Chapter 2:

Configuring the Enhanced Interior Gateway Routing Protocol

- CCNP-RS ROUTE

Chapter 2 Objectives

- Describe the basic operation of EIGRP.
- Plan and implement EIGRP routing.
- Configure and verify EIGRP routing.
- Configure and verify basic EIGRP in an enterprise WAN.
- Configure and verify EIGRP Authentication.
- Describe and configure EIGRP optimization mechanisms; verify and troubleshoot the overall implementation.

Understanding EIGRP Terminology and Operation

EIGRP Capabilities and Attributes

- EIGRP is a Cisco-proprietary distance-vector protocol with link-state features.
- EIGRP features include:
  - Fast convergence
  - Partial updates
  - Multiple network layer support
  - Use of multicast and unicast communication
  - Variable-length subnet masking (VLSM) support
  - Seamless connectivity across all data link layer protocols and topologies
  - By default, don`t automatic route summarization.

EIGRP Terminology

- Neighbor table
- Topology table
- Routing table
- Advertised Distance (AD)
- Feasible Distance (FD)
- Successor
- Feasible successor (FS)
- Passive Versus Active Routes

EIGRP Tables

- **Neighbor table**
  - Contains EIGRP neighbor addresses and the interface through which they can be reached.
- **Topology table**
  - Contains all destinations advertised by neighboring routers.
- **Routing table**
  - Contains EIGRP successor routes.

AD versus FD

- **Advertised Distance (AD)**
  - Advertised distance (AD), also referred to as the Reported Distance, is the cost between the next-hop router and the destination.
- **Feasible Distance (FD)**
  - Feasible distance (FD) is the cost between the local router and the next-hop router plus the next-hop router's AD to the destination network.

Successor and Feasible Successor

- **Successor**
  - A successor is a neighboring router that has a least-cost path to a destination (the lowest FD) that is guaranteed not to be part of a routing loop.
  - Successor routes are offered to the routing table to be used for forwarding packets.
  - Multiple successors can exist if they have the same FD.
- **Feasible successor (FS)**
  - A feasible successor is a neighbor that is closer to the destination, but it is not the least-cost path.
  - A feasible successor ensures a loop-free topology because it must have an *AD less than the FD* of the current successor route.
  - Feasible successors are selected at the same time as successors but are kept in the topology table as backups to the successor routes.
  - The topology table can maintain multiple feasible successors for a destination.

Passive versus Active Routes

- **Passive Route**
  - A route is considered *passive* when the router is not performing recomputation on that route.
  - *Passive* is the operational, stable state.
- **Active route**
  - A route is *active* when it is undergoing recomputation.

Key EIGRP Technologies

- **Reliable Transport Protocol (RTP)**
  - Responsible for guaranteed, ordered delivery of EIGRP packets to all neighbors.
- **Neighbor discovery/recovery mechanism**
  - Enables EIGRP routers to dynamically learn when their neighbors become unreachable or inoperative by periodically sending small hello packets.
- **Protocol-dependent modules (PDMs)**
  - Responsible for network layer protocol-specific requirements such as IP, IPv6, AppleTalk, and Novell NetWare.
- **DUAL finite-state machine**
  - **Diffusing Update Algorithm (DUAL)** is the routing algorithm that tracks all routes advertised by all neighbors and uses *distance* information, known as the composite metric, to select efficient, loop-free paths to all destinations.

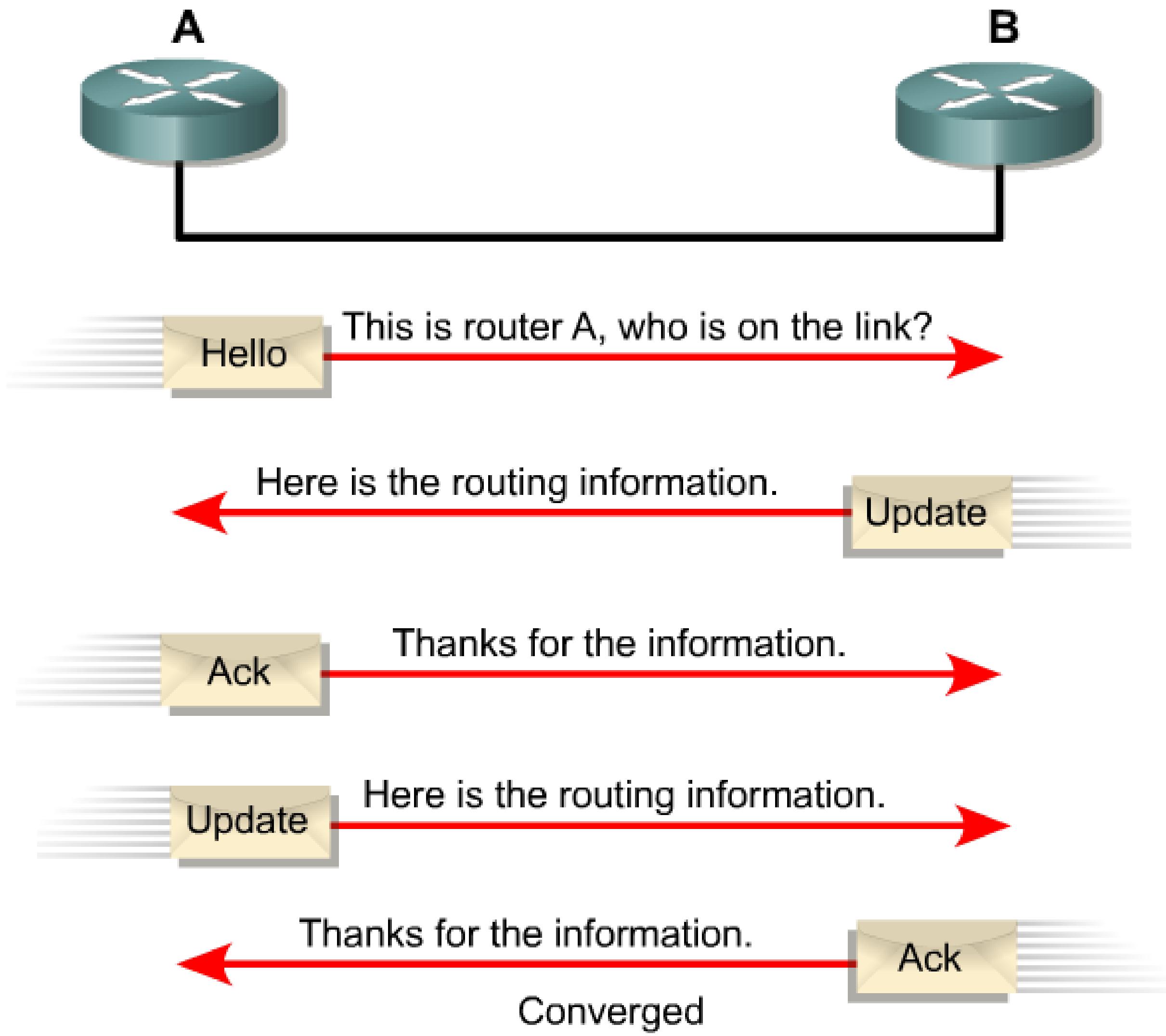
Reliable Transport Protocol

- EIGRP cannot use the services of UDP or TCP since IPX and Appletalk do not use the TCP/IP protocol suite.
- Reliable Transport Protocol (RTP) is the Transport layer protocol uniquely used by EIGRP for the delivery and reception of EIGRP packets.
  - RTP is similar to TCP but is a Cisco proprietary.
- RTP provides reliable or unreliable service as the situation warrants.
  - Reliable packets (Update, Query, Reply) require explicit acknowledgement while unreliable packets (Hello, ACK) do not.

Neighbor Discovery / Recovery

- EIGRP routers actively establish relationships with their neighbors.
- Adjacencies are established using small Hello packets which are sent every 5 or 60 seconds.
  - If a neighbor misses 3 consecutive Hello packets then the route is considered invalid.
  - Default = 15 seconds or 180 seconds.

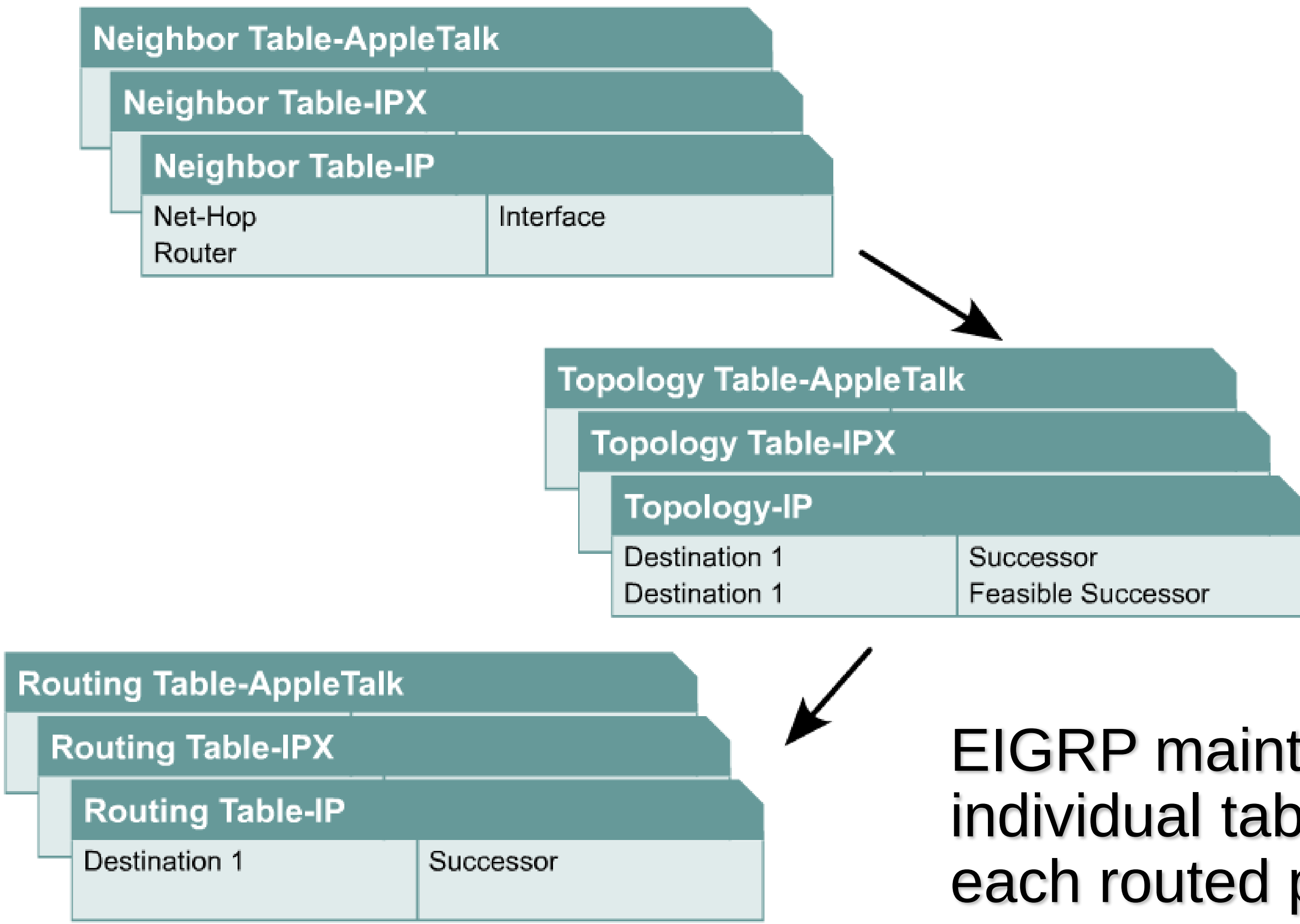
Neighbor Discovery / Recovery



Protocol-Dependent Modules

- Various routed protocols are supported through its PDMs.
  - Provides independence from routed protocols.
  - PDMs are modular, scalable and adaptable.
  - EIGRP can adapt to new or revised routed protocols.
  - PDMs protect EIGRP from painstaking revision.
- Each PDM is responsible for all functions related to its specific routed protocol.

Protocol-Dependent Modules



EIGRP maintains individual tables for each routed protocol.

DUAL finite-state machine

- DUAL uses the Neighbor and Topology tables to calculate route information.
- When a link fails, DUAL looks for a feasible successor in its Neighbor and Topology tables.
  - It compares all routes advertised by neighbors by using a composite metric for each route.
  - Lowest-cost paths are then inserted into the routing table.

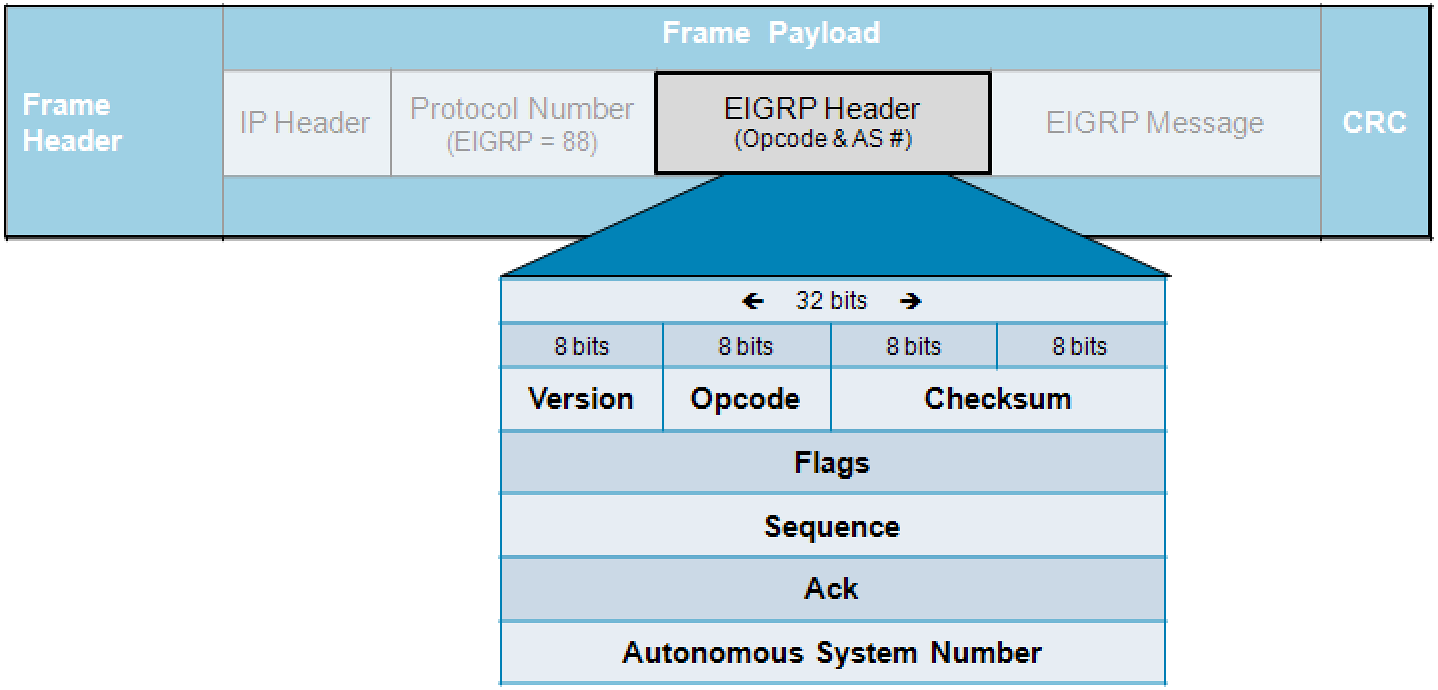
EIGRP Packet

Frame Header	Frame Payload				CRC
	IP Header	Protocol Number (EIGRP = 88)	EIGRP Header	EIGRP Message	
On a LAN, the EIGRP packet is encapsulated in an Ethernet frame with a destination multicast MAC address: **01-00-5E-00-00-0A**	The destination IP address is set to the multicast **224.0.0.10** and the EIGRP protocol field is **88**.		The EIGRP header identifies the type of EIGRP packet and autonomous system number.	The EIGRP message consists of the Type / Length / Value (TLV).	

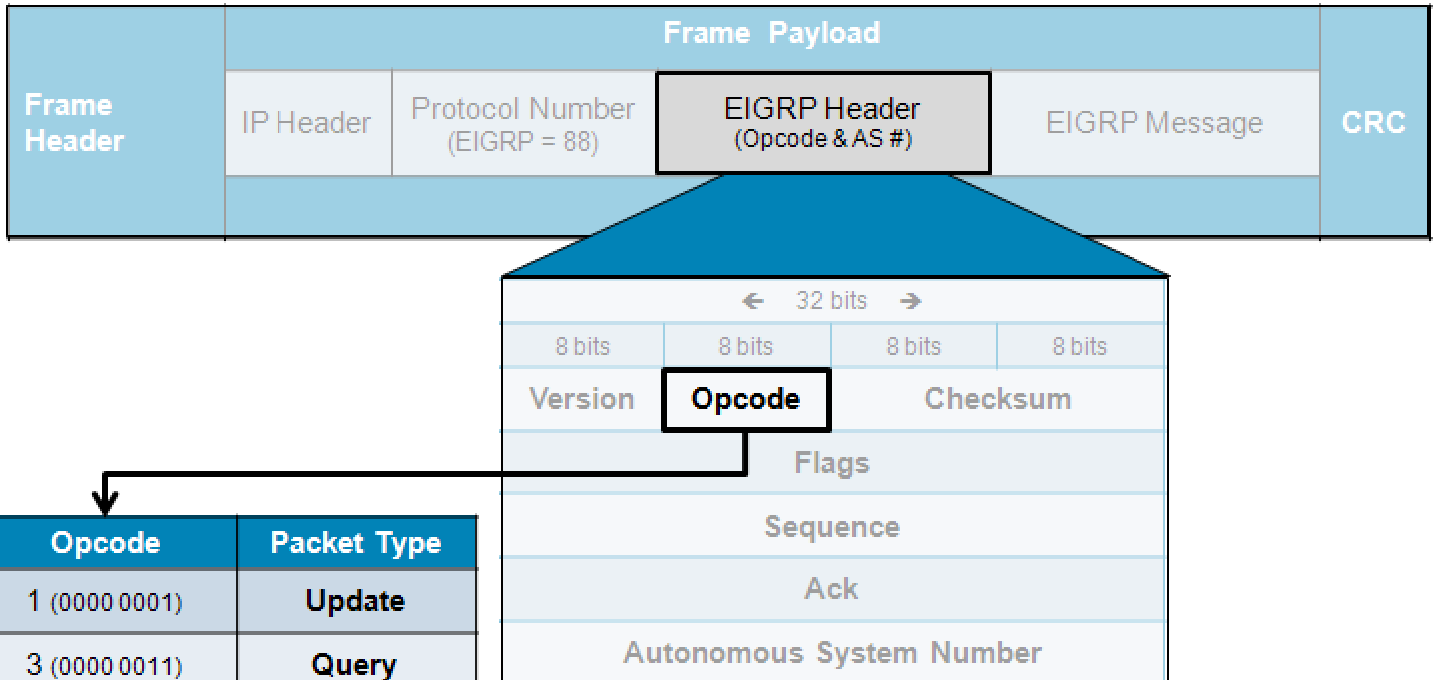
EIGRP Header

- EIGRP uses these 5 packet types to maintain its various tables and establish complex relationships with neighbor routers:
  - Hello
  - Acknowledgment
  - Update
  - Query
  - Reply

EIGRP Header



EIGRP Packet

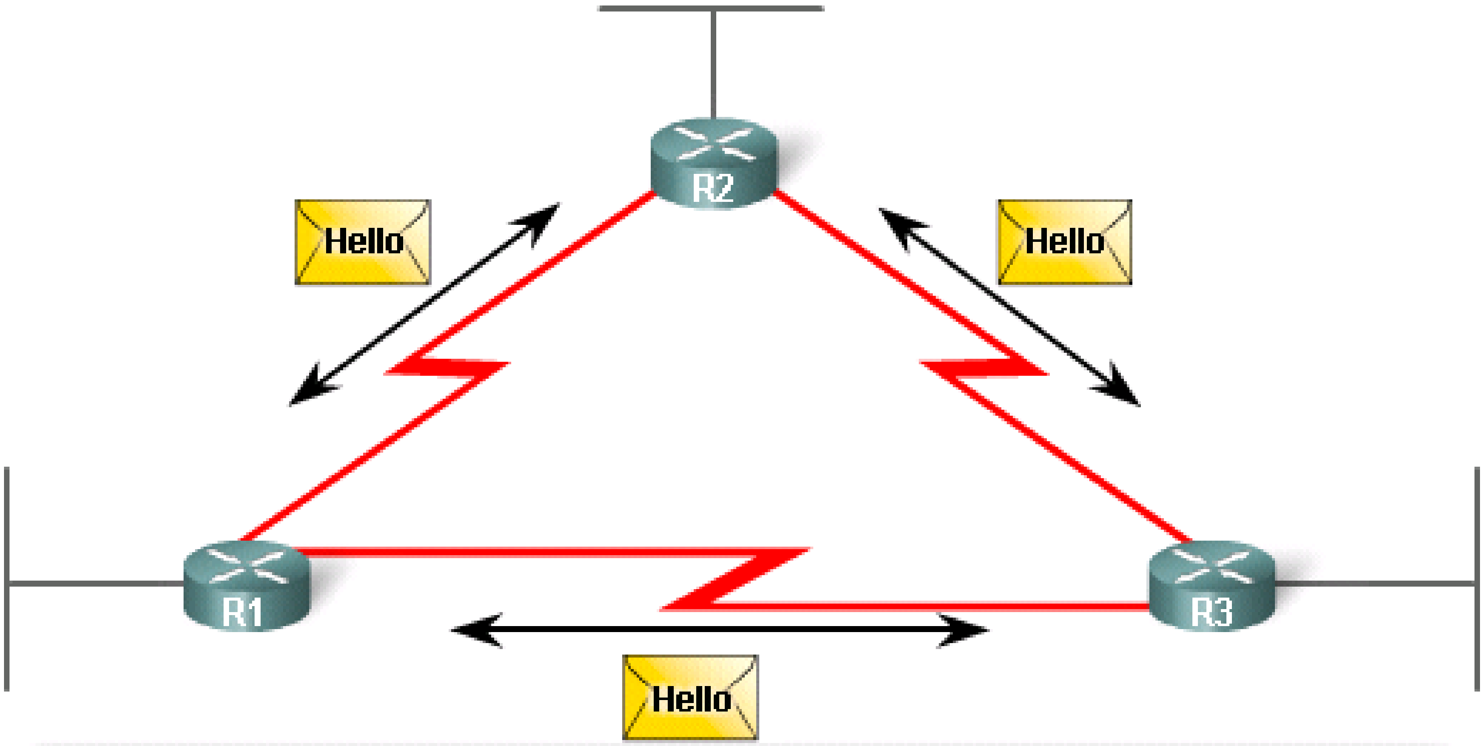


Opcode	Packet Type
1 (0000 0001)	**Update**
3 (0000 0011)	**Query**
4 (0000 0100)	**Reply**
5 (0000 0101)	**Hello**
6 (0000 0110)	**IPX SAP**

Hello Packets

- EIGRP relies on Hello packets to discover, verify, and rediscover neighbor routers.
- EIGRP Hello packets are multicast to 224.0.0.10.
- Hello packets are always sent unreliably and therefore do not require acknowledgment.

EIGRP Hello Packets



Hello packet

- Use to discover neighbors & form adjacencies
- Unreliable so no response required from recipient

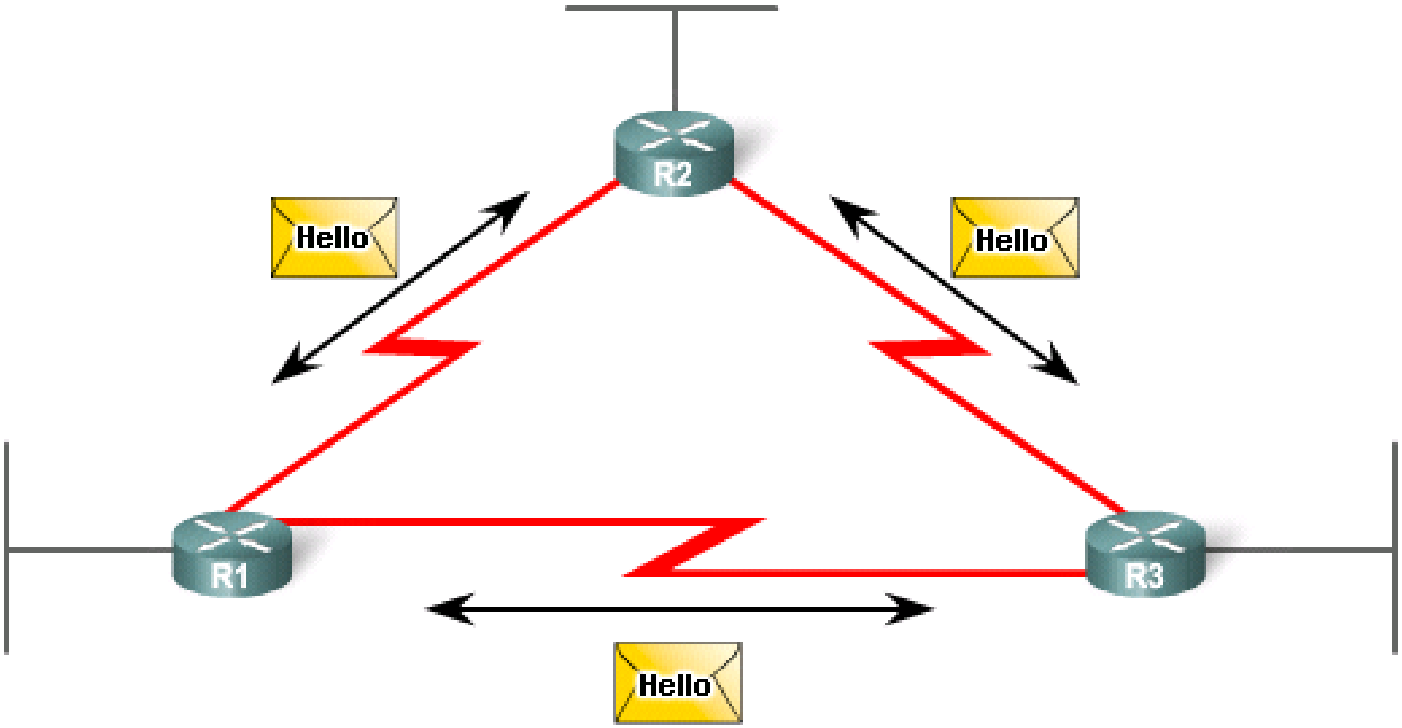
Hello Packets

- Hellos are sent at a fixed (and configurable) interval, called the Hello interval.
  - Hello/Hold timers do not need to match.
  - To reset the Hello interval: no ip hello-interval eigrp as#
- Hello interval depends on the interface’s bandwidth.
  - High bandwidth = 5 seconds
    - Default interval on point-to-point serial links, multipoint circuits with bandwidth greater than T1, and LANs.
  - Low Bandwidth = 60 seconds
    - Default interval on T1 or less multipoint WAN circuits.

Hello Packets

- On hearing Hellos, a router creates a neighbor table and the continued receipt of Hellos maintains the table.
- Holdtime is the maximum amount of allowed time that Hellos are not heard from a neighbor.
  - Three times the Hello Interval:
    - Low Bandwidth (3 x 60 sec.) = 180 seconds
    - High bandwidth (3 x 5 sec.) = 15 seconds

Hello Packets



Bandwidth	Example Link	Default Hello Interval	Default Hold Time
1.544 Mbps	Multipoint Frame Relay	60 seconds	180 seconds
Greater than 1.544 Mbps	T1. T3 Ethernet	5 seconds	15 seconds

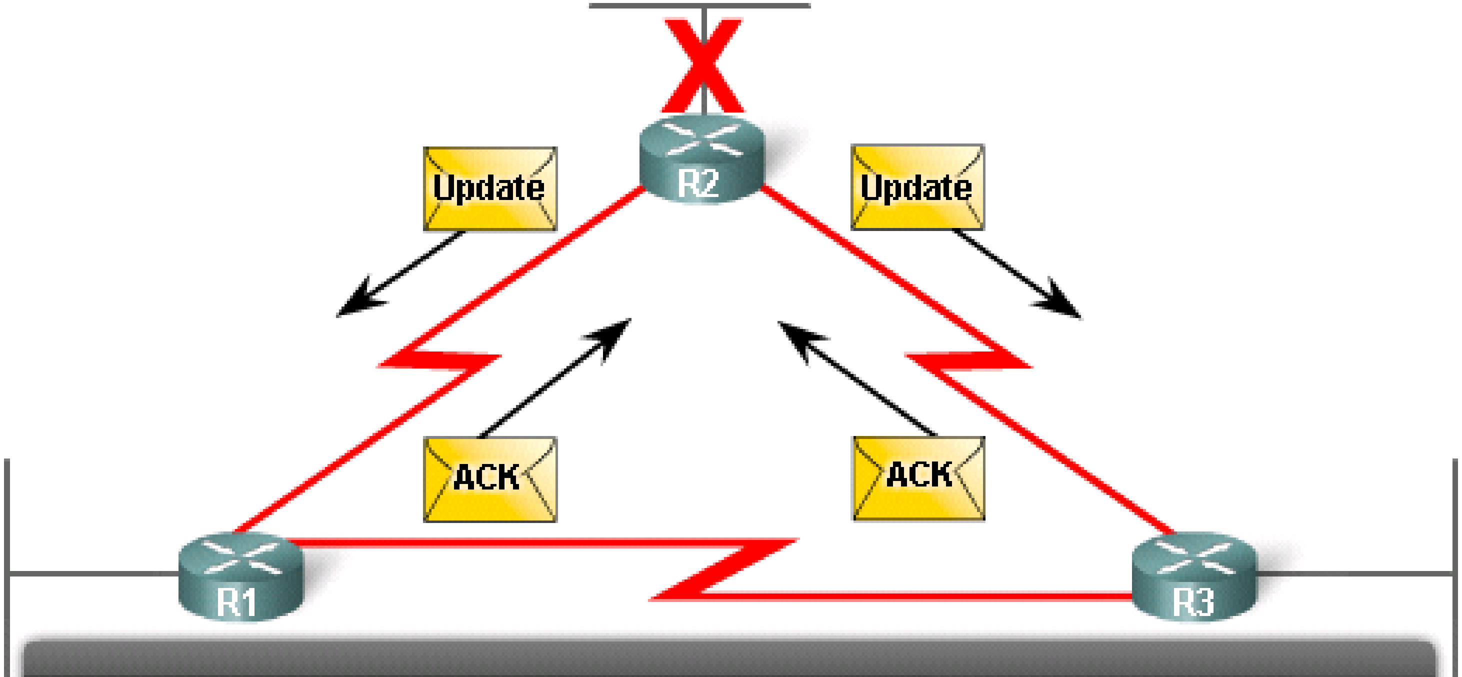
Acknowledgement Packets

- Are used to indicate receipt of any EIGRP packet during a "reliable" (i.e., RTP) exchange.
  - To be reliable, a sender's message must be acknowledged by the recipient.
- Acknowledgment packets are:
  - Dataless Hello packets.
  - Unicast.

