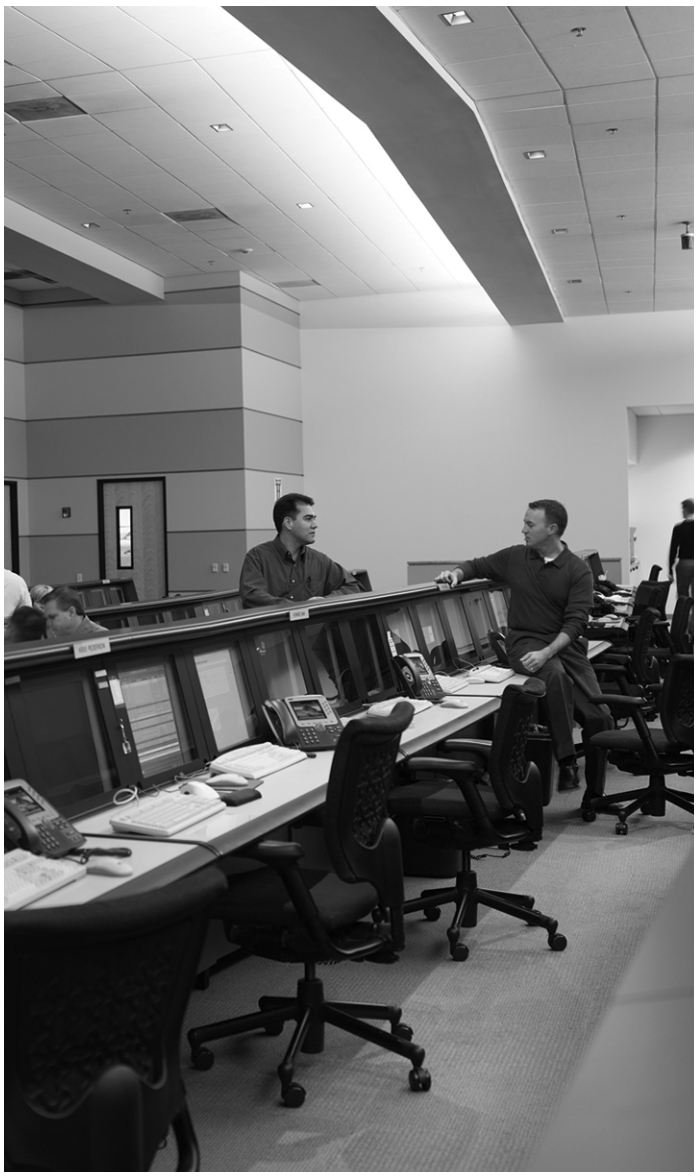


Blueprint topics covered in this chapter:

This chapter covers the following topics from the Cisco CCIE Routing and Switching written exam blueprint:

- IP Routing
  - OSPF
  - EIGRP
  - Route Filtering
  - RIPv2
  - The use of **show** and **debug** commands

CHAPTER 11

IGP Route Redistribution, Route Summarization, and Default Routing

This chapter covers several topics related to the use of multiple IGP routing protocols. IGPs can use default routes to pull packets toward a small set of routers, with those routers having learned routes from some external source. IGPs can use route summarization with a single routing protocol, but it is often used at redistribution points between IGPs as well. Finally, route redistribution by definition involves moving routes from one routing source to another. This chapter takes a look at each topic.

"Do I Know This Already?" Quiz

Table 11-1 outlines the major headings in this chapter and the corresponding "Do I Know This Already?" quiz questions.

Table 11-1 *"Do I Know This Already?" Foundation Topics Section-to-Question Mapping*

Foundation Topics Section	Questions Covered in This Section	Score
Route Maps, Prefix Lists, and Administrative Distance	1–2	
Route Redistribution	3–6	
Route Summarization	7–8	
Default Routes	9	
Total Score		

In order to best use this pre-chapter assessment, remember to score yourself strictly. You can find the answers in Appendix A, "Answers to the 'Do I Know This Already?' Quizzes."

1. A route map has several clauses. A route map's first clause has a **permit** action configured. The **match** command for this clause refers to an ACL that matches route 10.1.1.0/24 with a **permit** action, and matches route 10.1.2.0/24 with a **deny** action. If this route map is used for route redistribution, which of the following are true?
   a. The route map will attempt to redistribute 10.1.1.0/24.
   b. The question does not supply enough information to determine if 10.1.1.0/24 is redistributed.
   c. The route map will not attempt to redistribute 10.1.2.0/24.
   d. The question does not supply enough information to determine if 10.1.2.0/24 is redistributed.

2. Which of the following routes would be matched by this prefix list command: **ip prefix-list fred permit 10.128.0.0/9 ge 20**?
   a. 10.1.1.0 255.255.255.0
   b. 10.127.1.0 255.255.255.0
   c. 10.200.200.192 255.255.255.252
   d. 10.128.0.0 255.255.240.0
   e. None of these answers is correct.

3. A router is using the configuration shown below to redistribute routes. This router has several working interfaces with IP addresses in network 10.0.0.0, and has learned some network 10 routes with EIGRP and some with OSPF. Which of the following is true about the redistribution configuration?

```
router eigrp 1
 network 10.0.0.0
 redistribute ospf 2
!
router ospf 2
 network 10.0.0.0 0.255.255.255 area 3
 redistribute eigrp 1 subnets

R1# show ip route 15.0.0.0
Routing entry for 15.0.0.0/24, 5 known subnets
  Attached (2 connections)
  Redistributing via eigrp 1

O E1    10.6.11.0 [110/84] via 10.1.6.6, 00:21:52, Serial0/0/0.6
O E2    10.6.12.0 [110/20] via 10.1.6.6, 00:21:52, Serial0/0/0.6
C       10.1.6.0 is directly connected, Serial0/0/0.6
O IA    10.1.2.0 [110/65] via 10.1.1.5, 00:21:52, Serial0/0/0.5
C       10.1.1.0 is directly connected, Serial0/0/0.5
```

   a. EIGRP will not advertise any additional routes due to redistribution.
   b. OSPF will not advertise any additional routes due to redistribution.

c. Routes redistributed into OSPF will be advertised as E1 routes.

d. The **redistribute ospf 2** command would be rejected due to missing parameters.

4. Examine the following router configuration and excerpt from its IP routing table. Which routes could be redistributed into OSPF?

```
router eigrp 1
 network 12.0.0.0
router ospf 2
 redistribute eigrp 1 subnets
 network 13.0.0.0 0.255.255.255 area 3
An excerpt from the routing table is shown next:
C       12.1.6.0 is directly connected, Serial0/0/0.6
D       12.0.0.0/8 [90/2172416] via 13.1.1.1, 00:01:30, Serial0/0/0.5
C       13.1.1.0 is directly connected, Serial0/0/0.5
```

a. 12.1.6.0

b. 12.0.0.0

c. 13.1.1.0

d. None of the above

5. Two corporations merged. The network engineers decided to redistribute between one company's EIGRP network and the other company's OSPF network, using two mutually redistributing routers (R1 and R2) for redundancy. Assume that as many defaults as is possible are used for the redistribution configuration. Assume that one of the subnets in the OSPF domain is 10.1.1.0/24. Which of the following is true about a possible suboptimal route to 10.1.1.0/24 on R1—a route that sends packets through the EIGRP domain, and through R2 into the OSPF domain?

a. The suboptimal routes will occur unless the configuration filters routes at R1.

b. R1's administrative distance must be manipulated, such that OSPF routes have an administrative distance less than EIGRP's default of 90.

c. EIGRP prevents the suboptimal routes by default.

d. Using route tags is the only way to prevent the suboptimal routes.

6. Which of the following statements is true about the type of routes created when redistributing routes?

a. Routes redistributed into OSPF default to be external type 2.

b. Routes redistributed into EIGRP default to external, but can be set to internal with a route map.

c. Routes redistributed into RIP are external by default.

d. Routes redistributed into OSPF by a router in an NSSA area default to be external type 1.

7. Which of the following is not true about route summarization?

- **a.** The advertised summary is assigned the same metric as the lowest-metric component subnet.
- **b.** The router does not advertise the summary when its routing table does not have any of the component subnets.
- **c.** The router does not advertise the component subnets.
- **d.** Summarization, when used with redistribution, prevents all cases of suboptimal routes.

8. Which of the following is true of route summarization?

- **a.** OSPF can summarize routes only on ABRs and ASBRs.
- **b.** EIGRP summarization is configured with a **router eigrp** subcommand.
- **c.** RIPv2 summarization has the same features as EIGRP summarization, other than the syntax differences in the **ip summary-address** command.
- **d.** RIPv1 allows the **ip summary-address** command to be used.

9. Which of the following is/are true regarding the **default-information originate** router subcommand?

- **a.** It is not supported by EIGRP.
- **b.** It causes OSPF to advertise a default route, but only if a static route to 0.0.0.0/0 is in that router's routing table.
- **c.** The **always** keyword on the **default-information originate** command, when used for OSPF, means OSPF will originate a default route even if no default route exists in its own IP routing table.
- **d.** None of the other answers are correct.

Foundation Topics

Route Maps, Prefix Lists, and Administrative Distance

Route maps, IP prefix lists, and administrative distance (AD) must be well understood to do well with route redistribution topics on the CCIE Routing and Switching written exam. This section focuses on the tools themselves, followed by coverage of route redistribution.

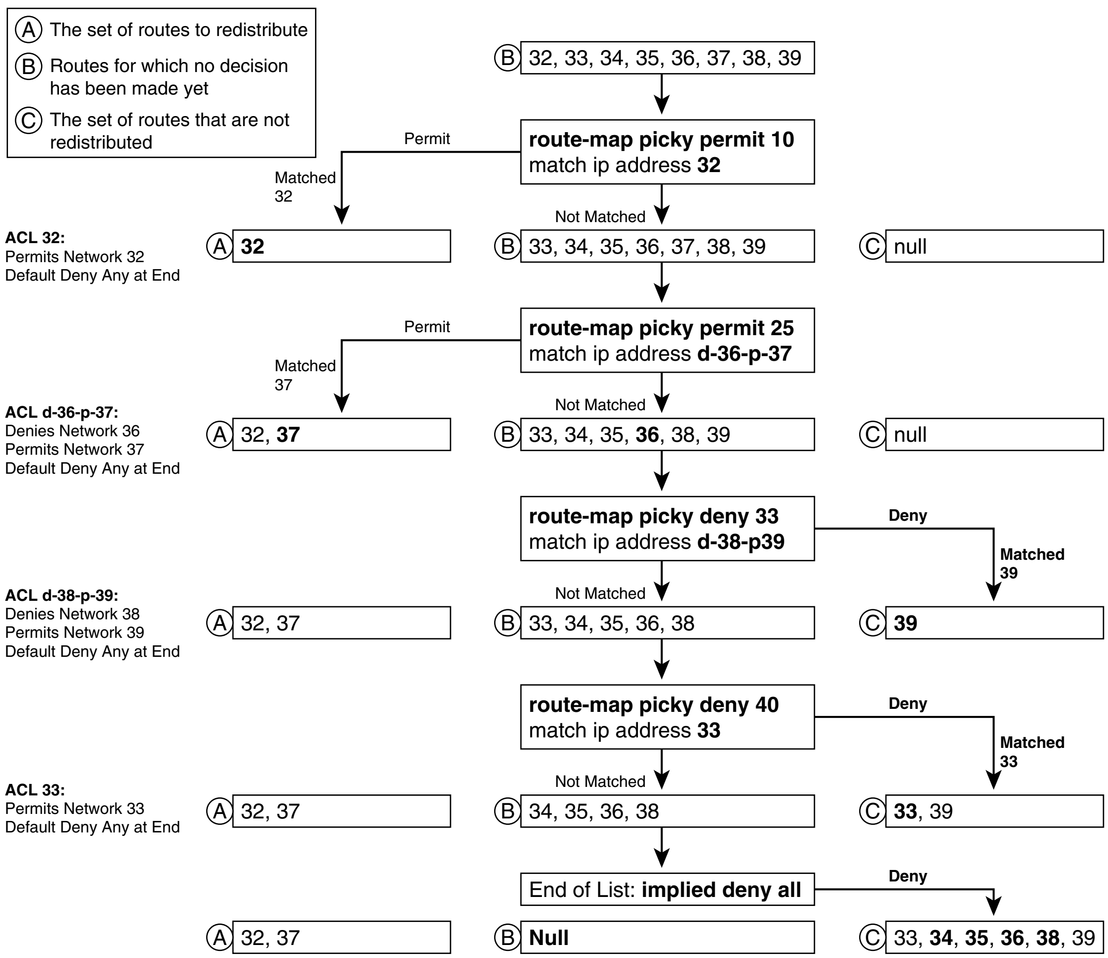
Configuring Route Maps with the route-map Command

