# Route Optimization

## BSCI Module 5

# Objectives

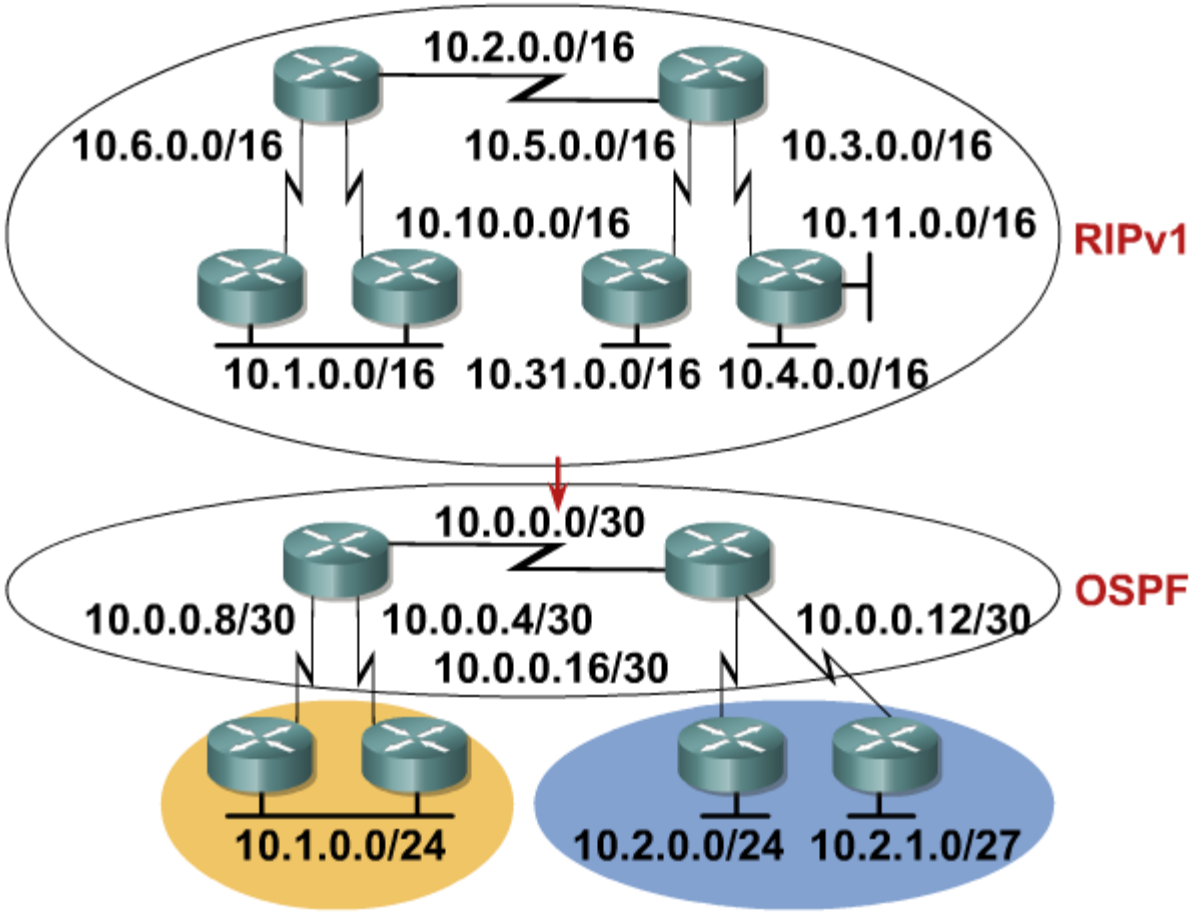
- Explain the purpose and use of seed metrics in route redistribution.
- Describe how to redistribute routes into RIP, OSPF, EIGRP, and IS-IS.
- Explain how to verify route redistribution.
- Explain how to control routing updates using the **`passive-interface default`** command and route maps.
- Describe new DHCP commands.

# Purpose of this Lesson

- Coverage of topics new to the “Route Optimization” module of BSCI.
- What’s new in this module?
  - Detailed explanation of seed metrics
  - Redistribute routes into RIP, OSPF, EIGRP, and IS-IS
  - Verify route redistribution
  - Controlling routing updates with the passive-interface default command and route maps
  - Using the distance command to avoid suboptimal routing
  - More DHCP commands

# Seed Metrics and Route Redistribution

# Using Multiple IP Routing Protocols




- FLSM to VLSM
- Hierarchical Addressing
- Hierarchical Areas

# Using Multiple Routing Protocols

- Interim during conversion
- Application-specific protocols
  - One size does not always fit all.
- Political boundaries
  - Groups that do not work well with others
- Mismatch between devices
  - Multivendor interoperability
  - Host-based routers

# Redistribution with Seed Metric

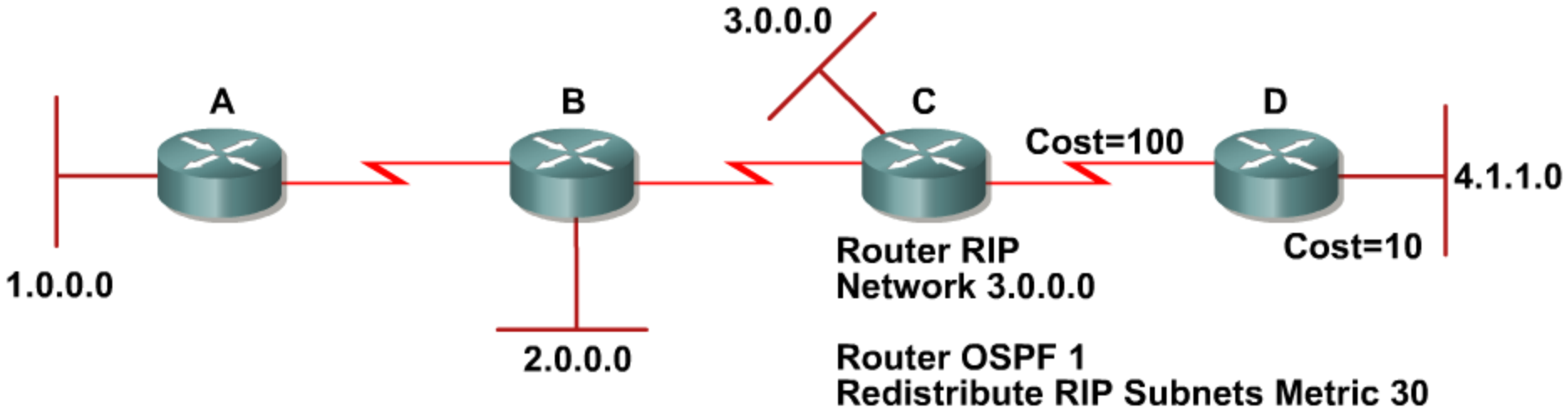



**TABLE A**

```
C 1.0.0.0
R 120/1 2.0.0.0
R 120/2 3.0.0.0
```

**TABLE B**

```
C 2.0.0.0
R 120/1 1.0.0.0
R 120/1 3.0.0.0
```

**TABLE C**

```
C 3.0.0.0
R 120/1    2.0.0.0
R 120/2    1.0.0.0
o 110/110  4.1.1.0
```

**TABLE D**

```
C  4.1.1.0/24
E2 110/130 1.0.0.0
E2 110/130 2.0.0.0
E2 110/130 3.0.0.0
```

# Default Seed Metrics

| Protocol | Default Seed Metrics |
|---|---|
| RIP | Infinity |
| IGRP/EIGRP | Infinity |
| OSPF | 20 for all except BGP, which is 1 |
| IS-IS | 0 |
| BGP | BGP metric is set to IGP metric value |

# Self Check

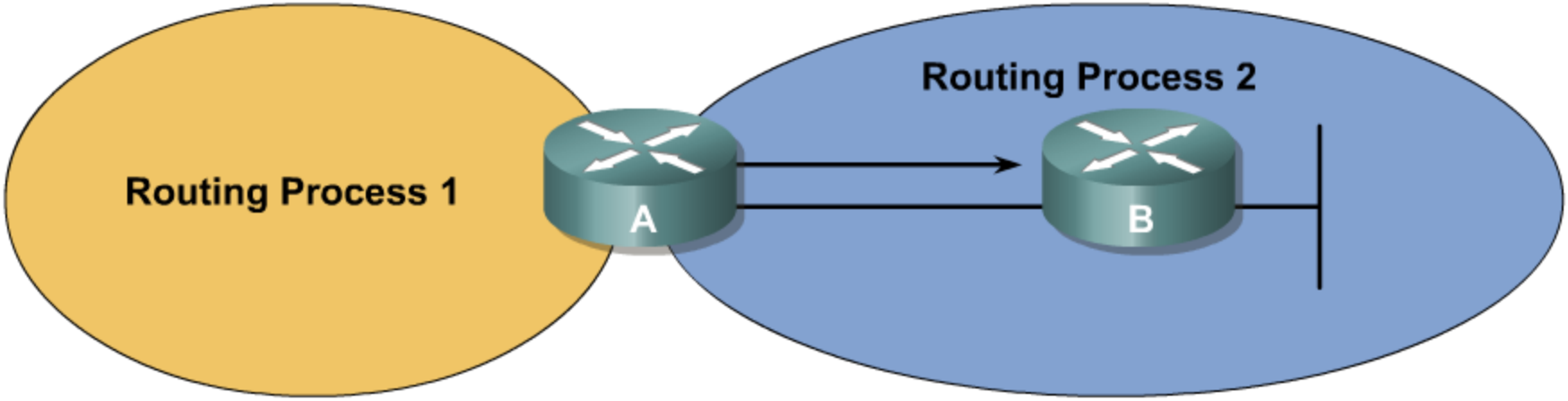
1. How is a seed metric used in redistribution?
2. What does a metric of infinity tell the router?
3. Which routing protocols should be configured with default metrics to prevent the default of infinity?