Update Packets

- After the local router discovers a new neighbor, update packets are sent to the new neighbor.
- Update packets are also used when a router detects a topology change.
  - The router sends a multicast Update packet to all neighbors, alerting them to the change.
- All Update packets are sent reliably.

Update Packets



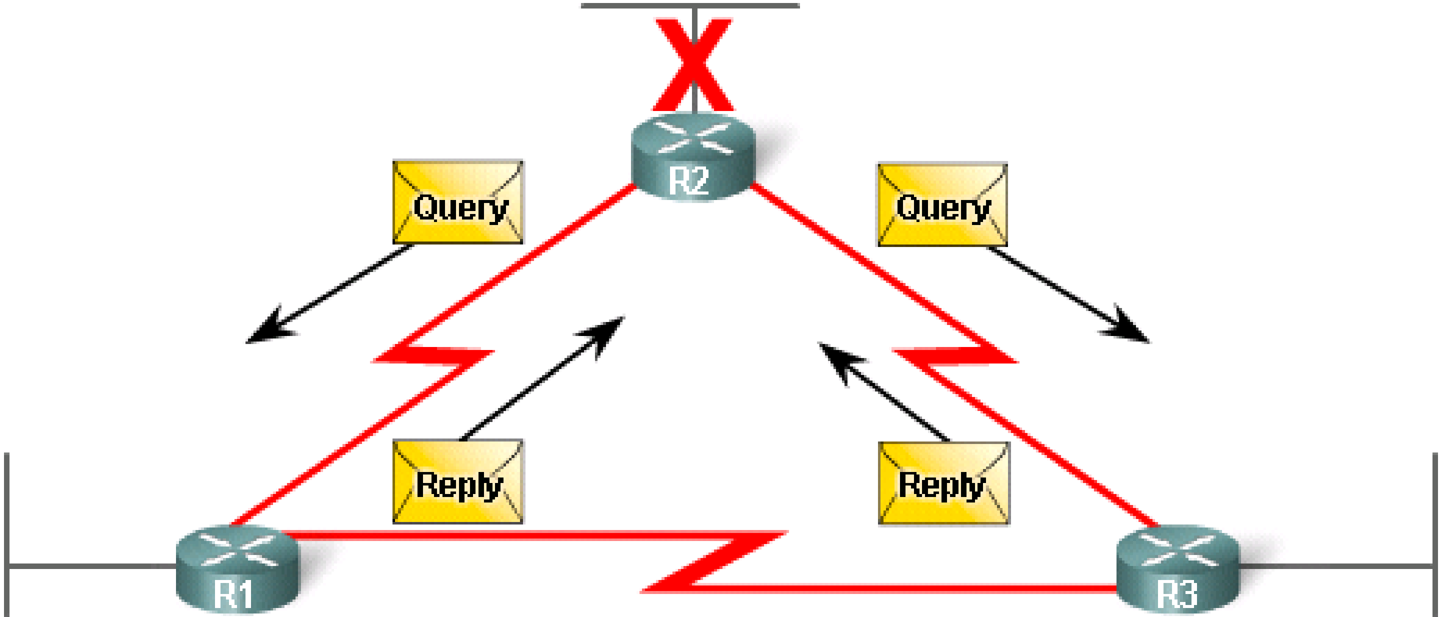
Update packet

- Initially sent after a new neighbor is discovered.
- Sent when a topology change has been detected.

Query and Reply Packets

- Query and Reply packets are sent when a destination has no feasible successors.
- Both packet types are sent reliably.
- A Query packet is multicasted to other EIGRP routers during the route re-computation process.
  - Query packets are always multicast.
- A Reply packet is used to respond to a query to instruct the originator not to recompute the route because feasible successors exist.
  - Reply packets are always unicast.

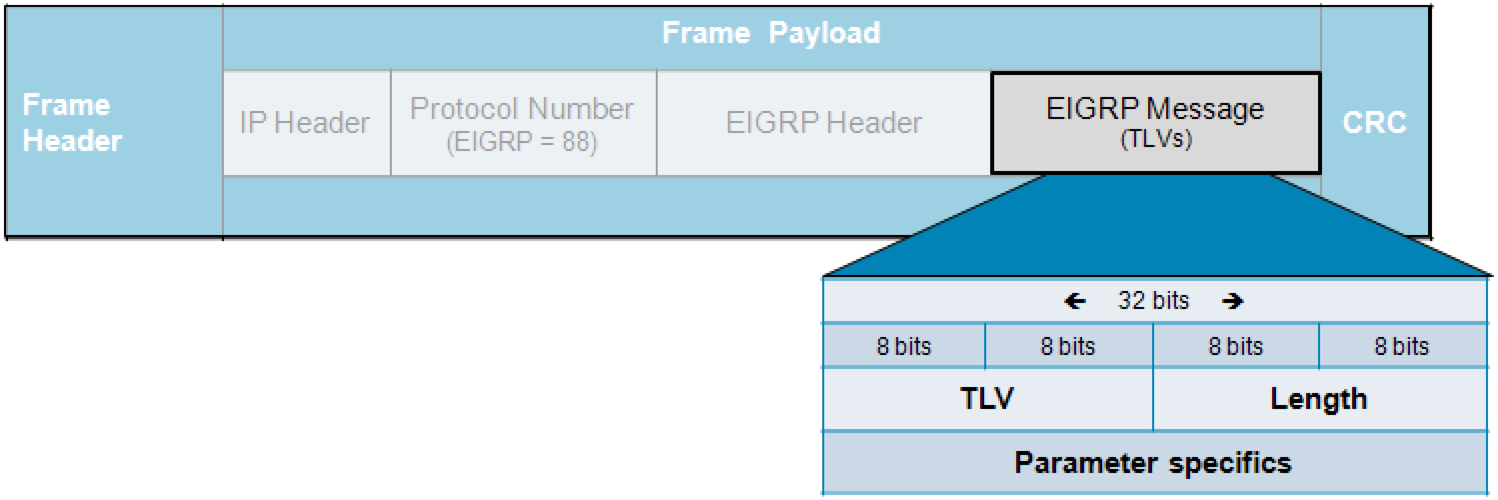
Query and Reply Packets



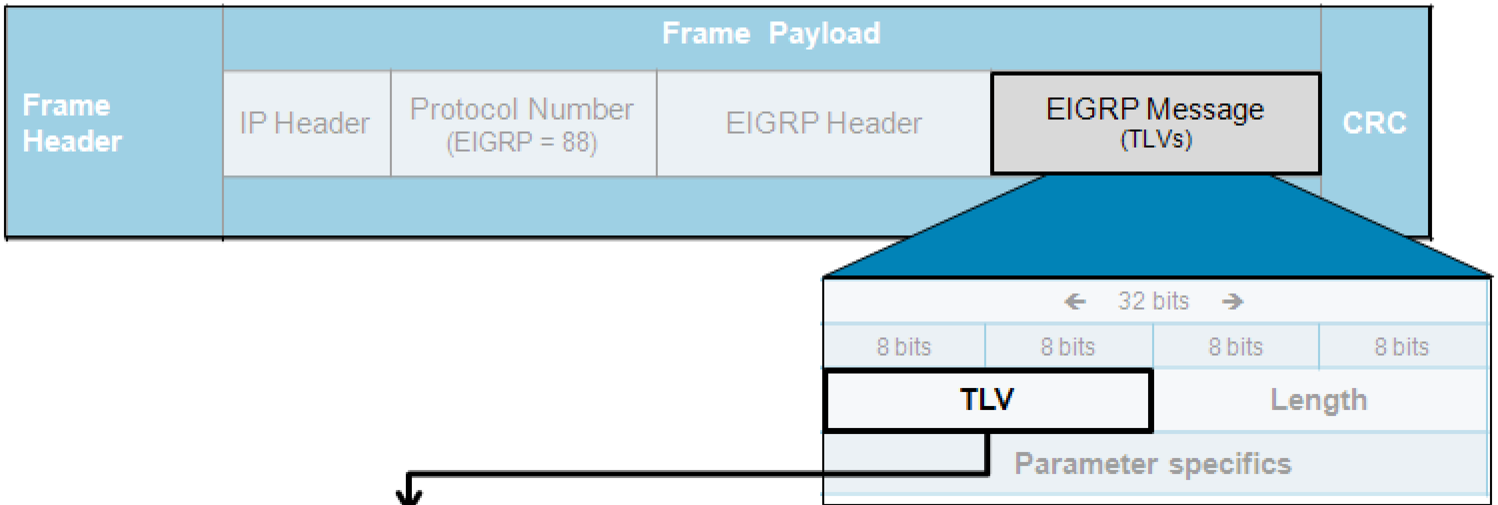
Query packet

- Used by DUAL when searching for networks or other tasks.Reply packet
- Automatically sent in response to Query packet Acknowledgement (ACK) packet
- Automatically sent back when reliable RTP is used

EIGRP Message

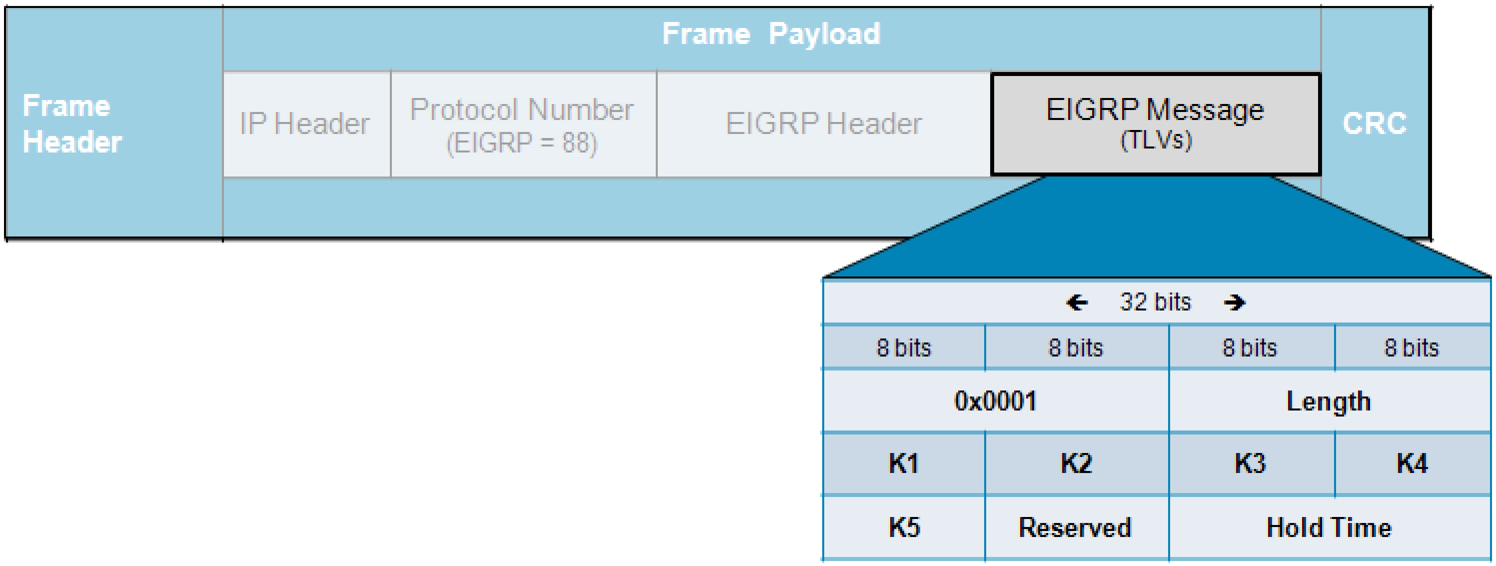


EIGRP Message - TLVs



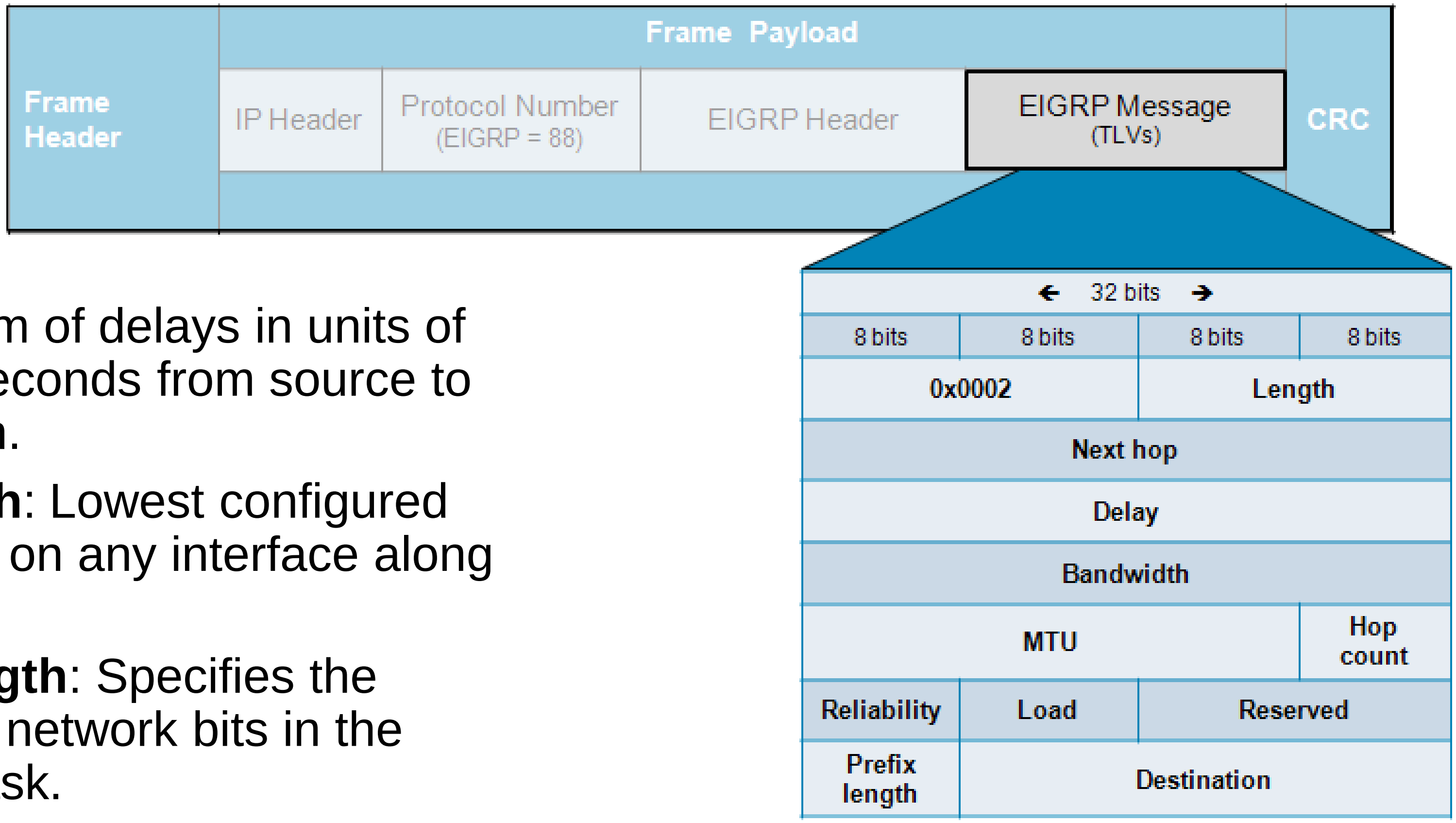
TLV	TLV Type
0x0001	**EIGRP Parameters**
0x0002	**Internal IP Routes**
0x0003	**External IP Routes**

TLV 0x0001 - EIGRP Parameters



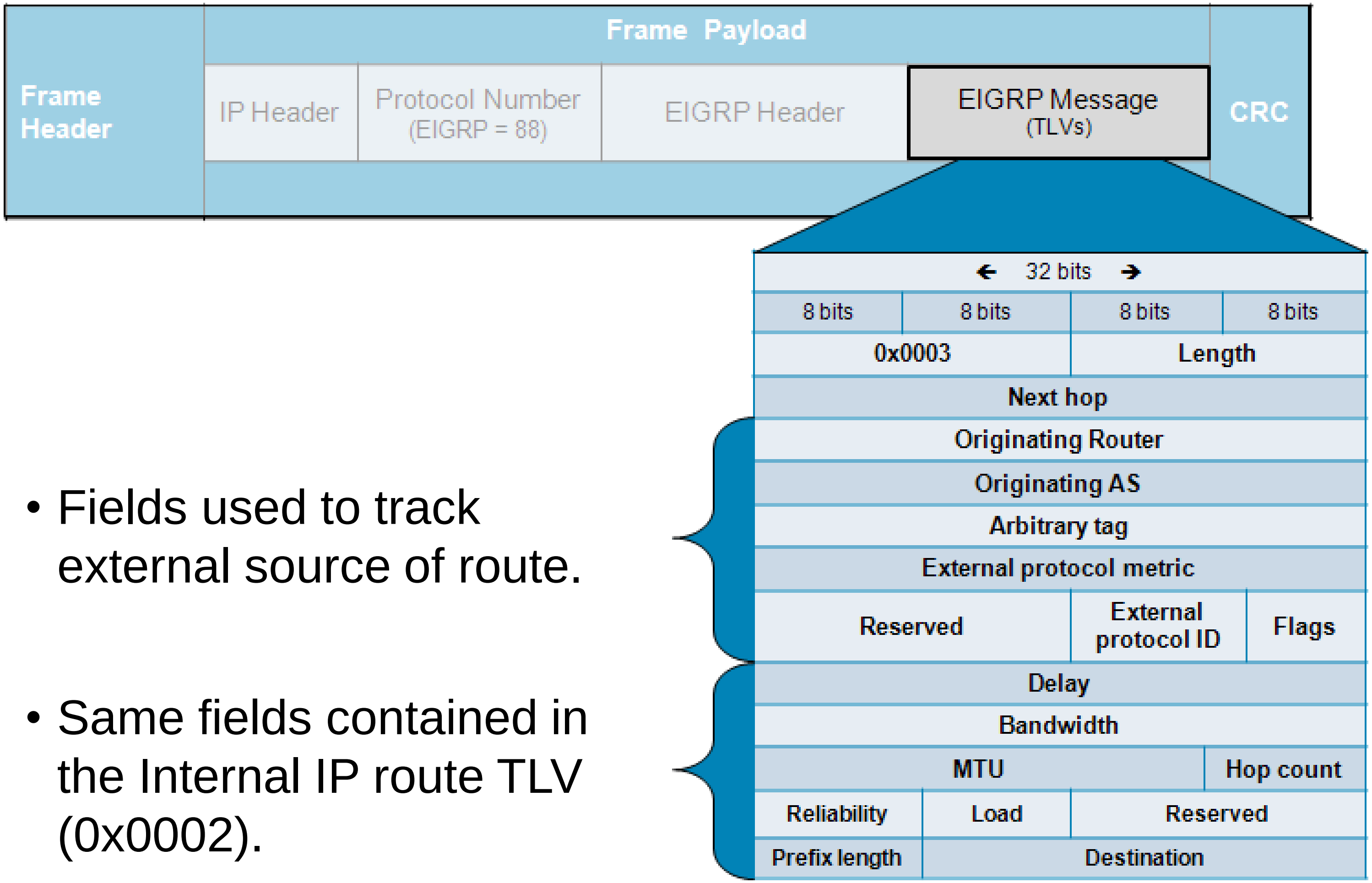
- K values are used to calculate the EIGRP metric.
- The Hold Time advertised by a neighbor is the maximum time a router should wait for any valid EIGRP message sent by that neighbor before declaring it dead.

TLV 0x0002 - Internal IP Routes



- **Delay**: Sum of delays in units of 10 microseconds from source to destination.
- **Bandwidth**: Lowest configured bandwidth on any interface along the route.
- **Prefix length**: Specifies the number of network bits in the subnet mask.
- **Destination**: The destination address of the route.

TLV 0x0003 - External IP Routes



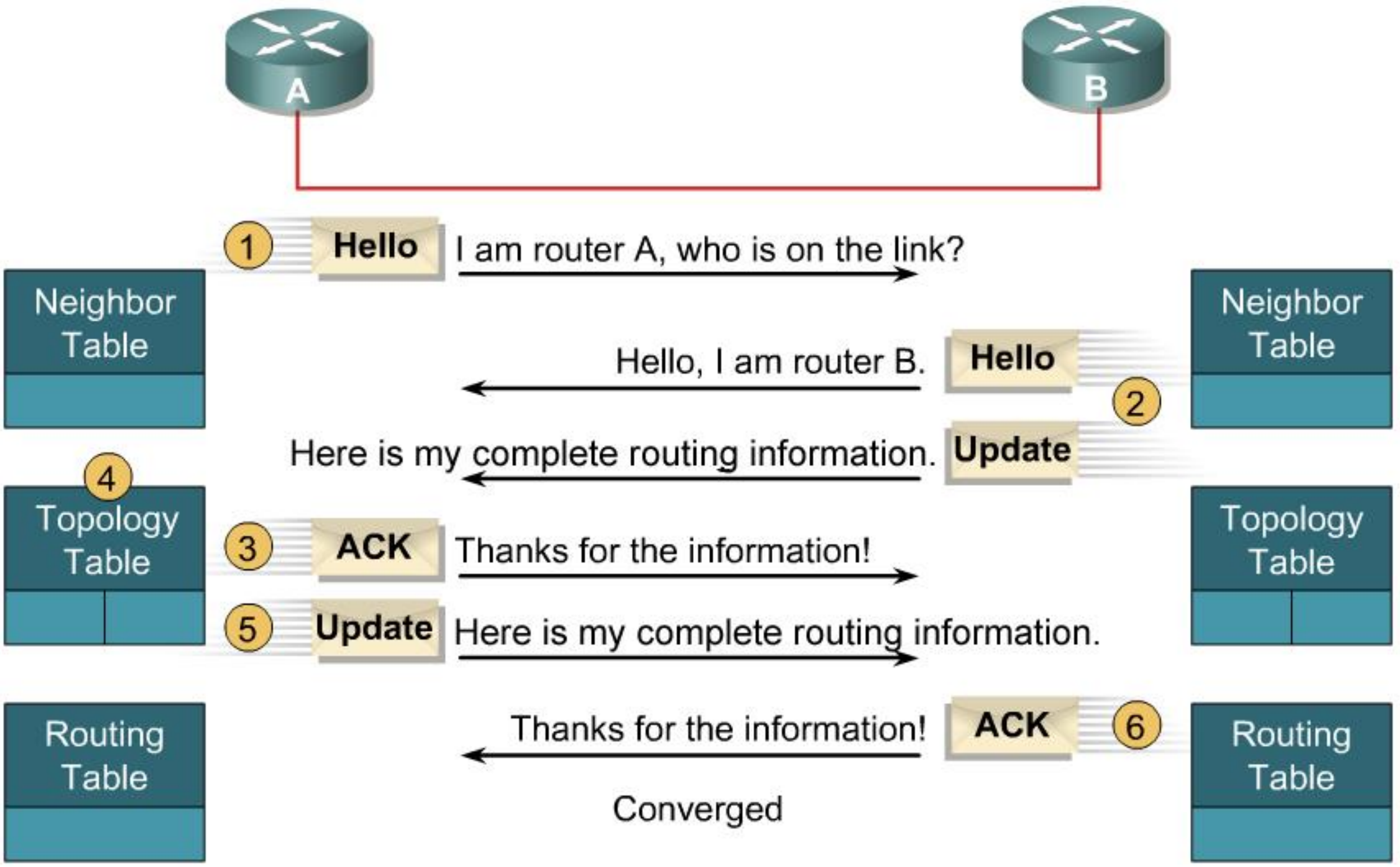
- Fields used to track external source of route.
- Same fields contained in the Internal IP route TLV (0x0002).

- IP external routes are routes which are imported into EIGRP through redistribution of a default route or other routing protocols.

Packet Types

Packet Type	Use
Hello	Used to discover other EIGRP routers in the network.
Acknowledgement	Used to acknowledge the receipt of any EIGRP packet.
Update	Convey routing information to known destinations.
Query	Used to get specific information from a neighbor router.
Reply	Used to respond to a query.

Initial Route Discovery



EIGRP Operations

- EIGRP selects primary (successor) and backup (feasible successor) routes and injects those into the topology table.
- The primary (successor) routes are then moved to the routing table.

IP EIGRP Neighbor Table	
Neighbor IP Address	Local router exit interface to neighbor

List of directly connected adjacent EIGRP neighbor routers and the local interface to exit to reach it.

IP EIGRP Topology Table	
Destination 1	FD / AD via each neighbor

List of all routes learned from each EIGRP neighbor and identifies successor routes and feasible successor routes.

IP Routing Table	
Destination 1	Best route

List of the best (successor) routes from the EIGRP topology table and other routing processes.

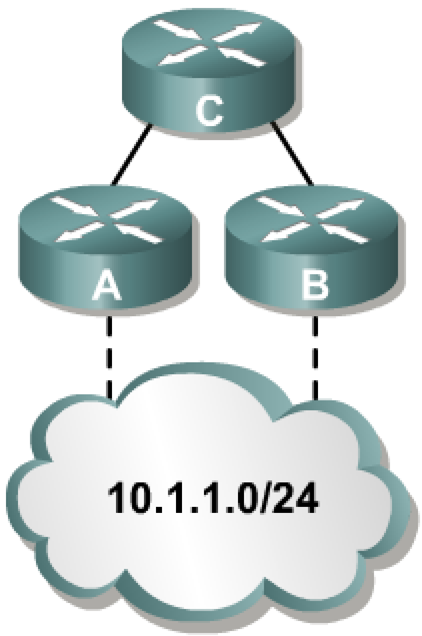
Example: EIGRP Tables

Router C's tables:

IP EIGRP Neighbor Table	
Next-Hop Router	Interface
Router A Router B	Ethernet 0 Ethernet 1

IP EIGRP Topology Table			
Network	Feasible Distance (EIGRP Metric)	Advertised Distance	EIGRP Neighbor
10.1.1.0 /24 10.1.1.0 /24	2000 2500	1000 1500	Router A (E0) Router B (E1)

The IP Routing Table			
Network	Metric (Feasible Distance)	Outbound Interface	Next Hop (EIGRP Neighbor)
10.1.1.0 /24	2000	Ethernet 0	Router A



EIGRP Neighbor Table

SRTT (Smooth Round Trip Timer) and RTO (Retransmit Interval) are used by RTP to manage reliable EIGRP packets.

SRTT indicates how long it takes for this neighbor to respond to reliable packets.

RTO indicates how long to wait before retransmitting if no ACK is received.

```
R1# show ip eigrp neighbors
IP-EIGRP neighbors for process 100
H   Address         Interface   Hold   Uptime     SRTT   RTO    Q    Seq
                                (sec)             (ms)          Cnt  Num
0   192.168.1.102   Se0/0/1     11     00:07:22   10     2280   0    5
R1#
```

- **H**: Lists the order in which a peering session was established with the specified neighbor, starting with 0.
- **Address**: Neighbor's IP address
- **Interface**: Local interface receiving EIGRP Hello packets.
- **Hold (sec)**: Seconds remaining before declaring neighbor down.

  The current hold time and is reset to the maximum hold time whenever a Hello packet is received.
- **Uptime**: Amount of time since this neighbor was added to the neighbor table.
- **Q Cnt**: Queue count should always be zero otherwise there's congestion on the link.
- **Seq Num**: The sequence number of the last update, query, or reply packet that was received from this neighbor.

EIGRP Topology Table

Destination network.

Number of successors

Feasible distance (FD) to the successor

```
R1# show ip eigrp topology
IP-EIGRP Topology Table for AS(100)/ID(192.168.1.101)
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status
P 172.17.0.0/16, 1 successors, FD is 40514560
       via 192.168.1.102 (40514560/28160), Serial0/0/1
R1#
```

Indicates if the route is in passive or active state.

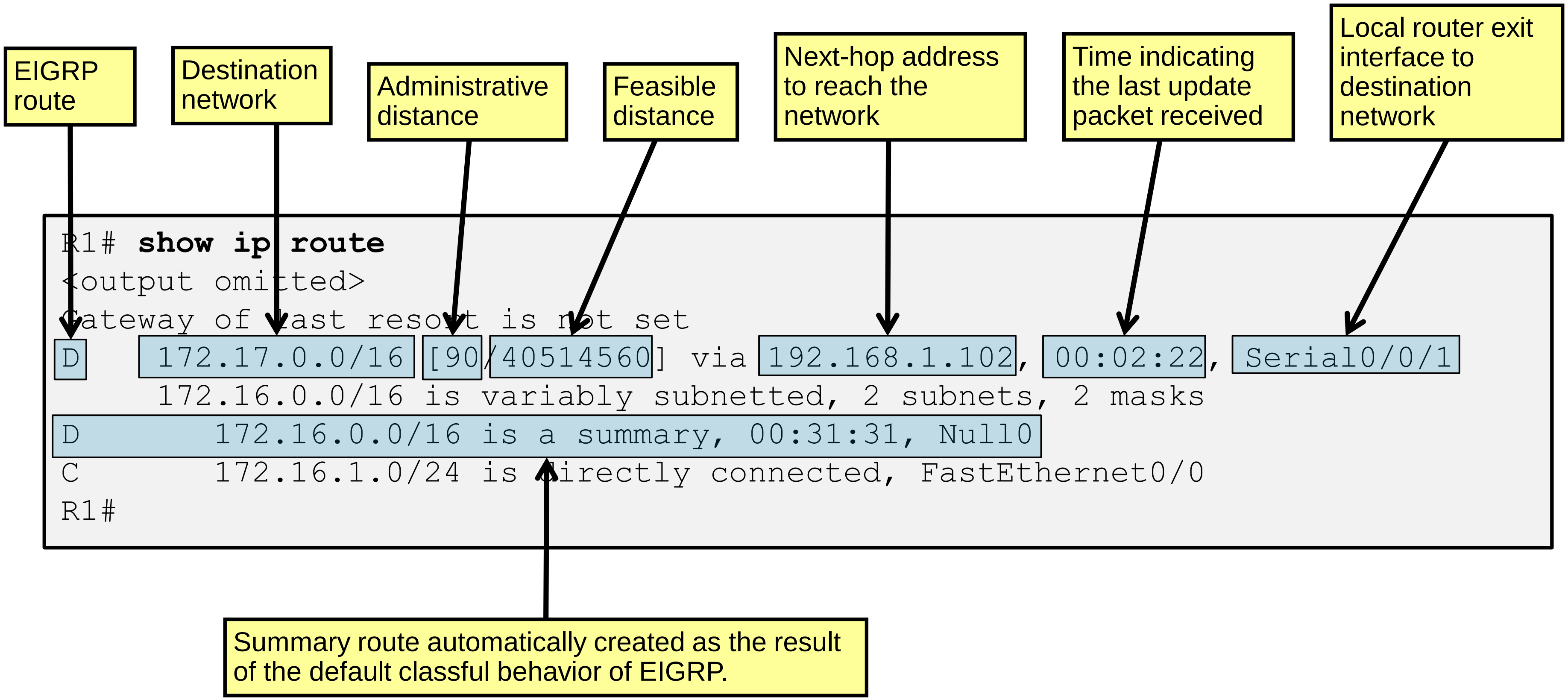
Next-hop address for successor.