Route maps provide programming logic similar to the If/Then/Else logic seen in other programming languages. A single route map has one or more **route-map** commands in it, and routers process **route-map** commands in sequential order based on sequence numbers. Each **route-map** command has underlying matching parameters, configured with the aptly named **match** command. (To match all packets, the **route-map** clause simply omits the **match** command.) Each **route-map** command also has one or more optional **set** commands that you can use to manipulate information—for instance, to set the metric for some redistributed routes. The general rules for route maps are as follows:

- Each **route-map** command must have an explicitly configured name, with all commands that use the same name being part of the same route map.
- Each **route-map** command has an action (**permit** or **deny**).
- Each **route-map** command in the same route map has a unique sequence number, allowing deletion and insertion of single **route-map** commands.
- When a route map is used for redistribution, the route map processes routes taken from the then-current routing table.
- The route map is processed sequentially based on the sequence numbers.
- Once a particular route is matched by the route map, it is not processed beyond that matching **route-map** command (specific to route redistribution).
- When a route is matched in a **route-map** statement, if the **route-map** command has a **permit** parameter, the route is redistributed (specific to route redistribution).
- When a route is matched in a **route-map** statement, if the **route-map** statement has a **deny** parameter, the route is not redistributed (specific to route redistribution).

Route maps can be confusing at times, especially when using the **deny** option on the **route-map** command. To help make sure the logic is clear before getting into redistribution, Figure 11-1 shows a logic diagram for an example route map. (This example is contrived to demonstrate some nuances of route map logic; a better, more efficient route map could be created to achieve the same results.) In the figure, R1 has eight loopback interfaces configured to be in class A networks 32 through 39. Figure 11-1 shows how the contrived **route-map picky** would process the routes.

Figure 11-1 *Route Map Logic Example*



First, a few clarifications about the meaning of Figure 11-1 are in order. The top of the figure begins with the set of connected networks (32 through 39), labeled with a "B," which is the set of routes still being considered for redistribution. Moving down the figure, four separate **route-map** commands sit inside this single route map. Each **route-map** clause (the clause includes the underlying **match** and **set** commands) in turn moves routes from the list of possible routes ("B") to either the list of routes to redistribute ("A") or the list to not redistribute ("C"). By the bottom of the figure, all routes will be noted as either to be redistributed or not to be redistributed.

The route map chooses to redistribute a route only if the **route-map** command has a **permit** option; the only time a **route-map** clause chooses to *not redistribute* a route is when the clause has a **deny** option. Ignoring the matching logic for a moment, the first two **route-map** commands (sequence numbers 10 and 25) use the **permit** option. As a result of those clauses, routes are either added to the list of routes to redistribute ("A") or left in the list of candidate routes ("B"). The third and fourth clauses (sequence numbers 33 and 40) use the **deny** option, so those clauses cause routes to be either added to the list of routes to not redistribute ("C"), or left in the list of candidate routes ("B"). In effect, once a **route-map** clause has matched a route, that route is flagged either as to be redistributed or as not to be redistributed, and the route is no longer processed by the route map.

One point that can sometimes be confused is that if a route is denied by an ACL used by a **match** command, it does not mean that the route is prevented from being redistributed. For instance, the **match ip address 32** command in clause 10 refers to ACL 32, which has one explicit *access control entry (ACE)* that matches network 32, with a **permit** action. Of course, ACL 32 has an implied deny all at the end, so ACL 32 permits network 32, and denies 33 through 39. However, denying networks 33 through 39 in the ACL does not mean that those routes are not redistributed—it simply means that those routes do not match **route-map** clause 10, so those routes are eligible for consideration by a later **route-map** clause.

The following list summarizes the key points about route map logic when used for redistribution:

KEY POINT

- **route-map** commands with the **permit** option either cause a route to be redistributed or leave the route in the list of routes to be examined by the next **route-map** clause.
- **route-map** commands with the **deny** option either filter the route or leave the route in the list of routes to be examined by the next **route-map** clause.
- If a clause's **match** commands use an ACL, an ACL match with the **deny** action does not cause the route to be filtered. Instead, it just means that route does not match that particular **route-map** clause.
- The **route-map** command includes an implied deny all clause at the end; to configure a permit all, use the **route-map** command, with a **permit** action, but without a **match** command.

Route Map match Commands for Route Redistribution

Route maps use the **match** command to define the fields and values used for matching the routes being processed. If more than one **match** command is configured in a single **route-map** clause, a route is matched only if all the **match** commands' parameters match the route. The logic in each

match command itself is relatively straightforward. Table 11-2 lists the **match** command options when used for IGP route redistribution.

Table 11-2 **match** *Command Options for IGP Redistribution*

match Command	Description
match interface *interface-type interface-number* [... *interface-type interface-number*]	Looks at outgoing interface of routes
***match ip address** {[*access-list-number* | *access-list-name*] | *prefix-list prefix-list-name*}	Examines route prefix and prefix length
***match ip next-hop** {*access-list-number* | *access-list-name*}	Examines route's next-hop address
***match ip route-source** {*access-list-number* | *access-list-name*}	Matches advertising router's IP address
match metric *metric-value*	Matches route's metric
match route-type {**internal** | **external** [**type-1** | **type-2**] | **level-1** | **level-2**}	Matches route type
match tag *tag-value* [...*tag-value*]	Tag must have been set earlier

*Can reference multiple numbered and named ACLs on a single command.

Route Map set Commands for Route Redistribution

When used for redistribution, route maps have an implied action—either to allow the route to be redistributed or to filter the route so that it is not redistributed. As described earlier in this chapter, that choice is implied by the **permit** or **deny** option on the **route-map** command. Route maps can also change information about the redistributed routes by using the **set** command. Table 11-3 lists the **set** command options when used for IGP route redistribution.

Table 11-3 **set** *Command Options for IGP Redistribution*

set Command	Description
set level {**level-1** | **level-2** | **level-1-2** | **stub-area** | **backbone**}	Defines database(s) into which the route is redistributed
set metric *metric-value*	Sets the route's metric for OSPF, RIP, and IS-IS
set metric *bandwidth delay reliability loading mtu*	Sets the IGRP/EIGRP route's metric values
set metric-type {**internal** | **external** | **type-1** | **type-2**}	Sets type of route for IS-IS and OSPF
set tag *tag-value*	Sets the unitless tag value in the route

IP Prefix Lists

IP prefix lists provide mechanisms to match two components of an IP route:

- The route prefix (the subnet number)
- The prefix length (the subnet mask)

The **redistribute** command cannot directly reference a prefix list, but a route map can refer to a prefix list by using the **match** command.

A prefix list itself has similar characteristics to a route map. The list consists of one or more statements with the same text name. Each statement has a sequence number to allow deletion of individual commands, and insertion of commands into a particular sequence position. Each command has a **permit** or **deny** action—but because it is used only for matching packets, the **permit** or **deny** keyword just implies whether a route is matched (**permit**) or not (**deny**). The generic command syntax is as follows:

```
ip prefix-list list-name [seq seq-value] {deny network/length | permit network/
length}[ge ge-value] [le le-value]
```

The sometimes tricky and interesting part of working with prefix lists is that the meaning of the *network/length*, *ge-value*, and *le-value* parameters changes depending on the syntax. The *network/length* parameters define the values to use to match the route prefix. For example, a *network/length* of 10.0.0.0/8 means "any route that begins with a 10 in the first octet." The **ge** and **le** options are used for comparison to the prefix length—in other words, to the number of binary 1s in the subnet mask. For instance, **ge 20 le 22** matches only routes whose masks are /20, /21, or /22. So, prefix list logic can be summarized into a two-step comparison process for each route:

1. The *route's prefix* must be within the range of addresses implied by the **prefix-list** command's *network/length* parameters.
2. The *route's prefix length* must match the *range of prefixes* implied by the **prefix-list** command.

The potentially tricky part of the logic relates to knowing the range of prefix lengths checked by this logic. The range is defined by the *ge-value* and *le-value* parameters, which stand for *greater-than-or-equal-to* and *less-than-or-equal-to*. Table 11-4 formalizes the logic, including the default values for *ge-value* and *le-value*. In the table, note that *conf-length* refers to the prefix length configured in the *network/prefix* (required) parameter, and *route-length* refers to the prefix length of a route being examined by the prefix list.

Table 11-4 *LE and GE Parameters on IP Prefix List, and the Implied Range of Prefix Lengths*

Prefix List Parameters	Range of Prefix Lengths
Neither	*conf-length* = *route-length*
Only **le**	*conf-length* <= *route-length* <= *le-value*
Only **ge**	*ge-value* <= *route-length* <= 32
Both **ge** and **le**	*ge-value* <= *route-length* <= *le-value*

Several examples can really help nail down prefix list logic. The following routes will be examined by a variety of prefix lists, with the routes numbered for easier reference:

1. 10.0.0.0/8
2. 10.128.0.0/9
3. 10.1.1.0/24
4. 10.1.2.0/24
5. 10.128.10.4/30
6. 10.128.10.8/30

Next, Table 11-5 shows the results of seven different one-line prefix lists applied to these six example routes. The table lists the matching parameters in the **prefix-list** commands, omitting the first part of the commands. The table explains which of the six routes would match the listed prefix list, and why.

Table 11-5 *Example Prefix Lists Applied to the List of Routes*

prefix-list Command Parameters	Routes Matched	Results
10.0.0.0/8	1	Without **ge** or **le** configured, both the prefix (10.0.0.0) and length (8) must be an exact match.
10.128.0.0/9	None	Without **ge** or **le** configured, the prefix (10.128.0.0) and length (9) must be an exact match, so none of the routes match.
10.0.0.0/8 ge 9	2–6	The 10.0.0.0/8 means "all routes whose first octet is 10," effectively representing an address range. The prefix length must be between 9 and 32, inclusive.
10.0.0.0/8 ge 24 le 24	3, 4	The 10.0.0.0/8 means "all routes whose first octet is 10," and the prefix range is 24 to 24—meaning only routes with prefix length 24.

Table 11-5 *Example Prefix Lists Applied to the List of Routes (Continued)*

prefix-list Command Parameters	Routes Matched	Results
10.0.0.0/8 le 28	1–4	The prefix length needs to be between 8 and 28, inclusive.
0.0.0.0/0	None	0.0.0.0/0 means "match all prefixes, with prefix length of exactly 0." So, it would match all routes' prefixes, but none of their prefix lengths. Only a default route would match this prefix list.
0.0.0.0/0 le 32	All	The range implied by 0.0.0.0/0 is all IPv4 addresses. The **le 32** then implies any prefix length between 0 and 32, inclusive. This is the syntax for "match all" prefix list logic.

Administrative Distance

A single router can learn routes using multiple IP routing protocols, as well as via connected and static routes. When a router learns a particular route from multiple sources, the router cannot use the metrics to determine the best route, because the metrics are based on different units. So, the router uses each route's *administrative distance (AD)* to determine which is best, with the lower number being better. Table 11-6 lists the default AD values for the various routing sources.

Table 11-6 *Administrative Distances*

KEY POINT

Route Type	Administrative Distance
Connected	0
Static	1
EIGRP summary route	5
EBGP	20
EIGRP (internal)	90
IGRP	100
OSPF	110
IS-IS	115
RIP	120
EIGRP (external)	170
iBGP (external)	200
Unreachable	255

The defaults can be changed by using the **distance** command. The command differs amongst all three IGPs covered in this book. The generic versions of the **distance** router subcommand for RIP, EIGRP, and OSPF, respectively, are as follows:

```
distance distance
distance eigrp internal-distance external-distance
distance ospf {[intra-area dist1] [inter-area dist2] [external dist3]}
```

As you can see, EIGRP and OSPF can set a different AD depending on the type of route as well, whereas RIP cannot. You can also use the **distance** command to set a router's view of the AD per route, as is covered later in this chapter.

