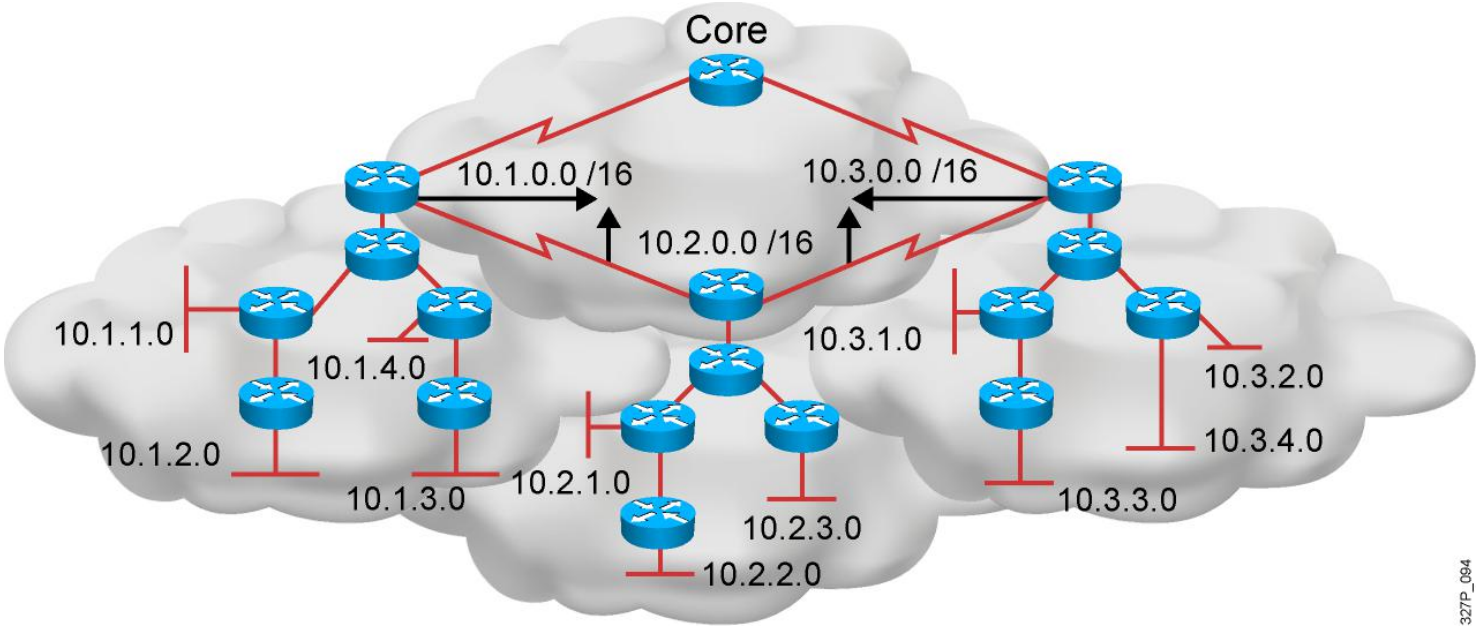
Implementing EIGRP

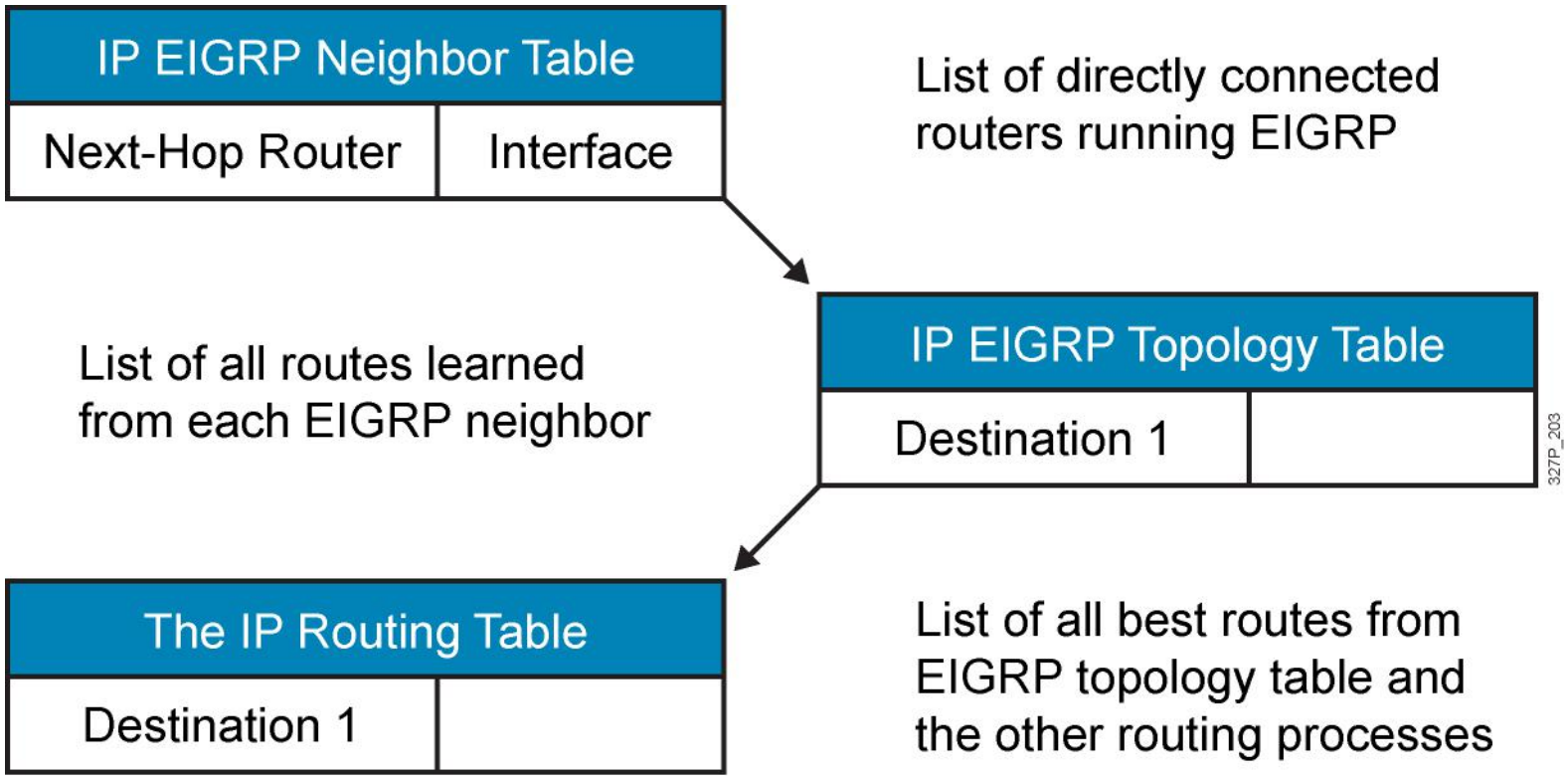
EIGRP Implementation

EIGRP Features

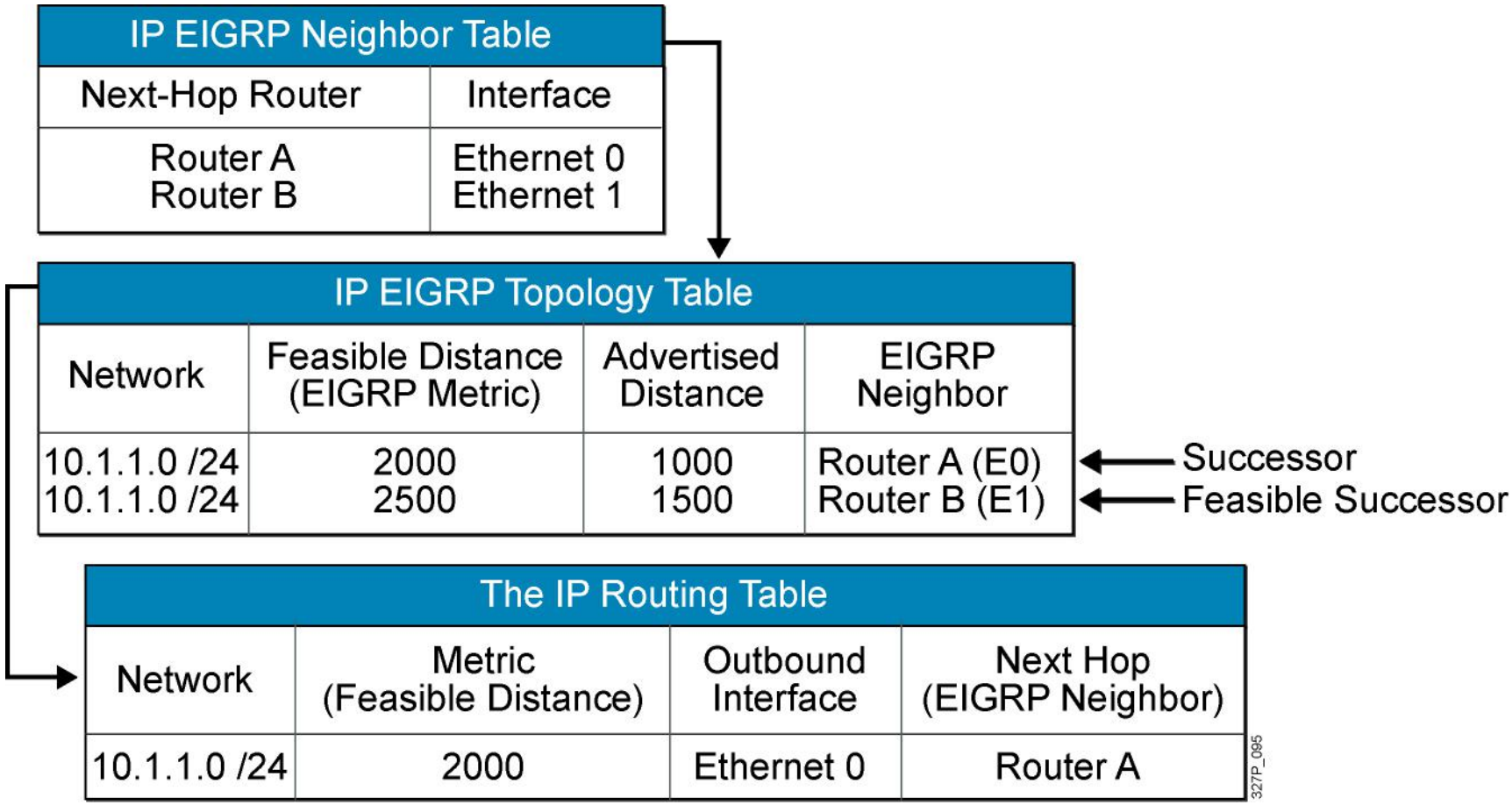


- Advanced distance vector
- Rapid convergence
- 100% loop-free classless routing
- Easy configuration
- Incremental updates
- Load balancing across equal- and unequal-cost pathways
- Flexible network design
- Multicast and unicast instead of broadcast address
- Support for VLSM and discontiguous subnets
- Manual summarization at any point in the internetwork
- Support for multiple network layer protocols

