CISCO Live!

#CiscoLive

cisco The bridge to possible

EIGRP - The Usual Suspects

The Who, The What, The Where

Steven Moore Leader, Solution Architecture
@smoore_bits
BRKENT-2050

Cisco Webex App

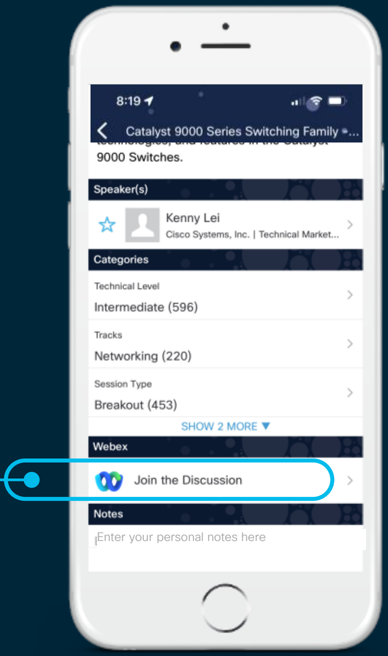
Questions?

Use Cisco Webex App to chat
with the speaker after the session

How

1. Find this session in the Cisco Live Mobile App
2. Click “Join the Discussion”
3. Install the Webex App or go directly to the Webex space
4. Enter messages/questions in the Webex space

Webex spaces will be moderated
by the speaker until June 17, 2022.



8:19

Catalyst 9000 Series Switching Family ...

9000 Switches.

Speaker(s)

Kenny Lei
Cisco Systems, Inc. | Technical Market...

Categories

Technical Level
Intermediate (596)

Tracks
Networking (220)

Session Type
Breakout (453)

SHOW 2 MORE ▼

Webex

Join the Discussion

Notes

Enter your personal notes here

https://ciscolive.ciscoevents.com/ciscolivebot/#BRKENT-2050

Agenda

- How this Session Will Help You
- Introducing EIGRP – The Usual Suspects
  - Who are they?
  - What are their motives?
  - What is your strategy?
- Conclusion & Next Steps

> *"Nothing matters but the facts. Without them, the science of criminal investigation is nothing more than a guessing game."*

Blake Edwards

Director, The Pink Panther

How This Session Will Help You

- Develop A Tactical Approach to Interrogating Your Network
- Minimize the Guesswork, Offer Practical Takeaways
- Increase Your Speed in Identifying Issues
- *This Will Not:*
  - Teach you the fundamentals of EIGRP (DGTL-BRKENT-1102)
  - Provide deep, specific troubleshooting into EIGRP (BRKRST-2331)

The Usual Suspects

Who Are They?

- Neighbors
- Packets
- Computations

What Are Their Motives?

- Establish Communication
- Exchange Information
- Make Decisions

What Is Your Strategy?

- Neighbor Table, Logs
- Topology, Event Log
- DUAL

How Things Should Work

Know What is Right, So You Can Spot What is Wrong

- Routing protocols share the same fundamental, essential components.
  - Establish Communication

    Who are they exchanging information with, and how?
  - Exchange Routes

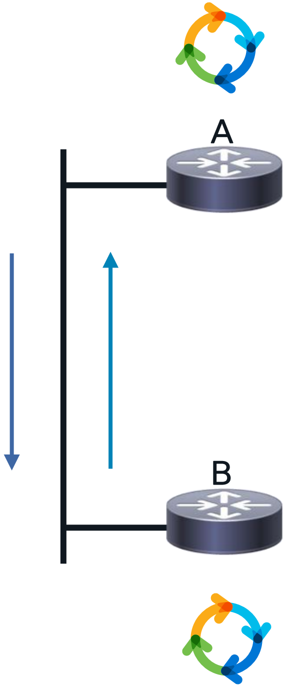
    What information is sent, and how?
  - Perform Computation

    What algorithm is used to compute loop free paths?
  - Route Installation

    What routes are the best? Can we install them?
- EIGRP is no exception!
- Understanding how EIGRP implements each of these will help us learn, use, and operate networks with EIGRP.