Route Redistribution

Although using a single routing protocol throughout an enterprise might be preferred, many enterprises use multiple routing protocols due to business mergers and acquisitions, organizational history, or in some cases for technical reasons. Route redistribution allows one or more routers to take routes learned via one routing protocol and advertise those routes via another routing protocol so that all parts of the internetwork can be reached.

To perform redistribution, one or more routers run both routing protocols, with each routing protocol placing routes into that router's routing table. Then, each routing protocol can take all or some of the other routing protocol's routes from the routing table and advertise those routes. This section begins by looking at the mechanics of how to perform simple redistribution on a single router, and ends with discussion of tools and issues that matter most when redistributing on multiple routers.

The Mechanics of the redistribute Command

The **redistribute** router subcommand tells one routing protocol to take routes from another routing protocol. This command can simply redistribute all routes or, by using matching logic, redistribute only a subset of the routes. The **redistribute** command also supports actions for setting some parameters about the redistributed routes—for example, the metric.

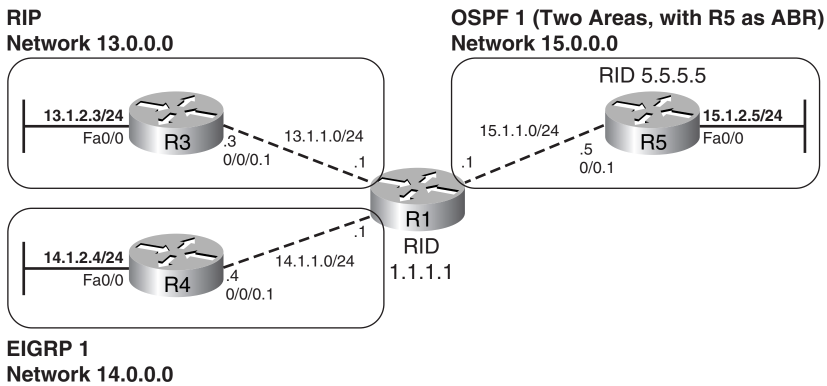
The full syntax of the **redistribute** command is as follows:

```
redistribute protocol [process-id] [level-1 | level-1-2 | level-2] [as-number] [metric
metric-value] [metric-type type-value] [match {internal | external 1 | external 2}] [tag
tag-value] [route-map map-tag] [subnets]
```

The **redistribute** command identifies the routing source from which routes are taken, and the **router** command identifies the routing process into which the routes are advertised. For example, the command **redistribute eigrp 1** tells the router to *take routes from* EIGRP process 1; if that command were under **router rip**, the routes would be redistributed into RIP, enabling other RIP routers in the network to see some or all routes coming from EIGRP AS 1.

The **redistribute** command has a lot of other parameters as well, most of which will be described in upcoming examples. The first few examples use the network shown in Figure 11-2. In this network, each IGP uses a different class A network just to make the results of redistribution more obvious. Also note that the numbering convention is such that each of R1's connected WAN subnets has 1 as the third octet, and each LAN subnet off R3, R4, and R5 has 2 as the third octet.

Figure 11-2 *Sample Network for Default Route Examples*



Redistribution Using Default Settings

The first example configuration meets the following design goals:

- R1 redistributes between each pair of IGPs—RIP, EIGRP, and OSPF.
- Default metrics are used whenever possible; when required, the metrics are configured on the **redistribute** command.
- Redistribution into OSPF uses the non-default **subnets** parameter, which causes subnets to be advertised into OSPF.
- All other settings use default values.

Example 11-1 shows R1's configuration for each routing protocol, along with **show** commands from all four routers to highlight the results of the redistribution.

Example 11-1 *Route Redistribution with Minimal Options*

```
! EIGRP redistributes from OSPF (process ID 1) and RIP. EIGRP must
! set the metric, as it has no default values. It also uses the
! no auto-summary command so that subnets will be redistributed into
! EIGRP.
```

continues

Example 11-1 *Route Redistribution with Minimal Options (Continued)*

```
router eigrp 1
 redistribute ospf 1 metric 1544 5 255 1 1500
 redistribute rip metric 1544 5 255 1 1500
 network 14.0.0.0
 no auto-summary
! OSPF redistributes from EIGRP (ASN 1) and RIP. OSPF defaults the
! metric to 20 for redistributed IGP routes. It must also use the
! subnets option in order to redistribute subnets.
router ospf 1
 router-id 1.1.1.1
 redistribute eigrp 1 subnets
 redistribute rip subnets
 network 15.0.0.0 0.255.255.255 area 0
! RIP redistributes from OSPF (process ID 1) and EIGRP (ASN 1). RIP
! must set the metric, as it has no default values. It also uses the
! no auto-summary command so that subnets will be redistributed into
! EIGRP.
router rip
 version 2
 redistribute eigrp 1 metric 2
 redistribute ospf 1 metric 3
 network 13.0.0.0
 no auto-summary
! R1 has a connected route (x.x.1.0) in networks 13, 14, and 15, as well as
! an IGP-learned route (x.x.2.0).
R1# show ip route
! lines omitted for brevity
     10.0.0.0/24 is subnetted, 1 subnets
C       10.1.1.0 is directly connected, FastEthernet0/0
     13.0.0.0/24 is subnetted, 2 subnets
C       13.1.1.0 is directly connected, Serial0/0/0.3
R       13.1.2.0 [120/1] via 13.1.1.3, 00:00:07, Serial0/0/0.3
     14.0.0.0/24 is subnetted, 2 subnets
D       14.1.2.0 [90/2172416] via 14.1.1.4, 00:58:20, Serial0/0/0.4
C       14.1.1.0 is directly connected, Serial0/0/0.4
     15.0.0.0/24 is subnetted, 2 subnets
O IA    15.1.2.0 [110/65] via 15.1.1.5, 00:04:25, Serial0/0/0.5
C       15.1.1.0 is directly connected, Serial0/0/0.5
! R3 learned two routes each from nets 14 and 15.
! Compare the metrics set on R1's RIP redistribute command to the metrics below.
R3# show ip route rip
     14.0.0.0/24 is subnetted, 2 subnets
R       14.1.2.0 [120/2] via 13.1.1.1, 00:00:19, Serial0/0/0.1
R       14.1.1.0 [120/2] via 13.1.1.1, 00:00:19, Serial0/0/0.1
     15.0.0.0/24 is subnetted, 2 subnets
R       15.1.2.0 [120/3] via 13.1.1.1, 00:00:19, Serial0/0/0.1
R       15.1.1.0 [120/3] via 13.1.1.1, 00:00:19, Serial0/0/0.1
```

Example 11-1 *Route Redistribution with Minimal Options (Continued)*

```
! R4 learned two routes each from nets 13 and 15.
! EIGRP injected the routes as external (EX), which are considered AD 170.
R4# show ip route eigrp
     13.0.0.0/24 is subnetted, 2 subnets
D EX    13.1.1.0 [170/2171136] via 14.1.1.1, 00:09:57, Serial0/0/0.1
D EX    13.1.2.0 [170/2171136] via 14.1.1.1, 00:09:57, Serial0/0/0.1
     15.0.0.0/24 is subnetted, 2 subnets
D EX    15.1.2.0 [170/2171136] via 14.1.1.1, 01:00:27, Serial0/0/0.1
D EX    15.1.1.0 [170/2171136] via 14.1.1.1, 01:00:27, Serial0/0/0.1
```

```
! R5 learned two routes each from nets 13 and 14.
! OSPF by default injected the routes as external type 2, cost 20.
R5# show ip route ospf
     13.0.0.0/24 is subnetted, 2 subnets
O E2    13.1.1.0 [110/20] via 15.1.1.1, 00:36:12, Serial0/0.1
O E2    13.1.2.0 [110/20] via 15.1.1.1, 00:36:12, Serial0/0.1
     14.0.0.0/24 is subnetted, 2 subnets
O E2    14.1.2.0 [110/20] via 15.1.1.1, 00:29:56, Serial0/0.1
O E2    14.1.1.0 [110/20] via 15.1.1.1, 00:36:12, Serial0/0.1
! As a backbone router, OSPF on R1 created type 5 LSAs for the four E2 subnets.
! If R1 had been inside an NSSA stub area, it would have created type 7 LSAs.
R5# show ip ospf data ¦ begin Type-5
        Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
13.1.1.0        1.1.1.1         1444        0x80000002 0x000785 0
13.1.2.0        1.1.1.1         1444        0x80000002 0x00FB8F 0
14.1.1.0        1.1.1.1         1444        0x80000002 0x00F991 0
14.1.2.0        1.1.1.1         1444        0x80000002 0x00EE9B 0
```

Metrics must be set via configuration when redistributing into RIP and EIGRP, whereas OSPF uses default values. In the example, the two **redistribute** commands under **router rip** used hop counts of 2 and 3 just so the metrics could be easily seen in the **show ip route** command output on R3. The EIGRP metric in the **redistribute** command must include all five metric components, even if the last three are ignored by EIGRP's metric calculation (as they are by default). The command **redistribute rip metric 1544 5 255 1 1500** lists EIGRP metric components of bandwidth, delay, reliability, load, and MTU, in order. OSPF defaults to cost 20 when redistributing from an IGP, and 1 when redistributing from BGP.

The **redistribute** command redistributes only routes in that router's current IP routing table. When redistributing from a given routing protocol, the **redistribute** command takes routes listed in the IP routing table as being learned from that routing protocol. Interestingly, the **redistribute** command can also pick up connected routes. For example, R1 has an OSPF route to 15.1.2.0/24, and a connected route to 15.1.1.0/24. However, R3 (RIP) and R4 (EIGRP) redistribute both of these routes—the OSPF-learned route and one connected route—as a result of their respective

redistribute ospf commands. As it turns out, the **redistribute** command causes the router to use the following logic to choose which routes to redistribute from a particular IGP protocol:

KEY POINT

1. Take all routes in my routing table that were learned by the routing protocol from which routes are being redistributed.
2. Take all connected subnets matched by that routing protocol's **network** commands.

Example 11-1 shows several instances of exactly how this two-part logic works. For instance, R3 (RIP) learns about connected subnet 14.1.1.0/24, because RIP redistributes from EIGRP, and R1's EIGRP **network 14.0.0.0** command matches that subnet.

The **redistribute** command includes a **subnets** option, but only OSPF needs to use it. By default, when redistributing into OSPF, OSPF redistributes only routes for classful networks, ignoring subnets. By including the **subnets** option, OSPF redistributes subnets as well. The other IGPs redistribute subnets automatically; however, if at a network boundary, the RIP or EIGRP **auto-summary** setting would still cause summarization to use the classful network. In Example 11-1, if either RIP or EIGRP had used **auto-summary**, each redistributed network would show just the classful networks. For example, if RIP had configured **auto-summary** in Example 11-1, R3 would have a route to networks 14.0.0.0/8 and 15.0.0.0/8, but no routes to subnets inside those class A networks.

Setting Metrics, Metric Types, and Tags

Cisco IOS provides three mechanisms for setting the metrics of redistributed routes, as follows:

KEY POINT

1. Call a route map from the **redistribute** command, with the route map using the **set metric** command. This method allows different metrics for different routes.
2. Use the **metric** option on the **redistribute** command. This sets the same metric for all routes redistributed by that **redistribute** command.
3. Use the **default-metric** command under the **router** command. This command sets the metric for all redistributed routes whose metric was not set by either of the other two methods.

The list implies the order of precedence if more than one method defines a metric. For instance, if a route's metric is set by all three methods, the route map's metric is used. If the metric is set on the **redistribute** command and there is a **default-metric** command as well, the setting on the **redistribute** command takes precedence.

The **redistribute** command also allows a setting for the *metric-type* option, which really refers to the route type. For example, routes redistributed into OSPF must be OSPF external routes, but they can be either external type 1 (E1) or type 2 (E2) routes. Table 11-7 summarizes the defaults for metrics and metric types.