EIGRP Tables



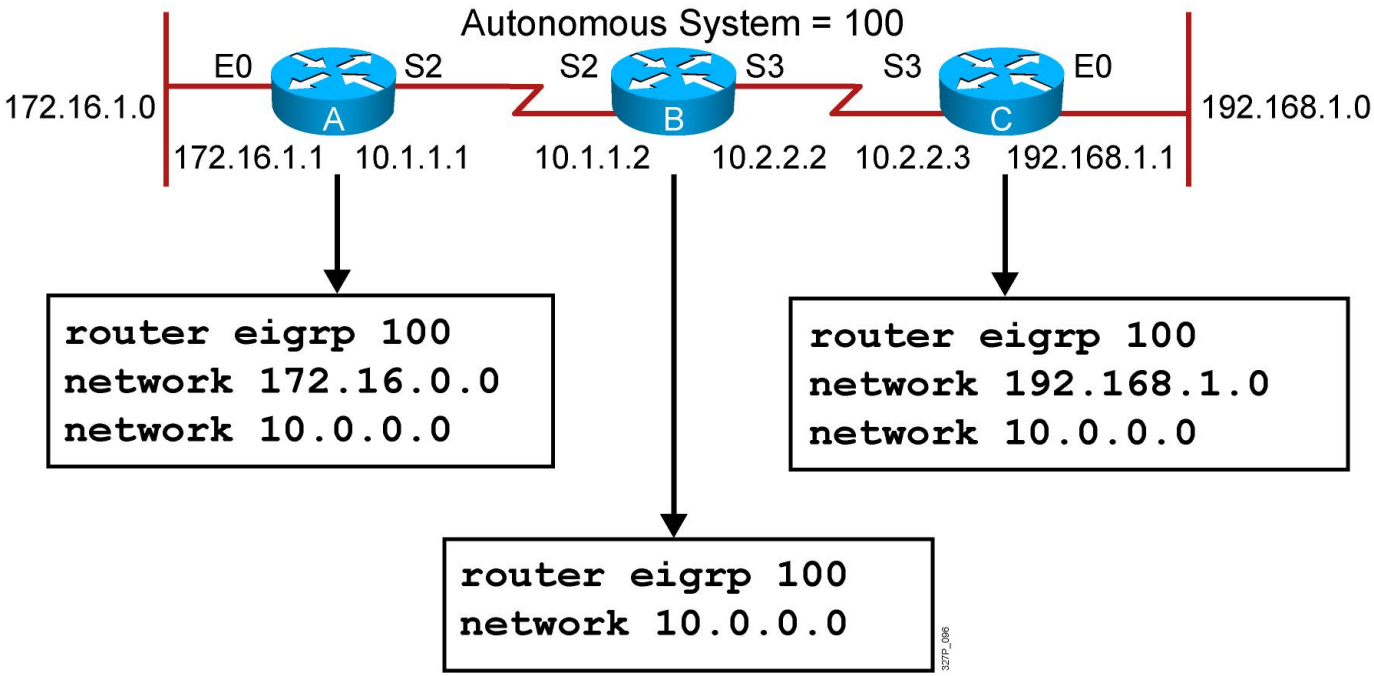
EIGRP Path Calculation (Router C)



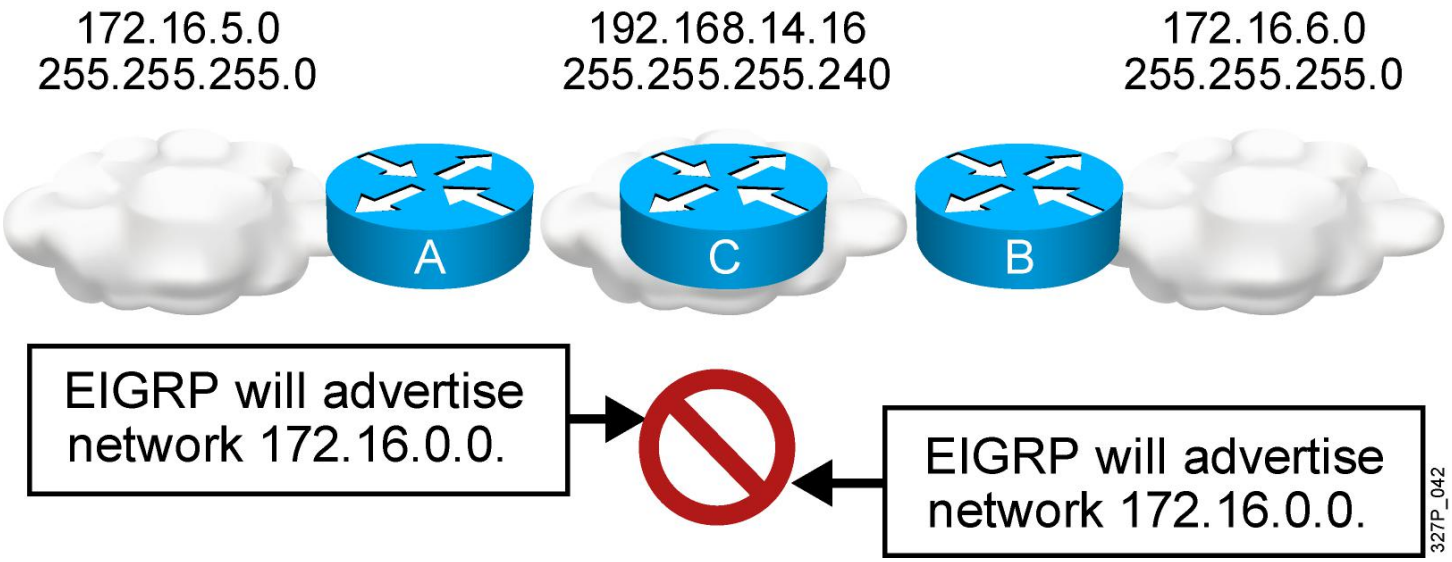
EIGRP Configuration

```
RouterX(config)# router eigrp autonomous-system
```

```
RouterX(config-router)# network network-number
```