Routing Protocol Background

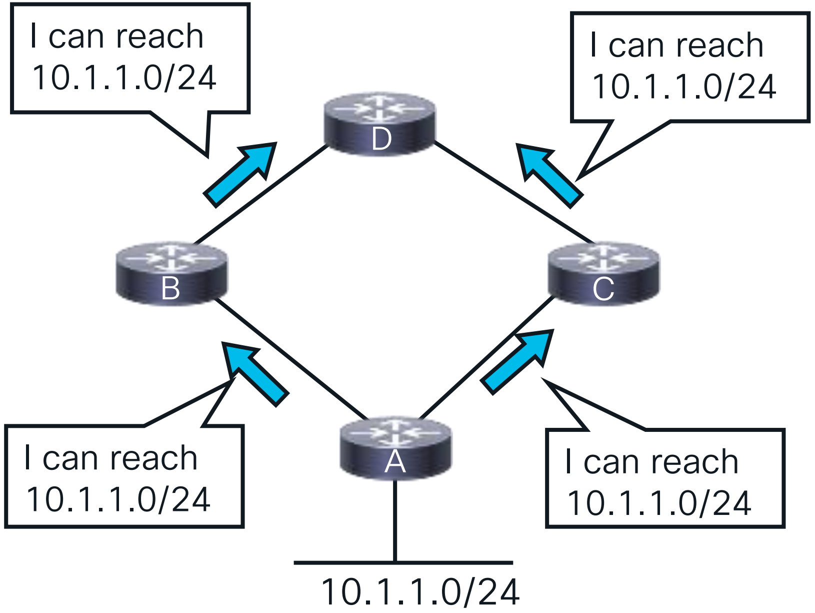
- Logical Sequence of Events
- EIGRP:
  - Peers Form – 3 way handshake
  - Routes Exchanged – Reliable Transport
  - Path Computation – Topology Table, DUAL
  - Routing Table Updated (if necessary)
  - Peers Updated (if necessary)



Distance Vector Routing

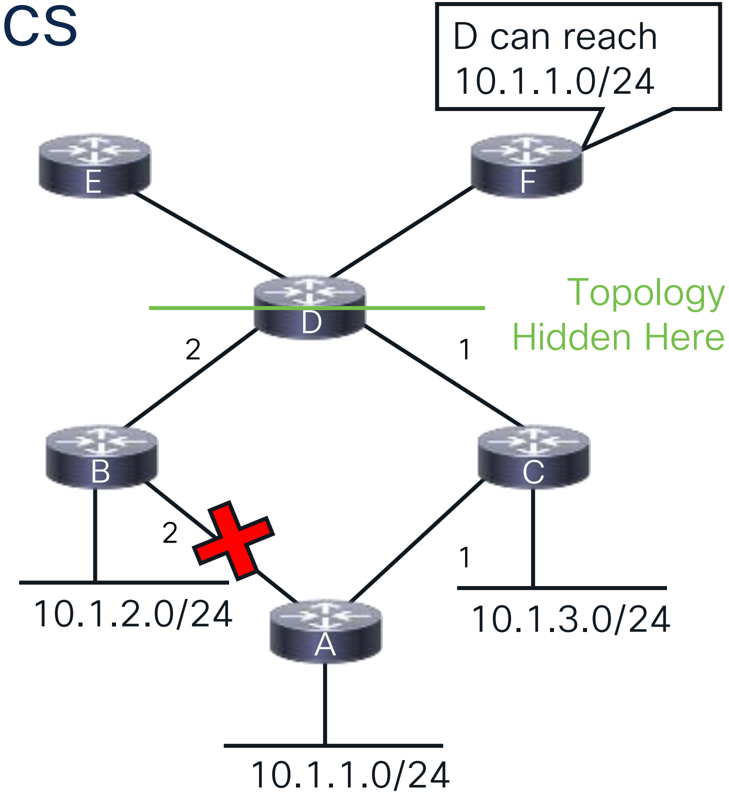
Distance Vector Routing Basics

- Topology information beyond the next hop is naturally hidden in distance vector protocols
- EIGRP only knows prefix and next-hop information
- A advertises that it can reach 10.1.1.0/24
- B and C only advertise to D that they can reach
  10.1.1.0/24, not that they are connected to A, which is then connected to 10.1.1.0/24
- D now knows to reach 10.1.1.0/24 it can use B or C, but D does not know what routers or connections exist beyond B and C