Table 11-7 *Default Metrics and Route Metric Types in IGP Route Redistribution*

KEY POINT

IGP into Which Routes Are Redistributed	Default Metric	Default (and Possible) Metric Types
RIP	None	RIP has no concept of external routes
EIGRP	None	External
OSPF	20/1*	E2 (E1 or E2)
IS-IS	0	L1 (L1, L2, L1/L2, or external)

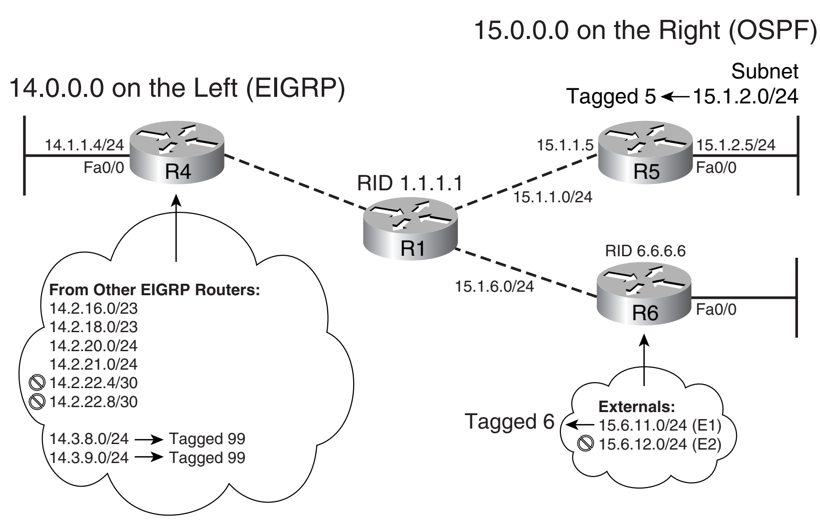
* OSPF uses cost 20 when redistributing from an IGP, and cost 1 when redistributing from BGP.

Redistributing a Subset of Routes Using a Route Map

Route maps can be referenced by any **redistribute** command. The route map may actually let all the routes through, setting different route attributes (for example, metrics) for different routes. Or, it may match some routes with a **deny** clause, which prevents the route from being redistributed. (Refer to Figure 11-1 for a review of route map logic.)

Figure 11-3 and Example 11-2 show an example of mutual redistribution between EIGRP and OSPF, with some routes being either filtered or changed using route maps.

Figure 11-3 *OSPF and EIGRP Mutual Redistribution Using Route Maps*



The following list details the requirements for redistribution from OSPF into EIGRP. These requirements use R1's perspective, because it is the router doing the redistribution.

- Routes with next-hop address 15.1.1.5 (R5) should be redistributed, with route tag 5.

- E1 routes sourced by R6 (RID 6.6.6.6) should be redistributed, and assigned a route tag of 6.
- No other routes should be redistributed.

The requirements for redistributing routes from EIGRP into OSPF are as follows, again from R1's perspective:

- Routes beginning with 14.2, and with masks /23 and /24, should be redistributed, with metric set to 300.
- Other routes beginning with 14.2 should not be redistributed.
- Routes beginning with 14.3 should be redistributed, with route tag 99.
- No other routes should be redistributed.

Most of the explanation of the configuration is provided in the comments in Example 11-2, with a few additional comments following the example.

Example 11-2 *Route Redistribution Using Route Maps*

```
! No metrics are set on the redistribute commands; either the default metric
! is used, or the route maps set the metrics. The default-metric command
! sets the unused EIGRP metric parameters to "1" because something must be
! configured, but the values are unimportant.
router eigrp 1
 redistribute ospf 1 route-map ospf-into-eigrp
 network 14.0.0.0
 default-metric 1544 5 1 1 1
 no auto-summary
! While this configuration strives to use other options besides the options
! directly on the redistribute command, when used by OSPF, you must still
! include the subnets keyword for OSPF to learn subnets from other IGPs.
router ospf 1
 router-id 1.1.1.1
 redistribute eigrp 1 subnets route-map eigrp-into-ospf
 network 15.0.0.0 0.255.255.255 area 0
! ACL A-14-3-x-x matches all addresses that begin 14.3. ACL A-15-1-1-5 matches
! exactly IP address 15.1.1.5. ACL A-6-6-6-6 matches exactly address 6.6.6.6.
ip access-list standard A-14-3-x-x
 permit 14.3.0.0 0.0.255.255
ip access-list standard A-15-1-1-5
 permit 15.1.1.5
ip access-list standard A-6-6-6-6
 permit 6.6.6.6
! The prefix lists matches prefixes in the range 14.2.0.0 through 14.2.255.255,
! with prefix length 23 or 24.
ip prefix-list e-into-o seq 5 permit 14.2.0.0/16 ge 23 le 24
```

Example 11-2 *Route Redistribution Using Route Maps (Continued)*

```
! route-map ospf-into-eigrp was called by the redistribute command under router
! eigrp, meaning that it controls redistribution from OSPF into EIGRP.
! Clause 10 matches OSPF routes whose next hop is 15.1.1.5, which is R5's serial
! IP address. R1's only route that meets this criteria is 15.1.2.0/24. This route
! will be redistributed because the route-map clause 10 has a permit action.
! The route tag is also set to 5.
route-map ospf-into-eigrp permit 10
 match ip next-hop A-15-1-1-5
 set tag 5
! Clause 15 matches OSPF routes whose LSAs are sourced by router with RID 6.6.6.6,
! namely R6, and also have metric type E1. R6 sources two external routes, but
! only 15.6.11.0/24 is E1. The route is tagged 6.
route-map ospf-into-eigrp permit 15
 match ip route-source A-6-6-6-6
 match route-type external type-1
 set tag 6
! route-map eigrp-into-ospf was called by the redistribute command under router
! ospf, meaning that it controls redistribution from EIGRP into OSPF.
! Clause 10 matches using a prefix list, which in turn matches prefixes that begin
! with 14.2, and which have either a /23 or /24 prefix length. By implication, it
! does not match prefix length /30. The metric is set to 300 for these routes.
route-map eigrp-into-ospf permit 10
 match ip address prefix-list e-into-o
 set metric 300
! Clause 18 matches routes that begin 14.3. They are tagged with a 99.
route-map eigrp-into-ospf permit 18
 match ip address A-14-3-x-x
 set tag 99
! Next, the example shows the routes that could be redistributed, and then
! shows the results of the redistribution, pointing out which routes were
! redistributed. First, the example shows, on R1, all routes that R1 could
! try to redistribute into EIGRP.
R1# show ip route 15.0.0.0
Routing entry for 15.0.0.0/24, 5 known subnets
Attached (2 connections)
Redistributing via eigrp 1

O E1    15.6.11.0 [110/84] via 15.1.6.6, 00:21:52, Serial0/0/0.6
O E2    15.6.12.0 [110/20] via 15.1.6.6, 00:21:52, Serial0/0/0.6
C       15.1.6.0 is directly connected, Serial0/0/0.6
O IA    15.1.2.0 [110/65] via 15.1.1.5, 00:21:52, Serial0/0/0.5
C       15.1.1.0 is directly connected, Serial0/0/0.5
```

```
! R4 sees only two of the five routes from 15.0.0.0, because only two matched either of
! the route-map clauses. The other three routes matched the default deny clause.
R4# show ip route 15.0.0.0
Routing entry for 15.0.0.0/24, 2 known subnets
```

continues

Example 11-2 *Route Redistribution Using Route Maps (Continued)*

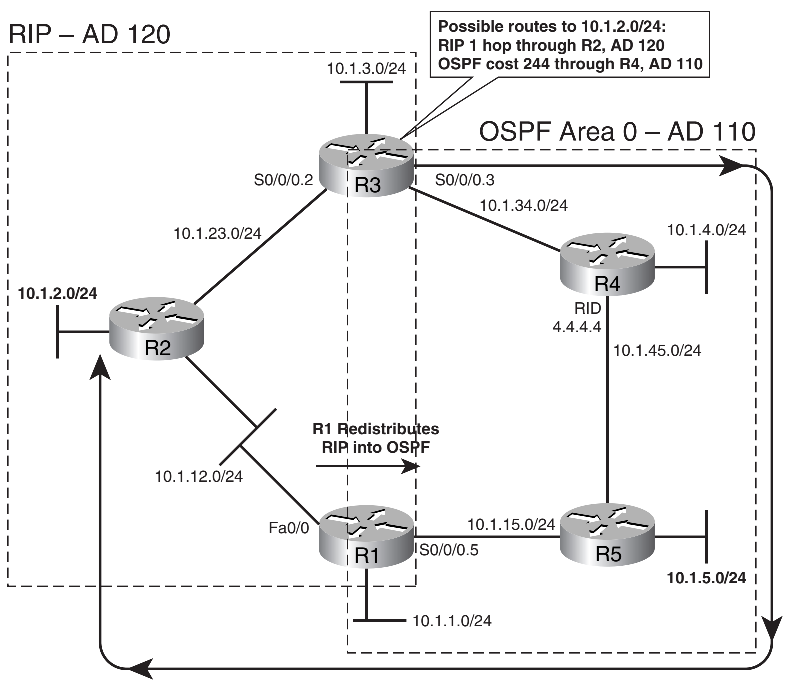
```
Redistributing via eigrp 1
D EX    15.6.11.0 [170/2171136] via 14.1.1.1, 00:22:21, Serial0/0/0.1
D EX    15.1.2.0 [170/2171136] via 14.1.1.1, 00:22:21, Serial0/0/0.1
! Still on R4, the show ip eigrp topology command displays the tag. This command
! filters the output so that just one line of output lists the tag values.
R4# sho ip eigrp topo 15.6.1.0 255.255.255.0 ¦ incl tag
        Administrator tag is 5 (0x00000005)
R4# sho ip eigrp topo 15.6.11.0 255.255.255.0 ¦ incl tag
        Administrator tag is 6 (0x00000006)
! Next, the example shows the possible routes that could be redistributed from
! EIGRP into OSPF.
! The next command (R1) lists all routes that could be redistributed into OSPF.
R1# show ip route 14.0.0.0
Routing entry for 14.0.0.0/8, 10 known subnets
  Attached (1 connections)
  Variably subnetted with 3 masks
  Redistributing via eigrp 1, ospf 1

D       14.3.9.0/24 [90/2297856] via 14.1.1.4, 00:34:48, Serial0/0/0.4
D       14.3.8.0/24 [90/2297856] via 14.1.1.4, 00:34:52, Serial0/0/0.4
D       14.1.2.0/24 [90/2172416] via 14.1.1.4, 00:39:27, Serial0/0/0.4
C       14.1.1.0/24 is directly connected, Serial0/0/0.4
D       14.2.22.8/30 [90/2297856] via 14.1.1.4, 00:35:49, Serial0/0/0.4
D       14.2.20.0/24 [90/2297856] via 14.1.1.4, 00:36:12, Serial0/0/0.4
D       14.2.21.0/24 [90/2297856] via 14.1.1.4, 00:36:08, Serial0/0/0.4
D       14.2.16.0/23 [90/2297856] via 14.1.1.4, 00:36:34, Serial0/0/0.4
D       14.2.22.4/30 [90/2297856] via 14.1.1.4, 00:35:53, Serial0/0/0.4
D       14.2.18.0/23 [90/2297856] via 14.1.1.4, 00:36:23, Serial0/0/0.4
! Next, on R5, note that the two /30 routes beginning with 14.2 were correctly
! prevented from getting into OSPF. It also filtered the redistribution of the
! two routes that begin with 14.1. As a result, R5 knows only 6 routes in
! network 14.0.0.0, whereas R1 had 10 subnets of that network it could have
! redistributed. Also below, note that the /23 and /24 routes inside 14.2 have
! metric 300.
R5# show ip route 14.0.0.0
Routing entry for 14.0.0.0/8, 6 known subnets
  Variably subnetted with 2 masks

O E2    14.3.9.0/24 [110/20] via 15.1.1.1, 00:22:41, Serial0/0.1
O E2    14.3.8.0/24 [110/20] via 15.1.1.1, 00:22:41, Serial0/0.1
O E2    14.2.20.0/24 [110/300] via 15.1.1.1, 00:22:41, Serial0/0.1
O E2    14.2.21.0/24 [110/300] via 15.1.1.1, 00:22:41, Serial0/0.1
O E2    14.2.16.0/23 [110/300] via 15.1.1.1, 00:22:41, Serial0/0.1
O E2    14.2.18.0/23 [110/300] via 15.1.1.1, 00:22:41, Serial0/0.1
! The show ip ospf database command confirms that the route tag was set
! correctly.
R5# show ip ospf data external 14.3.8.0 | incl Tag
External Route Tag: 99
```

NOTE Route maps have an implied **deny** clause at the end of the route map. This implied **deny** clause matches all packets. As a result, any routes not matched in the explicitly configured **route-map** clauses match the implied **deny** clause, and are filtered. Both route maps in the example used the implied **deny** clause to actually filter the routes.