# Configuring and Verifying Route Redistribution

# Redistribution Supports All Protocols

```
RtrA(config)#router rip
RtrA(config-router)#redistribute ?
  bgp         Border Gateway Protocol (BGP)
  connected   Connected
  eigrp       Enhanced Interior Gateway Routing Protocol
(EIGRP)
  isis        ISO IS-IS
  iso-igrp    IGRP for OSI networks
  metric      Metric for redistributed routes
  mobile      Mobile routes
  odr         On Demand stub Routes
  ospf        Open Shortest Path First (OSPF)
  rip         Routing Information Protocol (RIP)
  route-map   Route map reference
  static      Static routes
  <cr>
```

# Planning Redistribution




- Locate the boundary router between two routing processes.
- Determine which routing process is the core or backbone process
- Determine which routing process is the edge or migration process
- Select a method for injecting the required edge protocol routes into the core.

# Configuring Redistribution into RIP

- Use this command to redistribute routes into RIP:

```
Router(config-router)# redistribute protocol
[process-id] [match route-type] [metric metric-
value] [route-map map-tag]
```

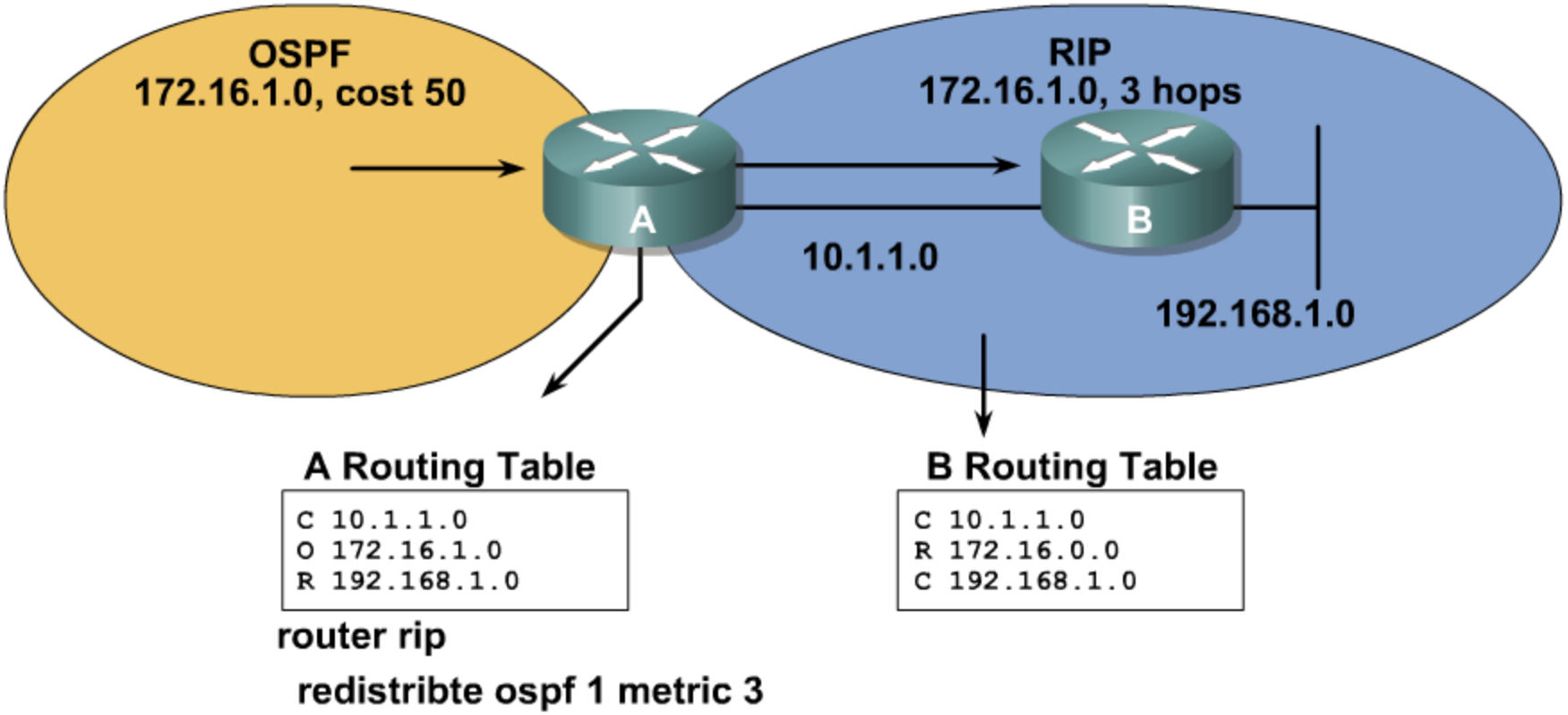
```
RtrA(config)# router rip
RtrA(config-router)# redistribute ospf ?

 <1-65535>  Process ID
RtrA(config-router)# redistribute ospf 1 ?

  match       Redistribution of OSPF routes
  metric      Metric for redistributed routes
  route-map   Route map reference
  …
  <cr>
```

- Default metric is infinity.

# The `redistribute` command parameters for RIP

| Parameter | Description |
|---|---|
| *protocol* | Source protocol from which routes are being redistributed. |
| *process-id* | This value is an AS number. For OSPF, this value is an OSPF process ID. |
| **match** *route-type* | (Optional) Command parameter used for redistributing OSPF routes into another routing protocol. For OSPF, the criterion by which OSPF routes are redistributed into other routing domains. |
| **metric** *metric-value* | (Optional) Parameter used to specify the RIP seed metric for the redistributed route. When you are redistributing into RIP, this value is not specified and no value is specified using the **default-metric** router configuration command, then the default metric is 0, which is interpreted as infinity, and routes will not be redistributed. The metric for RIP is the hop count. |
| **route-map** *map-tag* | (Optional) Identifier of a configured route map to be interrogated to filter the importation of routes from this source routing protocol to the current routing protocol. |

# Redistributing into RIP



# Configuring Redistribution into OSPF

- Use this command to redistribute routes into OSPF:

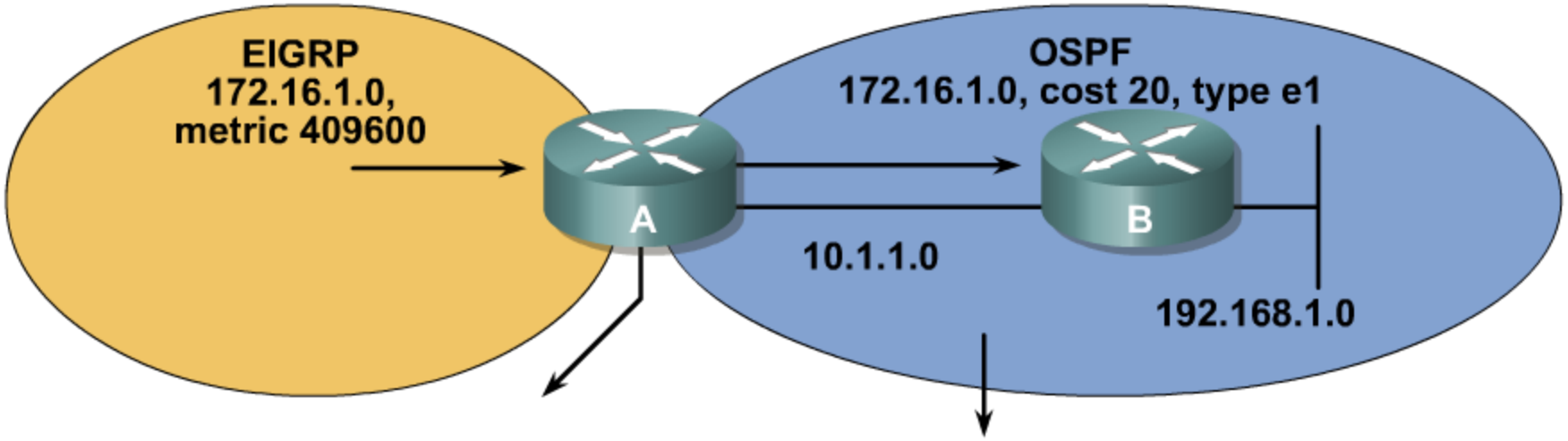
```
Router(config-router)# redistribute protocol
[process-id] [metric metric-value] [metric-type
type-value] [route-map map-tag] [subnets] [tag
tag-value]
```

- Default metric is 20.
- Default metric type is 2.
- Subnets do not redistribute by default.