Feasible distance (FD) to the successor

Advertised distance (AD) from the successor

Outbound interface to reach the network.

EIGRP Routing Table



EIGRP Administrative Distance (AD)

- EIGRP default administrative distances

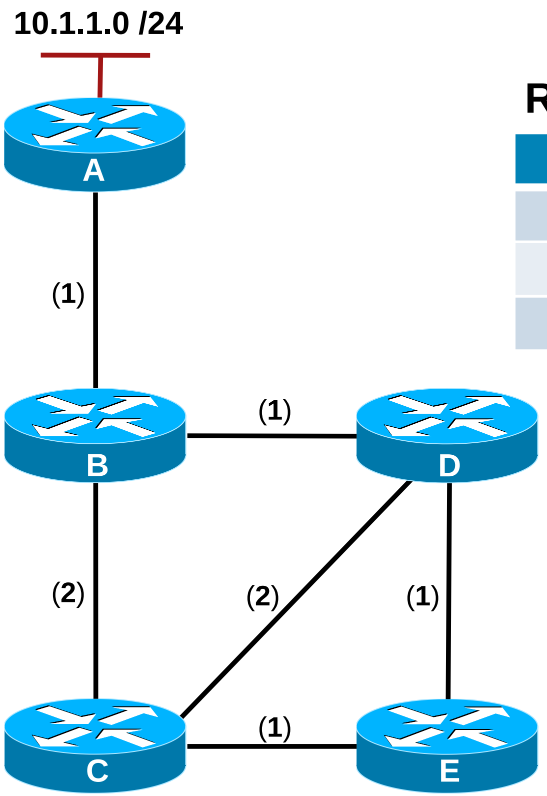
Default Administrative Distances

Route Source	Administrative Distance
Connected	0
Static	1
EIGRP summary route	5
External BGP	20
Internal EIGRP	90
IGRP	100
OSPF	110
IS-IS	115
RIP	120
External EIGRP	170
Internal BGP	200

Routes manually summarized. (Local Router)

Routes redistributed into EIGRP.

DUAL Example



Router D

EIGRP	FD	AD	Topology
10.1.1.0 /24	2		***** Passive *****
via B	2	1	Successor
via C	5	3	

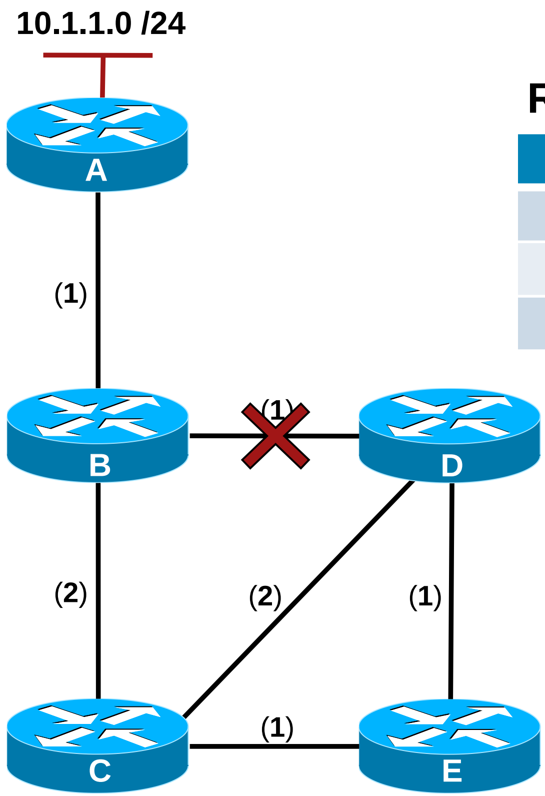
Router C

EIGRP	FD	AD	Topology
10.1.1.0 /24	3		***** Passive *****
via B	3	1	Successor
via D	4	2	Feasible Successor
via E	4	3	

Router E

EIGRP	FD	AD	Topology
10.1.1.0 /24	3		***** Passive *****
via D	3	2	Successor
via C	4	3	

DUAL Example



Router D

EIGRP	FD	AD	Topology
10.1.1.0 /24	2		***** Passive *****
~~via B~~	~~2~~	~~1~~	~~Successor~~
via C	5	3	

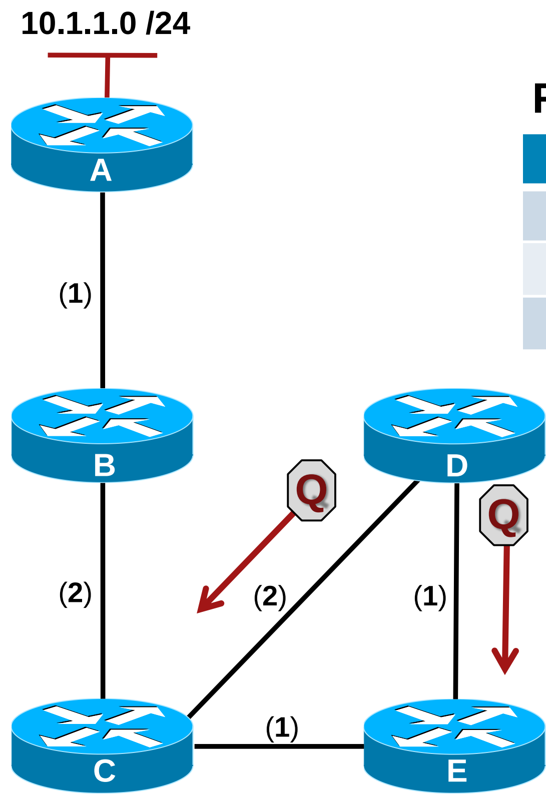
Router C

EIGRP	FD	AD	Topology
10.1.1.0 /24	3		***** Passive *****
via B	3	1	Successor
via D	4	2	Feasible Successor
via E	4	3	

Router E

EIGRP	FD	AD	Topology
10.1.1.0 /24	3		***** Passive *****
via D	3	2	Successor
via C	4	3	

DUAL Example



Router D

EIGRP	FD	AD	Topology
10.1.1.0 /24	-1		***** ACTIVE ******
via E			(Q) Query
via C	5	3	(Q) Query

 = Query

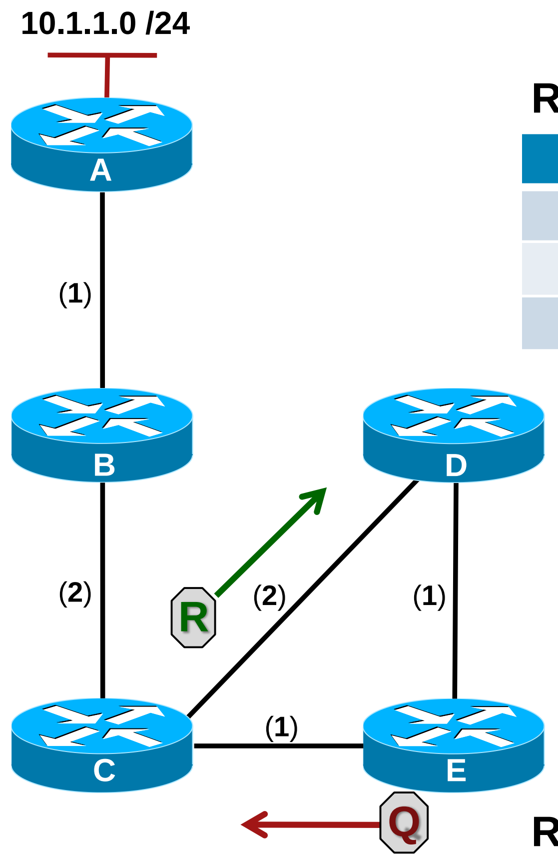
Router C

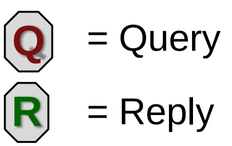
EIGRP	FD	AD	Topology
10.1.1.0 /24	3		***** Passive *****
via B	3	1	Successor
~~via D~~	~~4~~	~~2~~	~~Feasible Successor~~
via E	4	3	

Router E

EIGRP	FD	AD	Topology
10.1.1.0 /24	3		***** Passive *****
~~via D~~	~~3~~	~~2~~	~~Successor~~
via C	4	3	

DUAL Example





Router D

EIGRP	FD	AD	Topology
10.1.1.0 /24	-1		***** ACTIVE ******
via E			(Q) Query
via C	5	3	

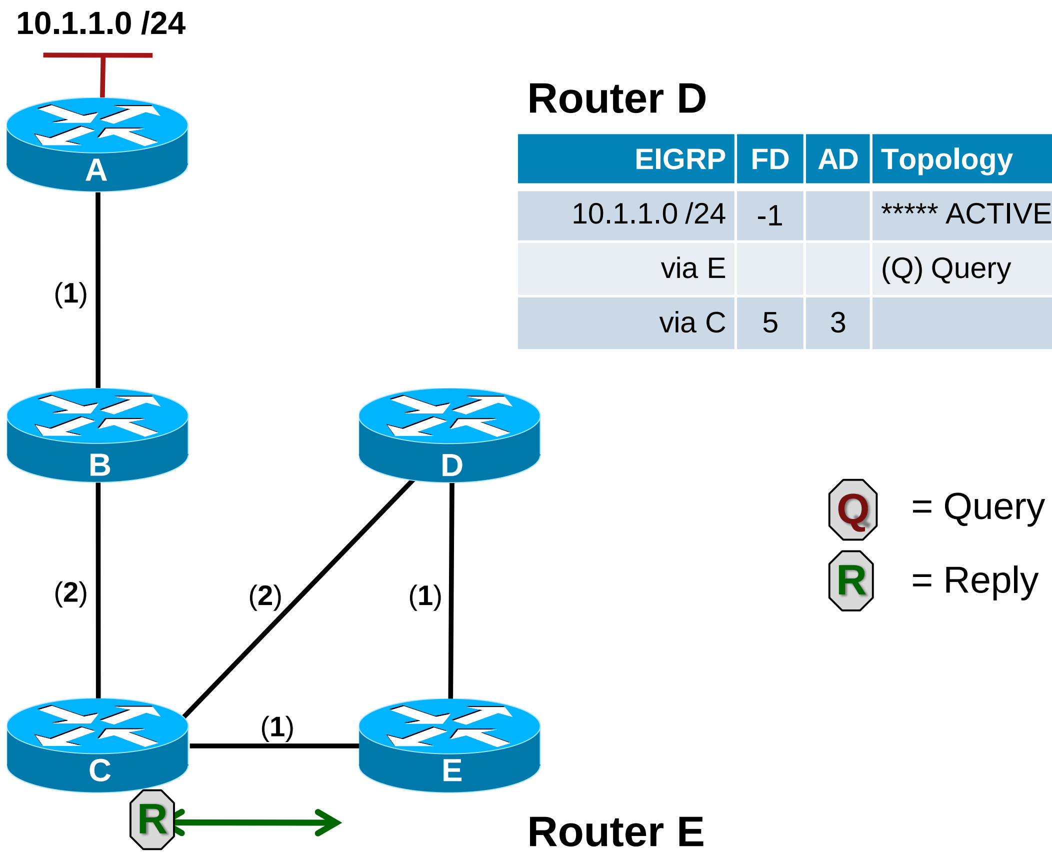
Router C

EIGRP	FD	AD	Topology
10.1.1.0 /24	3		***** Passive *****
via B	3	1	Successor
via D			
~~via E~~	~~4~~	~~3~~	

Router E

EIGRP	FD	AD	Topology
10.1.1.0 /24	-1		***** ACTIVE ******
via D			
via C	4	3	(Q) Query

DUAL Example



Router D

EIGRP	FD	AD	Topology
10.1.1.0 /24	-1		***** ACTIVE ******
via E			(Q) Query
via C	5	3	

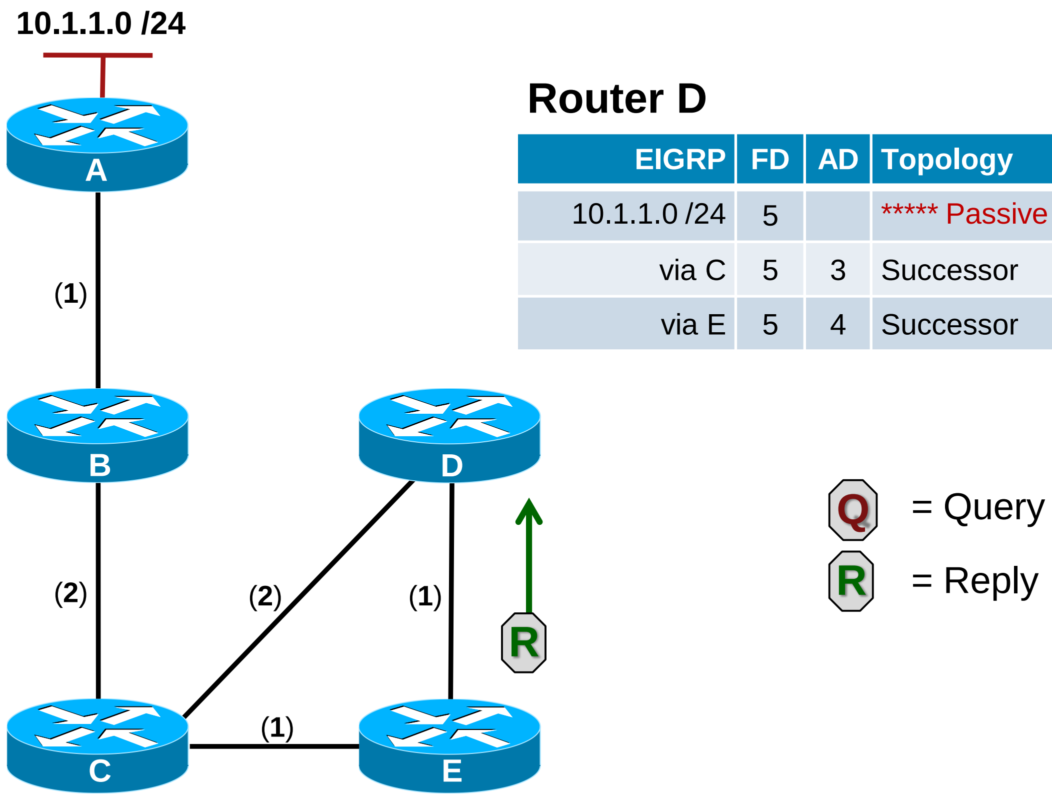
Router C

EIGRP	FD	AD	Topology
10.1.1.0 /24	3		***** Passive *****
via B	3	1	Successor
via D			
via E			

Router E

EIGRP	FD	AD	Topology
10.1.1.0 /24	4		***** Passive *****
via C	4	3	Successor
via D			

DUAL Example



Router D

EIGRP	FD	AD	Topology
10.1.1.0 /24	5		***** Passive *****
via C	5	3	Successor
via E	5	4	Successor

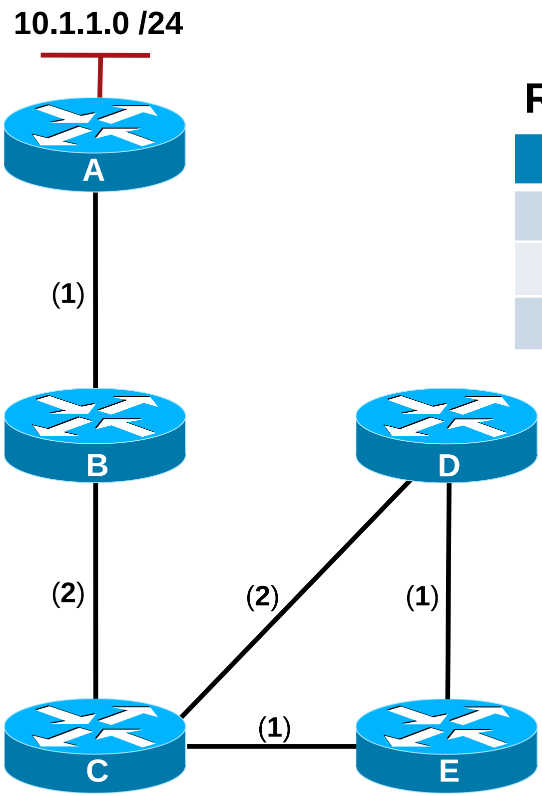
Router C

EIGRP	FD	AD	Topology
10.1.1.0 /24	3		***** Passive *****
via B	3	1	Successor
via D			
via E			

Router E

EIGRP	FD	AD	Topology
10.1.1.0 /24	4		***** Passive *****
via C	4	3	Successor
via D			

DUAL Example



Router D

EIGRP	FD	AD	Topology
10.1.1.0 /24	5		***** Passive *****
via C	5	3	Successor
via E	5	4	Successor

Router C

EIGRP	FD	AD	Topology
10.1.1.0 /24	3		***** Passive *****
via B	3	1	Successor
via D			
via E			

Router E

EIGRP	FD	AD	Topology
10.1.1.0 /24	4		***** Passive *****
via C	4	3	Successor
via D			

EIGRP Metric Calculation

- EIGRP uses a composite metric which can be based on the following metrics:
  - <u>Bandwidth</u>
  - <u>Delay</u>
  - Reliability
  - Load
  - MTU
- Only Bandwidth and Delay are used by default.

Note: It is often incorrectly stated that EIGRP can also use the smallest MTU in the path. In actual fact, the MTU is included in the EIGRP routing update, but is not actually used in the metric calculation.

EIGRP Metric Calculation

$$\text{Metric} = \left[\left(K1 * BW_{min} + \frac{K2 * BW_{min}}{256 - \text{load}} + K3 * \text{delay} \right) * \frac{K5}{K4 + \text{reliability}} \right] * 256$$

$$\text{Metric} = \left(\left(\frac{10^7}{\text{least-bandwidth}} \right) + \text{cumulative-delay} \right) * 256$$

EIGRP Bandwidth

- EIGRP uses the slowest bandwidth (BW) in its metric calculation.
  - Calculated BW = reference BW / slowest BW (kbps)

- The value of the bandwidth may or may not reflect the actual physical bandwidth of the interface.
  - For example, most serial interfaces use the default bandwidth value of 1.544 Mbps but this may not accurately reflect the links actual bandwidth.

$$BW_{min} = \frac{10^7}{\text{least-bandwidth}}$$

EIGRP Bandwidth

- Because both EIGRP and OSPF use bandwidth in default metric calculations, a correct value for bandwidth is very important to the accuracy of routing information.
  - If the actual bandwidth of the link differs from the default bandwidth value, then the bandwidth value should be modified.
- To modify the bandwidth value, use the **`bandwidth`** interface command.

Note: The bandwidth command does NOT change the physical bandwidth of the link.

EIGRP Delay

- Delay is a measure of the time it takes for a packet to traverse a route.
  - EIGRP uses the cumulative sum of all outgoing interfaces.
    - Calculated Delay = the sum of outgoing interface delays / 10
- The delay (DLY) metric is a static value based on the type of link to which the interface is connected and is expressed in microseconds.

Media	Delay
100M ATM	100 µS
Fast Ethernet	100 µS
FDDI	100 µS
1HSSI	20,000 µS
16M Token Ring	630 µS
Ethernet	1,000 µS
T1 (Serial Default)	20,000 µS
512K	20,000 µS
DS0	20,000 µS
56K	20,000 µS

Other EIGRP Metrics

- Reliability (not a default EIGRP metric) is a measure of the likelihood that a link will fail.
  - Measure dynamically & expressed as a fraction of 255.
  - The higher the fraction the better the reliability
- Load (not a default EIGRP metric) reflects how much traffic is using a link
  - Number is determined dynamically and is expressed as a fraction of 255
  - The lower the fraction the less the load on the link
- These optional criteria can be used but are not recommended, because they typically result in frequent recalculation of the topology table.

EIGRP Composite Metric Calculation

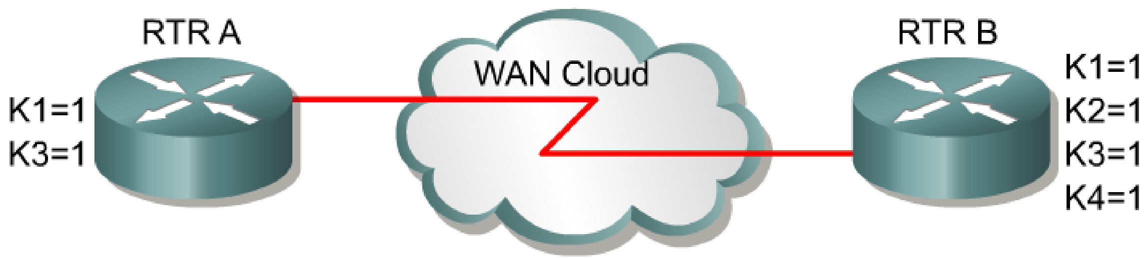
- The EIGRP composite metric formula consists of values K1 through K5, known as EIGRP metric weights.
  - By default, only K1 (bandwidth) and K3 (delay) are set to 1.
  - K2 (load), K4 (reliability), and K5 (MTU) are set to 0.
- K values can be changed with the EIGRP router command:

```
Router(config-router)# metric weights tos k1 k2 k3 k4 k5
```

$$\text{Metric} = \left[\left(K1 * BW_{min} + \frac{K2 * BW_{min}}{256 - load} + K3 * delay\right) * \frac{K5}{K4 + reliability}\right] * 256$$

Mismatched K Values

- EIGRP neighbors cannot use mismatched metric values.
  - All EIGRP neighbors must use the same metrics.
  - Metrics can be altered using the **`metric weights`** command.




```
RTRA(config)#router eigrp 1
RTRA(config-router)#network x.x.x.x
RTRA(config-router)#metric weights 0 1 1 1 1 0


RTRB(config)#router eigrp 1
RTRB(config-router)#network x.x.x.x
RTRB(config-router)#metric weights 0 1 1 1 1 0
```

EIGRP Metric Calculation Example

Default metric = [K1*bandwidth + K3*delay] * 256

Since K1 and K3 both equal 1, the formula simplifies to: **bandwidth + delay**

bandwidth = speed of slowest link in route to the destination
delay = sum of the delays of each link in route to the destination

Slowest bandwidth: (10,000,000/bandwidth kbps) * 256

Plus the sum of the delays + (sum of delay/10) * 256

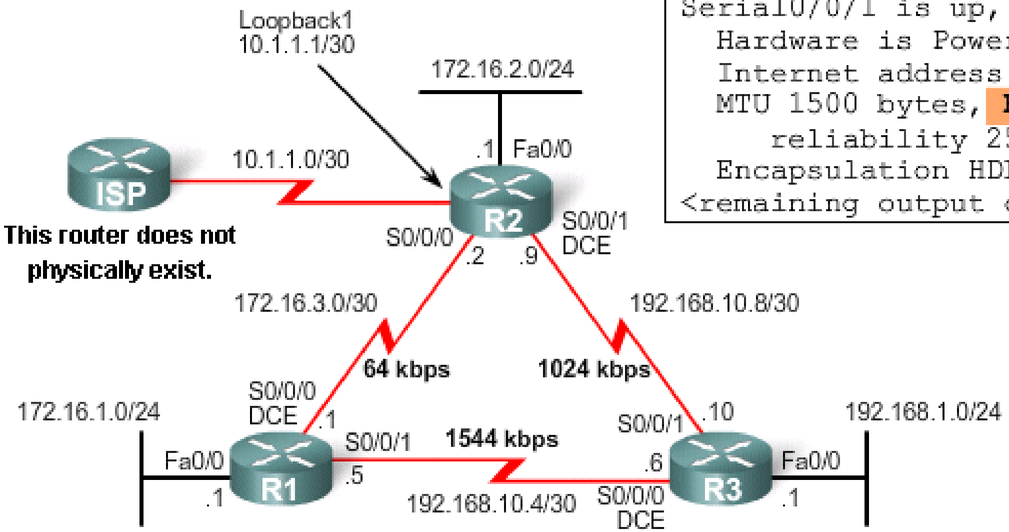
= EIGRP metric

```
R2#show ip route
<output omitted>

D    192.168.1.0/24 [90/3014400] via 192.168.10.10, 00:02:14, Serial0/0/1
```

EIGRP Bandwidth Calculation Example

- Bandwidth = 10,000,000 / 1024 = 9765 * 256 = 2499840

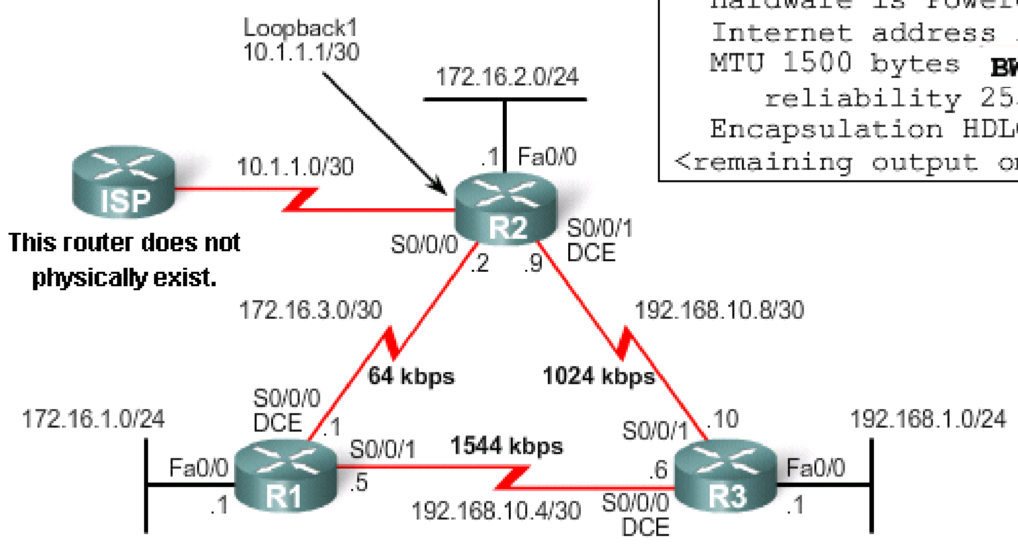
```
R2#show inter ser 0/0/1
Serial0/0/1 is up, line protocol is up
  Hardware is PowerQUICC Serial
  Internet address is 192.168.10.9/30
  MTU 1500 bytes, BW 1024 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set
<remaining output omitted>
```




```
R3#show inter fa 0/0
FastEthernet0/0 is up, line protocol is up
  Hardware is AmdFE, address is 0002.b9ee.5ee0 (bia 0002.b9ee.5ee0)
  Internet address is 192.168.1.1/24
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
<remaining output omitted>
```

EIGRP Delay Calculation Example

- Delay = 20,000 / 10 + (100 / 10) * 256 = 514560




```
R2#show inter ser 0/0/1
Serial0/0/1 is up, line protocol is up
  Hardware is PowerQUICC Serial
  Internet address is 192.168.10.9/30
  MTU 1500 bytes  BW 1024 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set
<remaining output omitted>
```

```
R3#show inter fa 0/0
FastEthernet0/0 is up, line protocol is up
  Hardware is AmdFE, address is 0002.b9ee.5ee0 (bia 0002.b9ee.5ee0)
  Internet address is 192.168.1.1/24
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
<remaining output omitted>
```

EIGRP Metric Calculation Example

- EIGRP Metric = 2499840 + 514560 = 3014400

```
R2#show ip route
<code output omitted>

Gateway of last resort is not set

     192.168.10.0/24 is variably subnetted, 3 subnets, 2 masks
D       192.168.10.0/24 is a summary, 00:00:15, Null0
D       192.168.10.4/30 [90/21024000] via 192.168.10.10, 00:00:15, Serial0/0/1
C       192.168.10.8/30 is directly connected, Serial0/0/1
     172.16.0.0/16 is variably subnetted, 4 subnets, 3 masks
D       172.16.0.0/16 is a summary, 00:00:15, Null0
D       172.16.1.0/24 [90/40514560] via 172.16.3.1, 00:00:15, Serial0/0/0
C       172.16.2.0/24 is directly connected, FastEthernet0/0
C       172.16.3.0/30 is directly connected, Serial0/0/0
     10.0.0.0/30 is subnetted, 1 subnets
C       10.1.1.0 is directly connected, Loopback1
D    192.168.1.0/24 [90/3014400] via 192.168.10.10, 00:00:15, Serial0/0/1
```

Planning EIGRP Routing Implementations

Planning to Deploy EIGRP

- Prior to deploying an EIGRP routing solution, the following should be considered:
  - IP addressing plan
  - Network topology
  - EIGRP traffic engineering
- Once the requirements have been assessed, the implementation plan can be created.

Implementing EIGRP

- The information necessary to implement EIGRP routing includes the following:
  - The IP addresses to be configured on individual router interfaces
  - The EIGRP AS number, used to enable EIGRP.
  - A list of routers and interfaces on which EIGRP is to be enabled.
  - Metrics that need to be applied to specific interfaces, or EIGRP traffic engineering.
- In the implementation plan, EIGRP the tasks include the following:
  - Enabling the EIGRP routing protocol.
  - Configuring the proper network statements.
  - Optionally configuring the metric to appropriate interfaces.

Verifying EIGRP

- After implementing EIGRP, verification should confirm proper deployment on each router.
- Verification tasks include verifying:
  - The EIGRP neighbor relationships.
  - That the EIGRP topology table is populated with the necessary information.
  - That IP routing table is populated with the necessary information.
  - That there is connectivity in the network between routers and to other devices.
  - That EIGRP behaves as expected in a case of a topology change, by testing link failure and router failure events.

Documenting

- After a successful EIGRP deployment, the solution and verification process and results should be documented for future reference.
- Documentation should include:
  - A topology map
  - The IP addressing plan
  - The AS number used
  - The networks included in EIGRP on each router
  - Any special metrics configured

Configuring and Verifying EIGRP

Enable EIGRP Routing

- Define EIGRP as the IP routing protocol.

```
Router(config)#
```

router eigrp *autonomous-system-id*

- To exchange routing updates, EIGRP routers must have the same autonomous system ID.

Identify EIGRP Networks

- Define EIGRP networks to advertise to EIGRP neighbors.