Mutual Redistribution at Multiple Routers

When multiple routers redistribute between the same two routing protocol domains, several potential problems can occur. One type of problem occurs on the redistributing routers, because those routers will learn a route to most subnets via both routing protocols. That router uses the AD to determine the best route when comparing the best routes from each of the two routing protocols; this typically results in some routes using suboptimal paths. For example, Figure 11-4 shows a sample network, with R3 choosing its AD 110 OSPF route to 10.1.2.0/24 over the probably better AD 120 RIP route.

Figure 11-4 *OSPF and RIP Redistribution*



NOTE The OSPF configuration for this network matches only the interfaces implied by the OSPF box in Figure 11-4. RIP does not have a *wildcard-mask* option on the **network** command, so R1's and R3's **network** commands will match all of their interfaces, as all are in network 10.0.0.0.

In Figure 11-4, R3 learns of subnet 10.1.2.0/24 via RIP updates from R2. Also, R1 learns of the subnet with RIP and redistributes the route into OSPF, and then R3 learns of a route to 10.1.2.0/24 via OSPF. R3 chooses the route with the lower administrative distance; with all default settings, OSPF's AD of 110 is better that RIP's 120.

If both R1 and R3 mutually redistribute between RIP and OSPF, the suboptimal route problem would occur on either R1 or R3 for each RIP subnet, all depending on timing. Example 11-3 shows the redistribution configuration, along with R3 having the suboptimal route shown in Figure 11-4. However, after R1's fa0/0 interface flaps, R1 now has a suboptimal route to 10.1.2.0/24, but R3 has an optimal route.

Example 11-3 *Suboptimal Routing at Different Redistribution Points*

```
! R1's related configuration follows:
router ospf 1
 router-id 1.1.1.1
 redistribute rip subnets
 network 10.1.15.1 0.0.0.0 area 0
!
router rip
 redistribute ospf 1
 network 10.0.0.0
 default-metric 1
```

```
! R3's related configuration follows:
router ospf 1
 router-id 3.3.3.3
 redistribute rip subnets
 network 10.1.34.3 0.0.0.0 area 0
!
router rip
 redistribute ospf 1
 network 10.0.0.0
 default-metric 1
! R3 begins with an AD 120 OSPF route, and not a RIP route, to 10.1.2.0/24.
R3# sh ip route | incl 10.1.2.0
O E2    10.1.2.0 [110/20] via 10.1.34.4, 00:02:01, Serial0/0/0.4
```

```
! R1 has a RIP route to 10.1.2.0/24, and redistributes it into OSPF, causing R3
! to learn an OSPF route to 10.1.2.0/24.
R1# sh ip route | incl 10.1.2.0
R       10.1.2.0 [120/1] via 10.1.12.2, 00:00:08, FastEthernet0/0
! Next, R1 loses its RIP route to 10.1.2.0/24, causing R3 to lose its OSPF route.
R1# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)# int fa 0/0
R1(config-if)# shut
```

```
! R3 loses its OSPF route, but can then insert the RIP route into its table.
```

Example 11-3 *Suboptimal Routing at Different Redistribution Points (Continued)*

```
R3# sh ip route | incl 10.1.2.0
R       10.1.2.0 [120/1] via 10.1.23.2, 00:00:12, Serial0/0/0.2
! Not shown: R1 brings up its fa0/0 again
! However, R1 now has the suboptimal route to 10.1.2.0/24, through OSPF.
R1# sh ip route | incl 10.1.2.0
O E2    10.1.2.0 [110/20] via 10.1.15.5, 00:00:09, Serial0/0/0.5
```

The key concept behind this seemingly odd example is that a redistributing router processes only the current contents of its IP routing table. When this network first came up, R1 learned its RIP route to 10.1.2.0/24, and redistributed into OSPF, *before* R3 could do the same. So, R3 was faced with the choice of putting the AD 110 (OSPF) or AD 120 (RIP) route into its routing table, and R3 chose the lower AD OSPF route. Because R3 never had the RIP route to 10.1.2.0/24 in its routing table, R3 could not redistribute that RIP route into OSPF.

Later, when R1's fa0/0 failed (as shown in Example 11-3), R3 had time to remove the OSPF route and add the RIP route for 10.1.2.0/24 to its routing table—which then allowed R3 to redistribute that RIP route into OSPF, causing R1 to have the suboptimal route.

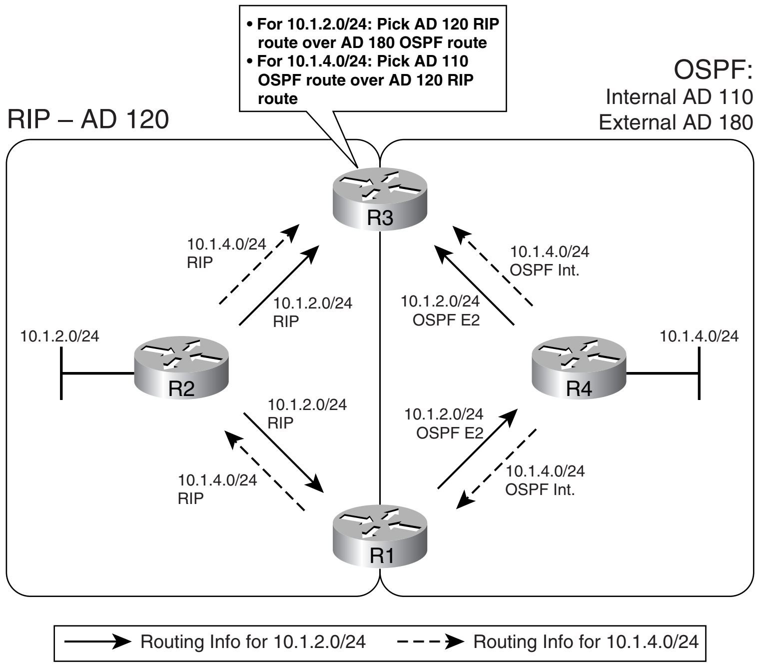
To solve this type of problem, the redistributing routers must have some awareness of which routes came from the other routing domain. In particular, the lower-AD routing protocol needs to decide which routes came from the higher-AD routing protocol, and either use a different AD for those routes or filter the routes. The next few sections show a few different methods of preventing this type of problem.

Preventing Suboptimal Routes by Setting the Administrative Distance

One simple and elegant solution to the problem of suboptimal routes on redistributing routers is to flag the redistributed routes with a higher AD. A route's AD is not advertised by the routing protocol; however, a single router can be configured such that it assigns different AD values to different routes, which then impacts that one router's choice of which routes end up in that router's routing table. For example, back in Figure 11-4 and Example 11-3, R3 could have assigned the OSPF-learned route to 10.1.2.0/24 an AD higher than 120, thereby preventing the original problem.

Figure 11-5 shows a more complete example, with a route from the RIP domain (10.1.2.0/24) and another from the OSPF domain (10.1.4.0/24). Redistributing router R3 will learn the two routes both from RIP and OSPF. By configuring R3's logic to treat OSPF internal routes with default AD 110, and OSPF external routes with AD 180 (or any other value larger than RIP's default of 120), R3 will choose the optimal path for both RIP and OSPF routes.

Figure 11-5 *The Effect of Differing ADs for Internal and External Routes*



Example 11-4 shows how to configure both R1 and R3 to use a different AD for external routes by using the **distance ospf external 180** command, under the **router ospf** process.

Example 11-4 *Preventing Suboptimal Routes with the* **distance** *Router Subcommand*

```
! Both R1's and R3's configurations look like they do in Example 11-3's, but with the
! addition of the distance command.
router ospf 1
 distance ospf external 180
! R3 has a more optimal RIP route to 10.1.2.0/24, as does R1.
R3# sh ip route | incl 10.1.2.0
R       10.1.2.0 [120/1] via 10.1.23.2, 00:00:19, Serial0/0/0.2
! R1 next…
R1# show ip route | incl 10.1.2.0_
R       10.1.2.0 [120/1] via 10.1.12.2, 00:00:11, FastEthernet0/0
! R1 loses its next-hop interface for the RIP route, so now its OSPF route, with
! AD 180, is its only and best route to 10.1.2.0/24.
R1# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)# int fa 0/0
R1(config-if)# shut
R1(config-if)# do sh ip route | incl 10.1.2.0
O E2    10.1.2.0 [180/20] via 10.1.15.5, 00:00:05, Serial0/0/0.5
```

EIGRP supports the exact same concept by default, using AD 170 for external routes and 90 for internal routes. In fact, if EIGRP were used instead of OSPF in this example, neither R1 nor R3 would have experienced any of the suboptimal routing. You can reset EIGRP's distance for internal and external routes by using the **distance eigrp** router subcommand. (At presstime, neither the IS-IS nor RIP **distance** commands support setting external route ADs and internal route ADs to different values.)

In some cases, the requirements may not allow for setting all external routes' ADs to another value. For instance, if R4 injected some legitimate external routes into OSPF, the configuration in Example 11-4 would result in either R1 or R3 having a suboptimal route to those external routes that pointed through the RIP domain. In those cases, the **distance** router subcommand can be used in a different way, influencing some or all of the routes that come from a particular router. The syntax is as follows:

distance {*distance-value ip-address* {*wildcard-mask*} [*ip-standard-list*] [*ip-extended-list*]

This command sets three key pieces of information: the AD to be set, the IP address of the router advertising the routes, and, optionally, an ACL with which to match routes. With RIP, EIGRP, and IS-IS, this command identifies a neighboring router's interface address using the *ip-address wildcard-mask* parameters. With OSPF, those same parameters identify the RID of the router owning (creating) the LSA for the route. The optional ACL then identifies the subset of routes for which the AD will be set. The logic boils down to something like this:

> Set this AD value for all routes, learned from a router that is defined by the IP address and wildcard mask, and for which the ACL permits the route.

Example 11-5 shows how the command could be used to solve the same suboptimal route problem on R1 and R3, while not causing suboptimal routing for other external routes. The design goals are summarized as follows:

- Set a router's local AD for its OSPF routes for subnets in the RIP domain to a value of 179, thereby making the RIP routes to those subnets better than the OSPF routes to those same subnets.
- Do not set the AD for any other routes.