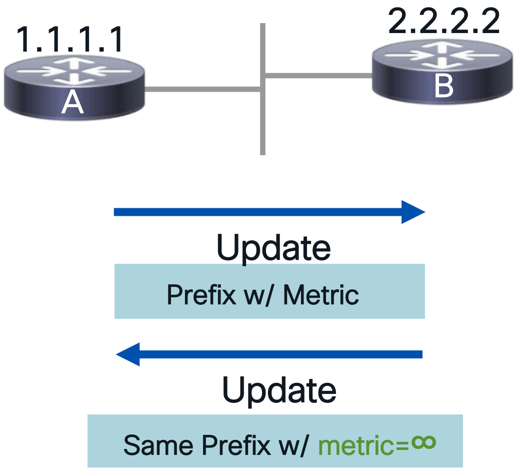
Distance Vector Routing Basics

- Hiding topology information hides information about changes in the topology
- D advertises reachability to 10.1.1.0/24 to E and F
  - If the A to B link fails, D can still reach 10.1.1.0/24 (although the metric might change)
  - If F continues to use D to reach 10.1.1.0/24
  - Does F need to know about the A to B link failure?
  - No!
- What's the issue if D advertises reachability?
  - When the A to B link fails, D will send an update to F
  - F may then go active, and potentially send a Query to its peers
  - This results in increased CPU, memory, and convergence time for a path F can only reach though D



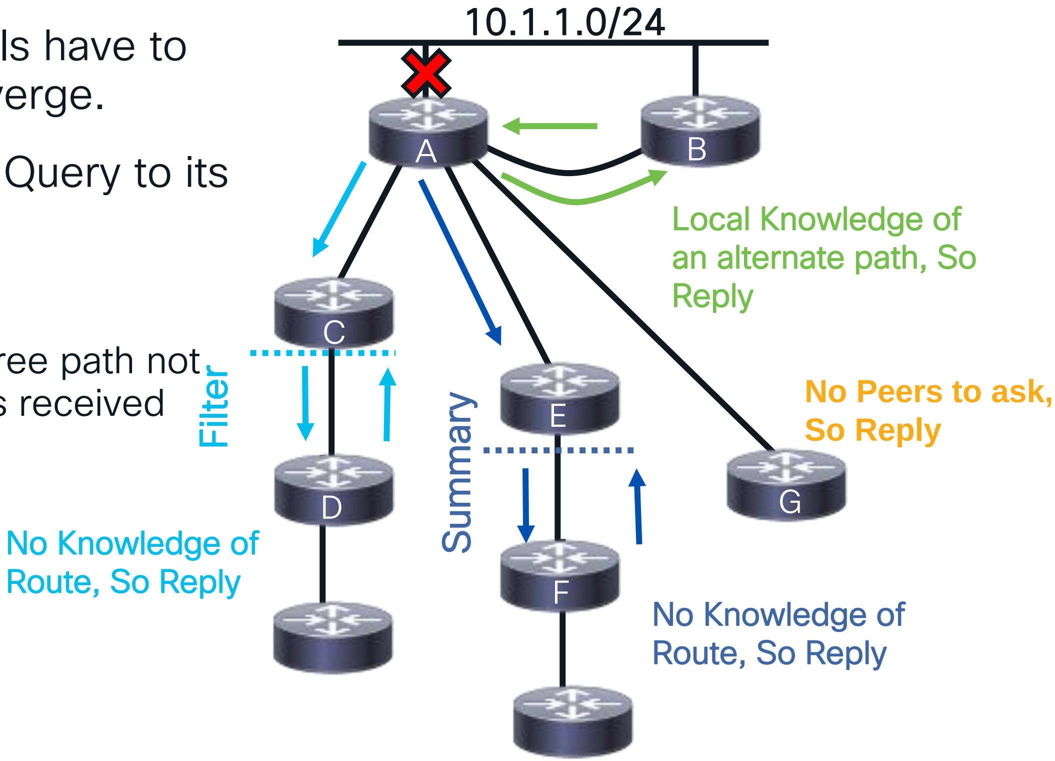
Distance Vector Routing Basics

- Common Concerns – But are they true?
  - Slower Convergence – Why?
  - Count to Infinity – What?
  - Topological “Blindness” - Tradeoffs
- Poison Reverse
- Split Horizon