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

```
network network [mask]
```

- The *network* parameter can be a network, a subnet, or the address of a directly connected interface.
- The *mask* is a wildcard mask (inverse mask) used to determine how to interpret the address.
  - The mask has wildcard bits, where 0 is a match and 1 is “don’t care.”
  - For example, 0.0.255.255 indicates a match in the first 2 octets.

Note on EIGRP Masks

- Most EIGRP references state that the wildcard mask is required.
- However, since IOS 12.0(4)T, the mask argument can actually be configured using wild card bits or a regular subnet mask.
- For example, either format could be used to configure the 10.10.10.0 network:

  `network 10.10.10.0 0.0.0.3`

  or

  `network 10.10.10.0 255.255.255.252`
- Best Practice to configure the 10.10.10.0 network:

  `network 10.10.10.1 0.0.0.0`

Define the Interface Bandwidth

- Defines the interface's bandwidth (optional).

```
Router(config-if)#
```

bandwidth *kilobits*

- The *kilobits* parameter indicates the intended bandwidth in kbps.
  - For example, to set the bandwidth to 512,000 bps, use the **bandwidth 512** command.
- The configured bandwidth is used by routing protocols in the metric calculation.
- The command does not actually change the speed of the interface.

Enable / Disable Automatic Summarization

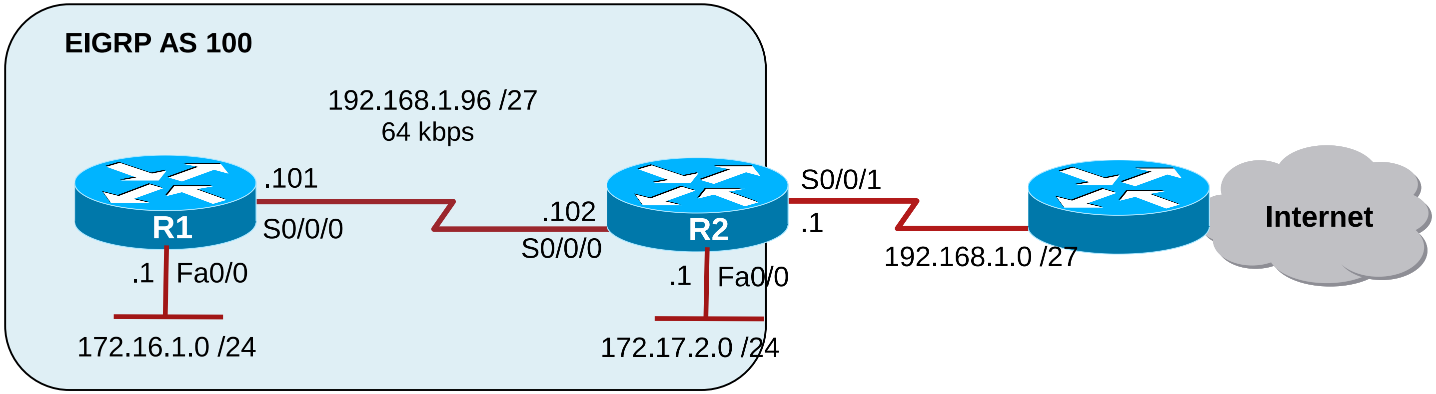
- By default, EIGRP don`t automatically summarizes subnets.

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

```
auto-summary
```

- This makes EIGRP behave like a classful routing protocol and therefore summarizes subnets on the classful boundary.
- Automatic summarization can be disabled using the **`no auto-summary`** router configuration command.

Configuring EIGRP Example: Classful

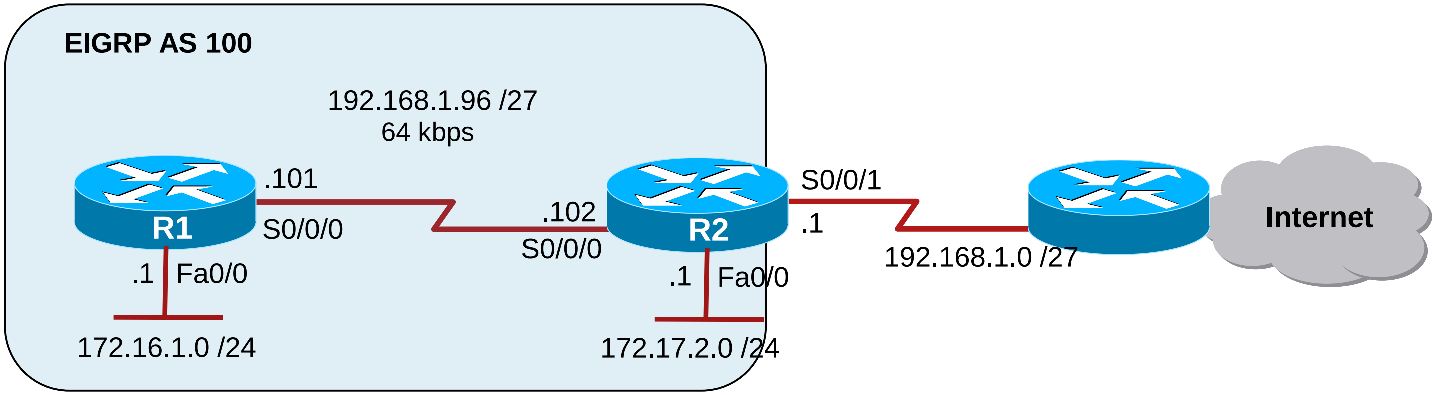


Classful configuration example:

```
R1(config)# interface Fa0/0
R1(config-if)# ip address 172.16.1.1 255.255.255.0
R1(config-if)# no shut
R1(config-if)# interface S0/0/0
R1(config-if)# ip address 192.168.1.101 255.255.255.224
R1(config-if)# bandwidth 64
R1(config-if)# no shut
R1(config-if)# exit
```

```
R2(config)# interface Fa0/0
R2(config-if)# ip address 172.16.2.1 255.255.255.0
R2(config-if)# no shut
R2(config-if)# interface S0/0/0
R2(config-if)# ip address 192.168.1.102 255.255.255.224
R2(config-if)# bandwidth 64
R2(config-if)# no shut
R2(config-if)# interface S0/0/1
R2(config-if)# ip address 192.168.1.1 255.255.255.224
R2(config-if)# bandwidth 64
R2(config-if)# no shut
R2(config-if)# exit
```

Configuring EIGRP Example: Classful

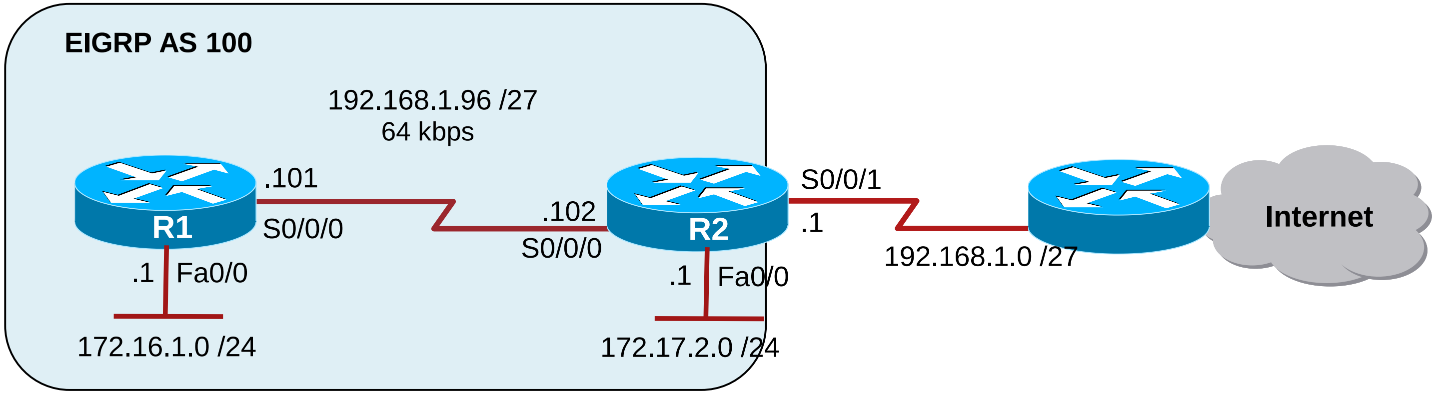


Classful configuration example:

```
R1(config)# router eigrp 100
R1(config-router)# network 192.168.1.96
R1(config-router)# network 172.16.1.0
R1(config-router)#
```

```
R2(config)# router eigrp 100
R2(config-router)# network 192.168.1.96
R2(config-router)# network 172.17.2.0
*Jul 26 10:02:25.963: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 100: Neighbor 192.168.1.101 (Serial0/0/0) is
   up: new adjacency 172.17.2.0
R2(config-router)#
R2#
```

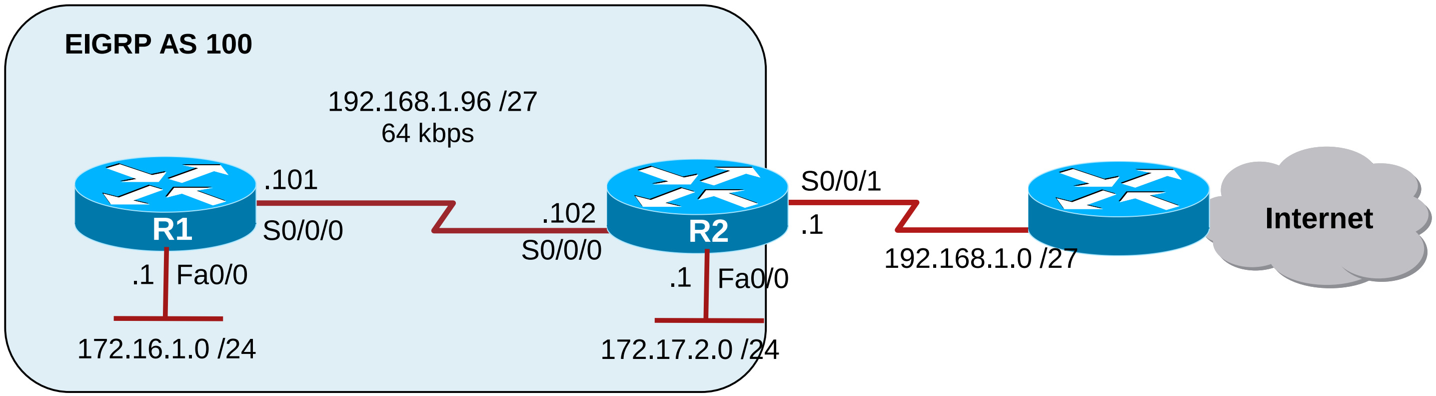
Verifying EIGRP Example



Classful configuration example:

```
R1# show running-config | section router eigrp
router eigrp 100
 network 172.16.0.0
 network 192.168.1.0

R1# show ip route
<output omitted>
Gateway of last resort is not set

D     172.17.0.0/16 [90/40514560] via 192.168.1.102, 00:24:02, Serial0/0/0
      172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
D        172.16.0.0/16 is a summary, 00:25:27, Null0
C        172.16.1.0/24 is directly connected, FastEthernet0/0
      192.168.1.0/24 is variably subnetted, 3 subnets, 2 masks
C        192.168.1.96/27 is directly connected, Serial0/0/0
D        192.168.1.0/27 [90/41024000] via 192.168.1.102, 00:16:56, Serial0/0/0
D        192.168.1.0/24 is a summary, 00:25:27, Null0
R1#
```

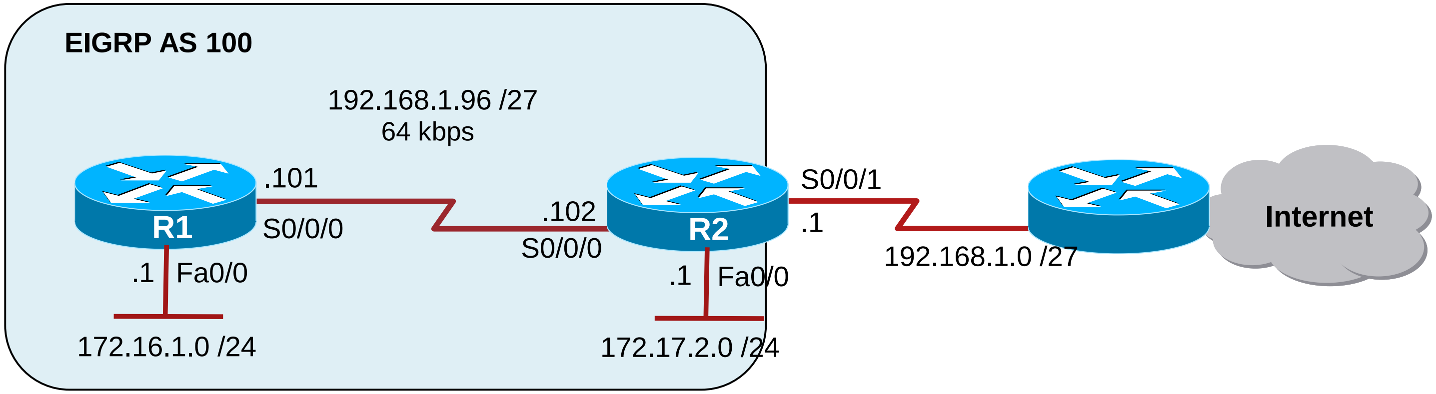
Verifying EIGRP Example



Classful configuration example:

```
R2# show running-config | section router eigrp
router eigrp 100
 network 172.17.0.0
 network 192.168.1.0

R2# show ip route
<output omitted>
Gateway of last resort is not set

     172.17.0.0/16 is variably subnetted, 2 subnets, 2 masks
D       172.17.0.0/16 is a summary, 00:13:10, Null0
C       172.17.2.0/24 is directly connected, FastEthernet0/0
D    172.16.0.0/16 [90/40514560] via 192.168.1.101, 00:13:26, Serial0/0/0
     192.168.1.0/24 is variably subnetted, 3 subnets, 2 masks
C       192.168.1.96/27 is directly connected, Serial0/0/0
C       192.168.1.0/27 is directly connected, Serial0/0/1
D       192.168.1.0/24 is a summary, 00:13:10, Null0
R2#
```

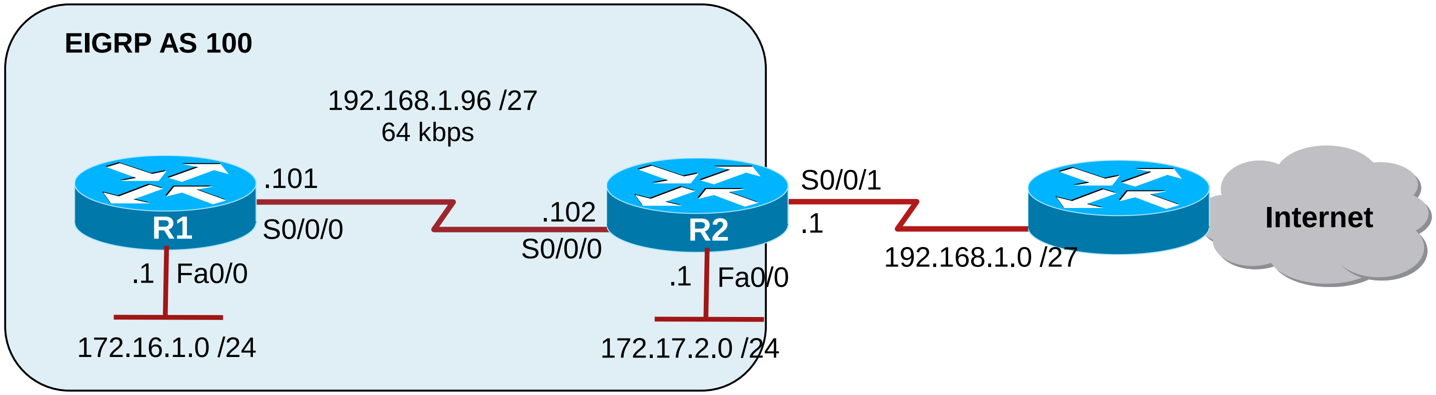
Verifying EIGRP Example



Classful configuration example:

```
R2# show ip protocols
Routing Protocol is "eigrp 100"
<output omitted>
  Automatic network summarization is in effect
  Automatic address summarization:
    192.168.1.0/24 for FastEthernet0/0
      Summarizing with metric 40512000
    172.17.0.0/16 for Serial0/0/0, Serial0/0/1
      Summarizing with metric 28160
  Maximum path: 4
  Routing for Networks:
    172.17.0.0
    192.168.1.0
  Routing Information Sources:
<output omitted>
R2#
```

Configuring EIGRP Example: Classless

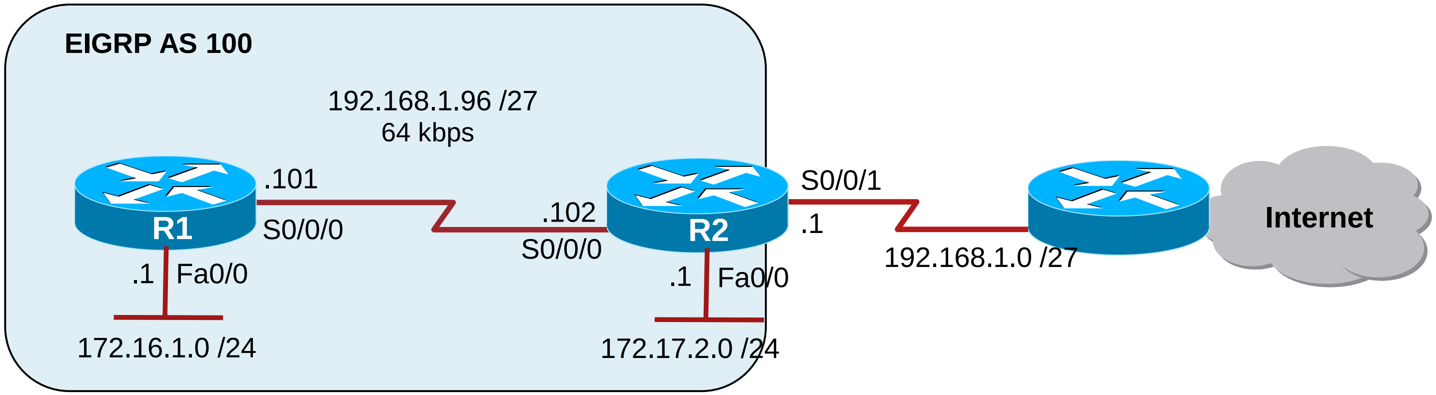


Classless configuration example:

```
R1(config)# no router eigrp 100
R1(config)# router eigrp 100
R1(config-router)# network 192.168.1.96 0.0.0.31
R1(config-router)# network 172.16.1.0 0.0.0.255
R1(config-router)# end
R1# show run | section router eigrp
router eigrp 100
 network 172.16.1.0 0.0.0.255
 network 192.168.1.96 0.0.0.31

R1#
```

```
R2(config)# no router eigrp 100
R2(config)# router eigrp 100
R2(config-router)# network 192.168.1.96 0.0.0.31
R2(config-router)# network 172.17.2.0 0.0.0.255
R2(config-router)# end
R2# show run | section router eigrp
router eigrp 100
 network 172.17.2.0 0.0.0.255
 network 192.168.1.96 0.0.0.31

R2#
```

Verifying EIGRP Example



Classful configuration example:

```
R2# show ip protocols
Routing Protocol is "eigrp 100"
<output omitted>
Automatic network summarization is in effect
  Automatic address summarization:
    192.168.1.0/24 for FastEthernet0/0
      Summarizing with metric 40512000
    172.17.0.0/16 for Serial0/0/0
      Summarizing with metric 28160
  Maximum path: 4
  Routing for Networks:
    172.17.2.0/24
    192.168.1.96/27
  Routing Information Sources:
    Gateway         Distance      Last Update
    (this router)         90      00:00:06
    Gateway         Distance      Last Update
    192.168.1.101         90      00:00:26
  Distance: internal 90 external 170
```

Verifying EIGRP: show ip protocols

Verify routing protocol information on the router.

```
R1# show ip protocols
Routing Protocol is "eigrp 100"
<output omitted>
 EIGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
  EIGRP maximum hopcount 100
  EIGRP maximum metric variance 1
  Redistributing: eigrp 100
  EIGRP NSF-aware route hold timer is 240s
  Automatic network summarization is in effect
  Automatic address summarization:
    192.168.1.0/24 for FastEthernet0/0
      Summarizing with metric 40512000
    172.16.0.0/16 for Serial0/0/0
      Summarizing with metric 28160
  Maximum path: 4
  Routing for Networks:
    172.16.1.0/24
    192.168.1.96/27
→
```

```
→
Routing Information Sources:
    Gateway          Distance      Last Update
    (this router)          90      00:08:56
    Gateway          Distance      Last Update
    192.168.1.102          90      00:07:59
  Distance: internal 90 external 170
```

Verifying EIGRP: show ip eigrp neighbors

EIGRP uses the Neighbor table to list adjacent routers.

```
R1# show ip eigrp neighbors
IP-EIGRP neighbors for process 100
H   Address                  Interface        Hold Uptime    SRTT   RTO  Q   Seq
                                              (sec)          (ms)        Cnt Num
0   192.168.1.102            Se0/0/0           11 00:09:17    22   2280  0   5
R1#
```

Verifying EIGRP: show ip eigrp topology

Verify routing protocol information on the router.

```
R1# show ip eigrp topology
IP-EIGRP Topology Table for AS(100)/ID(192.168.1.101)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 192.168.1.96/27, 1 successors, FD is 40512000
        via Connected, Serial0/0/0
P 192.168.1.0/24, 1 successors, FD is 40512000
        via Summary (40512000/0), Null0
P 172.16.0.0/16, 1 successors, FD is 28160
        via Summary (28160/0), Null0
P 172.17.0.0/16, 1 successors, FD is 40514560
        via 192.168.1.102 (40514560/28160), Serial0/0/0
P 172.16.1.0/24, 1 successors, FD is 28160
        via Connected, FastEthernet0/0
```

Verifying EIGRP: show ip route eigrp

Verify that the router recognizes EIGRP routes.

```
R1# show ip route eigrp
D    172.17.0.0/16 [90/40514560] via 192.168.1.102, 00:10:18, Serial0/0/0
     172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
D       172.16.0.0/16 is a summary, 00:11:19, Null0
     192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
D       192.168.1.0/24 is a summary, 00:11:19, Null0
R1#
R1# show ip route
<output omitted>
Gateway of last resort is not set


D    172.17.0.0/16 [90/40514560] via 192.168.1.102, 00:10:35, Serial0/0/0
     172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
D       172.16.0.0/16 is a summary, 00:11:37, Null0
C       172.16.1.0/24 is directly connected, FastEthernet0/0
     192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.1.96/27 is directly connected, Serial0/0/0
D       192.168.1.0/24 is a summary, 00:11:37, Null0
```

Verifying EIGRP: show ip eigrp interfaces

Verify EIGRP configured interfaces.

```
R1# show ip eigrp interfaces
IP-EIGRP interfaces for process 100

                        Xmit Queue   Mean   Pacing Time   Multicast    Pending
Interface        Peers  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Se0/0/0            1        0/0        22      10/380        468           0
Fa0/0              0        0/0         0       0/1            0           0
R1#
```

Verifying EIGRP: show ip eigrp traffic

Verify EIGRP traffic information.

```
R1# show ip eigrp traffic
IP-EIGRP Traffic Statistics for AS 100
  Hellos sent/received: 338/166
  Updates sent/received: 7/7
  Queries sent/received: 0/0
  Replies sent/received: 0/0
  Acks sent/received: 2/2
  SIA-Queries sent/received: 0/0
  SIA-Replies sent/received: 0/0
  Hello Process ID: 228
  PDM Process ID: 226
  IP Socket queue:   0/2000/1/0 (current/max/highest/drops)
  Eigrp input queue: 0/2000/1/0 (current/max/highest/drops)

R1#
```

Verifying EIGRP: debug eigrp packets

Traces transmission and receipt of EIGRP packets.

```
R2# debug eigrp packets
*Jul 26 10:51:24.051: EIGRP: Sending HELLO on Serial0/0/0
*Jul 26 10:51:24.051:   AS 100, Flags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0
*Jul 26 10:51:24.111: EIGRP: Sending HELLO on FastEthernet0/0
*Jul 26 10:51:24.111:   AS 100, Flags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0
*Jul 26 10:51:26.667: EIGRP: Received HELLO on Serial0/0/0 nbr 192.168.1.101
*Jul 26 10:51:26.667:   AS 100, Flags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
*Jul 26 10:51:28.451: EIGRP: Sending HELLO on FastEthernet0/0
*Jul 26 10:51:28.451:   AS 100, Flags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0
*Jul 26 10:51:29.027: EIGRP: Sending HELLO on Serial0/0/0
*Jul 26 10:51:29.027:   AS 100, Flags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0
*Jul 26 10:51:31.383: EIGRP: Received HELLO on Serial0/0/0 nbr 192.168.1.101
*Jul 26 10:51:31.383:   AS 100, Flags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
*Jul 26 10:51:33.339: EIGRP: Sending HELLO on FastEthernet0/0
*Jul 26 10:51:33.339:   AS 100, Flags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0
*Jul 26 10:51:33.511: EIGRP: Sending HELLO on Serial0/0/0
*Jul 26 10:51:33.511:   AS 100, Flags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0
```

EIGRP Passive-Interface

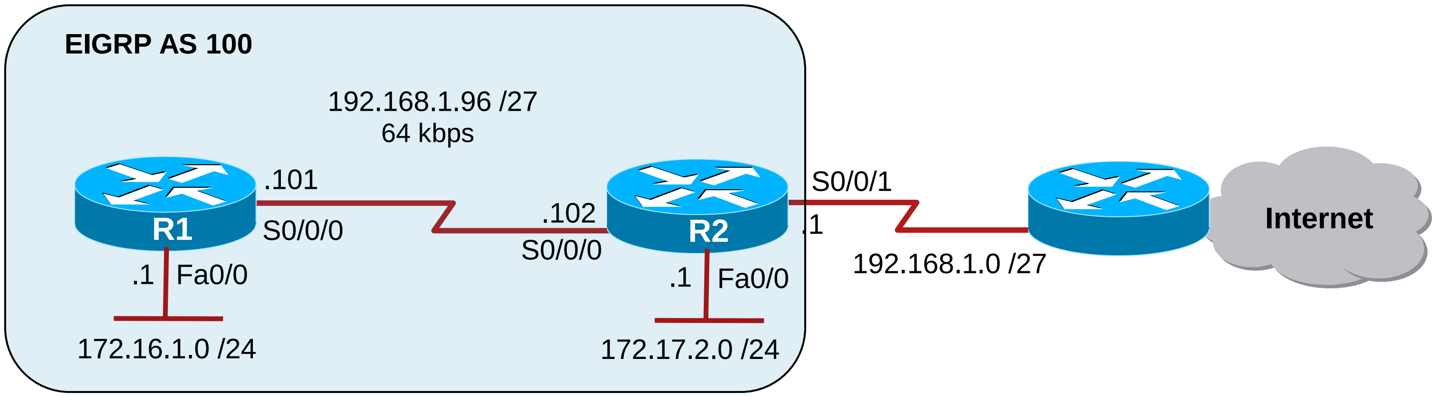
- Prevent EIGRP updates out a specified router interface.

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

passive-interface *type number* **[default]**

- Set a particular interface or all router interfaces to passive.
- The **default** option sets all router interfaces to passive.
- For EIGRP, the command:
  - Prevents neighbor relationships from being established.
  - Routing updates from a neighbor are ignored.
  - Allows a subnet on a passive interface to be announced in EIGRP

Passive-Interface Example




```
R1(config)# router eigrp 100
R1(config-router)# passive-interface fa0/0
R1(config-router)#
```

```
R2(config)# router eigrp 100
R2(config-router)# passive-interface fa0/0
R2(config-router)#
```

Alternate configuration:

```
R1(config)# router eigrp 100
R1(config-router)# passive-interface default
R1(config-router)# no passive-interface S0/0/0
```

```
R2(config)# router eigrp 100
R2(config-router)# passive-interface default
R2(config-router)# no passive-interface S0/0/0
```

Propagating a Default Route

- To propagate a default route in EIGRP, use either the:

```
ip summary-address eigrp as-number 0.0.0.0 0.0.0.0
```

or

```
ip route 0.0.0.0 0.0.0.0 next-hop | interface
&
redistribute static
```

- Once configured, the default route has to be propagated into the EIGRP AS.

ip route 0.0.0.0 0.0.0.0 *next-hop* **|** *interface* Command

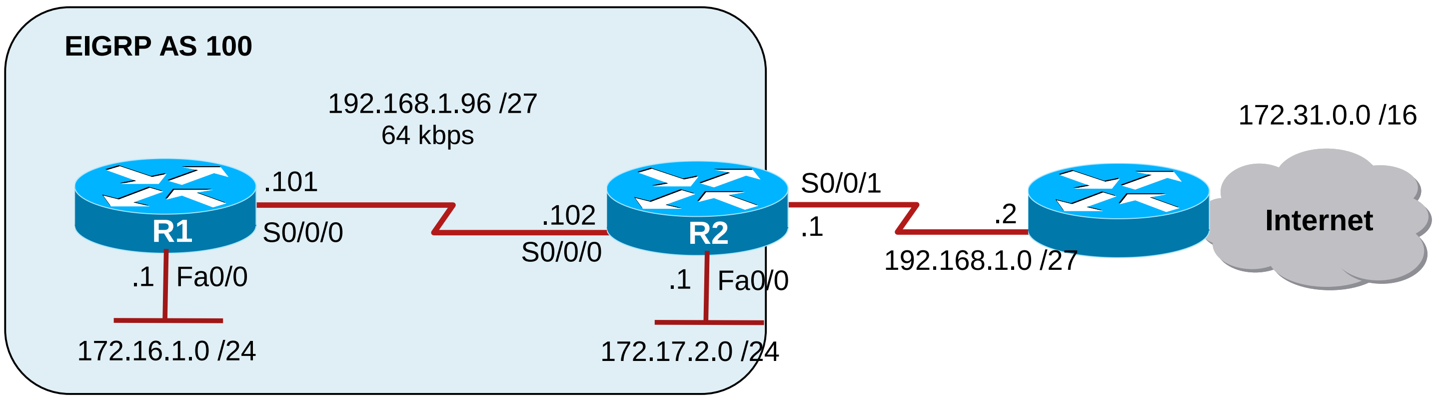
- Configures a router to advertise a default route as the gateway of last resort.

```
Router(config)#
```

ip route 0.0.0.0 0.0.0.0 *interface* **|** *next-hop*

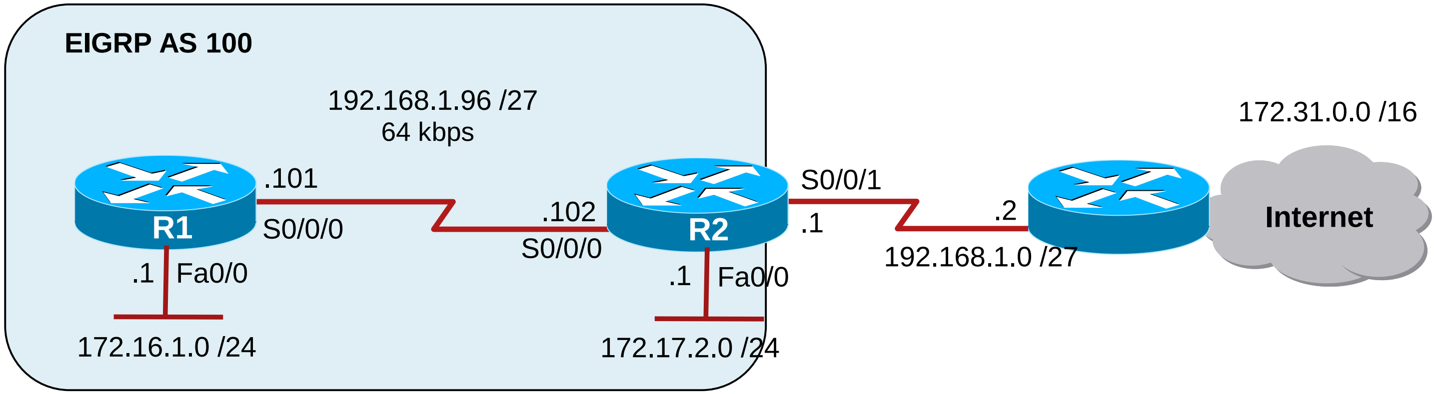
- The choice of parameter affects the next selection of commands.
  - If the *interface* parameter is used, then only the **network 0.0.0.0** needs to also be entered.
  - If the *next-hop* parameter is used, then the **network 0.0.0.0** and the **redistribute static** command must be configured.