# Example: Redistribution into OSPF

```
RtrA(config)# router ospf 1
RtrA(config-router)# redistribute eigrp ?

  <1-65535>  Autonomous system number
RtrA(config-router)# redistribute eigrp 100 ?

  metric          Metric for redistributed routes
  metric-type     OSPF/IS-IS exterior metric type for
redistributed routes
  route-map       Route map reference
  subnets         Consider subnets for redistribution into OSPF
  tag             Set tag for routes redistributed into OSPF
  …
  <cr>
```

# Redistributing into OSPF




**A Routing Table**

```
C 10.1.1.0
D 172.16.1.0
O 192.168.1.0
```

**B Routing Table**

```
C     10.1.1.0
O E1  172.16.1.0
C     192.168.1.0
```

**router ospf 1**

**redistribte eigrp 100 subnets metric-type 1**

# Configuring Redistribution into EIGRP

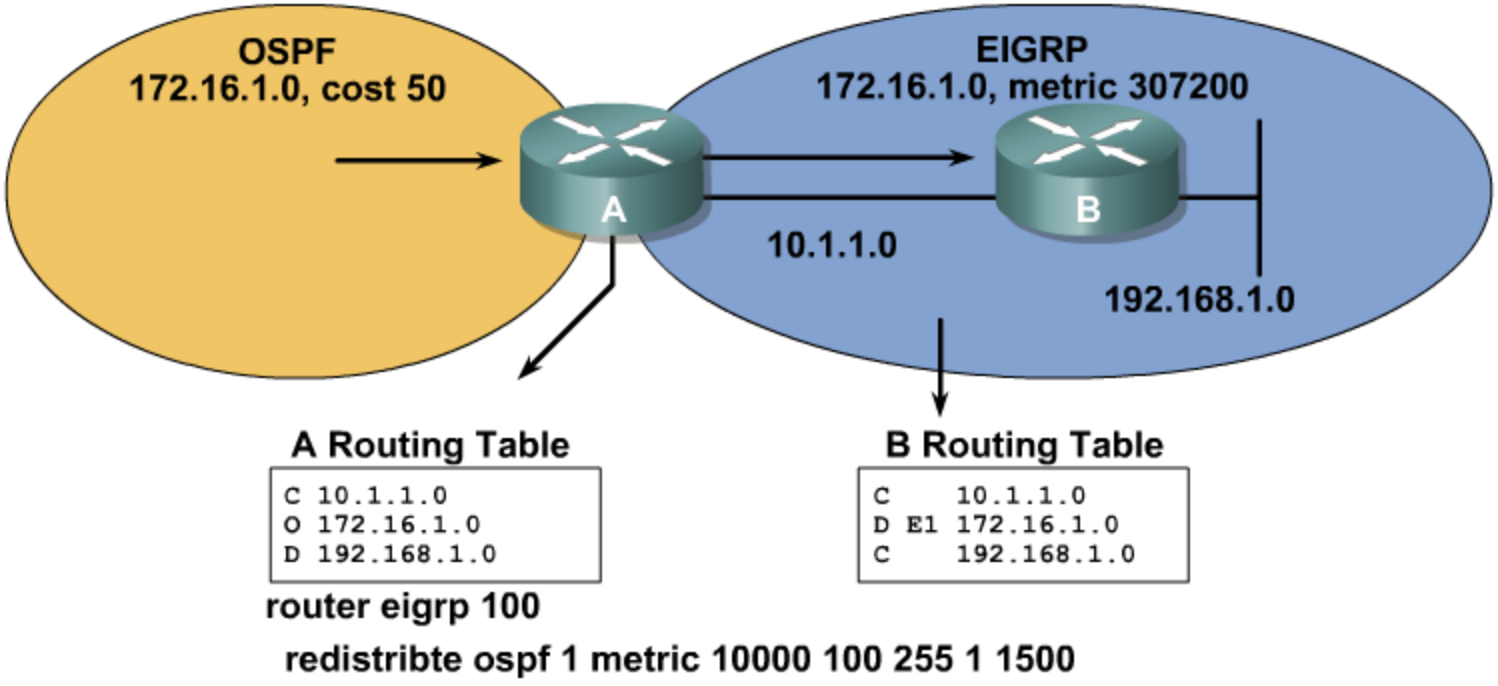
- Use this command to redistribute routes into EIGRP:

router(config-router)# **redistribute** *protocol* [*process-id*] [**match** {**internal** | **external 1** | **external 2**}] [**metric** *metric-value*] [**route-map** *map-tag*]

```
RtrA(config)# router eigrp 100
RtrA(config-router)# redistribute ospf ?

  <1-65535>  Process ID
RtrA(config-router)# redistribute ospf 1 ?

  match         Redistribution of OSPF routes
  metric        Metric for redistributed routes
  route-map     Route map reference

  …
<cr>
```

- Default metric is infinity.

# Redistributing into EIGRP




- Bandwidth in kilobytes = 10000
- Delay in tens of microseconds = 100
- Reliability = 255 (maximum)
- Load = 1 (minimum)
- MTU = 1,500 bytes

# Configuring Redistribution into IS-IS

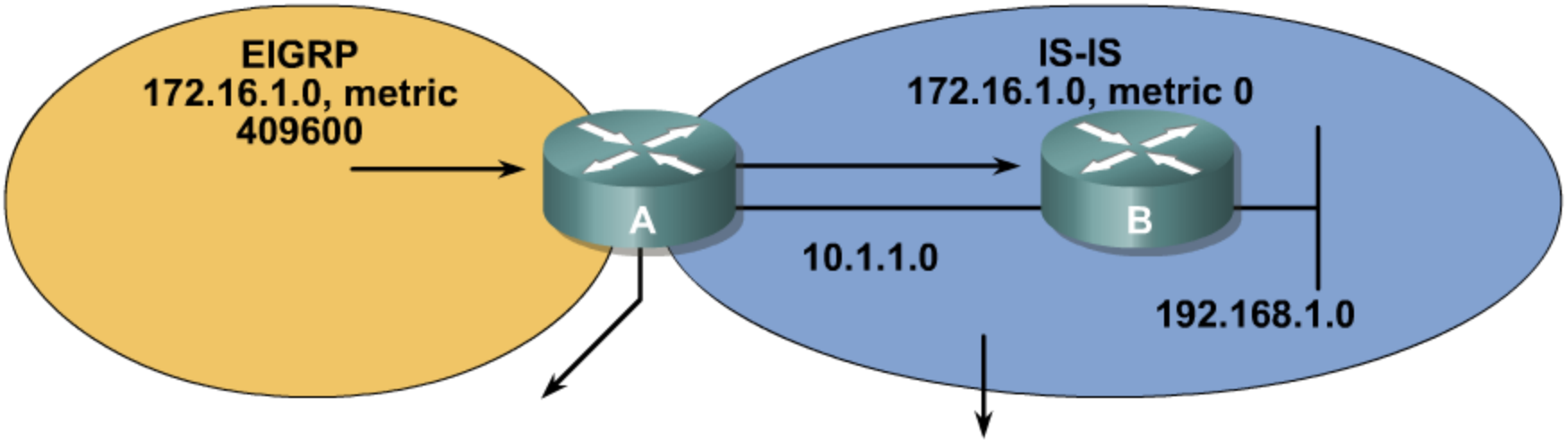
- Use this command to redistribute routes into IS-IS:

router(config-router)# **redistribute** *protocol* [*process-id*] [**level** *level-value*] [**metric** *metric-value*] [**metric-type** *type-value*] [**route-map** *map-tag*]

```
RtrA(config)# router isis
RtrA(config-router)# redistribute eigrp 100 ?

  level-1       IS-IS level-1 routes only
  level-1-2     IS-IS level-1 and level-2 routes
  level-2       IS-IS level-2 routes only
  metric        Metric for redistributed routes
  metric-type   OSPF/IS-IS exterior metric type for redistributed routes
  route-map     Route map reference
  ..
  Output Omitted
```

- Routes are introduced as level 2 with a metric of 0 by default.

# Redistributing into IS-IS




**A Routing Table**

```
C     10.1.1.0
D     172.16.1.0
i L1  192.168.1.0
```

**B Routing Table**

```
C     10.1.1.0
i L2  172.16.1.0
C     192.168.1.0
```

**router isis**

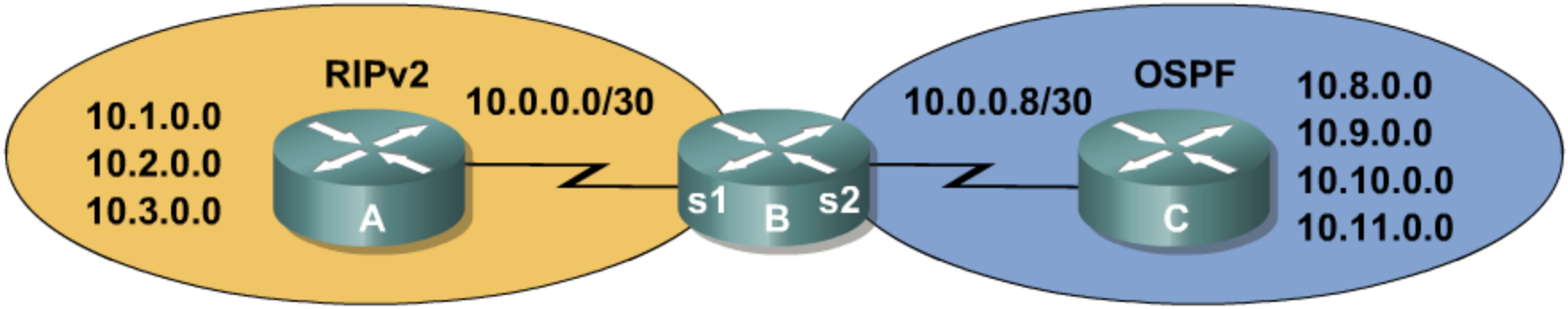
**redistribte eigrp 100**

# Redistributing IS-IS into Other Protocols

```
Router(config)# router ospf 1
Router(config-router)# redistribute isis ?
<output omitted>
  level-1                 IS-IS level-1 routes only
  level-1-2               IS-IS level-1 and level-2 routes
  level-2                 IS-IS level-2 routes only
<output omitted>
```