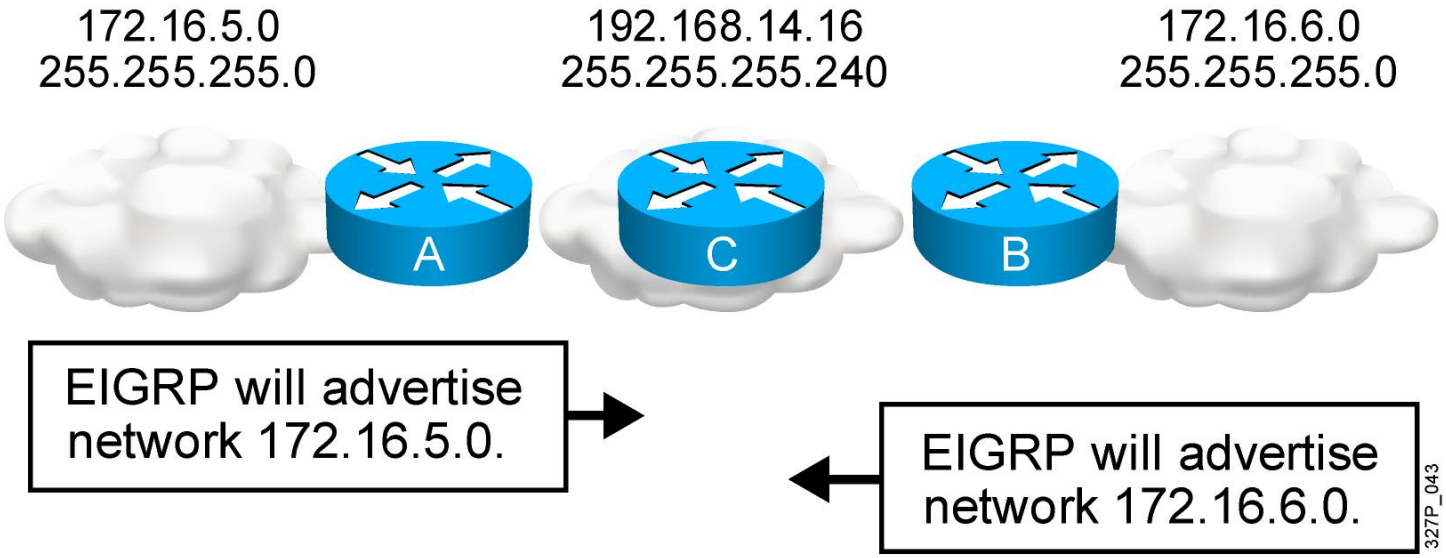


EIGRP and Discontiguous Networks Default Scenario Configuration



- EIGRP, by default, does not advertise subnets and, therefore, cannot support discontiguous subnets.

EIGRP and Discontiguous Networks with no auto-summary



- EIGRP with the **no auto-summary** parameter can advertise subnets and, therefore, can support discontiguous subnets.