`ip route 0.0.0.0 0.0.0.0` *`interface`* Example




```
R2(config)# ip route 0.0.0.0 0.0.0.0 S0/0/1
R2(config)# router eigrp 100
R2(config-router)# network 0.0.0.0
R2(config-router)# do show ip route
<output omitted>
Gateway of last resort is 0.0.0.0 to network 0.0.0.0

     172.17.0.0/16 is variably subnetted, 2 subnets, 2 masks
D       172.17.0.0/16 is a summary, 03:13:25, Null0
C       172.17.2.0/24 is directly connected, FastEthernet0/0
D    172.16.0.0/16 [90/40514560] via 192.168.1.101, 03:13:25, Serial0/0/0
     192.168.1.0/27 is subnetted, 2 subnets
C       192.168.1.96 is directly connected, Serial0/0/0
C       192.168.1.0 is directly connected, Serial0/0/1
S*   0.0.0.0/0 is directly connected, Serial0/0/1
R2(config-router)#
```

`ip route 0.0.0.0 0.0.0.0` *`next-hop`* **Example**

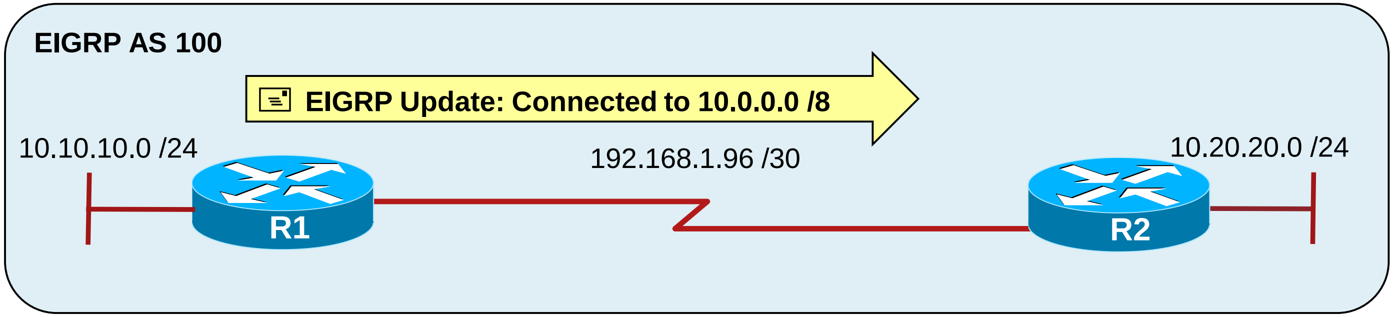



```
R2(config)# ip route 0.0.0.0 0.0.0.0 192.168.1.2
R2(config)# router eigrp 100
R2(config-router)# network 0.0.0.0
R2(config-router)# redistribute static
R2(config-router)# do show ip route
<output omitted>
Gateway of last resort is 192.168.1.2 to network 0.0.0.0

     172.17.0.0/16 is variably subnetted, 2 subnets, 2 masks
D       172.17.0.0/16 is a summary, 02:53:48, Null0
C       172.17.2.0/24 is directly connected, FastEthernet0/0
D    172.16.0.0/16 [90/40514560] via 192.168.1.101, 02:53:48, Serial0/0/0
     192.168.1.0/27 is subnetted, 2 subnets
C       192.168.1.96 is directly connected, Serial0/0/0
C       192.168.1.0 is directly connected, Serial0/0/1
S*   0.0.0.0/0 [1/0] via 192.168.1.2
R2(config-router)#
```

EIGRP Route Summarization

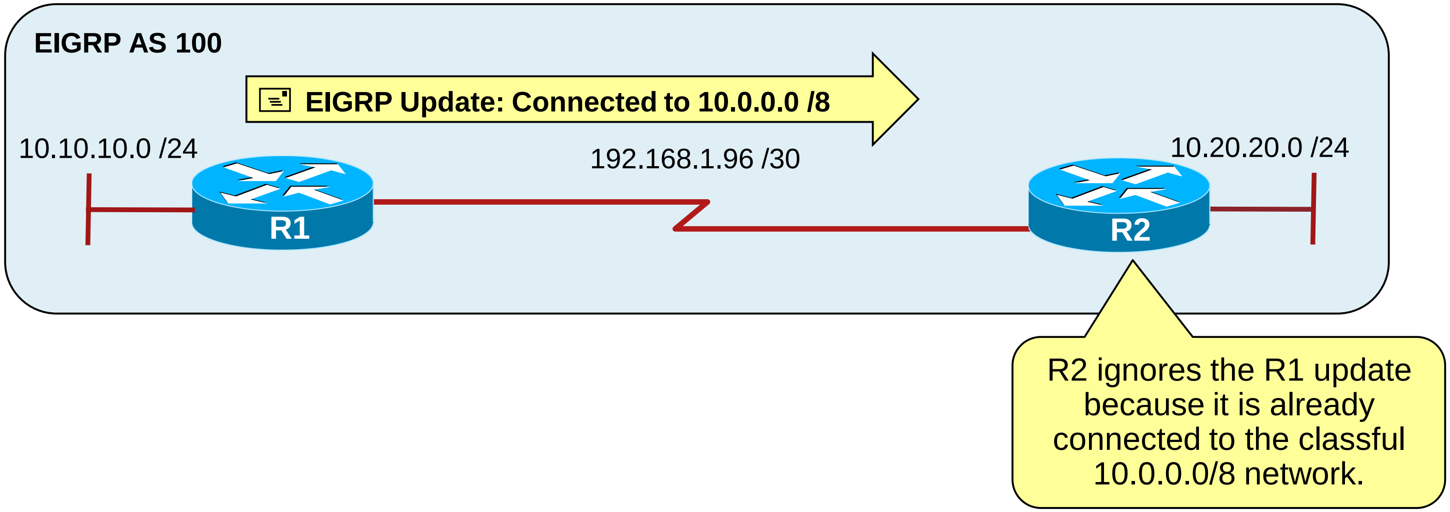
- EIGRP don`t automatically summarizes routes at a major network boundary by default.
  - Due to the configured **`auto-summary`** router configuration command.
  - In most cases, auto summarization is a good thing as it keeps routing tables as compact as possible.
  - Sometimes it's not a good thing such as when there is a discontiguous subnetwork.
- Typically for routing to work properly, auto-summarization should be disabled using the **`no auto-summary`** router configuration command its default behavior.

Summarization in Discontiguous Networks




```
R1# show running-config | section router eigrp
router eigrp 100
 passive-interface FastEthernet0/0
 network 10.10.10.0 0.0.0.255
 network 192.168.1.96 0.0.0.31
 auto-summary
R1# show ip protocols
Routing Protocol is "eigrp 100"
<output omitted>
  Automatic network summarization is in effect
  Automatic address summarization:
    10.0.0.0/8 for Serial0/0/0
      Summarizing with metric 28160
  Maximum path: 4
  Routing for Networks:
    10.10.10.0/24
    192.168.1.96/27
  Passive Interface(s):
    FastEthernet0/0
  <output omitted>
```

Summarization in Discontiguous Networks



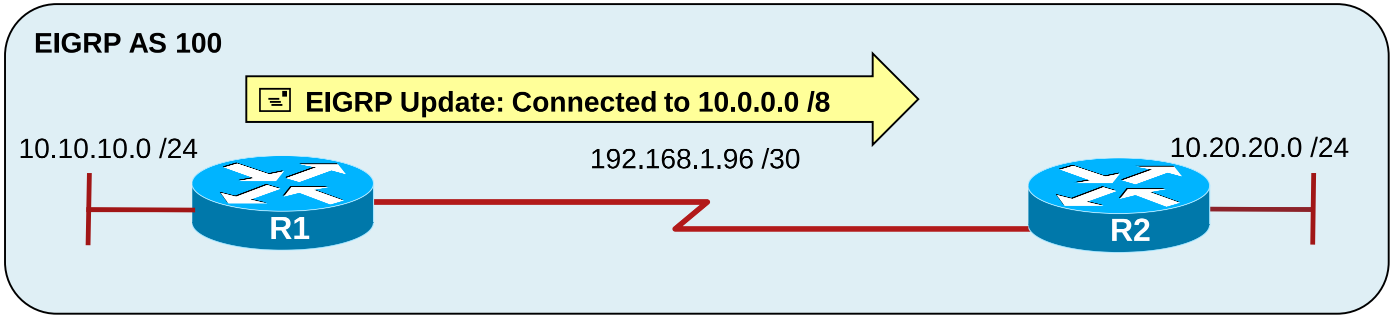

```
R2# show ip route
<output omitted>
Gateway of last resort is 192.168.1.2 to network 0.0.0.0

     10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       10.20.20.0/24 is directly connected, FastEthernet0/0
D       10.0.0.0/8 is a summary, 00:13:35, Null0
     192.168.1.0/27 is subnetted, 2 subnets
C       192.168.1.96 is directly connected, Serial0/0/0
C       192.168.1.0 is directly connected, Serial0/0/1
S*   0.0.0.0/0 [1/0] via 192.168.1.2
R2#
```

Null 0

- Notice that the summarized route (10.0.0.0/8) has an entry pointing to null0.
  - Null0 is automatically added to the table and are called summary routes.
  - Null 0 is a directly connected, software-only interface.
  - The use of the null0 interface prevents the router from trying to forward traffic to other routers in search of a more precise, longer match.

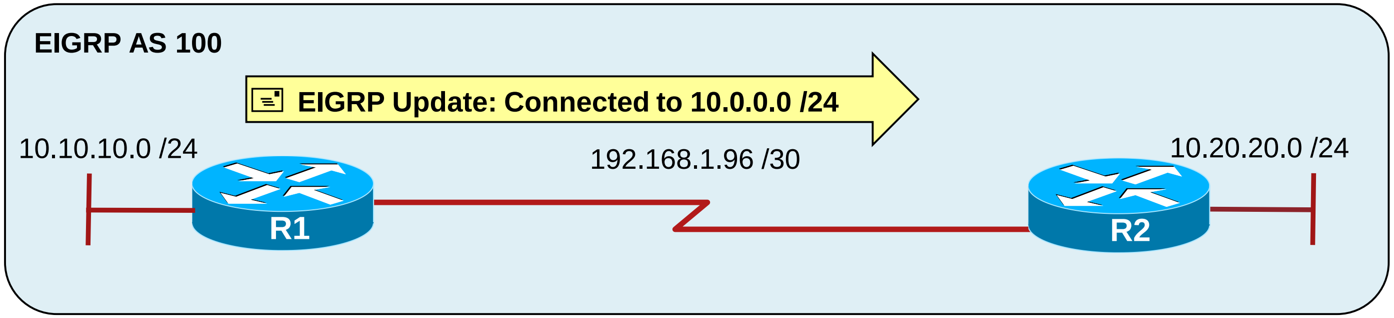
```
R2# show ip route
<output omitted>
Gateway of last resort is 192.168.1.2 to network 0.0.0.0

     10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       10.20.20.0/24 is directly connected, FastEthernet0/0
D       10.0.0.0/8 is a summary, 00:13:35, Null0
     192.168.1.0/27 is subnetted, 2 subnets
C       192.168.1.96 is directly connected, Serial0/0/0
C       192.168.1.0 is directly connected, Serial0/0/1
S*   0.0.0.0/0 [1/0] via 192.168.1.2
R2#
```

Disabling Automatic Summarization




```
R1(config)# router eigrp 100
R1(config-router)# no auto-summary
R1(config-router)#
*Jul 26 22:14:07.183: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 100: Neighbor 192.168.1.102
 (Serial0/0/0) is resync: summary configured
R1(config-router)# end
R1# show ip protocols
Routing Protocol is "eigrp 100"
<output omitted>
Automatic network summarization is not in effect
  Maximum path: 4
  Routing for Networks:
    10.10.10.0/24
    192.168.1.96/27
<output omitted>
```

Disable Automatic Summarization




```
R2(config)# router eigrp 100
R2(config)# no auto-summary
R2(config)# end
R2# show ip route
<output omitted>
Gateway of last resort is 192.168.1.2 to network 0.0.0.0

     10.0.0.0/24 is subnetted, 2 subnets
C       10.20.20.0 is directly connected, FastEthernet0/0
D       10.10.10.0 [90/40514560] via 192.168.1.101, 00:05:21, Serial0/0/0
     192.168.1.0/27 is subnetted, 2 subnets
C       192.168.1.96 is directly connected, Serial0/0/0
C       192.168.1.0 is directly connected, Serial0/0/1
S*   0.0.0.0/0 [1/0] via 192.168.1.2
R2#
```

Summarizing on an Interface

- Earlier distance vector protocols were unable to create summary routes other than the classful boundaries or /8, /16/ or /24.
- To address this shortcoming, EIGRP added the **`ip summary-address eigrp`** interface configuration command.
  - The command is used to create one or more summary routes within a network on any bit boundary (as long as a more specific route exists in the routing table).
- IP EIGRP summary routes are given an administrative distance value of 5.
  - Standard EIGRP routes receive an administrative distance of 90
  - External EIGRP routes receive an administrative distance of 170.

ip summary-address eigrp

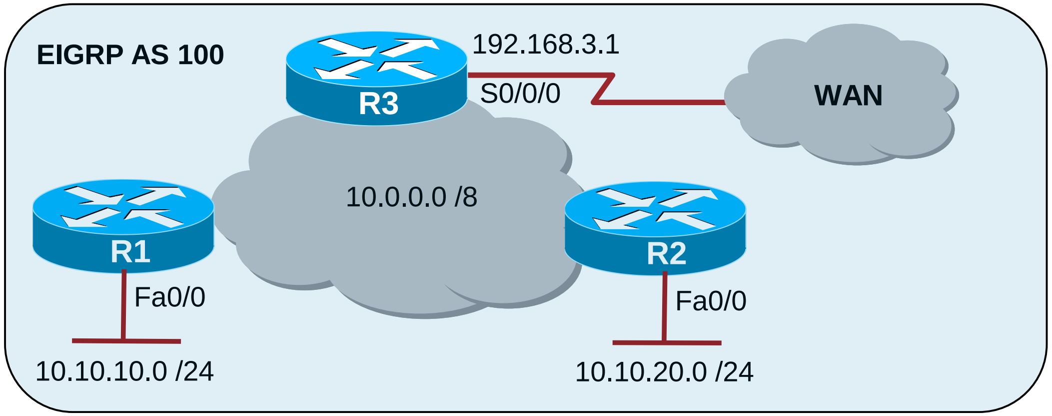
- Manually create a summary route at an arbitrary bit boundary.

```
Router(config-if)#
```

ip summary-address eigrp *as-number address mask* **[***admin-distance***]**

Parameter	Description
as-number	The number of the EIGRP AS is identified.
address	The IP address being advertised as the summary address. This address does not need to be aligned on Class A, B, or C boundaries.
mask	The IP subnet mask used to create the summary address.
admin-distance	(Optional) Administrative distance. A value from 0 to 255.

EIGRP Route Summarization




```
R1(config)# router eigrp 100
R1(config)# no auto-summary
```

```
R2(config)# router eigrp 100
R2(config)# no auto-summary
```

```
R3(config)# interface S0/0/0
R3(config-if)# ip address 192.168.3.1 255.255.255.0
R3(config-if)# ip summary-address eigrp 100 10.10.0.0 255.255.0.0
R3(config-if)# no shut
R3(config-if)# exit
R3# show ip protocols
Routing Protocol is "eigrp 100"
<output omitted>
Automatic network summarization is not in effect
  Address Summarization:
    10.10.0.0/16 for Serial0/0/0
<output omitted>
```

Configuring and Verifying EIGRP in an Enterprise WAN

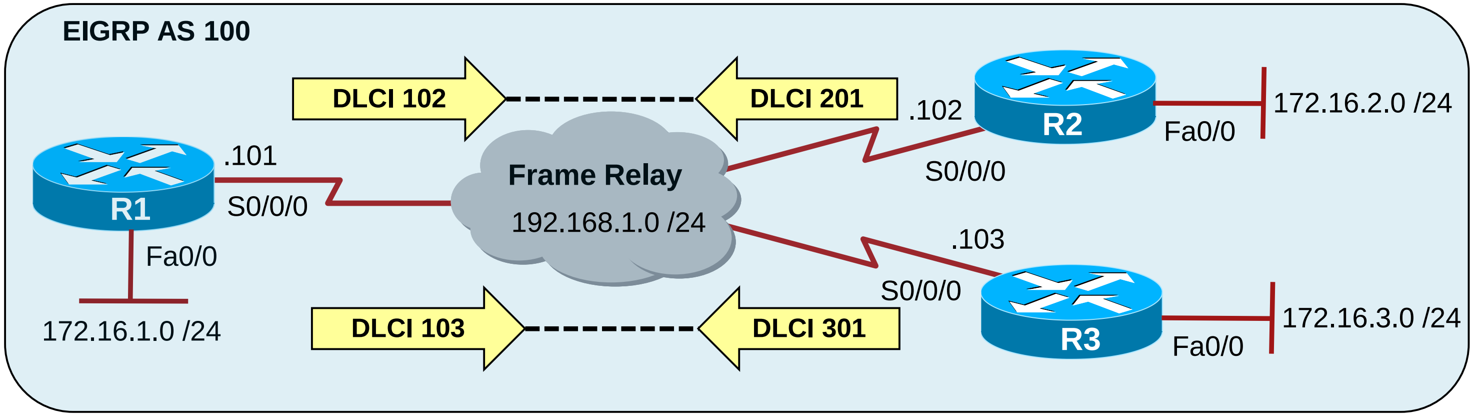
EIGRP and WAN Considerations

- There are various deployment options available for supporting EIGRP over a WAN including:
  - Frame Relay
    - Frame-Relay using dynamic mapping
    - Frame-Relay using static mapping
    - Multipoint and point-to-point Frame-Relay subinterfaces
  - Multiprotocol Label Switching (MPLS) virtual private networks (VPNs),
  - Ethernet over Multiprotocol Label Switching (EoMPLS)
- Other considerations include:
  - EIGRP load balancing
  - Limiting EIGRP bandwidth utilization on WAN links

Frame Relay Using Dynamic Mapping

- Easy deployment due to the use of inverse ARP.
- Auto detects most settings.
- Inverse-ARP will dynamically map the IP addresses of the devices at the other ends of the PVCs to the local DLCI number.
- Consists of three steps:
  1. Configure an IP address on the serial interface.
  2. Change the encapsulation on an interface using the **`encapsulation frame-relay`** command.
  3. Activate the interface.

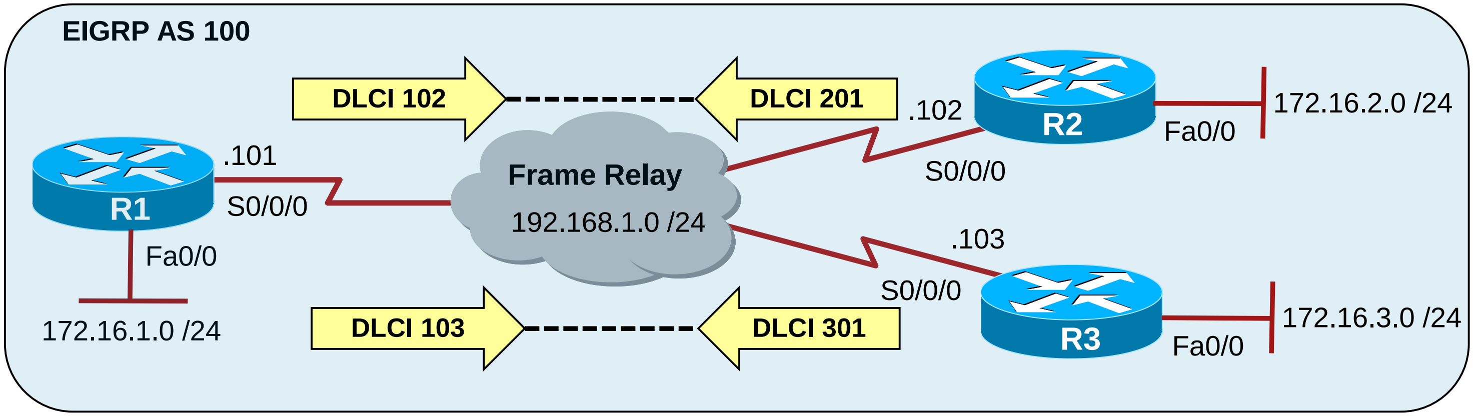
Frame Relay Using Dynamic Mapping




```
R1(config)# interface S0/0/0
R1(config-if)# encapsulation frame-relay
R1(config-if)# ip address 192.168.1.101 255.255.255.0
R1(config-if)# exit
R1(config)# router eigrp 100
R1(config-router)# network 172.16.1.0 0.0.0.255
R1(config-router)# network 192.168.1.0
R1(config-router)#
```

```
R3(config)# interface S0/0/0
R3(config-if)# encapsulation frame-relay
R3(config-if)# ip address 192.168.1.103 255.255.255.0
R3(config-if)# exit
R3(config)# router eigrp 100
R3(config-router)# network 172.16.3.0 0.0.0.255
R3(config-router)# network 192.168.1.0
R3(config-router)#
```

Frame Relay Using Dynamic Mapping




```
R1# show ip eigrp neighbors
IP-EIGRP neighbors for process 100
H   Address          Interface      Hold  Uptime    SRTT   RTO  Q   Seq
                                    (sec)           (ms)        Cnt Num
0   192.168.1.102    Se0/0/0        10    00:07:22   10    2280  0   5
1   192.168.1.103    Se0/0/0        10    00:09:34   10    2320  0   9
R1#
```

```
R3# show ip eigrp neighbors
IP-EIGRP neighbors for process 100
H   Address          Interface      Hold  Uptime    SRTT   RTO  Q   Seq
                                    (sec)           (ms)        Cnt Num
0   192.168.1.101    Se0/0/0        10    00:11:45   10    1910  0   6
R3#
```

Frame Relay Using Static Mapping

- Deploying static maps automatically disables the inverse-ARP feature.
- Consists of four steps:
  1. Configure an IP address on the serial interface.
  2. Change the encapsulation on an interface using the **`encapsulation frame-relay`** command.
  3. Map the IP-to-DLCI mapping commands on the interface using the **`frame-relay map`** command.
  4. Activate the interface.

`frame-relay map` Command

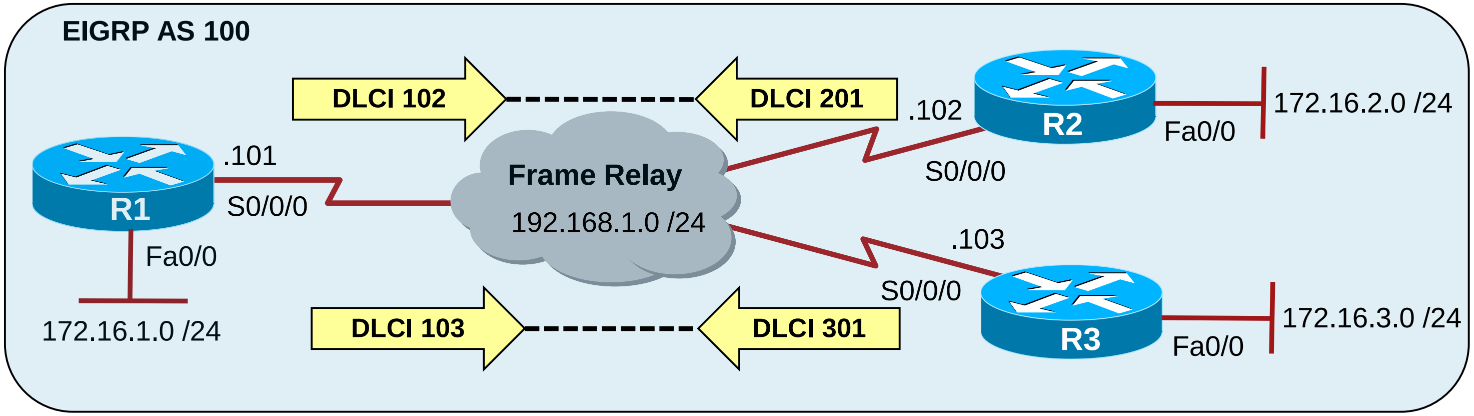
- Statically map the remote router's IP address to the local DLCI.

```
Router(config-if)#
```

```
frame-relay map protocol protocol-address dlci [broadcast] [ietf |
  cisco] [payload-compress {packet-by-packet | frf9 stack}]
```

Parameter	Description
protocol	Defines the supported protocol, bridging, or logical link control.
protocol-address	Defines the network layer address of the destination router interface.
dlci	Defines the local DLCI that is used to connect to the remote protocol address.
broadcast	(Optional) Allows broadcasts and multicasts over the VC, permitting the use of dynamic routing protocols over the VC.
ietf | cisco	Enables IETF or Cisco encapsulations.
payload-compress	(Optional) Enables payload compression.
packet-by-packet	(Optional) Enables packet-by-packet payload compression, using the Stacker method, a Cisco proprietary compression method.
frf9 stac	(Optional) Enables FRF.9 compression using the Stacker method.

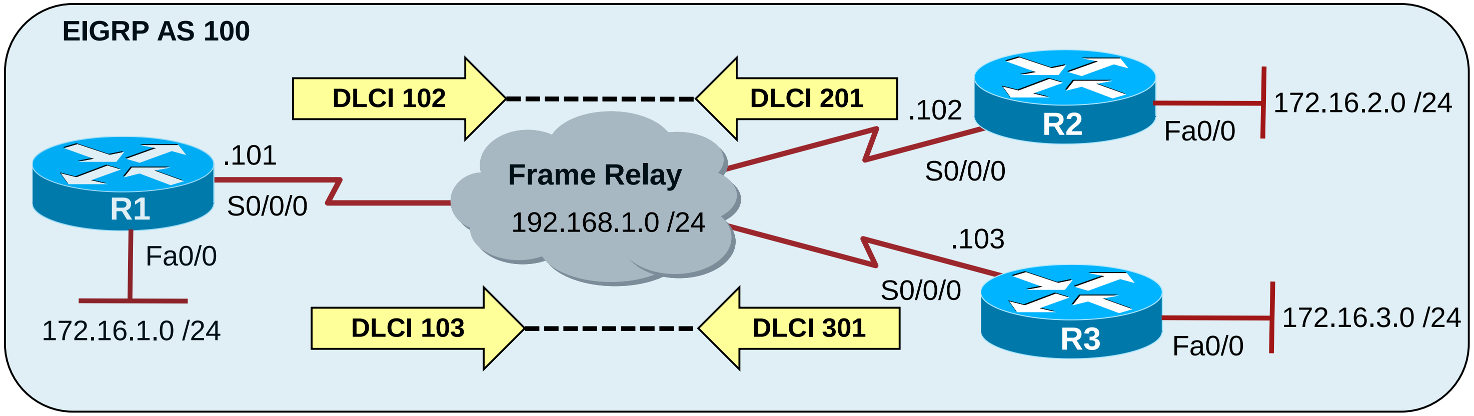
Frame Relay Using Static Mapping




```
R1(config)# interface S0/0/0
R1(config-if)# encapsulation frame-relay
R1(config-if)# ip address 192.168.1.101 255.255.255.0
R1(config-if)# frame-relay map ip 192.168.1.101 101
R1(config-if)# frame-relay map ip 192.168.1.102 102 broadcast
R1(config-if)# frame-relay map ip 192.168.1.103 103 broadcast
R1(config-if)#
```

```
R3(config)# interface S0/0/0
R3(config-if)# encapsulation frame-relay
R3(config-if)# ip address 192.168.1.103 255.255.255.0
R3(config-if)# frame-relay map ip 192.168.1.101 301
R3(config-if)# frame-relay map ip 192.168.1.102 301 broadcast
R3(config-if)# frame-relay map ip 192.168.1.103 301 broadcast
R3(config-if)#
```

Frame Relay Using Static Mapping




```
R1# show ip eigrp neighbors
IP-EIGRP neighbors for process 100
H   Address          Interface     Hold  Uptime    SRTT   RTO  Q   Seq
                                   (sec)           (ms)        Cnt Num
0   192.168.1.102    Se0/0/0        10    00:06:20   10   2280  0   5
1   192.168.1.103    Se0/0/0        10    00:08:31   10   2320  0   9
```

```
R3# show ip eigrp neighbors
IP-EIGRP neighbors for process 100
H   Address          Interface     Hold  Uptime    SRTT   RTO  Q   Seq
                                   (sec)           (ms)        Cnt Num
0   192.168.1.101    Se0/0/0       10    00:10:44   10   1910  0   6
1   192.168.1.102    Se0/0/0       10    00:03:02   10   2210  0   3
```

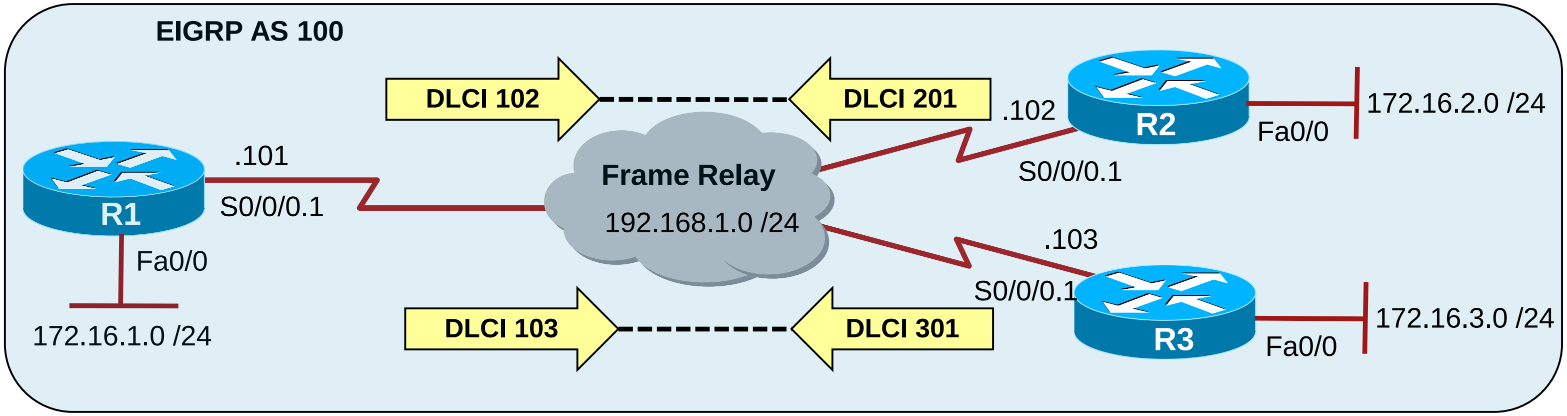
EIGRP over FR Multipoint Subinterfaces

- Multipoint subinterfaces can be created using a single Frame Relay physical interface.
  - Uses a single subnet, preserving the IP address space.
- Frame Relay multipoint is applicable to partial mesh and full mesh topologies.
- Consists of several steps:
  - Configure the physical interface with no IP address and change the encapsulation to Frame Relay.
  - Create a serial multipoint subinterface.
  - Configure an IP address on the serial interface.
  - Map the IP-to-DLCI mapping commands on the interface using the **`frame-relay map`** command.
  - Either rely on dynamic mapping or configure a local DLCI value using the **`frame-relay interface-dlci`** command.