# Route Redistribution Example

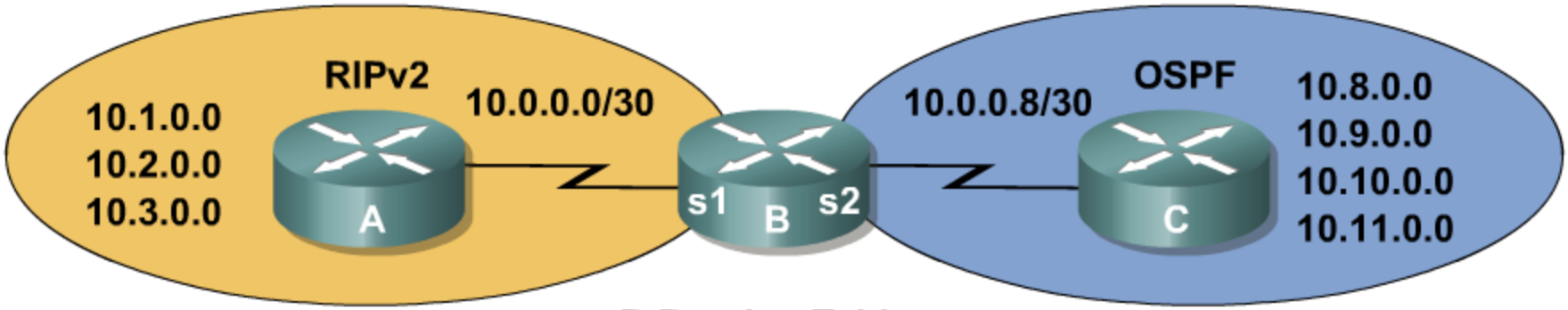
# Example: Before Redistribution




**Router B Configuration**

```
router ospf 1
   network 10.0.0.8 0.0.0.3 area 0

router rip
   network 10.0.0.0
   version 2
   passive-interface s2
```

# Example: Before Redistribution (Cont.)




**A Routing Table**

| | |
|---|---|
| C | 10.0.0.0 |
| R | 10.1.0.0 |
| R | 10.2.0.0 |
| R | 10.3.0.0 |

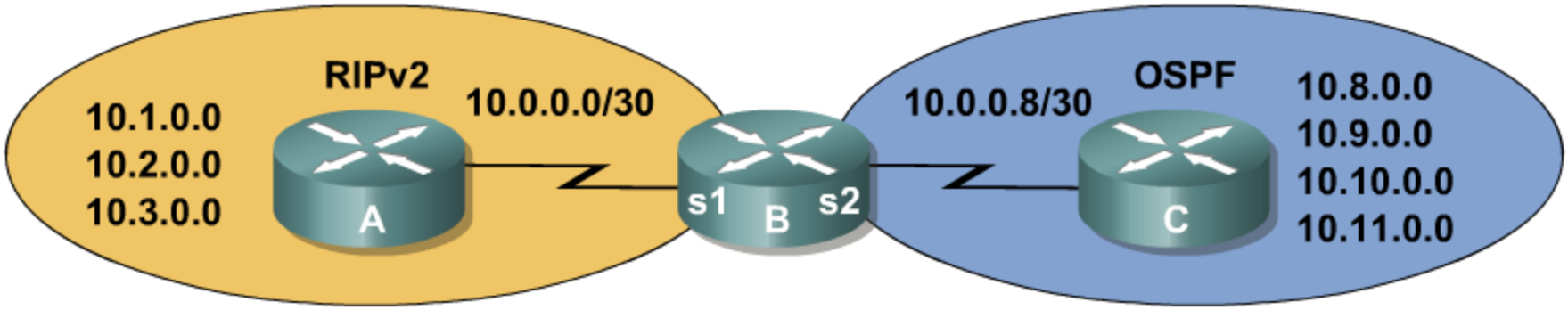
**B Routing Table**

| | |
|---|---|
| C | 10.0.0.0 |
| C | 10.0.0.8 |
| R | 10.1.0.0 |
| R | 10.2.0.0 |
| R | 10.3.0.0 |
| O | 10.8.0.0 |
| O | 10.9.0.0 |
| O | 10.10.0.0 |
| O | 10.11.0.0 |

**C Routing Table**

| | |
|---|---|
| C | 10.0.0.8 |
| O | 10.8.0.0 |
| O | 10.9.0.0 |
| O | 10.10.0.0 |
| O | 10.11.0.0 |

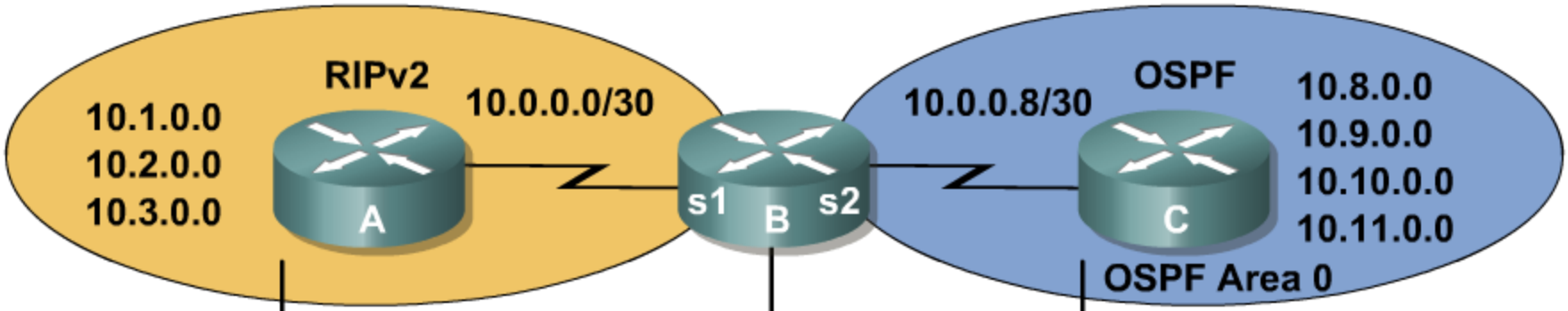
# Example: Configuring Redistribution at Router B




**Router B Configuration**

```
router ospf 1
   network 10.0.0.8 0.0.0.3 area 0
   redistribute rip subnets metric 300

router rip
   network 10.0.0.0
   version 2
   passive-interface s2
   redistribute ospf 1 metric 5
```

# Example: Routing Tables After Route Redistribution




**A Routing Table**

| | |
|---|---|
| C | 10.0.0.0 |
| R | 10.1.0.0 |
| R | 10.2.0.0 |
| R | 10.3.0.0 |
| R | 10.8.0.0 |
| R | 10.9.0.0 |
| R | 10.10.0.0 |
| R | 10.11.0.0 |

**B Routing Table**

| | |
|---|---|
| C | 10.0.0.0 |
| C | 10.0.0.8 |
| R | 10.1.0.0 |
| R | 10.2.0.0 |
| R | 10.3.0.0 |
| O | 10.8.0.0 |
| O | 10.9.0.0 |
| O | 10.10.0.0 |
| O | 10.11.0.0 |

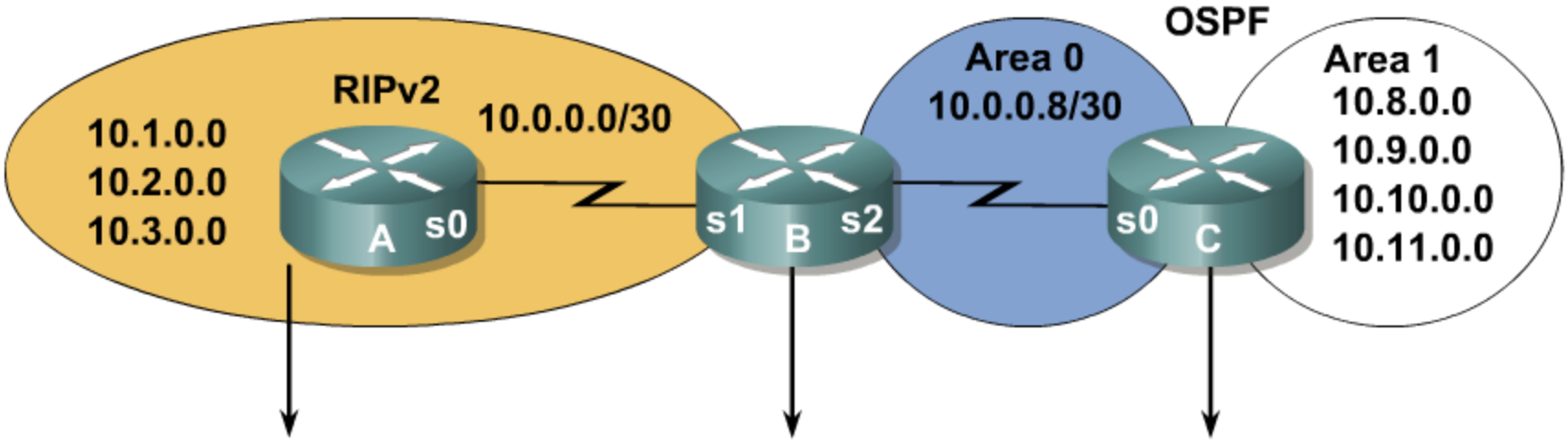
**C Routing Table**

| | |
|---|---|
| C | 10.0.0.8 |
| O E2 | 10.1.0.0 |
| O E2 | 10.2.0.0 |
| O E2 | 10.3.0.0 |
| O | 10.8.0.0 |
| O | 10.9.0.0 |
| O | 10.10.0.0 |
| O | 10.11.0.0 |

# Example: Routing Tables After Summarizing Routes and Redistributions

```
RouterA(config) #interface s0
RouterA(config-if) #ip summary-address rip 10.0.0.0 255.252.0.0
```

```
RouterC(config) #router ospf 1
RouterC(config-router) #area 1 range 10.8.0.0 255.252.0.0
```