Verifying the EIGRP Configuration

```
RouterX# show ip route eigrp
```

- Displays the current EIGRP entries in the routing table

```
RouterX# show ip protocols
```

- Displays the parameters and current state of the active process

```
RouterX# show ip eigrp interfaces
```

- Displays information about interfaces configured for EIGRP

```
RouterX# show ip eigrp interfaces
IP EIGRP interfaces for process 109

                      Xmit Queue     Mean   Pacing Time   Multicast    Pending
Interface     Peers   Un/Reliable    SRTT   Un/Reliable   Flow Timer   Routes
Di0             0       0/0             0      11/434          0          0
Et0             1       0/0           337      0/10            0          0
SE0:1.16        1       0/0            10      1/63          103          0
Tu0             1       0/0           330      0/16            0          0
```

Verifying the EIGRP Configuration (Cont.)

```
RouterX# show ip eigrp neighbors [detail]
```

- Displays the neighbors discovered by IP EIGRP

```
RouterX# show ip eigrp neighbors
IP-EIGRP Neighbors for process 77
```

Address	Interface	Holdtime (secs)	Uptime (h:m:s)	Q Count	Seq Num	SRTT (ms)	RTO (ms)
172.16.81.28	Ethernet1	13	0:00:41	0	11	4	20
172.16.80.28	Ethernet0	14	0:02:01	0	10	12	24
172.16.80.31	Ethernet0	12	0:02:02	0	4	5	20

Verifying the EIGRP Configuration (Cont.)

```
RouterX# show ip eigrp topology [all]
```

- Displays the IP EIGRP topology table
- Without the **[all]** parameter, shows successors and feasible successors

```
RouterX# show ip eigrp topology
IP-EIGRP Topology Table for process 77
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - Reply status
P 172.16.90.0 255.255.255.0, 2 successors, FD is 46251776
         via 172.16.80.28 (46251776/46226176), Ethernet0
         via 172.16.81.28 (46251776/46226176), Ethernet1
         via 172.16.80.31 (46277376/46251776), Serial0
P 172.16.81.0 255.255.255.0, 2 successors, FD is 307200
         via Connected, Ethernet1
         via 172.16.81.28 (307200/281600), Ethernet1
         via 172.16.80.28 (307200/281600), Ethernet0
         via 172.16.80.31 (332800/307200), Serial0
```

Verifying the EIGRP Configuration (Cont.)

```
RouterX# show ip eigrp traffic
```

- Displays the number of IP EIGRP packets sent and received

```
RouterX# show ip eigrp traffic
IP-EIGRP Traffic Statistics for process 77
  Hellos sent/received: 218/205
  Updates sent/received: 7/23
  Queries sent/received: 2/0
  Replies sent/received: 0/2
  Acks sent/received: 21/14
```

debug ip eigrp Command

```
RouterX# debug ip eigrp
IP-EIGRP: Processing incoming UPDATE packet
IP-EIGRP: Ext 192.168.3.0 255.255.255.0 M 386560 - 256000 130560 SM 360960 -
256000 104960
IP-EIGRP: Ext 192.168.0.0 255.255.255.0 M 386560 - 256000 130560 SM 360960 -
256000 104960
IP-EIGRP: Ext 192.168.3.0 255.255.255.0 M 386560 - 256000 130560 SM 360960 -
256000 104960
IP-EIGRP: 172.69.43.0 255.255.255.0, - do advertise out Ethernet0/1
IP-EIGRP: Ext 172.69.43.0 255.255.255.0 metric 371200 - 256000 115200
IP-EIGRP: 192.135.246.0 255.255.255.0, - do advertise out Ethernet0/1
IP-EIGRP: Ext 192.135.246.0 255.255.255.0 metric 46310656 - 45714176 596480
IP-EIGRP: 172.69.40.0 255.255.255.0, - do advertise out Ethernet0/1
IP-EIGRP: Ext 172.69.40.0 255.255.255.0 metric 2272256 - 1657856 614400
IP-EIGRP: 192.135.245.0 255.255.255.0, - do advertise out Ethernet0/1
IP-EIGRP: Ext 192.135.245.0 255.255.255.0 metric 40622080 - 40000000 622080
IP-EIGRP: 192.135.244.0 255.255.255.0, - do advertise out Ethernet0/1
```

Note: EIGRP routes are exchanged only when a change in topology occurs.