Example 11-5 *Using the* **distance** *Command to Reset Particular Routes' ADs*

```
! R1 config. Note that the command refers to 3.3.3.3, which is R3's RID. Other
! commands not related to resetting the AD are omitted. Of particular importance,
! the distance command on R1 refers to R3's OSPF RID, because R3 created the OSPF
! LSAs that we are trying to match—the LSAs created when R3 injected the
! routes redistributed from RIP.
router ospf 1
 distance 179 3.3.3.3 0.0.0.0 only-rip-routes
```

continues

Example 11-5 *Using the* **distance** *Command to Reset Particular Routes' ADs (Continued)*

```
!
ip access-list standard only-rip-routes
 permit 10.1.12.0
 permit 10.1.3.0
 permit 10.1.2.0
 permit 10.1.23.0
```

```
! R3 config. Note that the command refers to 1.1.1.1, which is R1's RID. Other
! commands not related to resetting the AD are omitted. Also, the only-rip-routes
! ACL is identical to R1's only-rip-routes ACL.
router ospf 1
distance 179 1.1.1.1 0.0.0.0 only-rip-routes
```

Preventing Suboptimal Routes by Using Route Tags

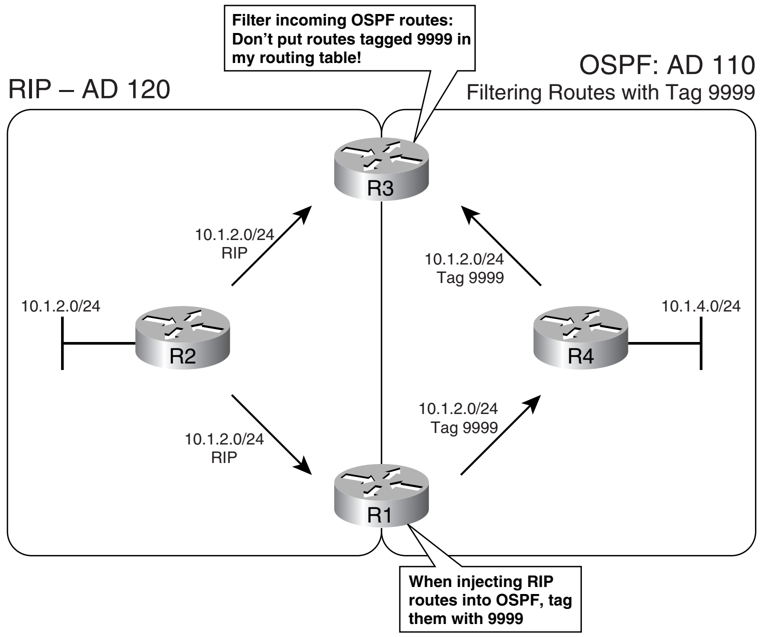
Another method of preventing suboptimal routing on the redistributing routers is to simply filter the problematic routes. Using subnet 10.1.2.0/24 as an example again, R3 could use an incoming **distribute-list** command to filter the OSPF route to 10.1.2.0/24, allowing R3 to use its RIP route to 10.1.2.0/24. R1 would need to perform similar route filtering as well to prevent its suboptimal route.

Performing simple route filtering based on IP subnet number works, but the redistributing routers will need to be reconfigured every time subnets change in the higher-AD routing domain. The administrative effort can be improved by adding *route tagging* to the process. By tagging all routes taken from the higher-AD domain and advertised into the lower-AD domain, the **distribute-list** command can make a simple check for that tag. Figure 11-6 shows the use of this idea for subnet 10.1.2.0/24.

Route tags are simply unitless integer values in the data structure of a route. These tags, typically either 16 or 32 bits long depending on the routing protocol, allow a router to imply something about a route that was redistributed from another routing protocol. For instance, R1 can tag its OSPF-advertised route to 10.1.2.0/24 with a tag—say, 9999. OSPF does not define what a tag of 9999 means, but the OSPF protocol includes the tag field in the LSA so that it can be used for administrative purposes. Later, R3 can filter routes based on their tag, solving the suboptimal route problem.

Figure 11-6 and Example 11-6 depict an example of route tagging and route filtering, used to solve the same old problem with suboptimal routes. R1 and R3 tag all redistributed RIP routes with tag 9999 as they enter the OSPF domain, and then R1 and R3 filter incoming OSPF routes based on the tags. This design works well because R1 can tag all redistributed RIP routes, thereby removing the need to change the configuration every time a new subnet is added to the RIP domain. (Note that both R1 and R3 will tag routes injected from RIP into OSPF as 9999, and both will then filter OSPF-learned routes with tag 9999. Figure 11-6 just shows one direction to keep the figure less cluttered.)

Figure 11-6 *Filtering with Reliance on Route Tags*



Example 11-6 *Using Route Tags and Distribute Lists to Prevent Suboptimal Routes at Redistributing Routers*

```
! R1 config. The redistribute command calls the route map that tags routes taken
! from RIP as 9999. distribute-list looks at routes learned in OSPF that were
! earlier tagged by R3.
router ospf 1
 redistribute rip subnets route-map tag-rip-9999
 network 10.1.15.1 0.0.0.0 area 0
 distribute-list route-map check-tag-9999 in
! Clause 10, a deny clause, matches all tagged 9999 routes—so those
! routes are filtered. Clause 20 permits all other routes, because with no match
! subcommand, the clause is considered to "match all."
route-map check-tag-9999 deny 10
 match tag 9999
!
route-map check-tag-9999 permit 20
! tag-rip-9999 matches all routes (it has no match command), and then
! tags them all with tag 9999. This route-map is used only for routes taken from
! RIP into OSPF.
route-map tag-rip-9999 permit 10
 set tag 9999
```
```
! R3 Config
! The R3 configuration does not have to use the same names for route maps, but
```

continues

Example 11-6 *Using Route Tags and Distribute Lists to Prevent Suboptimal Routes at Redistributing Routers (Continued)*

```
! the essential elements are identical, so the route maps are not repeated here.
router ospf 1
 redistribute rip subnets route-map tag-rip-9999
 network 10.1.34.3 0.0.0.0 area 0
 distribute-list route-map check-tag-9999 in
! R3 (shown) and R1 have RIP routes to 10.1.2.0, as well as other routes from the
! RIP domain. Also, note that the OSPF LSDB shows the tagged values on the routes.
R3# show ip route | incl 10.1.2.0
R       10.1.2.0 [120/1] via 10.1.23.2, 00:00:26, Serial0/0/0.2
R3# sh ip ospf data  begin Type-5
                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
10.1.1.0        1.1.1.1         834         0x80000006 0x00CE86 9999
10.1.1.0        3.3.3.3         458         0x80000003 0x0098B7 9999
10.1.2.0        1.1.1.1         834         0x80000006 0x00C390 9999
10.1.2.0        3.3.3.3         458         0x80000003 0x008DC1 9999
! lines omitted for brevity
! Next, the unfortunate side effect of filtering the routes—R3 does not have an
! alternative route to RIP subnets, although OSPF internal routers (like R4
! in Figure 11-6) will.
R3# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)# int s0/0/0.2
R3(config-subif)# shut
R3(config-subif)# ^Z
R3# sh ip route | incl 10.1.2.0
R3#
```

The last few lines of the example show the largest negative of using route filtering to prevent the suboptimal routes. When R3 loses connectivity to R2, R3 does not use the alternate route through the OSPF domain. R3's filtering of those routes occurs regardless of whether R3's RIP routes are available or not. As a result, using a solution that manipulates the AD may ultimately be the better solution to this suboptimal-routing problem.

Using Metrics and Metric Types to Influence Redistributed Routes

A different set of issues can occur for a router that is internal to a single routing domain, like R4 and R5 in Figure 11-4. The issue is simple—with multiple redistributing routers, an internal router learns multiple routes to the same subnet, so it must pick the best route. As covered earlier in the chapter, the redistributing routers can set the metrics; by setting those metrics with meaningful values, the internal routers can be influenced to use a particular redistribution point.

Interestingly, internal routers may not use metric as their first consideration when choosing the best route. For instance, an OSPF internal router will first take an intra-area route over an inter-area route, regardless of their metrics. Table 11-8 lists the criteria an internal router will use when picking the best route, before considering the metrics of the different routes.

Table 11-8 *IGP Order of Precedence for Choosing Routes Before Considering the Metric*

IGP	Order of Precedence of Metric
RIP	No other considerations
EIGRP	Internal, then external
OSPF	Intra-area, inter-area, E1, then E2*
IS-IS	L1, L2, external

* For E2 routes whose metric ties, OSPF also checks the cost to the advertising ASBR.

To illustrate some of these details, Example 11-7 focuses on R4 and its routes to 10.1.2.0/24 and 10.1.5.0/24 from Figure 11-4. The example shows the following, in order:

1. R1 and R3 advertise 10.1.2.0/24 as an E2 route, metric 20. R4 uses the route through R3, because R4's cost to reach ASBR R3 is lower than its cost to reach ASBR R1.
2. After changing R1 to advertise redistributed routes into OSPF as E1 routes, R4 uses the E1 routes through R1, even though the metric is larger than the E2 route through R3.
3. R4 uses it higher-metric intra-area route to 10.1.5.0/24 through R5. Then, the R4-R5 link fails, causing R4 to use the OSPF external E2 route to 10.1.5.0/24—the route that leads through the RIP domain and back into OSPF via the R3-R2-R1-R5 path.

Example 11-7 *Demonstration of the Other Decision Criteria for Choosing the Best Routes*

```
! R4 has E2 routes to all the subnets in the RIP domain, and they all point to R3.
R4# sh ip route ospf
10.0.0.0/24 is subnetted, 10 subnets
O       10.1.15.0 [110/128] via 10.1.45.5, 00:03:23, Serial0/0/0.5
O E2    10.1.12.0 [110/20] via 10.1.34.3, 00:03:23, Serial0/0/0.3
O E2    10.1.3.0 [110/20] via 10.1.34.3, 00:03:23, Serial0/0/0.3
O E2    10.1.2.0 [110/20] via 10.1.34.3, 00:03:23, Serial0/0/0.3
O E2    10.1.1.0 [110/20] via 10.1.34.3, 00:03:23, Serial0/0/0.3
O       10.1.5.0 [110/65] via 10.1.45.5, 00:03:23, Serial0/0/0.5
O E2    10.1.23.0 [110/20] via 10.1.34.3, 00:03:23, Serial0/0/0.3
! R4 chose the routes through R3 instead of R1 due to the lower cost to R3.
R4# show ip ospf border-routers
OSPF Process 1 internal Routing Table
Codes: i - Intra-area route, I - Inter-area route
```

continues

Example 11-7 *Demonstration of the Other Decision Criteria for Choosing the Best Routes (Continued)*

```
i 1.1.1.1 [128] via 10.1.45.5, Serial0/0/0.5, ASBR, Area 0, SPF 13
i 3.3.3.3 [64] via 10.1.34.3, Serial0/0/0.3, ASBR, Area 0, SPF 13
! (Not Shown): R1 is changed to redistribute RIP routes as E1 routes by
! adding the metric-type 1 option on the redistribute command on R1.
! R4 picks routes through R1 because they are E1 routes, even though the metric
! (148) is higher than the routes through R3 (cost 20)
R4# show ip route ospf
10.0.0.0/24 is subnetted, 10 subnets
O E1    10.1.2.0 [110/148] via 10.1.45.5, 00:00:11, Serial0/0/0.5
! lines omitted for brevity
! R4's route to 10.1.5.0/24 below is intra-area, metric 65
R4# show ip route | incl 10.1.5.0
O       10.1.5.0 [110/65] via 10.1.45.5, 00:04:48, Serial0/0/0.5
! (Not Shown): R4 shuts down link to R5
! R4's new route to 10.1.5.0/24 is E2, learned from R3, with metric 20
R4# show ip route | incl 10.1.5.0\
O E2    10.1.5.0 [110/20] via 10.1.34.3, 00:10:52, Serial0/0/0.3
```

Route Summarization

Route summarization creates a single route whose numeric range, as implied by the prefix/prefix length, is larger than the one or more smaller component routes. For example, 10.1.0.0/16 is a summary route that includes component subnets 10.1.1.0/24, 10.1.4.132/30, and any other subnets with the range 10.1.0.0 through 10.1.255.255.

> **NOTE** I use the term *component route* to refer to a route whose range of IP addresses is a subset of the range specified by a summary route; however, I have not seen this term in other reference materials from Cisco.