**A Routing Table**

```
C       10.0.0.0
R       10.1.0.0
R       10.2.0.0
R       10.3.0.0
R       10.8.0.0/14
```

**B Routing Table**

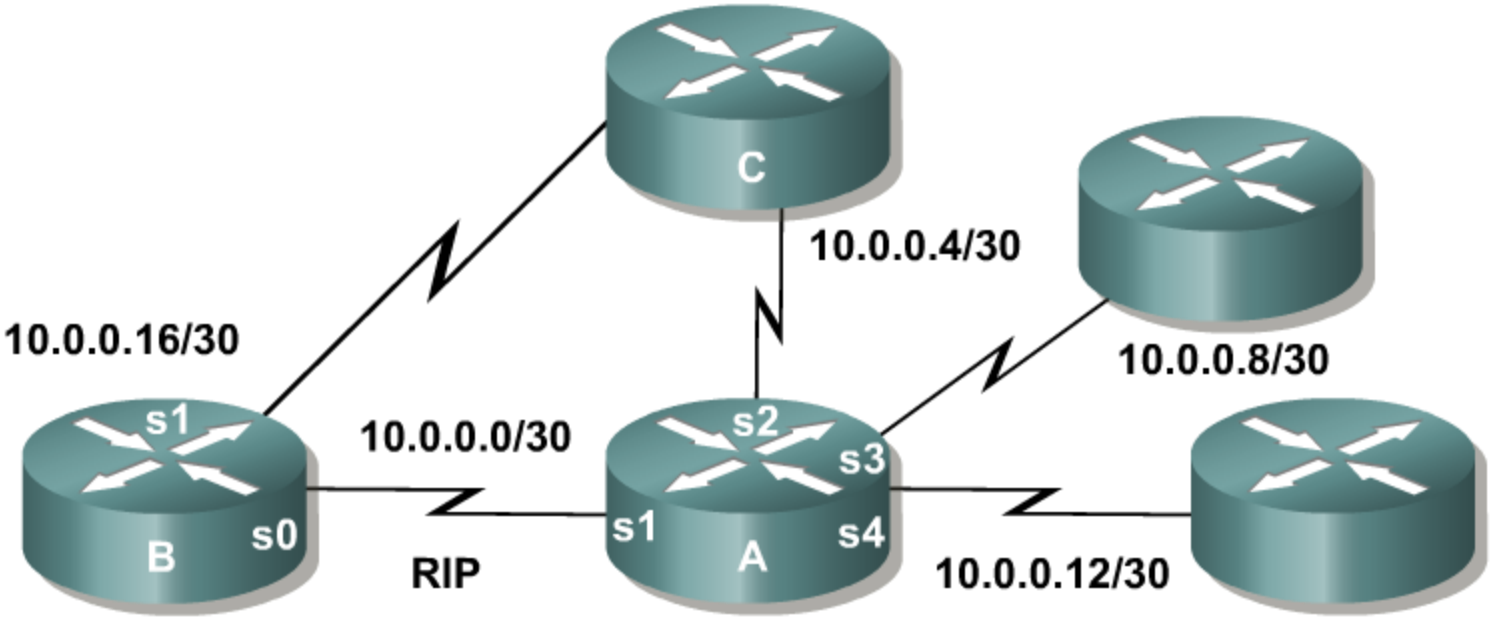
```
C       10.0.0.0
C       10.0.0.8
R       10.0.0.0/14
O IA    10.8.0.0/14
```

**C Routing Table**

```
C       10.0.0.8
O E2    10.0.0.0/14
O       10.8.0.0
O       10.9.0.0
O       10.10.0.0
O       10.11.0.0
```

# Controlling Routing Update Traffic

# Using the `passive-interface` Command




**Router B Configuration**

```
router rip
   network 10.0.0.0
   passive-interface s1
```

**Router A Configuration**

```
router rip
   network 10.0.0.0
   passive-interface default
   no passive-interface s1
```

# Route Maps

Route maps are similar to a scripting language for these reasons:

- They work like a more sophisticated access list:
  - Top-down processing
  - Once there is a match, leave the route map
- Lines are sequence-numbered for easier editing:
  - Insertion of lines
  - Deletion of lines
- Route maps are named rather than numbered for easier documentation.
- Match criteria and set criteria can be used, similar to the “if, then” logic in a scripting language.

# Route Map Applications

The common uses of route maps are as follows:

- Redistribution route filtering:

  A more sophisticated alternative to distribute lists

- Policy-based routing:

  The ability to determine routing policy based on criteria other than the destination network

- BGP policy implementation:

  The primary tool for defining BGP routing policies

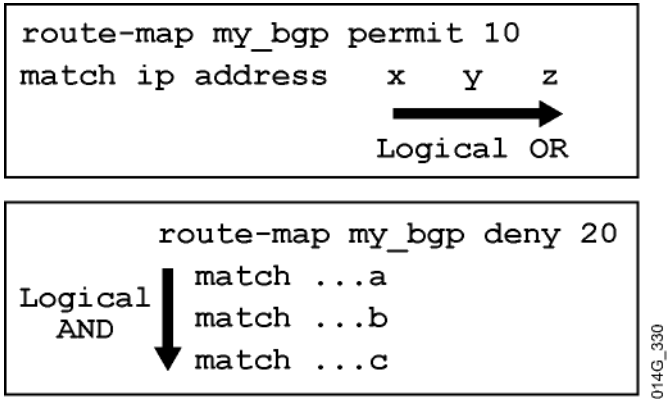
# Route Map Operation

- A list of statements composes a route map.
- The list is processed top-down like an access list.
- The first match found for a route is applied.
- The sequence number is used for inserting or deleting specific route map statements.

```
route-map my_bgp permit 10
       { match statements }
       { match statements }
       { set statements }
       { set statements }
route-map my_bgp deny 20
       ::            ::   ::
       ::            ::   ::
route-map my_bgp permit 30
       ::            ::   ::
       ::            ::   ::
```

# Route Map Operation (Cont.)

- The match statement may contain multiple references.
- Multiple match criteria in the same line use a logical OR.
- At least one reference must permit the route for it to be a candidate for redistribution.




- Each vertical match uses a logical AND.
- All match statements must permit the route for it to remain a candidate for redistribution.
- Route map permit or deny determines if the candidate will be redistributed.

# route-map Commands

```
router(config)#
```

```
route-map map-tag [permit | deny] [sequence-number]
```

- Defines the route map conditions

```
router(config-route-map)#
```

```
match {conditions}
```

- Defines the conditions to match

```
router(config-route-map)#
```

```
set {actions}
```

- Defines the action to be taken on a match

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

```
redistribute protocol [process id] route-map map-tag
```

- Allows for detailed control of routes being redistributed into a routing protocol

# The `match` Command

- The **`match`** commands specify criteria to be matched.
- The associated route map statement permits or denies the matching routes.

```
router(config-route-map)#
```

```
match {options}

  options :
  ip address ip-access-list
  ip route-source ip-access-list
  ip next-hop ip-access-list
  interface type number
  metric metric-value
  route-type [external | internal | level-1 | level-2 |local]
  …
```

# The `match` commands

| Command | Description |
|---|---|
| `match community` | Matches a BGP community |
| `match interface` | Matches any routes that have the next hop out of one of the interfaces specified |
| `match ip address` | Matches any routes that have a destination network number address that is permitted by a standard or extended ACL |
| `match ip next-hop` | Matches any routes that have a next-hop router address that is passed by one of the ACLs specified |
| `match ip route-source` | Matches routes that have been advertised by routers and access servers at the address that is specified by the ACLs |
| `match length` | Matches based on the layer 3 length of a packet |
| `match metric` | Matches routes with the metric specified |
| `match route-type` | Matches routes of the specified type |
| `match tag` | Matches tag of a route |

# The `set` Command

- The **`set`** commands modify matching routes.
- The command modifies parameters in redistributed routes.

```
router(config-route-map)#
```

```
set {options}
      options :
      metric metric-value
      metric-type [type-1 | type-2 | internal | external]
      level [level-1 | level-2 | level-1-2 |stub-area | backbone]
      ip next-hop next-hop-address
```

# The `set` commands

| Command | Description |
|---|---|
| `set as-path` | Modifies an AS path for BGP routes |
| `set automatic-tag` | Computes automatically the tag value |
| `set community` | Sets the BGP communities attribute |
| `set default interface` | Indicates where to output packets that pass a match clause of a route map for policy routing and have no explicit route to the destination |
| `set interface` | Indicates where to output packets that pass a match clause of a route map for policy routing |
| `set ip default next-hop` | Indicates where to output packets that pass a match clause of a route map for policy routing and for which the Cisco IOS software has no explicit route to a destination |
| `set ip next-hop` | Indicates where to output packets that pass a match clause of a route map for policy routing |
| `set level` | Indicates where to import routes for IS-IS and OSPF |
| `set local-preference` | Specifies a BGP local preference value |
| `set metric` | Sets the metric value for a routing protocol |
| `set metric-type` | Sets the metric type for the destination routing protocol |
| `set tag` | Sets tag value for destination routing protocol |
| `set weight` | Specifies the BGP weight value |

# Route Maps and Redistribution Commands

```
Router(config)# router ospf 10
Router(config-router)# redistribute rip route-map redis-rip
```

- Routes matching either access list 23 or 29 are redistributed with an OSPF cost of 500, external type 1.
- Routes permitted by access list 37 are not redistributed.
- All other routes are redistributed with an OSPF cost metric of 5000, external type 2.

```
Router(config)#
route-map redis-rip permit 10
match ip address 23  29
set metric 500
set metric-type type-1

route-map redis-rip deny 20
match ip address 37

route-map redis-rip permit 30
set metric 5000
set metric-type type-2
```

```
Router(config)#
access-list 23 permit 10.1.0.0 0.0.255.255
access-list 29 permit 172.16.1.0 0.0.0.255
access-list 37 permit 10.0.0.0 0.255.255.255
```