EIGRP Metric

The criteria that EIGRP uses by default to calculate its metric:

- Bandwidth
- Delay

The optional criteria that EIGRP can be configured to use when calculating its metric:

- Reliability
- Load

Note: Although MTU is exchanged in EIGRP packets between neighbor routers, MTU is not factored into the EIGRP metric calculation.

EIGRP Load Balancing

- By default, EIGRP does equal-metric load balancing:
  - By default, up to four routes with a metric equal to the minimum metric are installed in the routing table.
- There can be up to 16 entries in the routing table for the same destination:
  - The number of entries is configurable with the **maximum-paths** command.

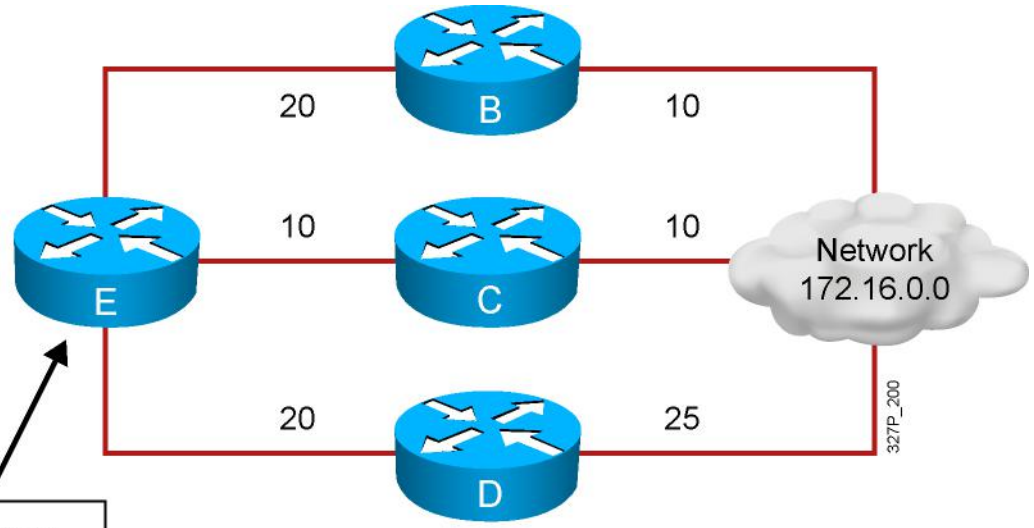
EIGRP Unequal-Cost Load Balancing

```
RouterX(config-router)#
variance multiplier
```

- Allows the router to load-balance across routes with a metric smaller than the *multiplier* value times the minimum metric route to that destination.
- The default variance is 1, which means equal-cost load balancing.

Variance Example

Network	Neighbor	FD	AD
172.16.0.0	B	30	10
	C	20	10
	D	45	25



- Router E chooses router C to route to network 172.16.0.0 because it has the lowest feasible distance of 20.
- With a variance of 2, router E also chooses router B to route to network 172.16.0.0 (20 + 10 = 30) < [2 * (FD) = 40].
- Router D is not considered to route to network 172.16.0.0 (because 25 > 20).

EIGRP MD5 Authentication

- EIGRP supports MD5 authentication.
- The router identifies itself for every EIGRP packet it sends.
- The router authenticates the source of each routing update packet that it receives.
- Each participating neighbor must have the same key configured.

EIGRP MD5 Authentication Configuration Steps

1. Create the keychain, a group of possible keys (passwords).
2. Assign a key ID to each key.
3. Identify the keys.
4. (Optional) Specify the duration a key will be valid.
5. Enable MD5 authentication on the interface.
6. Specify which keychain the interface will use.

Configuring EIGRP MD5 Authentication

```
RouterX(config)#
```

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

- Enters the configuration mode for the keychain

```
RouterX(config-keychain)#
```

```
key key-id
```

- Identifies the key and enters the configuration mode for the key ID

Configuring EIGRP MD5 Authentication (Cont.)