EIGRP over FR Multipoint Subinterfaces

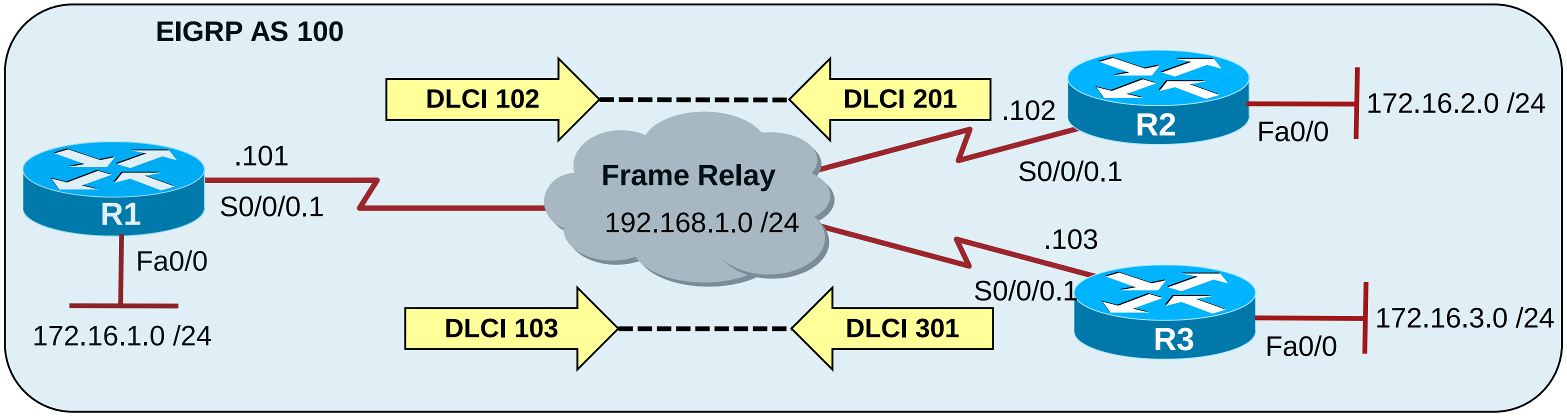
- Multipoint subinterfaces are configured with the **interface serial** *number.subinterface-number* **multipoint** command.
- The IP address-to-DLCI mapping is done by either:
  - Specifying the local DLCI value (using the **frame-relay interface-dlci** *dlci* command) and relying on Inverse ARP
  - Using manual IP address-to-DLCI mapping.

EIGRP over FR Multipoint Subinterfaces




```
R1(config)# interface S0/0/0
R1(config-if)# no ip address
R1(config-if)# encapsulation frame-relay
R1(config-if)# exit
R1(config)# interface Serial0/0/0.1 multipoint
R1(config-subif)# ip address 192.168.1.101 255.255.255.0
R1(config-subif)# no ip split-horizon eigrp 100
R1(config-subif)# frame-relay map ip 192.168.1.101 101
R1(config-subif)# frame-relay map ip 192.168.1.102 102 broadcast
R1(config-subif)# frame-relay map ip 192.168.1.103 103 broadcast
R1(config-subif)#
```

EIGRP over FR Multipoint Subinterfaces




```
R1# show ip eigrp neighbors
IP-EIGRP neighbors for process 100
H   Address          Interface    Hold  Uptime    SRTT   RTO  Q   Seq
                                  (sec)           (ms)        Cnt Num
0    192.168.1.102    Se0/0/0.1   10    00:06:41   10    2280  0   5
1    192.168.1.103    Se0/0/0.1   10    00:08:52   10    2320  0   9
```

```
R3# show ip eigrp neighbors
IP-EIGRP neighbors for process 100
H   Address          Interface    Hold  Uptime    SRTT   RTO  Q   Seq
                                  (sec)           (ms)        Cnt Num
0    192.168.1.101    Se0/0/0.1   10    00:10:37   10    1910  0   6
1    192.168.1.102    Se0/0/0.1   10    00:03:12   10    2210  0   3
```

EIGRP Unicast Neighbors

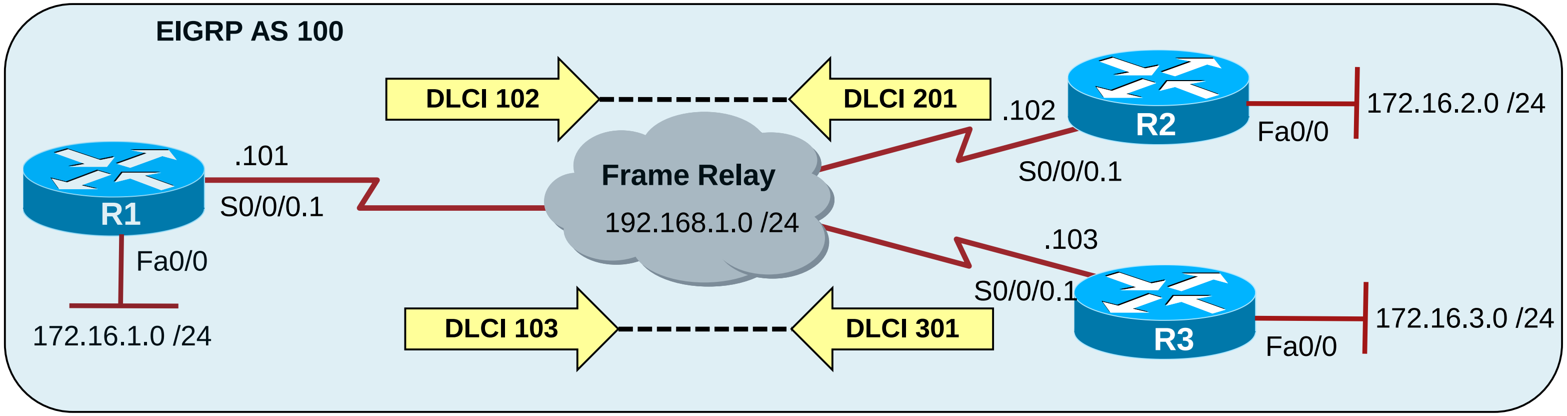
- Define a neighboring router to exchange EIGRP routing information.

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

```
neighbor {ip-address | ipv6-address} interface-type
  interface-number
```

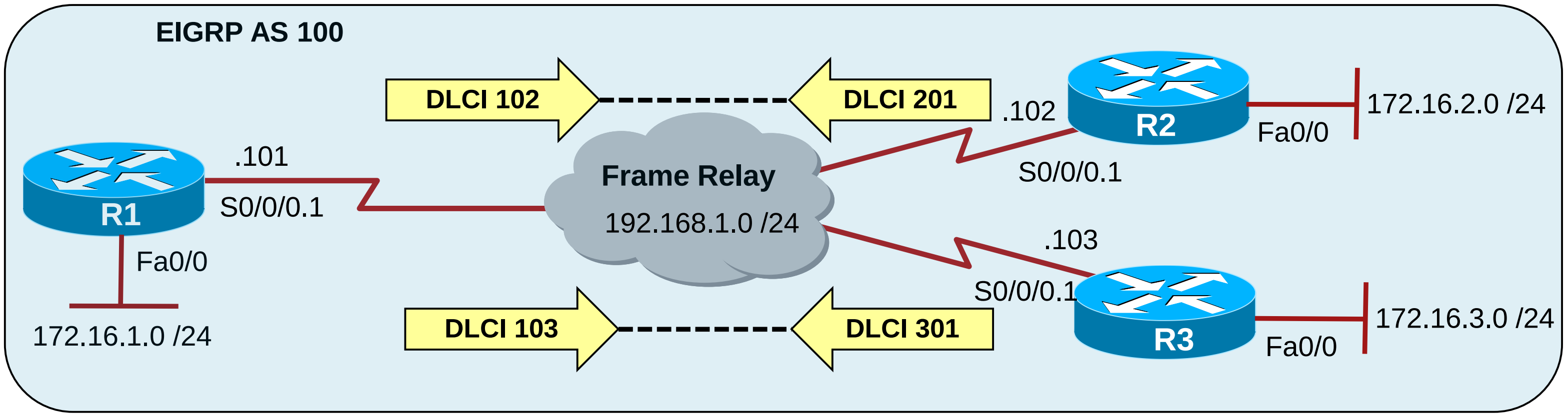
- EIGRP exchanges routing information with the specified neighbor using unicast packets.
- Multiple neighbor statements can be used to establish peering sessions with multiple specific EIGRP neighbors.
- The interface through which EIGRP will exchange routing updates must be specified in the neighbor statement.
- The interfaces through which two EIGRP neighbors exchange routing updates must be configured with IP addresses from the same network.

EIGRP Unicast Neighbors




```
R1(config)# interface S0/0/0
R1(config-if)# no ip address
R1(config-if)# encapsulation frame-relay
R1(config-if)# interface S0/0/0.1 multipoint
R1(config-subif)# ip address 192.168.1.101 255.255.255.0
R1(config-subif)# frame-relay map ip 192.168.1.102 102 broadcast
R1(config-subif)# frame-relay map ip 192.168.1.103 103 broadcast
R1(config-subif)# router eigrp 100
R1(config-router)# neighbor 192.168.1.102 S0/0/0.1
R1(config-router)#
```

EIGRP Unicast Neighbors




```
R2(config)# interface S0/0/0
R2(config-if)# no ip address
R2(config-if)# encapsulation frame-relay
R2(config-if)# interface S0/0/0.1 multipoint
R2(config-subif)# ip address 192.168.1.102 255.255.255.0
R2(config-subif)# frame-relay map ip 192.168.1.101 201 broadcast
R2(config-subif)# router eigrp 100
R2(config-router)# neighbor 192.168.1.101 S0/0/0.1
R2(config-router)#
```

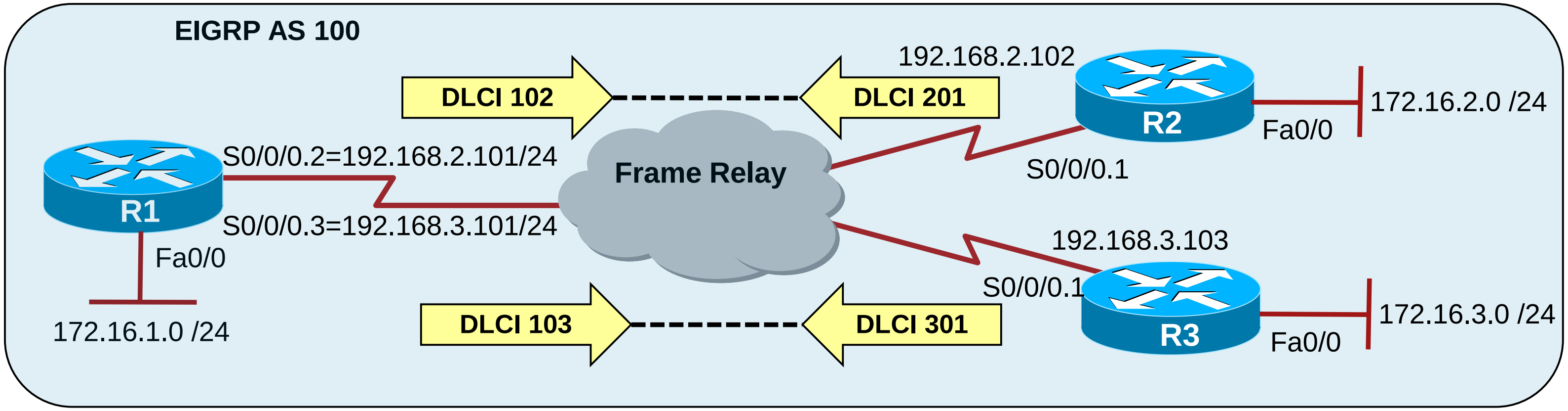
EIGRP over FR Point-to-Point Subinterfaces

- Point-to-point subinterfaces can be created using a single Frame Relay physical interface.
  - Uses multiple subnets.
- Frame Relay point-to point is applicable to hub and spoke topologies.
- Consists of several steps:
  - Configure the physical interface with no IP address and change the encapsulation to Frame Relay.
  - Create a serial point-to-point subinterface.
  - Configure an IP address on the serial interface.
  - Configure a local DLCI value using the **`frame-relay interface-dlci`** command.

EIGRP over FR Point-to-Point Subinterfaces

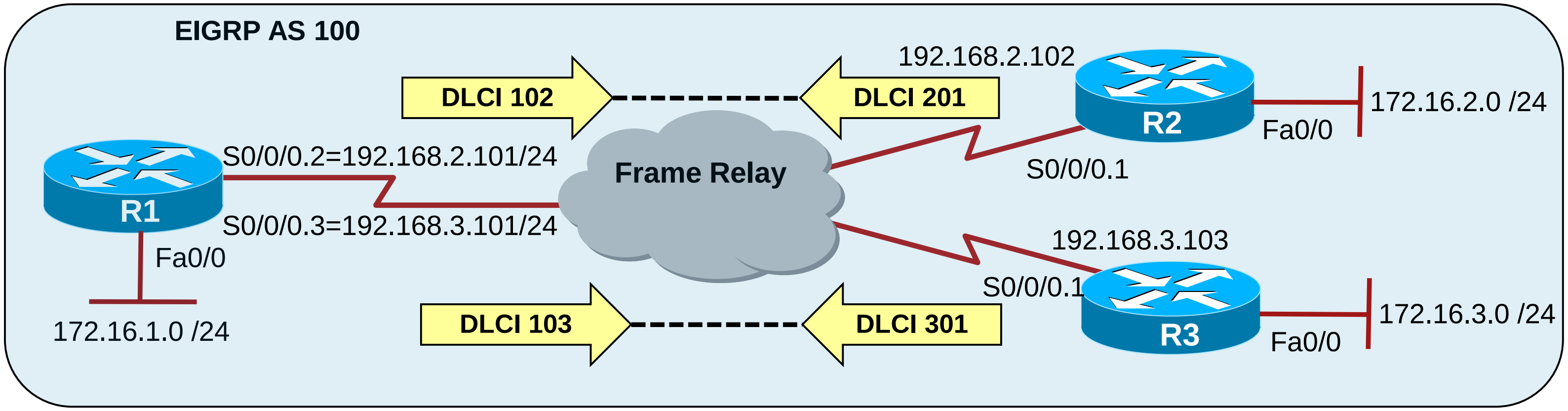
- Multipoint subinterfaces are configured with the **interface serial** *number.subinterface-number* **multipoint** command.
- The IP address-to-DLCI mapping is done by either:
  - Specifying the local DLCI value (using the **frame-relay interface-dlci** *dlci* command) and relying on Inverse ARP.

EIGRP over FR Point-to-Point Subinterfaces




```
R1(config)# interface S0/0/0
R1(config-if)# no ip address
R1(config-if)# encapsulation frame-relay
R1(config-if)# exit
R1(config)# interface Serial0/0/0.2 point-to-point
R1(config-subif)# ip address 192.168.2.101 255.255.255.0
R1(config-subif)# frame-relay interface-dlci 102
R1(config-subif)#
R1(config-subif)# interface Serial0/0/0.3 point-to-point
R1(config-subif)# ip address 192.168.3.101 255.255.255.0
R1(config-subif)# frame-relay interface-dlci 103
R1(config-subif)#
```

EIGRP over FR Point-to-Point Subinterfaces



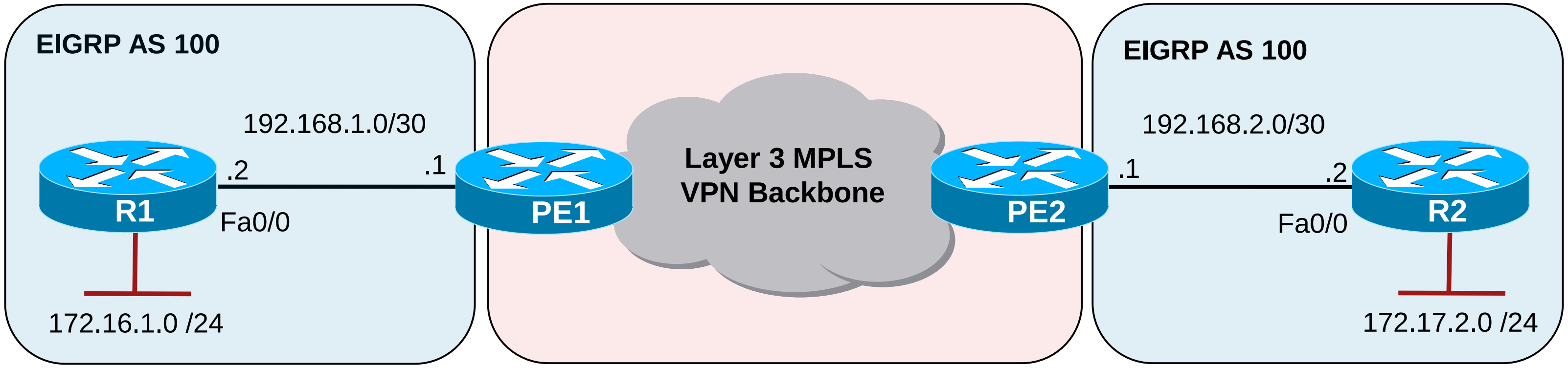

```
R1# show ip eigrp neighbors
IP-EIGRP neighbors for process 100
H   Address          Interface     Hold  Uptime    SRTT   RTO   Q   Seq
                                   (sec)           (ms)        Cnt  Num
0    192.168.2.102   Se0/0/0.2     10    00:08:04   10    2280   0   5
1    192.168.3.103   Se0/0/0.3       10    00:10:12   10    2320   0   9
```

```
R3# show ip eigrp neighbors
IP-EIGRP neighbors for process 100
H   Address          Interface     Hold  Uptime    SRTT   RTO   Q   Seq
                                   (sec)           (ms)        Cnt  Num
0    192.168.3.101   Se0/0/0.1     10    00:13:25   10    1910   0   6
```

EIGRP over MPLS

- Multi-Protocol Label Switching (MPLS) is an Internet Engineering Task Force (IETF) standard architecture that combines the advantages of Layer 3 routing with the benefits of Layer 2 switching.
- A unique feature of MPLS is its capability to perform label stacking, in which multiple labels can be carried in a packet.
- The top label, which is the last one in, is always processed first.
  - Label stacking enables multiple LSPs to be aggregated, thereby creating tunnels through multiple levels of an MPLS network.

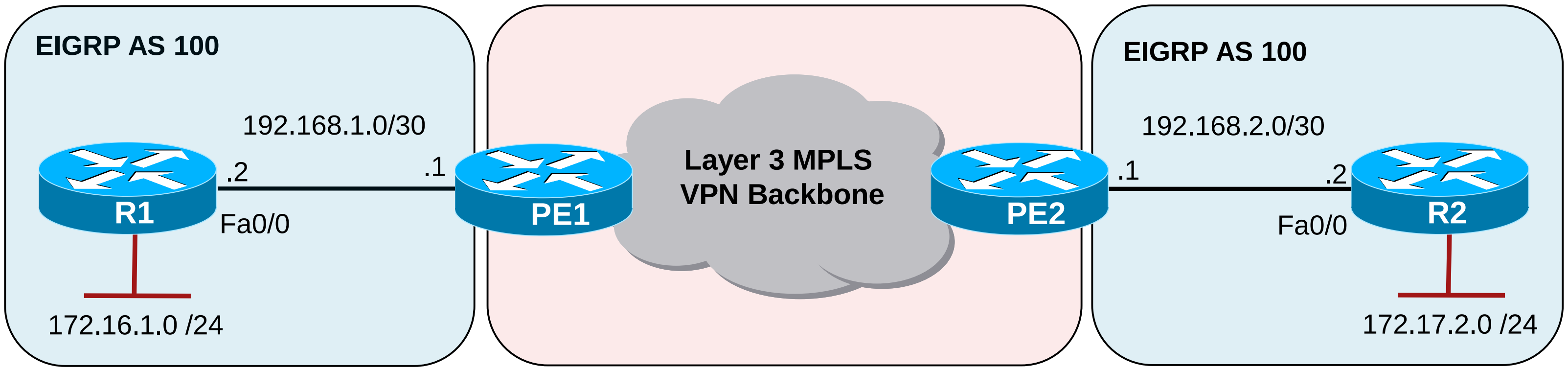
EIGRP over Layer 3 MPLS VPNs




```
R1(config)# interface FastEthernet0/0
R1(config-if)# ip address 192.168.1.2 255.255.255.252
R1(config-if)# exit
R1(config)# router eigrp 100
R1(config-router)# network 172.16.1.0 0.0.0.255
R1(config-router)# network 192.168.1.0
R1(config-router)#
```

```
R2(config)# interface FastEthernet0/0
R2(config-if)# ip address 192.168.2.2 255.255.255.252
R2(config-if)# exit
R2(config)# router eigrp 100
R2(config-router)# network 172.17.2.0 0.0.0.255
R2(config-router)# network 192.168.2.0
R2(config-router)#
```

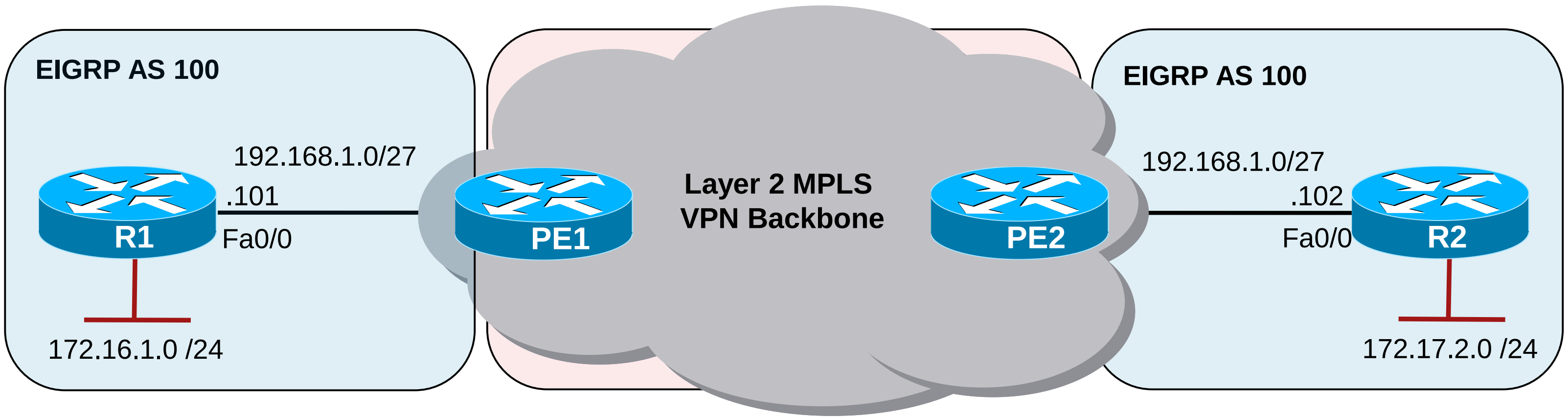
EIGRP over Layer 3 MPLS VPNs




```
R1# show ip eigrp neighbors
IP-EIGRP neighbors for process 100
H    Address          Interface    Hold  Uptime    SRTT    RTO  Q   Seq
                                   (sec)           (ms)         Cnt Num
0    192.168.1.1      Fe0/0        10    00:07:22   10    2280  0   5
```

```
R2# show ip eigrp neighbors
IP-EIGRP neighbors for process 100
H    Address          Interface    Hold  Uptime    SRTT    RTO  Q   Seq
                                   (sec)           (ms)         Cnt Num
0    192.168.2.1      Fe0/0        10    00:17:02   10    1380  0   5
```

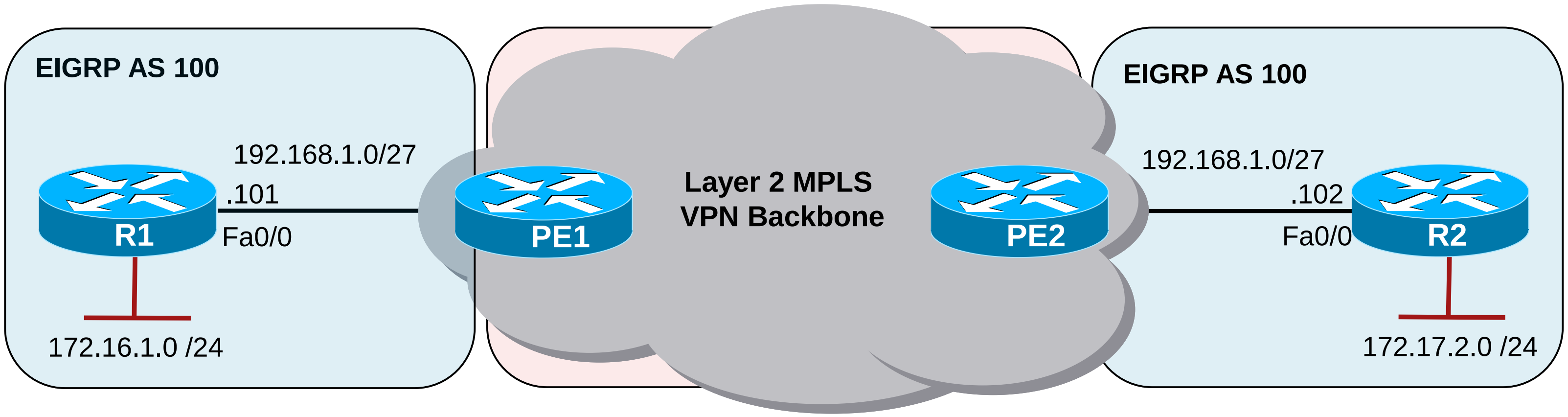
EIGRP over Layer 2 MPLS VPNs




```
R1(config)# interface FastEthernet0/0
R1(config-if)# ip address 192.168.1.101 255.255.255.224
R1(config-if)# exit
R1(config)# router eigrp 100
R1(config-router)# network 172.16.1.0 0.0.0.255
R1(config-router)# network 192.168.1.0
R1(config-router)#
```

```
R2(config)# interface FastEthernet0/0
R2(config-if)# ip address 192.168.1.102 255.255.255.224
R2(config-if)# exit
R2(config)# router eigrp 100
R2(config-router)# network 172.17.2.0 0.0.0.255
R2(config-router)# network 192.168.1.0
R2(config-router)#
```

EIGRP over Layer 2 MPLS VPNs



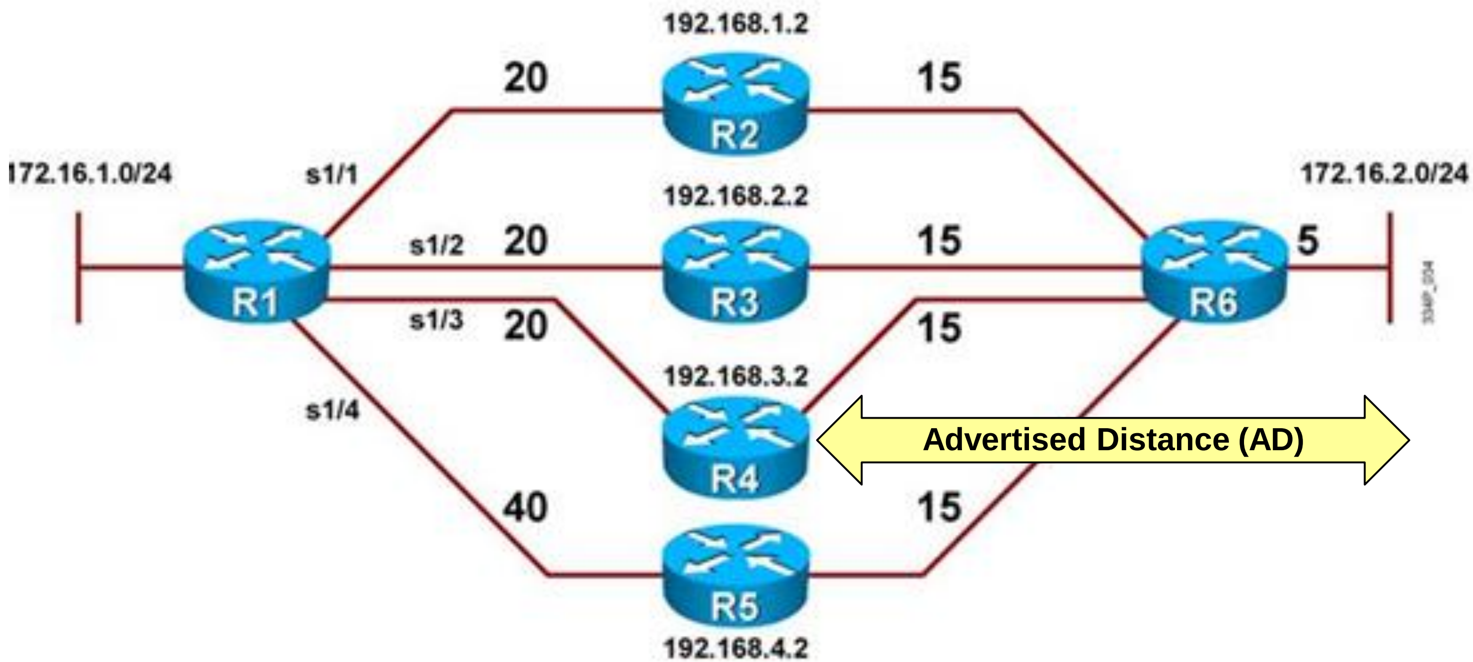

```
R1# show ip eigrp neighbors
IP-EIGRP neighbors for process 100
H    Address           Interface     Hold  Uptime    SRTT    RTO   Q   Seq
                                     (sec)           (ms)          Cnt  Num
0    192.168.1.102     Fe0/0         10    00:07:22   10     2280   0   5
```

```
R2# show ip eigrp neighbors
IP-EIGRP neighbors for process 100
H    Address           Interface     Hold  Uptime    SRTT    RTO   Q   Seq
                                     (sec)           (ms)          Cnt  Num
0    192.168.1.101     Fe0/0         10    00:17:02   10     1380   0   5
```

EIGRP Load Balancing

- Routes with a metric equal to the minimum metric are installed in the routing table.
  - Referred to as "equal-cost load balancing".
  - All IP routing protocols on Cisco routers can perform equal-cost load balancing.
- The **`maximum-paths`** *`maximum-path`* command can be used to allow up to 32 equal-cost paths.
  - Default is 4.
  - Setting the *`maximum-path`* option to 1 disables load balancing.