# Modifying Administrative Distance

```
Router(config-router)#
 distance administrative distance [address wildcard-mask
[access-list-number | name]]
```

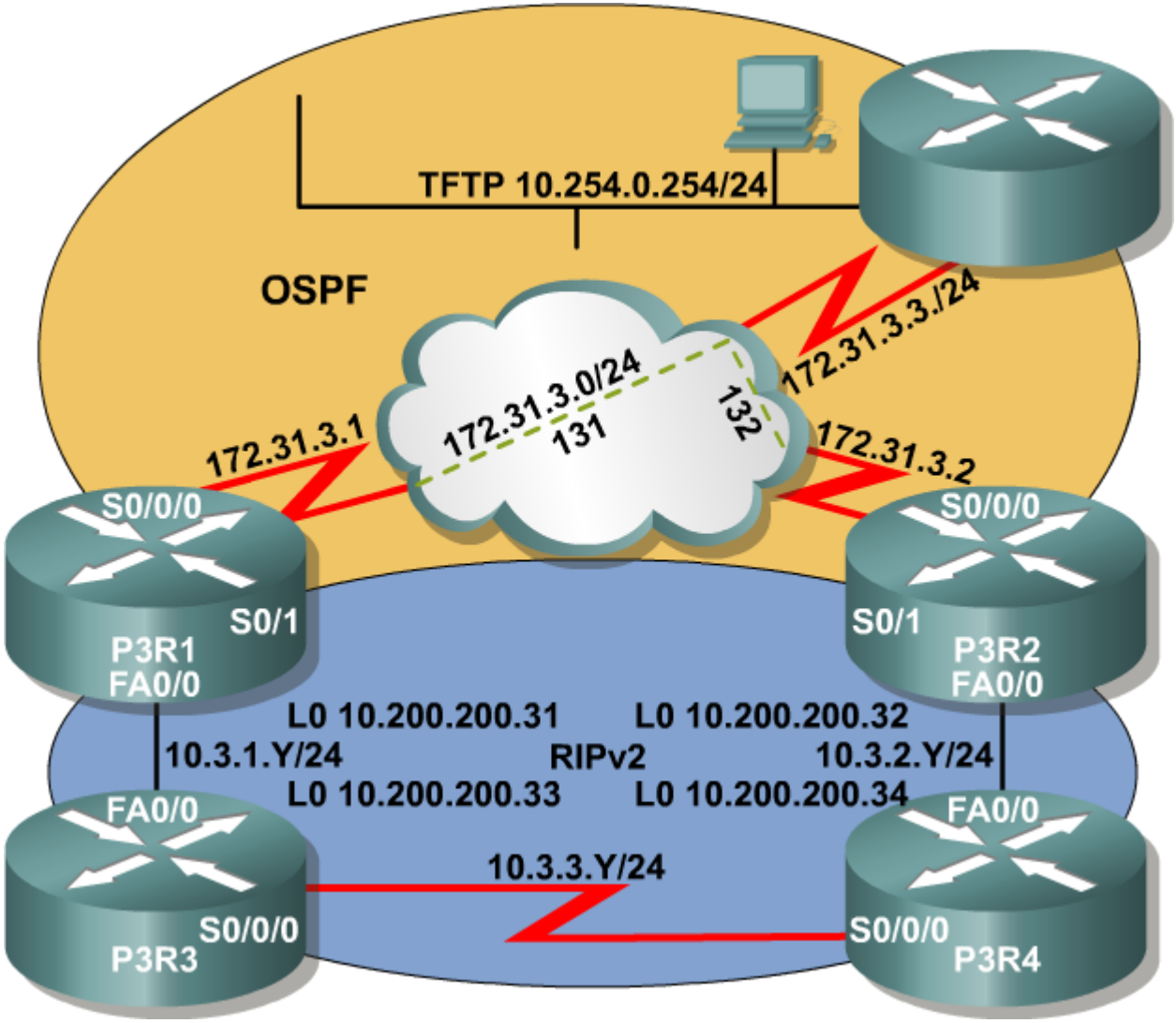
- Used for all protocols except EIGRP and BGP redistribution

```
Router(config-router)#
distance eigrp internal-distance external-distance
```

- Used for EIGRP

# Examples

# Example: Redistribution Using Administrative Distance



# Example: Redistribution Using Administrative Distance (Cont.)

Router P3R1

```
router ospf 1
 redistribute rip metric 10000 metric-type 1 subnets
 network 172.31.0.0 0.0.255.255 area 0
!
router rip
 version 2
 redistribute ospf 1 metric 5
 network 10.0.0.0
 no auto-summary
```

Router P3R2

```
router ospf 1
 redistribute rip metric 10000 metric-type 1 subnets
 network 172.31.3.2 0.0.0.0 area 0
!
router rip
 version 2
 redistribute ospf 1 metric 5
 network 10.0.0.0
 no auto-summary
```

# Example: Redistribution Using Administrative Distance (Cont.)

**With OSPF and RIP running:**


```
P3R2#show ip route
<Output Omitted>

Gateway of last resort is not set

     172.31.0.0/24 is subnetted, 7 subnets
O       172.31.55.0 [110/2342] via 172.31.3.3, 00:09:46, Serial0/0/0
C       172.31.3.0 is directly connected, Serial0/0/0
O       172.31.2.0 [110/1562] via 172.31.3.3, 00:09:46, Serial0/0/0
10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
O E1    10.3.1.0/24 [110/10781] via 172.31.3.1, 00:09:47, Serial0/0/0
O E1    10.3.3.0/24 [110/10781] via 172.31.3.1, 00:04:51, Serial0/0/0
C       10.3.2.0/24 is directly connected, fastethernet0/0
O E1    10.200.200.31/32 [110/10781] via 172.31.3.1, 00:09:48, Serial0/0/0
O E1    10.200.200.34/32 [110/10781] via 172.31.3.1, 00:04:52, Serial0/0/0
C       10.200.200.32/32 is directly connected, Loopback0
O E1    10.200.200.33/32 [110/10781] via 172.31.3.1, 00:04:52, Serial0/0/0
O E2    10.254.0.0/24 [110/50] via 172.31.3.3, 00:09:48, Serial0/0/0
```

- **P3R2 includes suboptimal paths and loops.**

# Example: Redistribution Using Administrative Distance (Cont.)

```
hostname P3R1
!
router ospf 1
 redistribute rip metric 10000 metric-type 1
subnets
 network 172.31.0.0 0.0.255.255 area 0
 distance 125 0.0.0.0 255.255.255.255 64
!
router rip
 version 2
 redistribute ospf 1 metric 5
 network 10.0.0.0
 no auto-summary
!
access-list 64 permit 10.3.1.0 0.0.0.255
access-list 64 permit 10.3.3.0 0.0.0.255
access-list 64 permit 10.3.2.0 0.0.0.255
access-list 64 permit 10.200.200.31
access-list 64 permit 10.200.200.34
access-list 64 permit 10.200.200.32
access-list 64 permit 10.200.200.33
```

```
hostname P3R2
!
router ospf 1
redistribute rip metric 10000 metric-type 1
subnets
 network 172.31.3.2 0.0.0.0 area 0
 distance 125 0.0.0.0 255.255.255.255 64
!
router rip
 version 2
 redistribute ospf 1 metric 5
 network 10.0.0.0
 no auto-summary
!
access-list 64 permit 10.3.1.0 0.0.0.255
access-list 64 permit 10.3.3.0 0.0.0.255
access-list 64 permit 10.3.2.0 0.0.0.255
access-list 64 permit 10.200.200.31
access-list 64 permit 10.200.200.34
access-list 64 permit 10.200.200.32
access-list 64 permit 10.200.200.33
```

# Example: Redistribution Using Administrative Distance (Cont.)

```
hostname P3R1
!
router ospf 1
 redistribute rip metric 10000 metric-type 1
subnets
 network 172.31.0.0 0.0.255.255 area 0
 distance 125 0.0.0.0 255.255.255.255 64
!
router rip
 version 2
 redistribute ospf 1 metric 5
 network 10.0.0.0
 no auto-summary
!
access-list 64 permit 10.3.1.0 0.0.0.255
access-list 64 permit 10.3.3.0 0.0.0.255
access-list 64 permit 10.3.2.0 0.0.0.255
access-list 64 permit 10.200.200.31
access-list 64 permit 10.200.200.34
access-list 64 permit 10.200.200.32
access-list 64 permit 10.200.200.33
```

```
hostname P3R2
!
router ospf 1
redistribute rip metric 10000 metric-type 1
subnets
 network 172.31.3.2 0.0.0.0 area 0
 distance 125 0.0.0.0 255.255.255.255 64
!
router rip
 version 2
 redistribute ospf 1 metric 5
 network 10.0.0.0
 no auto-summary
!
access-list 64 permit 10.3.1.0 0.0.0.255
access-list 64 permit 10.3.3.0 0.0.0.255
access-list 64 permit 10.3.2.0 0.0.0.255
access-list 64 permit 10.200.200.31
access-list 64 permit 10.200.200.34
access-list 64 permit 10.200.200.32
access-list 64 permit 10.200.200.33
```