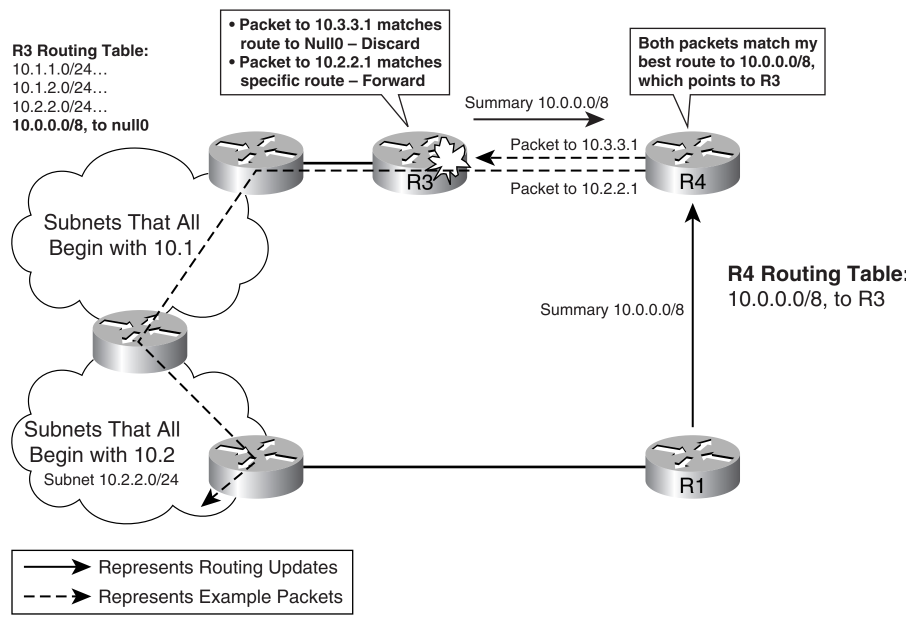
The following list details some of the key features that the three IGPs covered in this book have in common with regard to how route summarization works (by default):

- The advertised summary is assigned the same metric as the currently lowest-metric component subnet.
- The router does not advertise the component subnets.
- The router does not advertise the summary when its routing table does not have any of the component subnets.
- The summarizing router creates a local route to the summary, with destination null0, to prevent routing loops.

- Summary routes reduce the size of routing tables and topology databases, indirectly improving convergence.
- Summary routes decrease the amount of specific information in routing tables, sometimes causing suboptimal routing.

Figure 11-7 depicts the suboptimal-routing side effect when using route summarization. It also depicts the effect of using a summary to null0 on the summarizing router.

Figure 11-7 *Route Summarization Suboptimal Routing and Routing to Null0*



In Figure 11-7, R4 learned two paths to summary route 10.0.0.0/8, and picked the route through R3 based on the metric. Because R4 does not have a route for 10.2.2.0/24, R4 then sends any packets to that subnet based on its route to network 10.0.0.0/8, through R3. So, although subnets like 10.2.2.0/24 may be topologically closer to R4 through R1, R4 sends the packets via the scenic, suboptimal route through R3.

Also note that R4's summary route to 10.0.0.0/8 matches packets for which the component subnet does not exist anywhere in the network. In that case, routers like R4 forward the packets based on the larger summary, but once the packet reaches the router that created the summary, the packet is discarded by the summarizing router due to its null route. For instance, Figure 11-7 shows R4 forwarding a packet destined to 10.3.3.1 to R3. R3 does not have a more specific route than its route to 10.0.0.0/8, with next-hop interface null0. As a result, R3 discards the packet.

The sections that follow provide a few details about summarization with each routing protocol.

EIGRP Route Summarization

EIGRP provides the easiest and most straightforward rules for summarizing routes as compared with RIPv2, OSPF, and IS-IS. To summarize routes, the **ip summary-address eigrp** *as-number network-address subnet-mask* [*admin-distance*] command is placed under an interface. If any of the component routes are in that router's routing table, EIGRP advertises the summary route *out* that interface. The summary is defined by the *network-address subnet-mask* parameters.

One of the more interesting features of the EIGRP summary is the ability to set the AD of the summary route. The AD is not advertised with the route; the summarizing router, however, uses the configured AD to determine whether the null route for the summary should be put into its routing table. The EIGRP AD for summary routes defaults to 5.

OSPF Route Summarization

All OSPF routers in the same area must have identical LSDBs after flooding is complete. As a result, all routers in the same OSPF area must have the same summary routes, and must be missing the same component subnets of each summary. To make that happen, OSPF allows route summarization only as routes are injected into an area, either by an ABR (inter-area routes) or by an ASBR (external routes).

OSPF uses two different configuration commands to create the summary routes, depending on whether the summary is for inter-area or external routes. Table 11-9 lists the two commands. Both commands are configured under **router ospf**.

Table 11-9 *OSPF Route Summarization Commands*

KEY POINT

Where used	Command
ASBR	**summary-address** {{*ip-address mask*} | {*prefix mask*}} [**not-advertise**] [**tag** *tag*]
ABR	**area** *area-id* **range** *ip-address mask* [**advertise** | **not-advertise**] [**cost** *cost*]

The commands have a couple of important attributes. First, the **area range** command specifies an area; this area is the area in which the component subnets reside, with the summary being advertised into *all other areas*. Also, the **area range** command can set the cost for the summary route, instead of using the lowest cost of all component routes. Also, the **not-advertise** keyword can essentially be used to filter the subnets implied by the summary, as covered in Chapter 10, "OSPF."

The **summary-address** command summarizes external routes as they are injected into OSPF as an ASBR. The cost can be assigned, and the routes can be filtered using the **not-advertise** keyword.

RIP Route Summarization

RIP route summarization is weird in comparison to route summarization in the other IGPs, but at first glance, it appears to work just like EIGRP. To summarize routes, RIP uses the interface subcommand **ip summary-address rip** *ip-address ip-network-mask*. It can be used on any interface out which RIP advertises routes. If any of the component routes are in that router's routing table, RIP advertises the summary route out the interface, as defined by the *ip-address ip-network-mask* parameters. It also ceases advertising the component routes out that interface.

So far its sounds just like EIGRP summarization; however, there are a couple of unique restrictions. First, RIP route summarization works only with RIPv2, because RIPv1 does not support VLSM. Also, RIP does not allow supernetting—for instance, the command **ip summary-address rip 172.16.0.0 255.254.0.0**, which would combine two class B networks into one summary, is not supported by RIP. Finally, on a single interface, only one **ip summary-address rip** command is allowed per classful network. In other words, RIP would not allow a router to create two summary routes, one for 10.1.0.0/16 and one for 10.2.0.0/16, and advertise both out the same interface. EIGRP has none of these restrictions.

Default Routes

Routers forward packets using a default route when there are no specific routes that match a packet's destination IP address in the IP routing table. Routing protocols can advertise default routes, with each router choosing the best default route to list as that router's *gateway of last resort*. This section covers how a router can create a default route and then cause an IGP to advertise the default route.

In addition to the advertisement of default routes, each router may use one of two options for how the default route is used. As described in Chapter 7, "IP Forwarding (Routing)," each router's configuration includes either the (default) **ip classless** command or the **no ip classless** command. With **ip classless**, if a packet's destination does not match a specific route in the IP routing table, the router uses the default route. With **no ip classless**, the router first checks to see if any part of the destination address's classful network is in the routing table. If so, that router will not use the default route for forwarding that packet.

> **NOTE** The topic of default routing requires discussion of the configuration on one router, plus configuration of the other routers using the same IGP. For this section, I will call the router with the default routing configuration the "local" router, and other routers using the same IGP "other" routers.

Cisco IOS supports five basic methods of advertising default routes with IGPs, four of which are covered here. One method for advertising a default route is for one routing protocol to redistribute another routing protocol's default route. Because route redistribution has already been covered heavily, this section of the chapter covers other methods. Of the other four methods, not all are supported by all IGPs, as you can see in Table 11-10.

Table 11-10 *Four Methods for Learning Default Routes*

KEY POINT

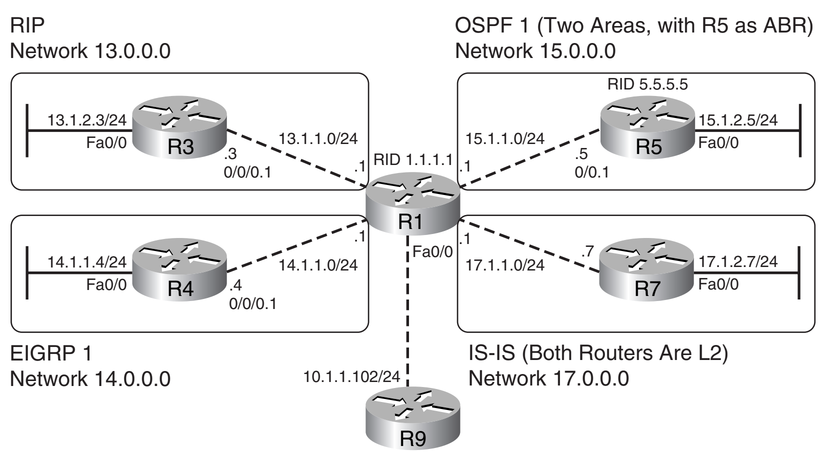
Feature	RIP	EIGRP	OSPF
Static route to 0.0.0.0, with the **redistribute static** command	Yes	Yes	No
The **default-information originate** command	Yes	No	Yes
The **ip default-network** command	Yes	Yes	No
Using summary routes	No	Yes	No

Interestingly, when a router learns of multiple default routes, using any of these methods, it will use the usual process for choosing the best route: administrative distance, route type (per Table 11-9, earlier in this chapter), and lowest metric, in that order.

NOTE Table 11-10 has details that may be difficult to memorize. To make it easier, you could start by ignoring the use of summary static routes, because it is not recommended by Cisco. Then, note that RIP supports the other three methods, whereas EIGRP supports two methods and OSPF supports only one—with EIGRP and OSPF not supporting any of the same options.

Figure 11-8 shows a sample network used with all the default route examples, in which R1 is the local router that configures the default routing commands.

Figure 11-8 *Sample Network for Default Route Examples*



Using Static Routes to 0.0.0.0, with redistribute static

Routers consider a route to 0.0.0.0/0 as a default route. RIP and EIGRP support redistribution of static routes, including such a default static route. The rules and conditions for redistributing static defaults into RIP and EIGRP are as follows:

KEY POINT

- The static **ip route 0.0.0.0 0.0.0.0** and **redistribute static** commands need to be configured on the same local router.
- The metric must be defaulted or set, using the same methods covered earlier in this chapter.
- The **redistribute** command can refer to a route map, which examines all static routes (not just the default).
- EIGRP treats the default route as an external route by default, with default AD 170.
- This method is not supported by OSPF.

Example 11-8 shows how R1 can inject defaults via RIP to R3 and via EIGRP to R4. The EIGRP configuration refers to a route map that examines all static routes, matching only static default routes. If other static routes existed, EIGRP would not advertise those routes based on the route map.

Example 11-8 *Static Default Route with Route Redistribution*

```
! R1 Config—note that ip classless is configured, but it does not impact the
! advertisement of the static route at all.
router eigrp 1
 redistribute static route-map just-default
 network 10.0.0.0
 network 14.0.0.0
 default-metric 1544 10 1 1 1
!
router rip
 version 2
 redistribute static
 network 13.0.0.0
 default-metric 1
!
ip classless
! The static route is configured next, followed by the prefix list that matches
! the default route, and the route map that refers to the prefix list.
ip route 0.0.0.0 0.0.0.0 10.1.1.102
!
ip prefix-list zero-prefix seq 5 permit 0.0.0.0/0
!
route-map just-default permit 10
 match ip address prefix-list zero-prefix
!
```

continues

Example 11-8 *Static Default Route with Route Redistribution (Continued)*

```
route-map just-default deny 20
! Next, R3, the RIP router, lists R1 (13.1.1.1) as its gateway of last resort,
! based on the RIP route to 0.0.0.0/0, next hop 13.1.1.1.
R3# sh ip route
! Lines omitted for brevity
Gateway of last resort is 13.1.1.1 to network 0.0.0.0

     13.0.0.0/24 is subnetted, 2 subnets
C       13.1.1.0 is directly connected, Serial0/0/0.1
C       13.1.2.0 is directly connected, FastEthernet0/0
R*   0.0.0.0/0 [120/1] via 13.1.1.1, 00:00:12, Serial0/0/0.1
! Next, R4, the EIGRP router, lists R1 (14.1.1.1) as its gateway of last resort,
! based on the EIGRP route to 0.0.0.0/0, next hop 14.1.1.1. Note that the default
! points to 0.0.0.0/0, AD 170, as it is an external route, due to the EX listed
! in the output of the show ip route command.
R4# sh ip route
! lines omitted for brevity
Gateway of last resort is 14.1.1.1 to network 0.0.0.0

D    10.0.0.0/8 [90/2172416] via 14.1.1.1, 00:01:30, Serial0/0/0.1
     14.0.0.0/24 is subnetted, 2 subnets
C       14.1.2.0 is directly connected, FastEthernet0/0
C       14.1.1.0 is directly connected, Serial0/0/0.1
D*EX 0.0.0.0/0 [170/2172416] via 14.1.1.1, 00:01:30, Serial0/0/0.1
```

Using the default-information originate Command

OSPF does not support redistribution of statically defined default routes. Instead, OSPF requires the **default-information originate** router subcommand, which essentially tells OSPF to redistribute any default routes found in the routing table, either static routes or routes from another routing protocol. The following list summarizes the default routing features when using the **default-information originate** command with OSPF:

KEY POINT

- Redistributes any default route (0.0.0.0/0) in the routing table.
- The command can set the metric and metric type directly, with OSPF defaulting to cost 1 and type E2.
- OSPF allows the use of the **always** keyword, which means a default is sourced regardless of whether a default route is in the routing table.
- Not supported by EIGRP.
- Supported by RIP, with some differences. (Refer to the text following Example 11-9 for an explanation of the differences.)