EIGRP Equal-Cost Load Balancing




```
R1(config)# router eigrp 100
R1(config-router)# network 172.16.1.0 0.0.0.255
R1(config-router)# network 192.168.1.0
R1(config-router)# network 192.168.2.0
R1(config-router)# network 192.168.3.0
R1(config-router)# network 192.168.4.0
R1(config-router)# maximum-paths 2
R1(config-router)#
```

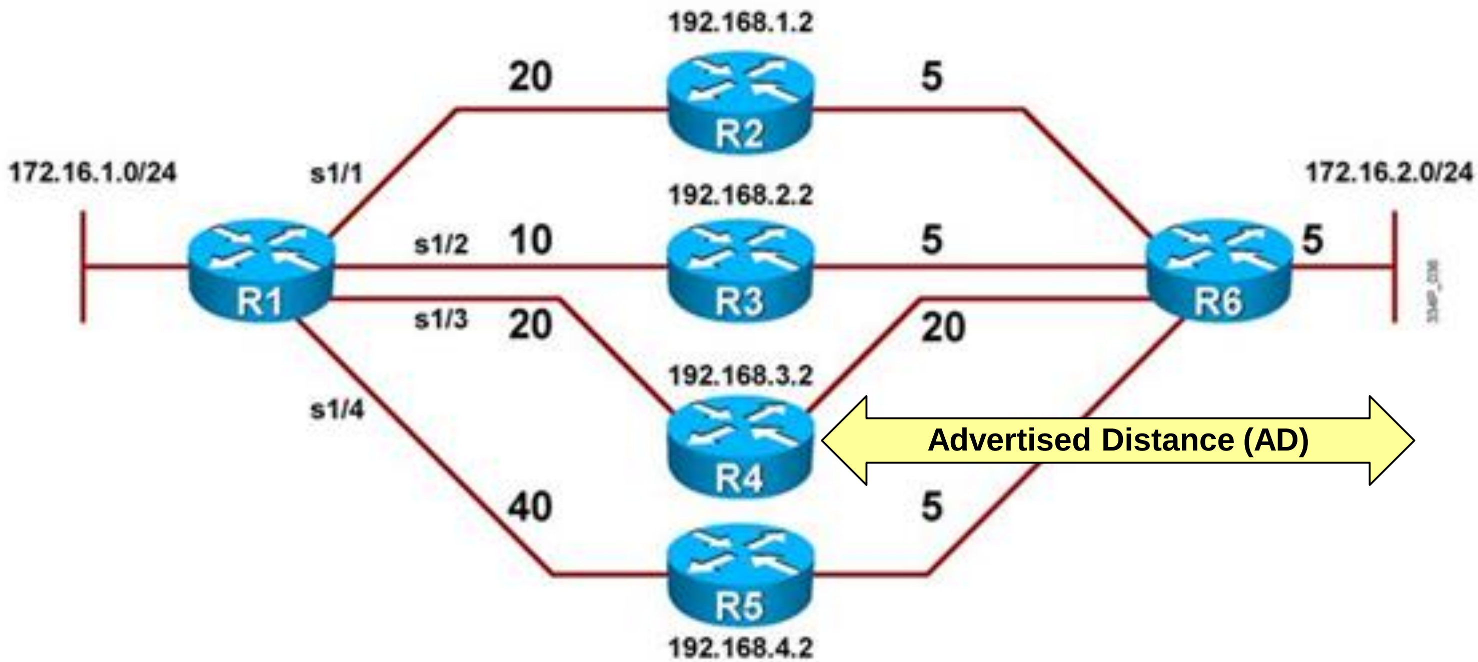
R1 Topology Table

Network	Neighbor	AD	FD
172.16.2.0/24	R2	20	40
	R3	20	40
	R4	20	40
	R5	20	60

Unequal Cost Load Balancing

- EIGRP can also balance traffic across multiple routes that have different metrics.
  - Referred to as unequal-cost load balancing.
- The degree to which EIGRP performs load balancing is controlled with the **`variance`** *`multiplier`* command.
  - The multiplier is a value, between 1 and 128, used for load balancing.
  - The default is 1, which means equal-cost load balancing.
  - Setting a variance value greater than 1 allows EIGRP to install multiple loop-free routes with unequal cost in the routing table.
  - EIGRP will always install successors (the best routes) in the routing table.
    - The variance allows feasible successors (and only feasible successor routes) as candidate routes to potentially be installed in the routing table.

EIGRP Unequal-Cost Load Balancing

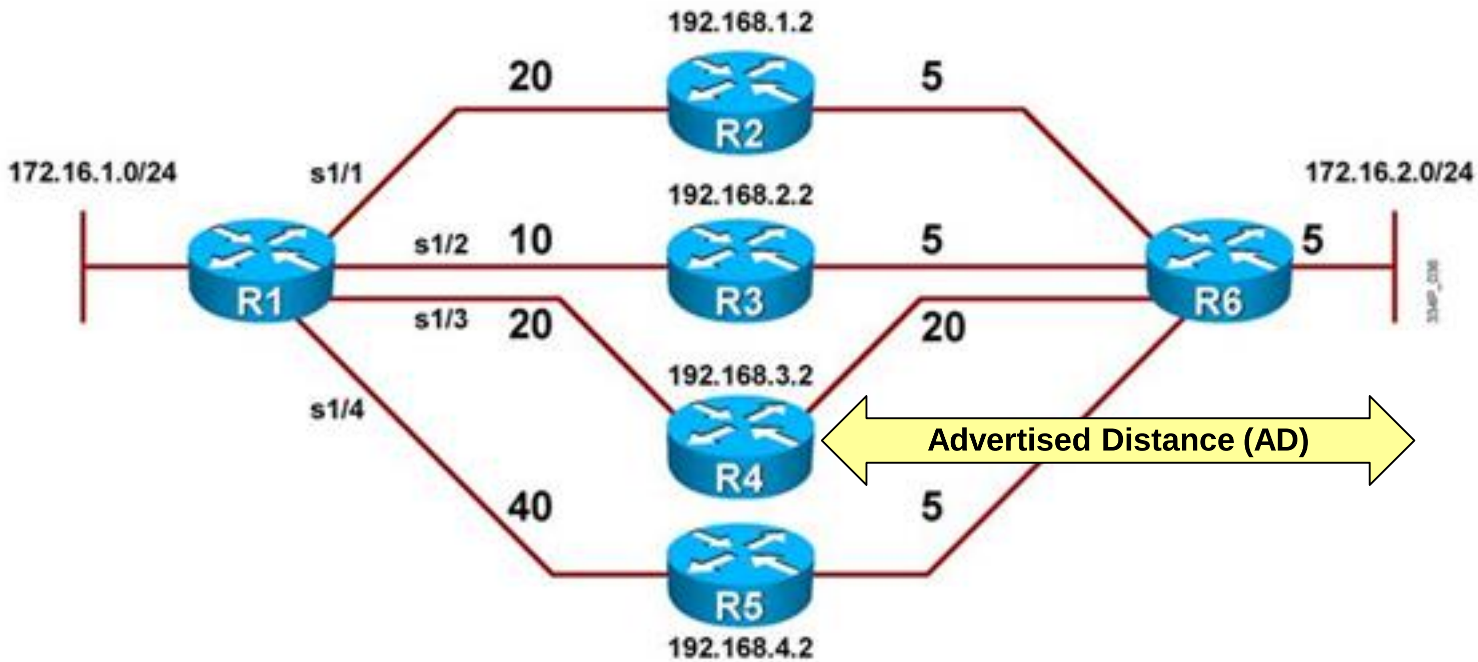



```
R1(config)# router eigrp 100
R1(config-router)# variance 2
R1(config-router)#
```

R1 Topology Table

Network	Neighbor	AD	FD
172.16.2.0/24	R2	10	30
	R3	10	20
	R4	25	45
	R5	10	50

EIGRP Unequal-Cost Load Balancing




```
R1(config)# router eigrp 100
R1(config-router)# variance 3
R1(config-router)#
```

R1 Topology Table

Network	Neighbor	AD	FD
172.16.2.0/24	R2	10	30
	R3	10	20
	R4	25	45
	R5	10	50

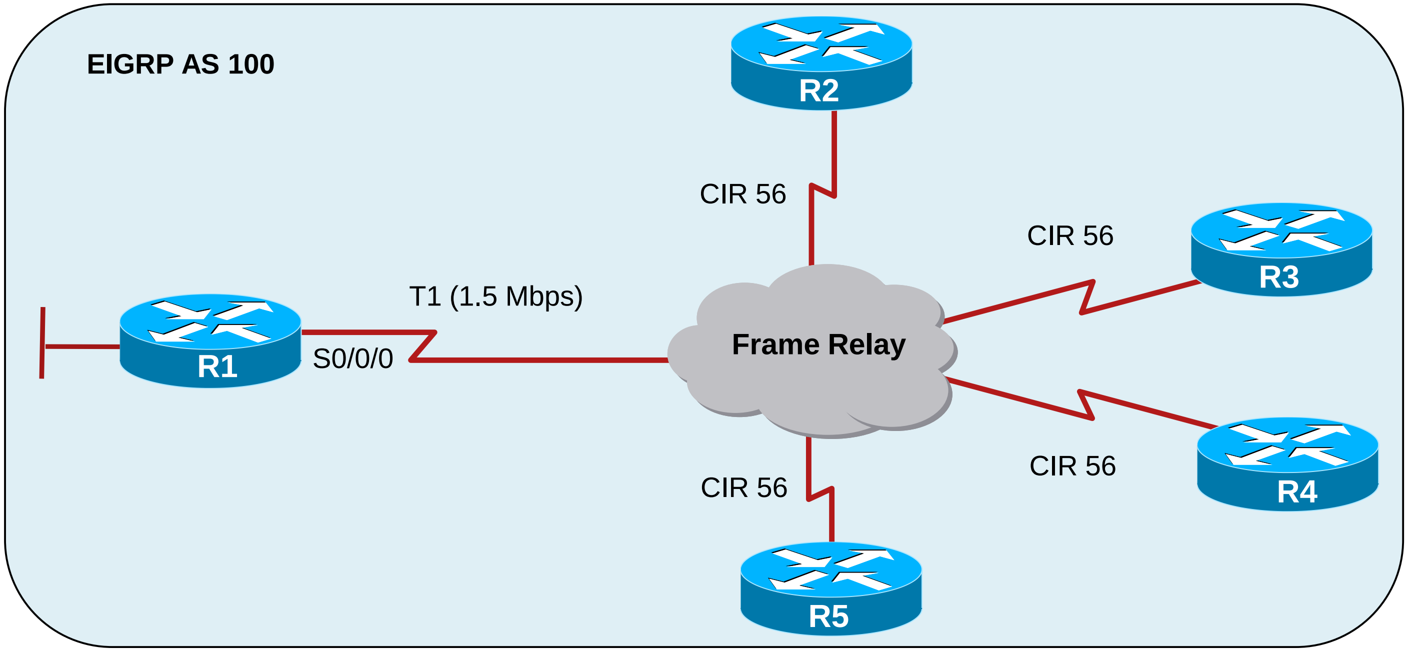
EIGRP Bandwidth Use Across WAN Links

- EIGRP operates efficiently in WAN environments and is scalable on both point-to-point links and NBMA multipoint and point-to-point links.
- However, default configuration of WAN connections may not be optimal therefore a solid understanding of EIGRP operation coupled with knowledge of link speeds can yield an efficient, reliable, scalable router configuration.
- There are two commands which could be configured to improve EIGRP operation:
  - **`bandwidth`**
  - **`ip percent-bandwidth`**

Setting EIGRP Bandwidth on a WAN

- EIGRP assumes the default bandwidth on the link instead of the true bandwidth, therefore suboptimal path selection may result.
  - For example, Serial links commonly default to 1.5 Mbps however the actual CIR may be 128 Kbps.
  - DUAL would use the 1.5 Mbps value instead of the actual slower 128 Kbps value in its metric calculation.
- It is recommended to configure the bandwidth setting using the **`bandwidth`** *`kilobits`* on serial interfaces.
- An important WAN consideration is the fact that multipoint interfaces physical bandwidth setting is shared equally by all neighbors.
  - EIGRP uses the **`bandwidth`** setting of the physical interface divided by the number of Frame Relay neighbors connected on that physical interface to get the bandwidth attributed to each neighbor.
  - The EIGRP configuration should reflect the correct percentage of the actual available bandwidth on the line.

EIGRP WAN Configuration – Example #1

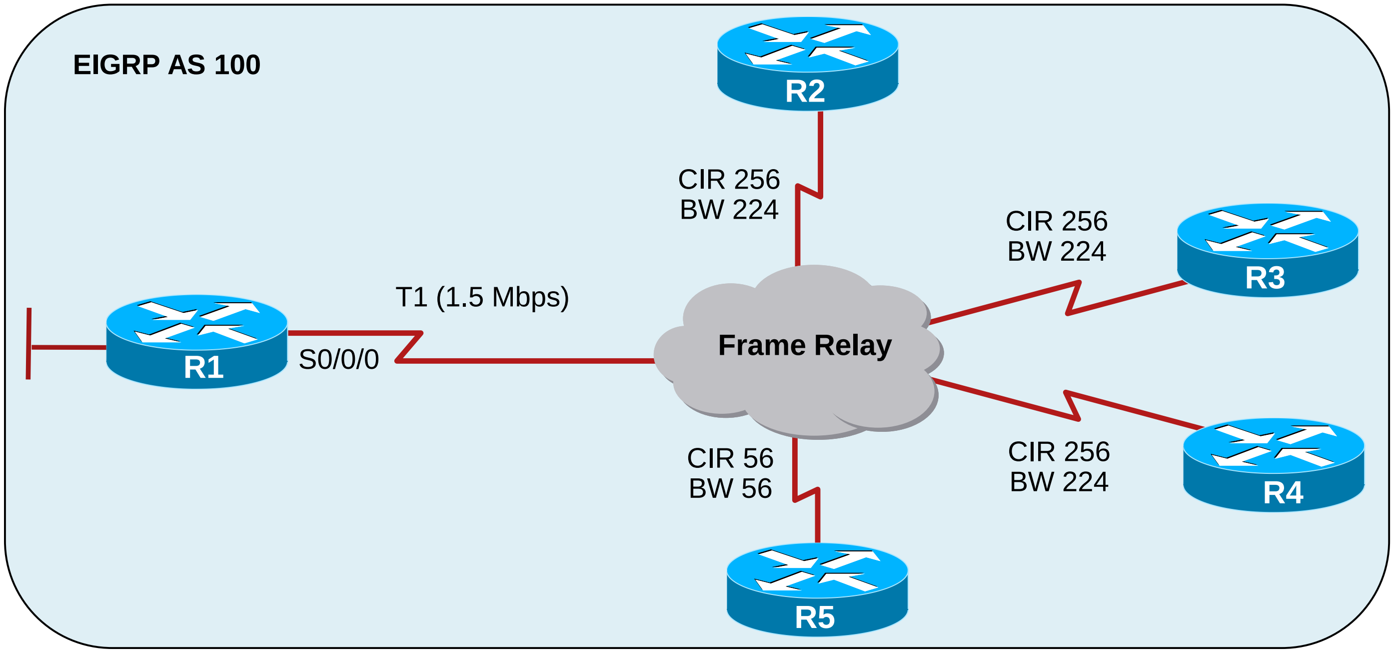



```
R1(config)#  interface S0/0/0
R1(config-if)# encapsulation frame-relay
R1(config-if)# bandwidth 224
```

- All VCs share the bandwidth evenly:

  4 (VC) x 56 (CIR) = 224

EIGRP WAN Configuration – Example #2a

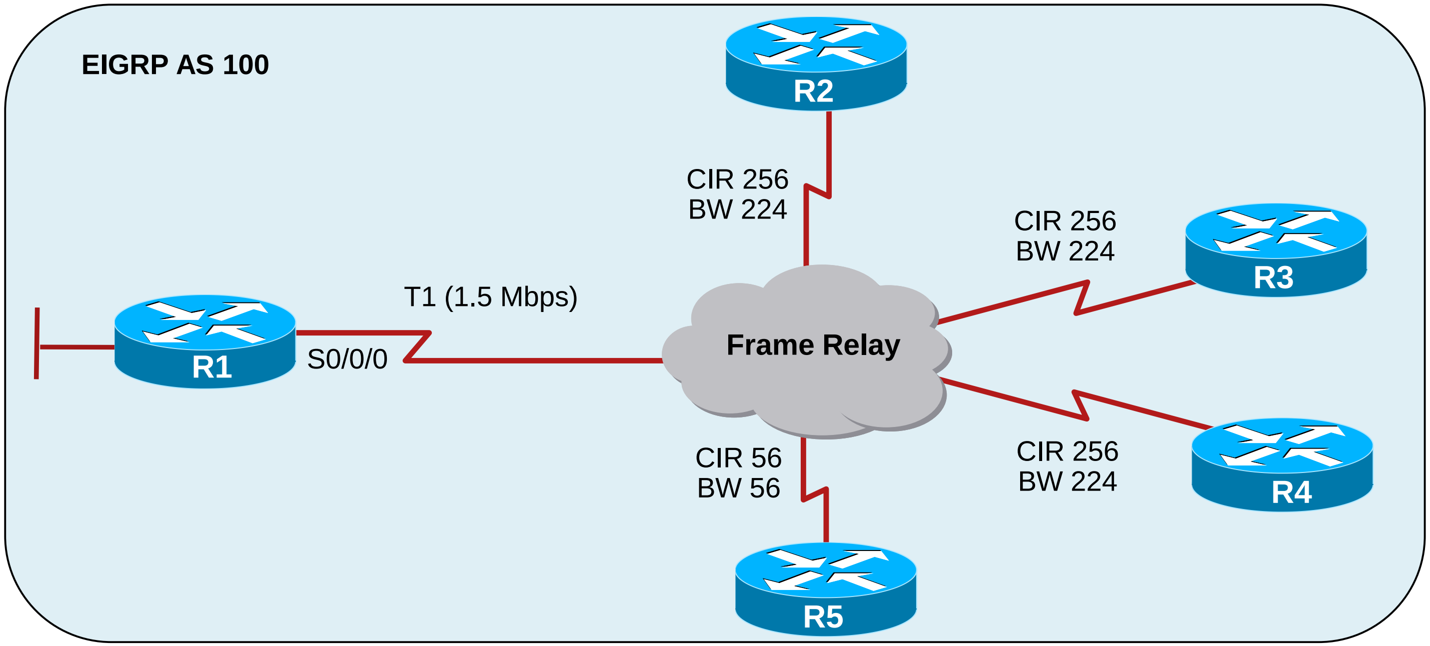



```
R1(config)#  interface S0/0/0
R1(config-if)# encapsulation frame-relay
R1(config-if)# bandwidth 224
```

- R2, R3, and R4 share the same CIR. To protect against overwhelming the slowest-speed circuit (to R5) the bandwidth should be configured to the lowest CIR multiplied by the number of circuits.

  4 (VC) x 56 (CIR) = 224

EIGRP WAN Configuration – Example #2b




```
R1(config)#  interface S0/0/0
R1(config-if)# encapsulation frame-relay
R1(config-if)# interface S0/0/0/0.1 multipoint
R1(config-subif)# bandwidth 768
R1(config-subif)# exit
R1(config)# interface S0/0/0/0.2 point-to-point
R1(config-subif)# bandwidth 56
R1(config-subif)#
```

- An alternate solution is to configure a multipoint subinterface for routers R2, R3, and R4 and configure a point-to-point subinterface network for R5.

EIGRP ip bandwith-percent

- By default, EIGRP is set to use up to 50% of the bandwidth of an interface to exchange routing information.
- The **`ip bandwidth-percent eigrp`** command can be configured to control the amount of bandwidth available to EIGRP.
  - This is not the same as the **`bandwidth`** command.
  - However, this command relies on the value set by the bandwidth command.

ip bandwidth-percent eigrp

- Configure the amount of bandwidth available to EIGRP.

```
Router(config-if)#
```

```
ip bandwidth-percent eigrp as-number percent
```

- The *as-number* is the EIGRP AS number.
- The *percent* parameter is the percentage of the configured bandwidth that EIGRP can use.
- The percentage value can be set to greater than 100.

Configuring and Verifying EIGRP Authentication

Router Authentication

- Many routing protocols support authentication such that a router authenticates the source of each routing update packet that it receives.

- Simple password authentication is supported by:
  - IS-IS
  - OSPF
  - RIPv2

- MD5 authentication is supported by:
  - OSPF
  - RIPv2
  - BGP
  - EIGRP

Simple Password vs. MD5 Authentication

- Simple password authentication:
  - Router sends packet and key.
  - Neighbor checks if received key matches its key.
  - Is not secure.

- MD5 authentication:
  - Configure a “key” (password) and key-id; router generates a message digest, or hash, of the key, key-id and message.
  - Message digest is sent with packet; key is not sent.
  - Is secure.

EIGRP MD5 Authentication

- EIGRP supports MD5 authentication.
- Router generates and checks every EIGRP packet. Router authenticates the source of each routing update packet that it receives.
- Configure a “key” (password) and key-id; each participating neighbor must have same key configured.

MD5 Authentication

- EIGRP MD5 authentication:
  - Router generates a message digest, or hash, of the key, key-id, and message.
  - EIGRP allows keys to be managed using key chains.
  - Specify key-id (number, key, and lifetime of key).
  - First valid activated key, in order of key numbers, is used.

Planning for EIGRP

- The following key parameters must be defined in enough detail before configuring EIGRP authentication:
  - The EIGRP AS number
  - The authentication mode (MD5)
  - The definition of one or more keys to authenticate EIGRP packets, according to the network security plan.
  - The keys’ lifetime, if multiple keys are defined.

- Once defined, the following steps may be implemented:
  1. Configure the authentication mode for EIGRP.
  2. Configure the key chain.
  3. Optionally configure the keys’ lifetime parameters.
  4. Enable authentication to use the key(s) in the key chain.

Configure the Authentication Mode for EIGRP

- Specify MD5 authentication for EIGRP packets.

```
Router(config-if)#
```

ip authentication mode eigrp *autonomous-system* **md5**

- Enable EIGRP packet authentication using key in the *key-chain*.

```
Router(config-if)#
```

ip authentication key-chain eigrp *autonomous-system name-of-chain*

Configure the Key Chain

- Define the keychain in key chain configuration mode.

```
Router(config)#
```

```
key chain name-of-chain
```

- Identify the key and enter the *key-id* configuration mode.

```
Router(config-keychain)#
```

```
key key-id
```

- Identify key string (password)

```
Router(config-keychain-key)#
```

```
key-string text
```

Configure Keys Lifetime Parameters (Optional)

- Specify when the key will be accepted for received packets.

```
Router(config-keychain-key)#
```

accept-lifetime *start-time* **{infinite |** *end-time* **|**
 duration *seconds***}**

- Specify when the key can be used for sending EIGRP packets.

```
Router(config-keychain-key)#
```

send-lifetime *start-time* **{infinite |** *end-time* **|**
 duration *seconds***}**

Enable Authentication to Use the Key Chain

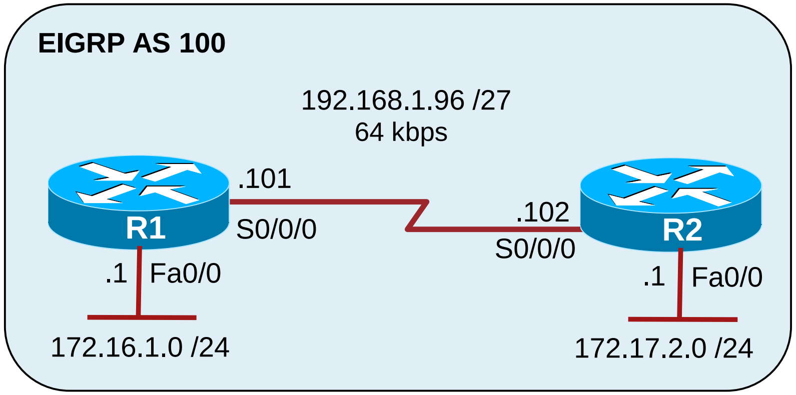
- Enable EIGRP packet authentication using key in the key-chain.

```
Router(config-if)#
```

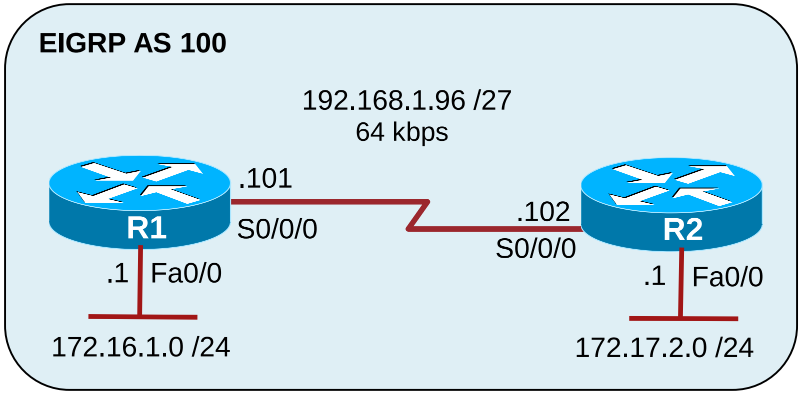
```
ip authentication key-chain eigrp autonomous-system
  name-of-chain
```

Configuring EIGRP MD5 Authentication

```
R1# show running-config
!
<output omitted>
!
key chain R1chain
 key 1
  key-string FIRST-KEY
  accept-lifetime 04:00:00 Jan 1 2015 infinite
  send-lifetime 04:00:00 Jan 1 2015 04:00:00 Jan 31 2015
 key 2
  key-string SECOND-KEY
  accept-lifetime 04:00:00 Jan 25 2015 infinite
  send-lifetime 04:00:00 Jan 25 2015 infinite
!
<output omitted>
!
interface FastEthernet0/0
ip address 172.16.1.1 255.255.255.0
!
interface Serial0/0/0
 bandwidth 64
 ip address 192.168.1.101 255.255.255.224
 ip authentication mode eigrp 100 md5
 ip authentication key-chain eigrp 100 R1chain
!
router eigrp 100
 network 172.16.1.0 0.0.0.255
 network 192.168.1.0
```



Configuring EIGRP MD5 Authentication




```
R2# show running-config
!
<output omitted>
!
key chain R2chain
 key 1
  key-string FIRST-KEY
  accept-lifetime 04:00:00 Jan 1 2015 infinite
  send-lifetime 04:00:00 Jan 1 2015 infinite
 key 2
  key-string SECOND-KEY
  accept-lifetime 04:00:00 Jan 25 2015 infinite
  send-lifetime 04:00:00 Jan 25 2015 infinite
!
<output omitted>
!
interface FastEthernet0/0
 ip address 172.17.2.2 255.255.255.0
!
interface Serial0/0/0
 bandwidth 64
 ip address 192.168.1.102 255.255.255.224
 ip authentication mode eigrp 100 md5
 ip authentication key-chain eigrp 100 R2chain
!
router eigrp 100
 network 172.17.2.0 0.0.0.255
 network 192.168.1.0
```

Verifying MD5 Authentication

```
R1#
*Apr 21 16:23:30.517: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 100: Neighbor 192.168.1.102
(Serial0/0/0) is up: new adjacency
R1#
R1# show ip eigrp neighbors
IP-EIGRP neighbors for process 100
H   Address                 Interface       Hold Uptime   SRTT   RTO  Q  Seq
                                            (sec)         (ms)       Cnt Num
0   192.168.1.102           Se0/0/0           12 00:03:10   17   2280  0  14
R1#
R1# show ip route

<output omitted>

Gateway of last resort is not set
D    172.17.0.0/16 [90/40514560] via 192.168.1.102, 00:02:22, Serial0/0/0
     172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
D       172.16.0.0/16 is a summary, 00:31:31, Null0
C       172.16.1.0/24 is directly connected, FastEthernet0/0
     192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.1.96/27 is directly connected, Serial0/0/0
D       192.168.1.0/24 is a summary, 00:31:31, Null0
R1#
R1# ping 172.17.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.17.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 12/15/16 ms
```

Verifying MD5 Authentication

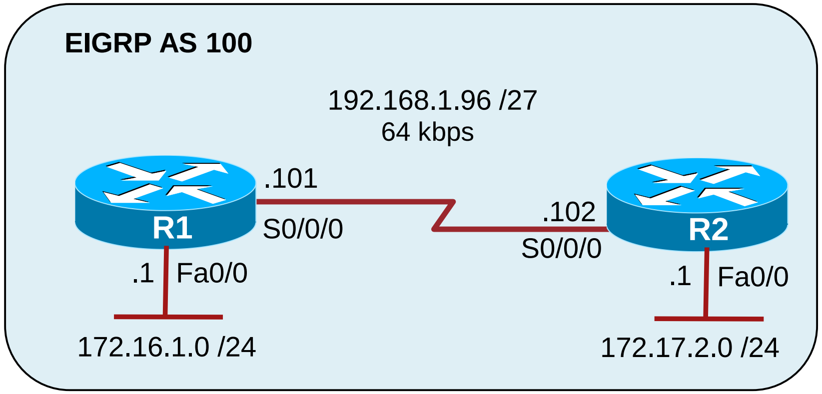
```
R1# show key chain
Key-chain R1chain:
    key 1 -- text "FIRST-KEY"
        accept lifetime (04:00:00 Jan 1 2015) - (always valid) [valid now]
        send lifetime (04:00:00 Jan 1 2015) - (04:00:00 Jan 31 2015)
    key 2 -- text "SECOND-KEY"
        accept lifetime (04:00:00 Jan 25 2015) - (always valid) [valid now]
        send lifetime (04:00:00 Jan 25 2015) - (always valid) [valid now]
```