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

```
key-string text
```

- Identifies the key string (password)

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

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

- (Optional) Specifies when the key is accepted for received packets

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

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

- (Optional) Specifies when the key can be used for sending packets

Configuring EIGRP MD5 Authentication (Cont.)

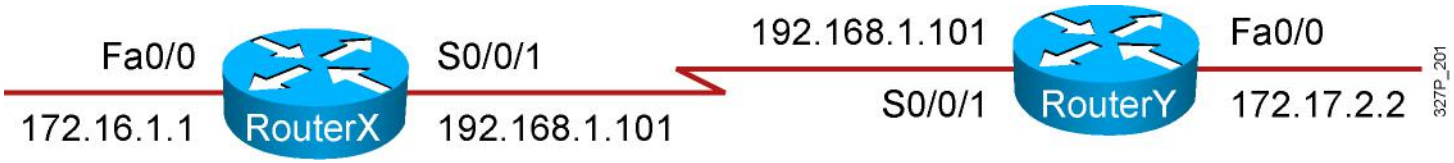
```
RouterX(config-if)#
ip authentication mode eigrp autonomous-system md5
```

- Specifies MD5 authentication for EIGRP packets

```
RouterX(config-if)#
ip authentication key-chain eigrp autonomous-system
name-of-chain
```

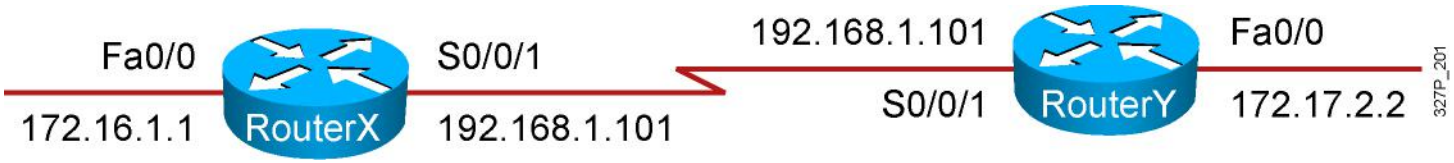
- Enables authentication of EIGRP packets using the key in the keychain

Example EIGRP MD5 Authentication Configuration




```
RouterX
<output omitted>
key chain RouterXchain
 key 1
  key-string firstkey
  accept-lifetime 04:00:00 Jan 1 2006 infinite
  send-lifetime 04:00:00 Jan 1 2006 04:01:00 Jan 1 2006
 key 2
  key-string secondkey
  accept-lifetime 04:00:00 Jan 1 2006 infinite
  send-lifetime 04:00:00 Jan 1 2006 infinite
<output omitted>
!
interface Serial0/0/1
 bandwidth 64
 ip address 192.168.1.101 255.255.255.224
 ip authentication mode eigrp 100 md5
 ip authentication key-chain eigrp 100 RouterXchain
```

Example EIGRP MD5 Authentication Configuration (Cont.)




```
RouterY
<output omitted>
key chain RouterYchain
 key 1
  key-string firstkey
  accept-lifetime 04:00:00 Jan 1 2006 infinite
  send-lifetime 04:00:00 Jan 1 2006 infinite
 key 2
  key-string secondkey
  accept-lifetime 04:00:00 Jan 1 2006 infinite
  send-lifetime 04:00:00 Jan 1 2006 infinite
<output omitted>
!
interface Serial0/0/1
 bandwidth 64
 ip address 192.168.1.102 255.255.255.224
 ip authentication mode eigrp 100 md5
 ip authentication key-chain eigrp 100 RouterYchain
```

Verifying MD5 Authentication

```
RouterX#
*Jan 21 16:23:30.517: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 100: Neighbor 192.168.1.102
(Serial0/0/1) is up: new adjacency

RouterX#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           12 00:03:10   17   2280  0  14

RouterX#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, Serial0/0/1
     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, Serial0/0/1
D       192.168.1.0/24 is a summary, 00:31:31, Null0

RouterX#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
```