Example 11-9 shows an example of using the **default-information originate** command with OSPF. In this case, R1 has learned a route to 0.0.0.0/0 via BGP from R9 in Figure 11-8.

Example 11-9 *Static Default Route with Route Redistribution*

```
router ospf 1
 network 15.0.0.0 0.255.255.255 area 0
 default-information originate
```

```
! R5 has a default route, defaulting to type E2, cost 1. It as advertised as a
! type 5 LSA.
R5# show ip route ospf
O*E2 0.0.0.0/0 [110/1] via 15.1.1.1, 00:18:07, Serial0/0.1
R5# sh ip ospf data | begin Type-5
                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
0.0.0.0         1.1.1.1         1257        0x80000001 0x008C12 1
```

As mentioned earlier, RIP does support the **default-information originate** command; however, the command behaves slightly differently in RIP than it does in OSPF. With RIP, this command creates and advertises a default route if either no default route exists or a default route was learned from another routing protocol. However, if a static route to 0.0.0.0/0 is in the local routing table, the **default-information originate** command does *not* cause RIP to inject a default—the reason behind this behavior is that RIP already supports redistribution of static routes, so **redistribute static** should be used in that case.

Using the ip default-network Command

RIP and EIGRP can inject default routes by using the **ip default-network** command. To do so, the following must be true on the local router:

- The local router must configure the **ip default-network** *net-number* command, with *net-number* being a classful network number.
- The classful network must be in the local router's IP routing table, via any means.
- For EIGRP only, the classful network must be advertised by the local router into EIGRP, again through any means.
- This method is not supported by OSPF.

When using the **ip default-network** command, RIP and EIGRP differ in how they advertise the default. RIP advertises a route to 0.0.0.0/0, but EIGRP flags its route to the classful network as a candidate default route. Because EIGRP flags these routes as candidates, EIGRP must then also be advertising those classful networks. However, because RIP does not flag the classful network as a candidate default route, RIP does not actually have to advertise the classful network referenced in the **ip default-network** command.

Example 11-10 shows the key difference between RIP and EIGRP with regard to the **ip default-network** command. In this case, R1 will advertise about classful network 10.0.0.0 using EIGRP due to the **auto-summary** command.

Example 11-10 *Static Default Route with Route Redistribution*

```
! EIGRP will advertise classful network 10.0.0.0/8 due to its network command,
! matching R1's fa0/0 interface, and the auto-summary command. Also, R1 must have
! a route to classful network 10.0.0.0/8, in this case due to a static route.
! RIP will not advertise classful network 10.0.0.0/8, but it will still be able
! to inject a default route based on the ip default-network command.
router eigrp 1
 network 10.0.0.0
 network 14.0.0.0
 auto-summary
!
router rip
 version 2
 network 13.0.0.0
!
ip classless
ip default-network 10.0.0.0
ip route 10.0.0.0 255.0.0.0 10.1.1.102
```

```
! On R3, RIP learns a route to 0.0.0.0/0 as its default.
R3# show ip route rip
R*   0.0.0.0/0 [120/1] via 13.1.1.1, 00:00:19, Serial0/0/0.1
```

```
! On R4, note that EIGRP learned a route to 10.0.0.0/8, shown with a * that
! flags the route as a candidate default route.
R4# show ip route
! lines omitted for brevity
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 14.1.1.1 to network 10.0.0.0

D*   10.0.0.0/8 [90/2172416] via 14.1.1.1, 00:05:35, Serial0/0/0.1
     14.0.0.0/24 is subnetted, 2 subnets
C       14.1.2.0 is directly connected, FastEthernet0/0
C       14.1.1.0 is directly connected, Serial0/0/0.1
```

Using Route Summarization to Create Default Routes

Generally speaking, route summarization combines smaller address ranges into a small number of larger address ranges. From that perspective, 0.0.0.0/0 is the largest possible summary, because it includes all possible IPv4 addresses. And, as it turns out, EIGRP route summarization supports summarizing the 0.0.0.0/0 supernet, effectively creating a default route.

Because route summarization causes a null route to be created for the summary, some Cisco documentation advises against using route summarization to create a default route. For example, in Figure 11-8, imagine that R9 is owned by this network's ISP, and R1 learns a default route (0.0.0.0/0) via EBGP from R9. However, when R1 configures an EIGRP default route using route summarization, R1 will also create a local route to 0.0.0.0/0 as well, but with destination null0. The EBGP route has a higher AD (20) than the EIGRP summary route to null0 (AD 5), so R1 will now replace its BGP-learned default route with the summary route to null0—preventing R1 from being able to send packets to the Internet.

Route summarization can still be used to create default routes with the proper precautions. The following list details a few of the requirements and options:

- The local router creates a local summary route, destination null0, using AD 5 (EIGRP), when deciding if its route is the best one to add to the local routing table.
- EIGRP advertises the summary to other routers as AD 90 (internal).
- This method is not supported by RIP and OSPF.
- To overcome the caveat of EIGRP's default route being set to null by having a low AD, set the AD higher (as needed) with the **ip summary-address** command.

Example 11-11 lists a sample configuration on R1 again, this time creating summary routes to 0.0.0.0/0 for EIGRP.

Example 11-11 *EIGRP and IS-IS Configuration for Creating Default Summary Routes*

```
! EIGRP route summarization is done under s0/0/0.4, the subnet connected to R4. In this
! example, the AD was changed to 7 (default 5) just to show how to change the AD. To
! avoid the problem with the default route to null0 on R1, the AD should have been set
! higher than the default learned via BGP.
interface Serial0/0/0.4 point-to-point
 ip address 14.1.1.1 255.255.255.0
 ip summary-address eigrp 1 0.0.0.0 0.0.0.0 7
! In this example, R1 has two sources for a local route to 0.0.0.0/0: EIGRP
! (AD 7, per the ip summary-address command), and BGP from R9
! (AD 20). R1 installs the EIGRP route based on the lowest AD.
R1# show ip route eigrp
      14.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
D       14.1.2.0/24 [90/2172416] via 14.1.1.4, 00:01:03, Serial0/0/0.4
D       14.0.0.0/8 is a summary, 05:53:19, Null0
D*   0.0.0.0/0 is a summary, 00:01:08, Null0
! Next, R4's EIGRP route shows AD 90, instead of the AD 7 configured at R1. AD is
! a local parameter—R4 uses its default AD of 90 for internal routes.
R4# show ip route eigrp
D*   0.0.0.0/0 [90/2172416] via 14.1.1.1, 00:01:14, Serial0/0/0.1
```

Foundation Summary

This section lists additional details and facts to round out the coverage of the topics in this chapter. Unlike most of the Cisco Press *Exam Certification Guides*, this book does not repeat information presented in the "Foundation Topics" section of the chapter. Please take the time to read and study the details in this section of the chapter, as well as review the items in the "Foundation Topics" section noted with Key Point icons.

Table 11-11 lists some of the most relevant Cisco IOS commands related to the topics in this chapter. Also refer to Tables 11-2 and 11-3 for the **match** and **set** commands.

Table 11-11 *Command Reference for Chapter 11*

Command	Command Mode and Description
redistribute *protocol* [*process-id*] {**level-1** | **level-1-2** | **level-2**} [*as-number*] [**metric** *metric-value*] [**metric-type** *type-value*] [**match** {**internal** | **external 1** | **external 2**}] [**tag** *tag-value*] [**route-map** *map-tag*] [**subnets**]	Router config mode; defines the routing protocol from which to take routes, several matching parameters, and several things that can be marked on the redistributed routes.
ip prefix-list *list-name* [**seq** *seq-value*] {**deny** *network/length* | **permit** *network/length*} [**ge** *ge-value*] [**le** *le-value*]	Global config mode; defines members of a prefix list, which match a prefix (subnet) and prefix length (subnet mask).
ip prefix-list *list-name sequence-number* **description** *text*	Global config; sets a description to a line in a prefix list.
distance {*ip-address* {*wildcard-mask*}} [*ip-standard-list*] [*ip-extended-list*]	Router config mode; identifies the route source, and an optional ACL to define a subnet of routes, for which this router's AD is changed. Influences the selection of routes by selectively overriding default AD.
distance eigrp *internal-distance external-distance*	EIGRP config; sets the AD for all internal and external routes.
distance ospf {[**intra-area** *dist1*] [**inter-area** *dist2*] [**external** *dist3*]}	OSPF config; sets the AD for all intra-area, interarea, and external routes.
ip summary-address eigrp *as-number network-address subnet-mask* [*admin-distance*]	Interface mode; configures an EIGRP route summary.
ip summary-address rip *ip-address ip-network-mask*	Interface mode; configures a RIP route summary.
area *area-id* **range** *ip-address mask* [**advertise** | **not-advertise**] [**cost** *cost*]	OSPF mode; configures an OSPF summary between areas.
summary-address *address mask* {**level-1** | **level-1-2** | **level-2**}	IS-IS mode; configures an IP summary route.

Table 11-11 *Command Reference for Chapter 11 (Continued)*

Command	Command Mode and Description
summary-address {{*ip-address mask*} | {*prefix mask*}} [**not-advertise**] [**tag** *tag*]	OSPF mode; configures an OSPF summary of external routes.
ip default-network *network-number*	Global config; sets a network from which to derive default routes.
default-information originate [**route-map** *map-name*]	IS-IS config; tells IS-IS to advertise a default route if it is in the routing table.
default-information originate [**always**] [**metric** *metric-value*] [**metric-type** *type-value*] [**route-map** *map-name*]	OSPF config; tells OSPF to advertise a default route, either if it is in the routing table or always.
ip route *prefix mask* {*ip-address* | *interface-type interface-number* [*ip-address*]} [*distance*] [*name*] [**permanent**] [**tag** *tag*]	Global config; used to create static IP routes, including static routes to 0.0.0.0 0.0.0.0, which denotes a default route.

Memory Builders

The CCIE Routing and Switching written exam, like all Cisco CCIE written exams, covers a fairly broad set of topics. This section provides some basic tools to help you exercise your memory about some of the broader topics covered in this chapter.

Fill in Key Tables from Memory

First, take the time to print Appendix F, "Key Tables for CCIE Study," which contains empty sets of some of the key summary tables from the "Foundation Topics" section of this chapter. Then, simply fill in the tables from memory, checking your answers when you review the "Foundation Topics" sections tables that have a Key Point icon beside them. The PDFs can be found on the CD in the back of the book, or at http://www.ciscopress.com/title/1587201410.

Definitions

Next, take a few moments to write down the definitions for the following terms:

> default route, route redistribution, external route, aggregate route, route map, IP prefix list, summary route, component route, gateway of last resort

Refer to the CD-based glossary to check your answers.

Further Reading

Routing TCP/IP, Volume I, Second Edition, by Jeff Doyle and Jennifer DeHaven Carroll

CCIE Practical Studies, Volume II, by Karl Solie and Leah Lynch