Troubleshooting MD5 Authentication

```
R1# debug eigrp packets
EIGRP Packets debugging is on
    (UPDATE, REQUEST, QUERY, REPLY, HELLO, IPXSAP, PROBE, ACK, STUB, SIAQUERY, SIAREPLY)
*Jan 21 16:38:51.745: EIGRP: received packet with MD5 authentication, key id = 1
*Jan 21 16:38:51.745: EIGRP: Received HELLO on Serial0/0/0 nbr 192.168.1.102
*Jan 21 16:38:51.745:   AS 100, Flags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0 peerQ un/rely
0/0
```

```
R2# debug eigrp packets
EIGRP Packets debugging is on
    (UPDATE, REQUEST, QUERY, REPLY, HELLO, IPXSAP, PROBE, ACK, STUB, SIAQUERY, SIAREPLY)
R2#
*Jan 21 16:38:38.321: EIGRP: received packet with MD5 authentication, key id = 2
*Jan 21 16:38:38.321: EIGRP: Received HELLO on Serial0/0/0 nbr 192.168.1.101
*Jan 21 16:38:38.321:   AS 100, Flags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0 peerQ un/rely
0/0
```

Configuring EIGRP MD5 Authentication




```
R1(config-if)# key chain R1chain
R1(config-keychain)# key 2
R1(config-keychain-key)# key-string wrongkey
R1(config-keychain-key)#
```

```
R2# debug eigrp packets
EIGRP Packets debugging is on
    (UPDATE, REQUEST, QUERY, REPLY, HELLO, IPXSAP, PROBE, ACK, STUB, SIAQUERY, SIAREPLY)
*Jan 21 16:50:18.749: EIGRP: pkt key id = 2, authentication mismatch
*Jan 21 16:50:18.749: EIGRP: Serial0/0/0: ignored packet from 192.168.1.101, opcode = 5 (invalid
authentication)
*Jan 21 16:50:18.749: EIGRP: Dropping peer, invalid authentication
*Jan 21 16:50:18.749: EIGRP: Sending HELLO on Serial0/0/0
*Jan 21 16:50:18.749:   AS 100, Flags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0
*Jan 21 16:50:18.753: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 100: Neighbor 192.168.1.101
 (Serial0/0/0) is down: Auth failure
R2#
R2# show ip eigrp neighbors
IP-EIGRP neighbors for process 100
R2#
```

Optimizing EIGRP Implementations

Factors That Influence EIGRP Scalability

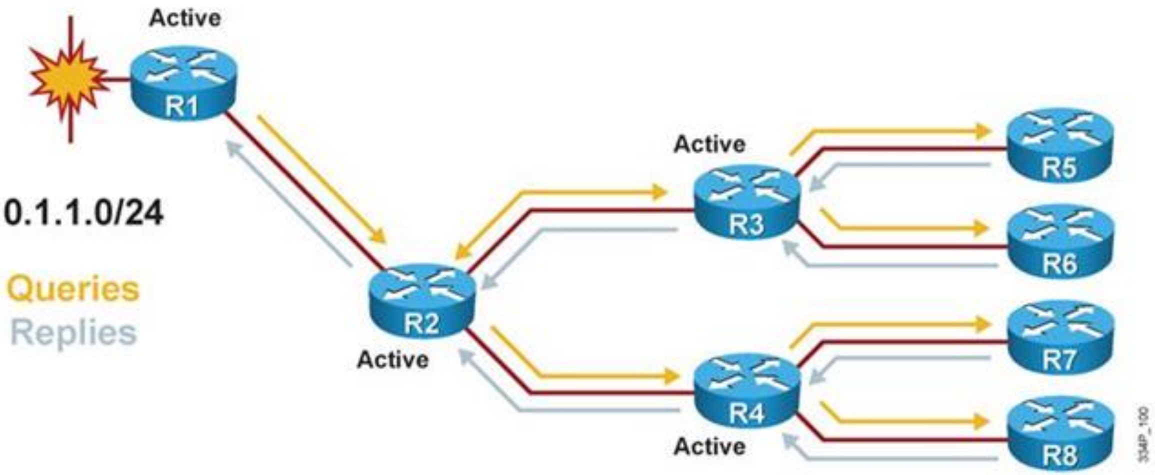
- Quantity of routing information exchanged between peers: without proper route summarization, this can be excessive.
- Number of routers that must be involved when a topology change occurs.
- Depth of topology: the number of hops that information must travel to reach all routers.
- Number of alternate paths through the network.

EIGRP Query Process

- Queries are sent when a route is lost and no feasible successor is available.
- The lost route is now in "active" state.
- Queries are sent to all neighboring routers on all interfaces except the interface to the successor.
- If the neighbors do not have their lost-route information, queries are sent to their neighbors.
- If a router has an alternate route, it answers the query; this stops the query from spreading in that branch of the network.

Overwhelming EIGRP Query Process

- In a large internetwork EIGRP queries can generate many resources.



- Several solutions exist to optimize the query propagation process and to limit the amount of unnecessary EIGRP load on the links, including:
  - Summarization
  - Redistribution
  - EIGRP stub routing feature.

Stuck-in-Active

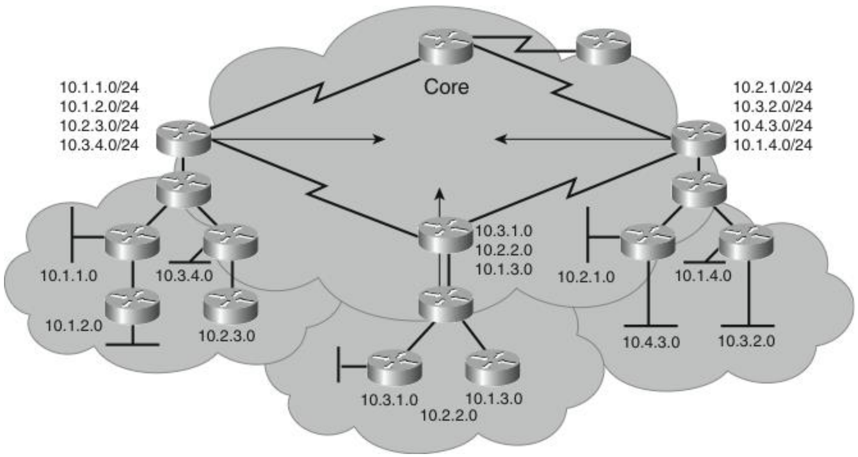
- **<Behavior in pre-12.1>** If a router does not receive a reply to all the outstanding queries within default 3 minutes (180 seconds), the route goes into Stuck-in-Active (SIA) state.

- **<Behavior in 12.1 and Later>** If a router does not receive a reply to all the outstanding queries within 90 seconds, the router sends Stuck-in-Active-Query (SIA-Query).

- Common SIA reasons:
  - A router is too busy to answer the query.
  - A router cannot allocate the memory to process the query.
  - The circuit between the two routers is not reliable.
  - The router has unidirectional links.

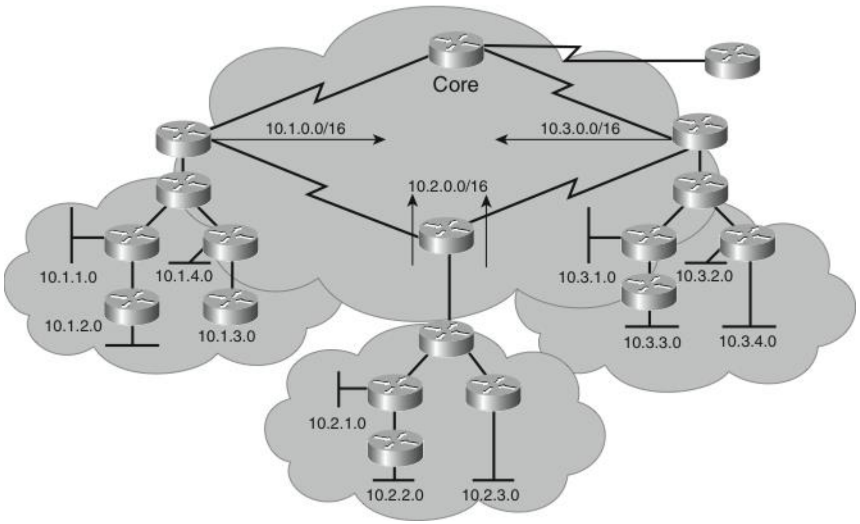
- SIA solutions:
  - Redesign the network to limit the query range by route summarization and the **`ip summary-address eigrp`** command.
  - Configure the remote routers as **stub** EIGRP routers.

SIA Solution: Summarization



- Poorly designed networks can make summarization difficult.
- Manually summarize the routes whenever possible to support a hierarchical network design.
- The more networks EIGRP summarizes, the lower the number of queries being sent out.
  - Ultimately reduces the occurrence of SIA errors.

SIA Solution: Summarization



- This network design is better because subnet addresses from individual major networks are localized within each cloud, allowing summary routes configured using the **`ip summary-address eigrp`** command to be injected into the core.
- As an added benefit, the summary routes act as a boundary for the queries generated by a topology change.

SIA Solution: Stub Networks

- The EIGRP Stub Routing feature:
  - Improves network stability
  - Reduces resource utilization and
  - Simplifies remote router (spoke) configuration

EIGRP Stub Routing

- Stub routing is commonly used in hub-and-spoke topology.
- Stub router sends a special peer information packet to all neighboring routers to report its status as a stub router.
  - Any neighbor that receives a packet informing it of the stub status does not query the stub router for any routes.
  - Stub routers are not queried and instead, hub routers connected to the stub router answer the query on behalf of the stub router.
- Only the remote routers are configured as stubs.

EIGRP Stub

- Configure a router as a stub router.

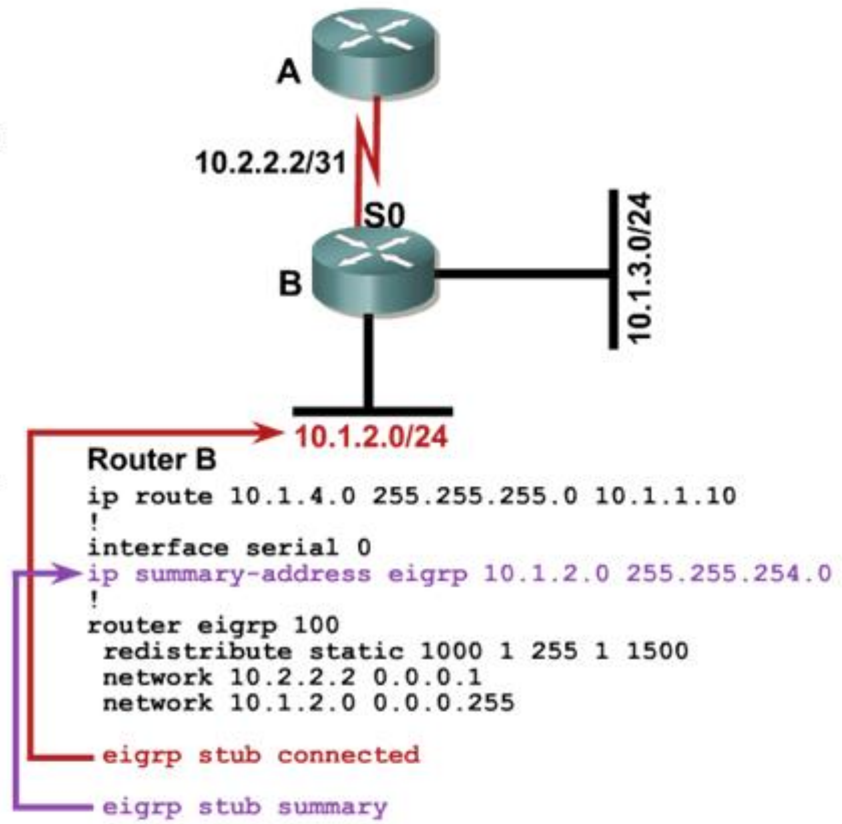
```
Router(config-router)#
```

```
eigrp stub [receive-only | connected | static | summary | redistributed]
```

Parameter	Description
receive-only	Restricts the router from sharing any of its routes with any other router within an EIGRP AS. Keyword cannot be combined with any other keyword.
connected	Permits the EIGRP stub routing feature to send connected routes. This option is enabled by default and is the most widely practical stub option.
static	Permits the EIGRP stub routing feature to send static routes. Redistributing static routes with the redistribute static command is still necessary.
summary	Permits the EIGRP stub routing feature to send automatically summarized and / or manually summarized routes. This option is enabled by default.
redistributed	Permits the EIGRP stub routing feature to send redistributed routes. Redistributing routes with the redistribute command is still necessary.

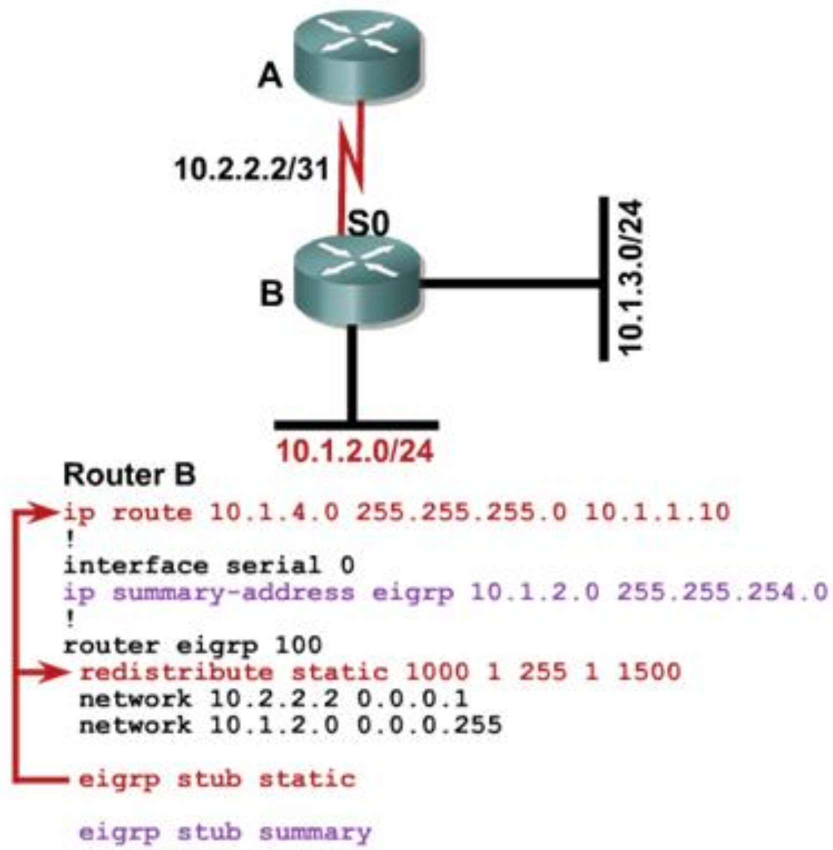
Example: EIGRP Stub Parameters

- If **`stub connected`** is configured:
  - B will advertise 10.1.2.0/24 to A.
  - B will not advertise 10.1.2.0/23, 10.1.3.0/23, or 10.1.4.0/24.
- If **`stub summary`** is configured:
  - B will advertise 10.1.2.0/23 to A.
  - B will not advertise 10.1.2.0/24, 10.1.3.0/24, or 10.1.4.0/24.

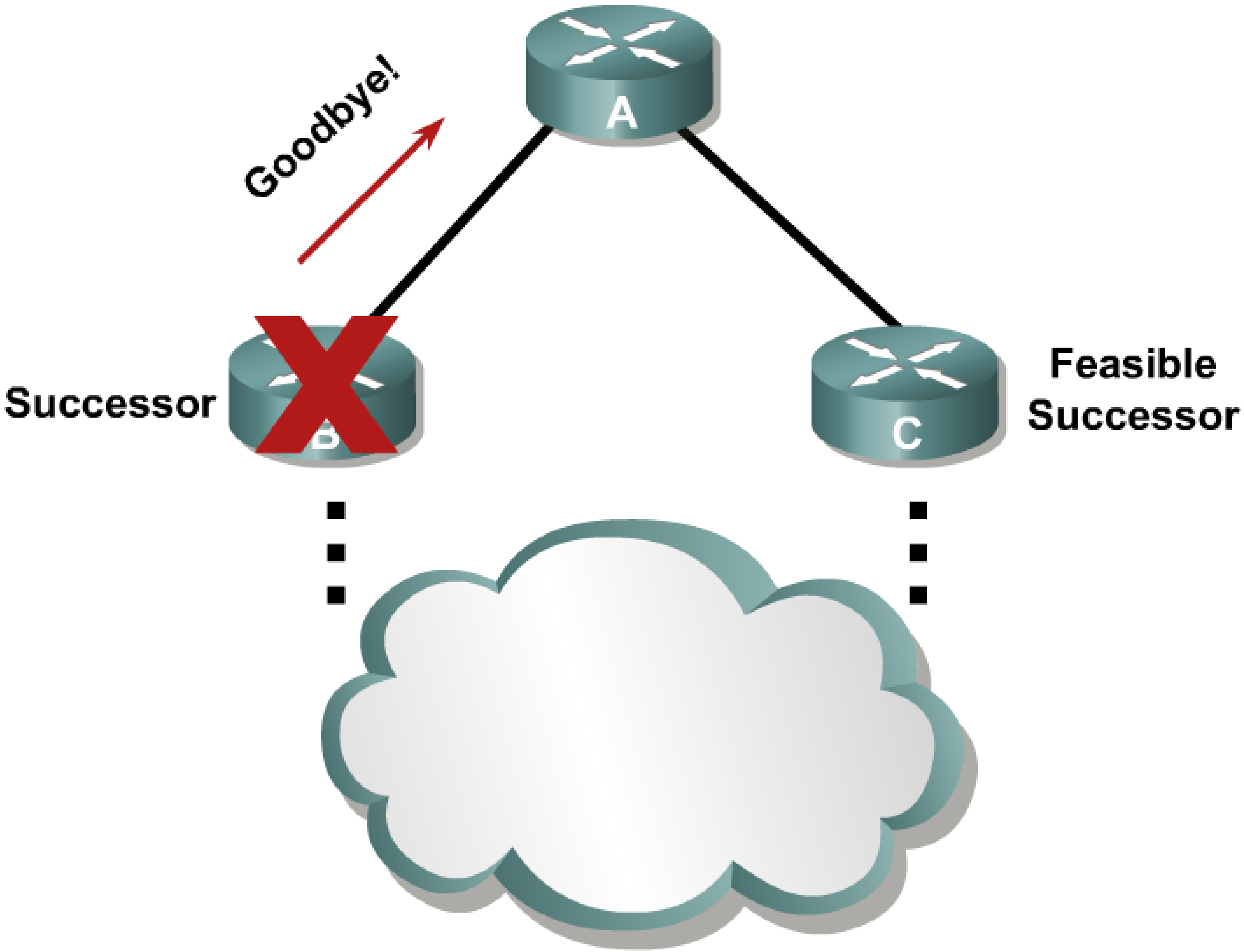


Example: EIGRP Stub Parameters (Cont.)

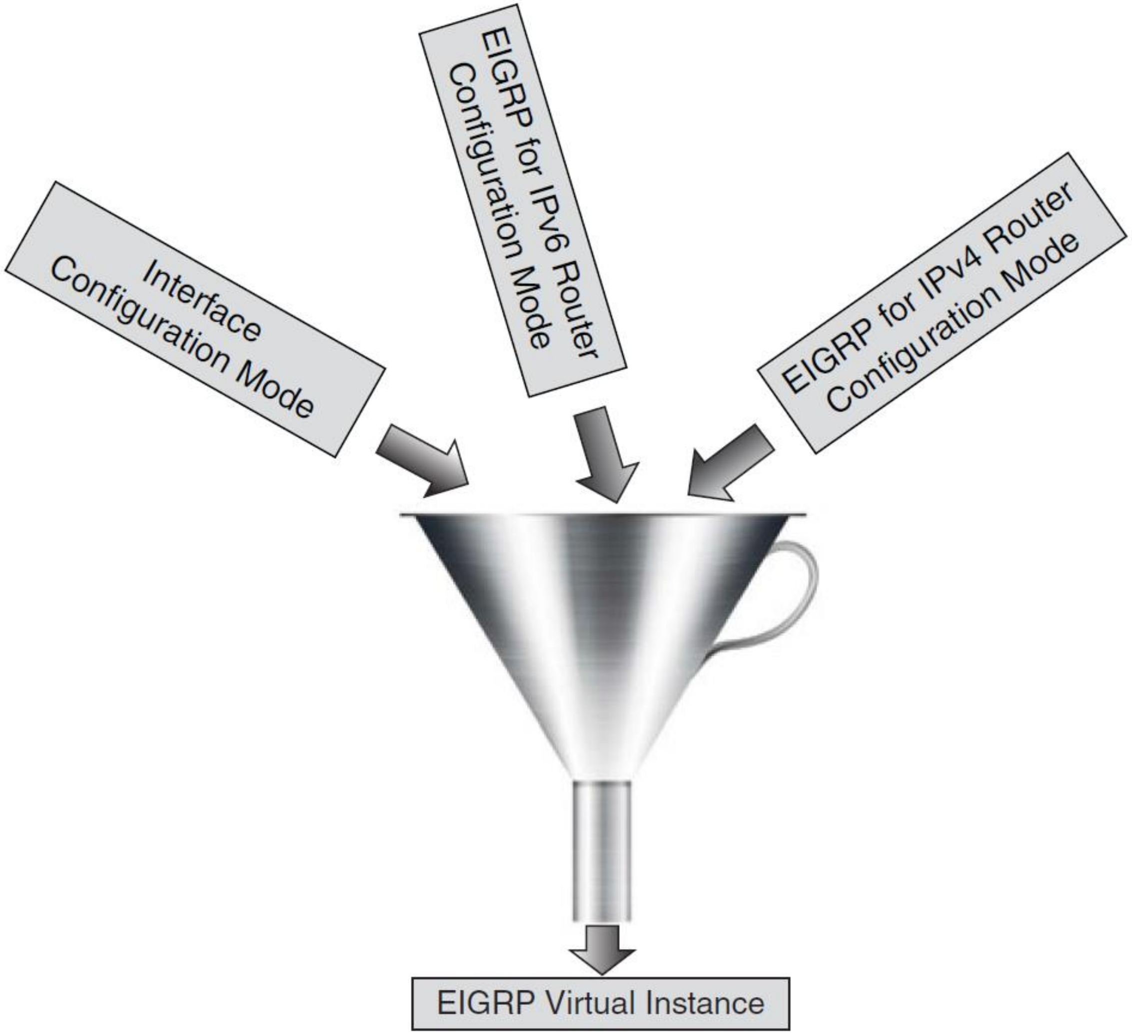
- If **`stub static`** is configured:
  - B will advertise 10.1.4.0/24 to A.
  - B will not advertise 10.1.2.0/24, 10.1.2.0/23, or 10.1.3.0/24.
- If **`stub receive-only`** is configured:
  - B won't advertise anything to A, so A needs to have a static route to the networks behind B to reach them.



Graceful Shutdown



EIGRP Named Mode



EIGRP Named Mode

Configuration Mode	Description
Address-Family	General EIGRP configuration commands are issued under this configuration mode. For example, router ID, network, and EIGRP stub router configurations are performed here. Multiple address families (for example, IPv4 and IPv6) can be configured under the same EIGRP virtual instance.
Address-Family-Interface	Commands entered under interface configuration mode with a traditional EIGRP configuration are entered here for Named EIGRP configuration. For example, timer and passive interface configurations are performed here.
Address-Family-Topology	Commands that have a direct impact on a router's EIGRP topology table are given in this configuration mode. For example, variance and redistribution are configured in this mode.

Traditional EIGRP vs Named EIGRP

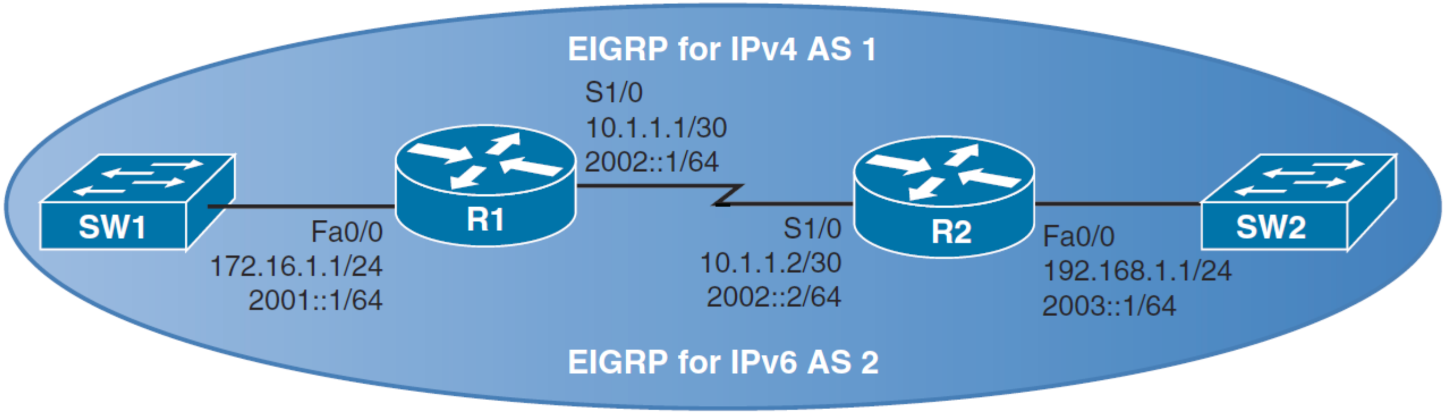
```
!Router R1 Configuration
R1# conf term
R1(config)# router eigrp 1
R1(config-router)# network 0.0.0.0
```

```
!Router R2 Configuration
R2# conf term
R2(config)# router eigrp 1
R2(config-router)# network 0.0.0.0
```

```
!R1 Router Configuration
R1# conf term
R1(config)# router eigrp R1DEMO
R1(config-router)# address-family ipv4 autonomous-system 1
R1(config-router-af)# network 0.0.0.0
```

```
!R2 Router Configuration
R2# conf term
R2(config)# router eigrp R2DEMO
R2(config-router)# address-family ipv4 autonomous-system 1
R2(config-router-af)# network 0.0.0.0
```

Traditional EIGRP vs Named EIGRP




```
!Router R1 Configuration

interface FastEthernet0/0
ip address 172.16.1.1 255.255.255.0
ipv6 address 2001::1/64
ipv6 eigrp 2
!
interface Serial1/0
ip address 10.1.1.1 255.255.255.252
ip hello-interval eigrp 1 2
ip hold-time eigrp 1 10
ipv6 address 2002::1/64
ipv6 eigrp 2
!
router eigrp 1
variance 2
network 0.0.0.0
passive-interface default
no passive-interface Serial1/0
!
ipv6 router eigrp 2
variance 2
```

```
!Router R2 Configuration

interface FastEthernet0/0
ip address 192.168.1.1 255.255.255.0
ipv6 address 2003::1/64
ipv6 eigrp 2
!
interface Serial1/0
ip address 10.1.1.2 255.255.255.252
ip hello-interval eigrp 1 2
ip hold-time eigrp 1 10
ipv6 address 2002::2/64
ipv6 eigrp 2
!
router eigrp 1
variance 2
network 0.0.0.0
passive-interface default
no passive-interface Serial1/0
!
ipv6 router eigrp 2
variance 2
```

Traditional EIGRP vs Named EIGRP

```
!Router R1 Configuration
router eigrp R1DEMO
!
address-family ipv4 unicast autonomous-system 1
!
af-interface default
hello-interval 2
hold-time 10
passive-interface
exit-af-interface
!
af-interface Serial1/0
no passive-interface
exit-af-interface
!
topology base
variance 2
exit-af-topology
network 0.0.0.0
exit-address-family
!
address-family ipv6 unicast autonomous-system 2
!
topology base
variance 2
exit-af-topology
exit-address-family
```

```
!Router R2 Configuration
router eigrp R2DEMO
!
address-family ipv4 unicast autonomous-system 1
!
af-interface default
hello-interval 2
hold-time 10
passive-interface
exit-af-interface
!
af-interface Serial1/0
no passive-interface
exit-af-interface
!
topology base
variance 2
exit-af-topology
network 0.0.0.0
exit-address-family
!
address-family ipv6 unicast autonomous-system 2
!
topology base
variance 2
exit-af-topology
exit-address-family
```

Chapter 2 Summary

- EIGRP initial route discovery process, started by a router sending hello packets. Neighboring routers reply with update packets, which populate the router's topology table. The router chooses the successor routes and offers them to the routing table.
- The DUAL process including selecting FSs. To qualify as an FS, a next-hop router must have an AD less than the FD of the current successor route for the particular network, to ensure a loop-free network.
- The EIGRP metric calculation, which defaults to bandwidth (the slowest bandwidth between the source and destination) + delay (the cumulative interface delay along the path).
- Planning EIGRP implementations, including:
  - IP addressing
  - Network topology
  - EIGRP traffic engineering.
- The list of tasks for each router in the network include:
  - Enabling the EIGRP routing protocol (with the correct AS number)
  - Configuring the proper network statements
  - Optionally configuring the metric to appropriate interfaces.

Chapter 2 Summary (continued)

- Basic EIGRP configuration commands.
- Commands for verifying EIGRP operation.
- Configuring a **passive-interface**.
- Propagating a default route.
- EIGRP summarization.
- EIGRP over Frame Relay.
- EIGRP over MPLS.
- EIGRP load-balancing
- EIGRP operation in WAN environments:
- Configuring, verifying, and troubleshooting EIGRP MD5 authentication.
- EIGRP scalability factors, including the amount of information exchanged, the number of routers, the depth of the topology, and the number of alternative paths through the network.
- The SIA state and how to limit the query range to help reduce SIAs.
- Configuring the remote routers as stub EIGRP routers.
- Graceful shutdown, which broadcasts a goodbye message (in a hello packet, with all K values set to 255) when an EIGRP routing process is shut down, to inform neighbors

Chapter 2 Labs

- **IGP-LAB-2.1 EIGRP**
- **IGP-LAB-2.2 EIGRP Extra**

Q&A