# Example: Redistribution Using Administrative Distance (Cont.)

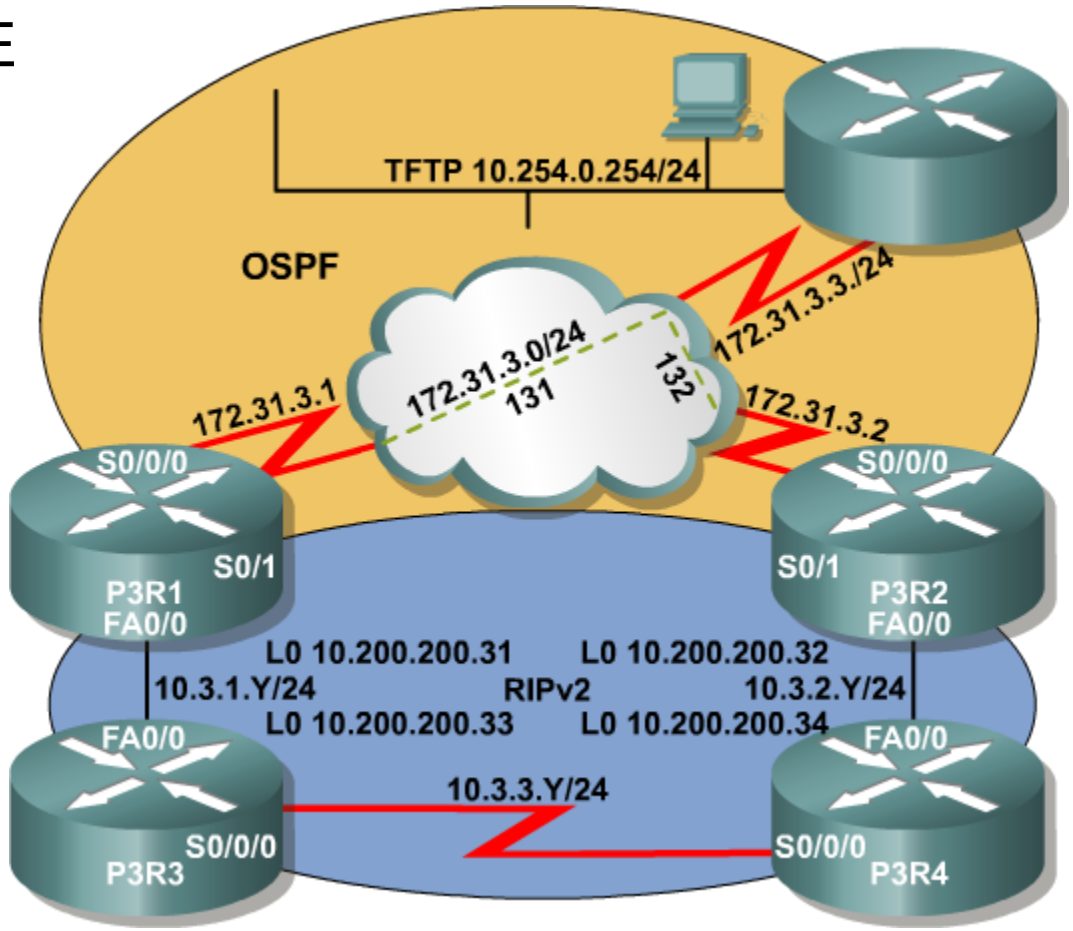
**With OSPF changing administrative distance:**


```
Gateway of last resort is not set

     172.31.0.0/16 is variably subnetted, 8 subnets, 2 masks
O       172.31.55.4/32 [110/781] via 172.31.33.4, 00:00:01, Serial0/0/0
C       172.31.33.0/24 is directly connected, Serial0/0/0
O       172.31.33.1/32 [110/1562] via 172.31.33.4, 00:00:01, Serial0/0/0
O       172.31.33.4/32 [110/781] via 172.31.33.4, 00:00:01, Serial0/0/0
O       172.31.44.4/32 [110/781] via 172.31.33.4, 00:00:01, Serial0/0/0
O       172.31.22.4/32 [110/781] via 172.31.33.4, 00:00:01, Serial0/0/0
O       172.31.11.4/32 [110/781] via 172.31.33.4, 00:00:03, Serial0/0/0
O       172.31.66.4/32 [110/781] via 172.31.33.4, 00:00:03, Serial0/0/0
     10.0.0.0/8 is variably subnetted, 8 subnets, 2 masks
R       10.3.1.0/24 [120/2] via 10.3.2.4, 00:00:03, FastEthernet0/0
R       10.3.3.0/24 [120/1] via 10.3.2.4, 00:00:03, FastEthernet0/0
C       10.3.2.0/24 is directly connected, FastEthernet0/0
R       10.200.200.31/32 [120/3] via 10.3.2.4, 00:00:04, FastEthernet0/0
R       10.200.200.34/32 [120/1] via 10.3.2.4, 00:00:04, FastEthernet0/0
C       10.200.200.32/32 is directly connected, Loopback0
R       10.200.200.33/32 [120/2] via 10.3.2.4, 00:00:04, FastEthernet0/0
O E2    10.254.0.0/24 [110/50] via 172.31.33.4, 00:00:04, Serial0/0/0
```

- **Router P3R2 prefers RIP routes.**

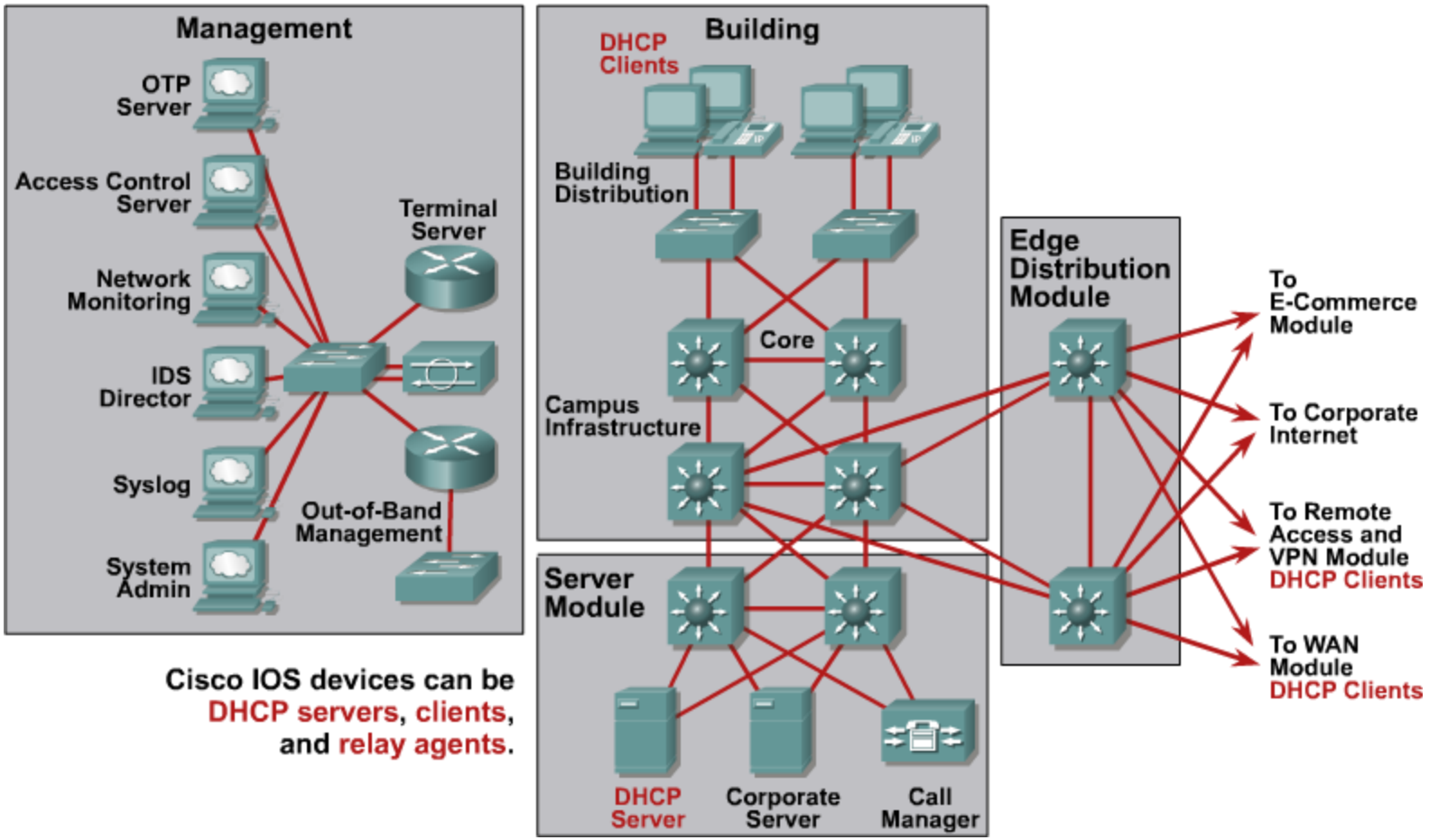
# Know Your Network

- Be very familiar with your network BEFORE implementing redistribution
- Focus on routers with redundant paths
- Make sure no path information is lost when using the **`distance`** command



# Configuring DHCP

# DHCP in an Enterprise Network



# Configuring an Cisco IOS DHCP Server

```
Router(config)#ip dhcp pool [pool name]
```

- Enable a DHCP pool for use by hosts.

```
Router(config-dhcp)#network [network address][subnet mask]
```

- Specify the network and subnet mask of the pool.

```
Router(config-dhcp)#default-router [host address]
```

- Specify the default router for the pool to use.

```
Router(config)#ip dhcp excluded-address low-address high-address
```

- Specify the IP address that should not assign to DHCP clients.

# Optional DHCP Server Commands

```
Router(config-dhcp)#domain-name domain
```

- Specifies the domain name for the client.

```
Router(config-dhcp)#dns-server address
```

- Specifies the IP address of a DNS server that is available to a DHCP client. One is required, but up to eight can be specified.

```
Router(config-dhcp)#netbios-name-server address
```

- Same as DNS, but for WINS.

```
Router(config-dhcp)#lease {days [hours] [minutes] | infinite}
```

- Specifies the duration of the lease. The default is a one-day lease.

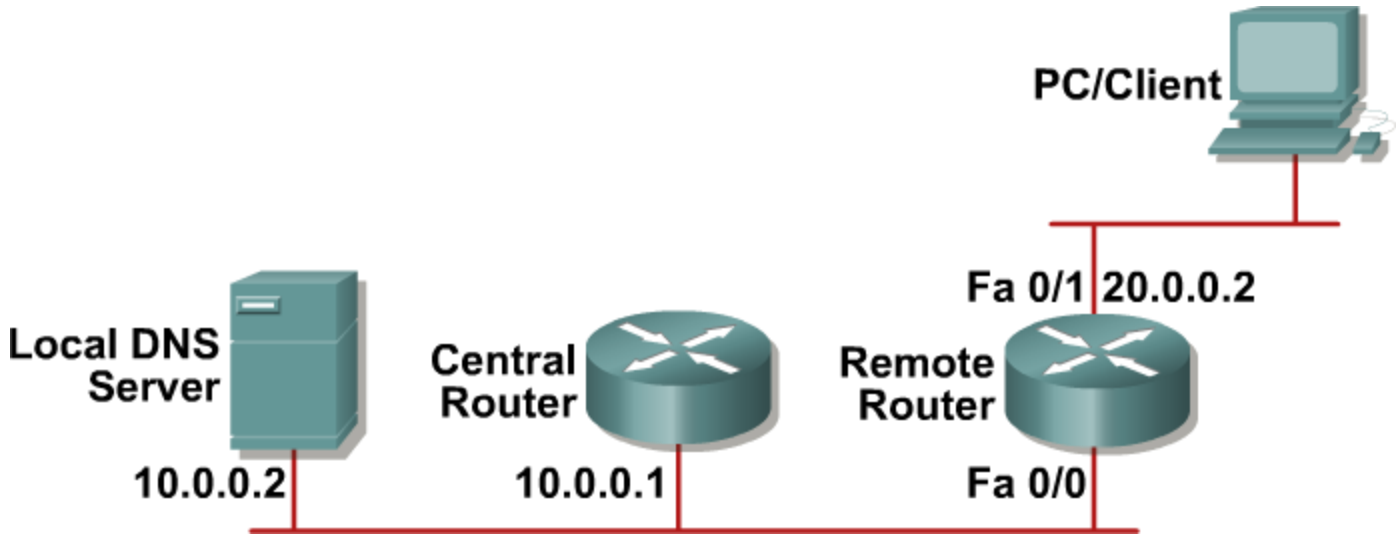
# DHCP Database Command and Configuration

```
Router(config)#ip dhcp database url [timeout seconds | write-
delay seconds]
```

- Configures the database agent and the interval between database updates and database transfers.

```
ipdhcp database ftp://user:passwords@172.16.4.253/router-dhcp write-delay 120
ip dhcp excluded-address 172.16.1.100 172.16.1.103
ip dhcp excluded-address 172.16.2.100 172.16.2.103
ip dhcp pool 0
  network 172.16.0.0/16
  domain-name global.com
  dns-server 172.16.1.102 172.16.2.102
  netbios-name-server 172.16.2.103 172.16.2.103
  default-router 172.16.1.100
```

# Importing and Autoconfiguration

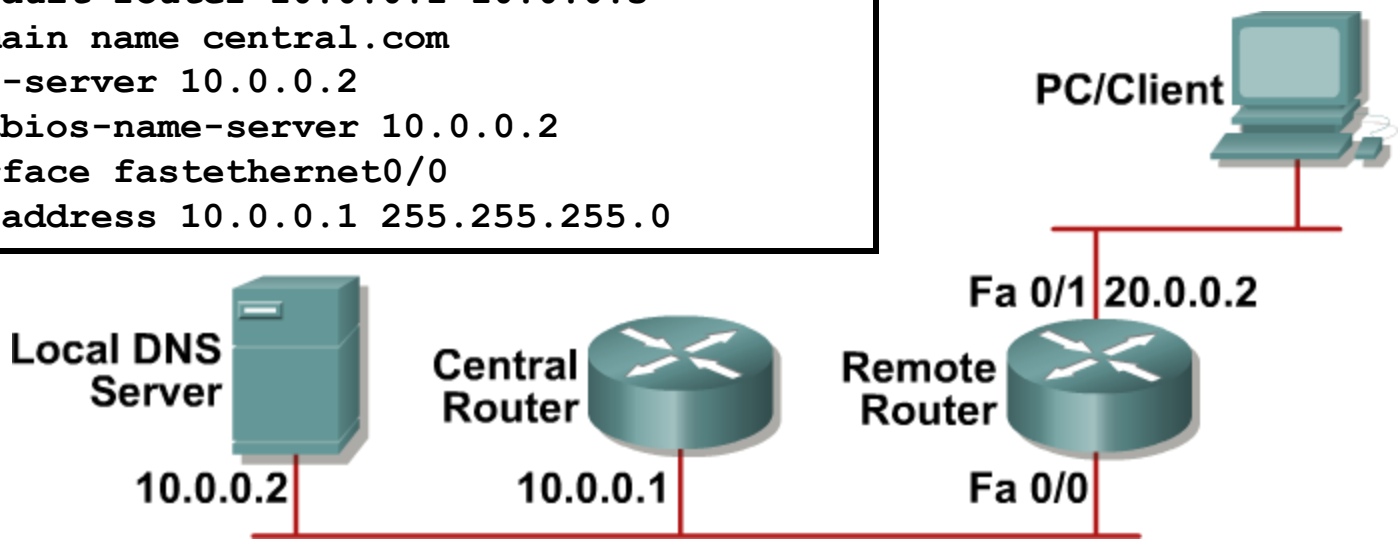



```
Router(config-dhcp)#import all
```

- Used to import DHCP option parameters into DHCP server database. Used for remote DHCP pools.

# Importing and Autoconfiguration (Cont.)

```
ip dhcp-excluded address 10.0.0.1 10.0.0.5
ip dhcp pool central
  network 10.0.0.0 255.255.255.0
  default-router 10.0.0.1 10.0.0.5
  domain name central.com
  dns-server 10.0.0.2
  netbios-name-server 10.0.0.2
interface fastethernet0/0
  ip address 10.0.0.1 255.255.255.0
```

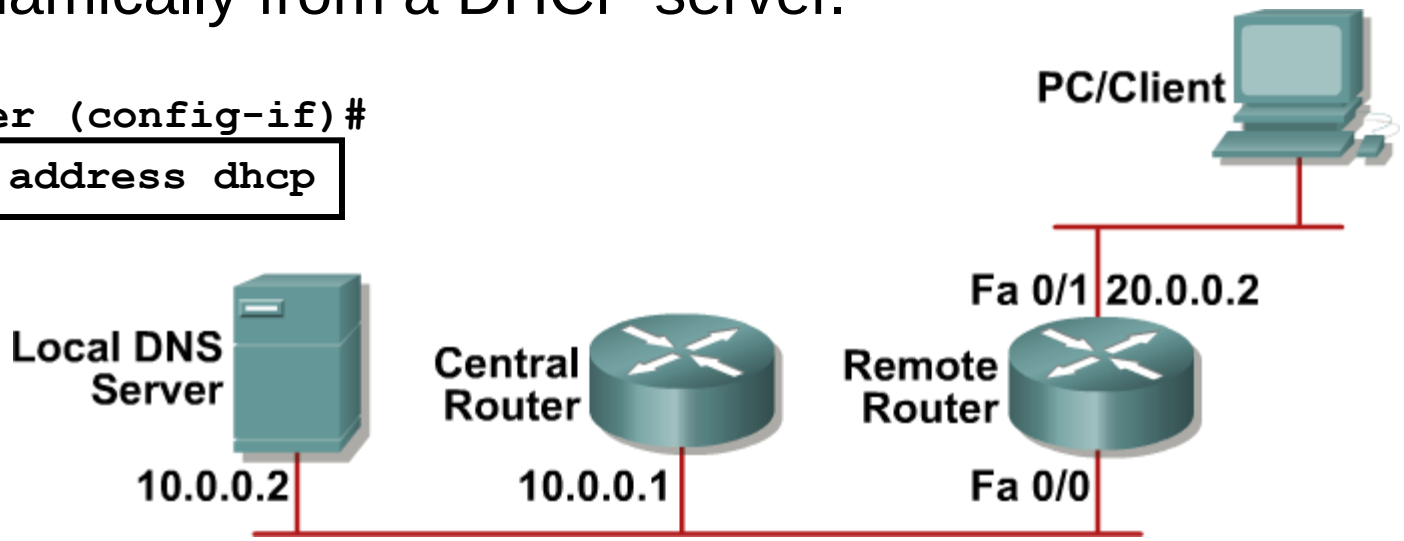



```
ip dhcp-excluded address 20.0.0.2
ip dhcp pool client
  network 20.0.0.0 255.255.255.0
  default-router 20.0.0.2
  import all
interface fastethernet0/0
 ip address dhcp
```

# DHCP Client

Enables an IOS device to obtain an IP address dynamically from a DHCP server.

```
Router (config-if)#
 ip address dhcp
```




```
ip dhcp-excluded address 20.0.0.2
ip dhcp pool client
  network 20.0.0.0 255.255.255.0
  default-router 20.0.0.2
  import all
interface fastethernet0/0
 ip address dhcp
```

# Relay Agent Option Support