Distance Vector Routing Basics - Resolution

- Without full topological info, protocols have to cooperatively resolve routes to converge.
- When EIGRP goes active, it sends a Query to its peers looking for the lost route.
- The Query is bounded by:
  - Local knowledge of an alternate loop-free path not learned through the peer the query was received from
  - No local knowledge of the route because of filtering
  - No local knowledge of the route because of summarization
  - No peers to query



Local Knowledge of an alternate path, So Reply

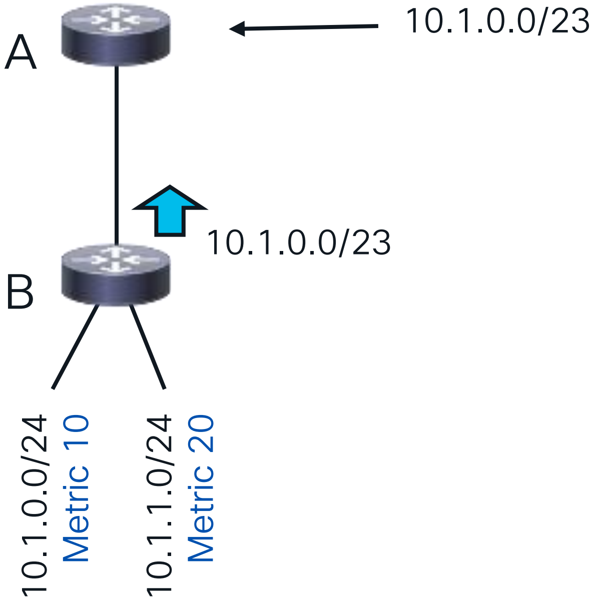
No Peers to ask, So Reply

No Knowledge of Route, So Reply

No Knowledge of Route, So Reply

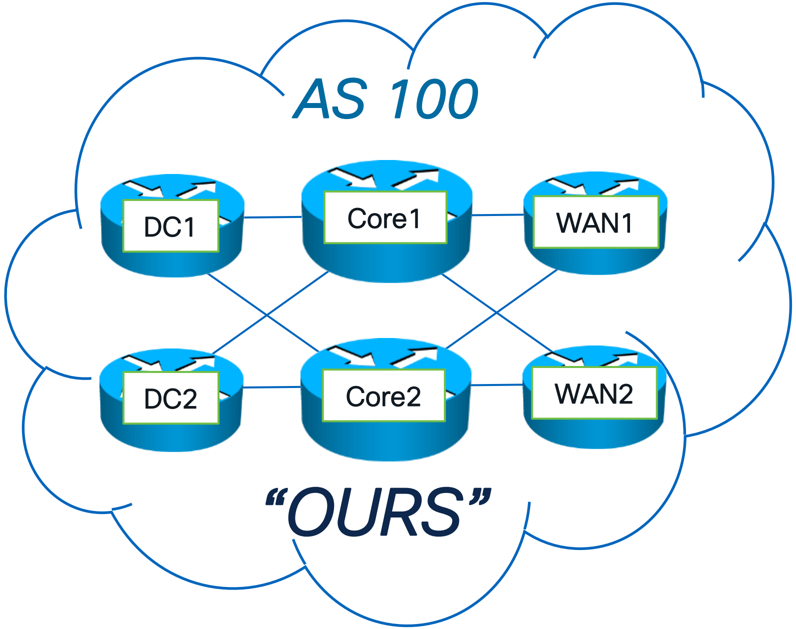
Route Summarization

- A summary:
  - Aggregates information – fewer, less specific routes downstream
  - Hides topology changes by filtering out components
- Only advertise the summary if a component route is present
  - Component is any route falling within the summary address range
  - Component Routes are automatically filtered and not sent downstream



EIGRP Autonomous System

- A collection of devices under the same administrative control
- Shares a consistent routing policy
- Creates the outermost edges of the network



Suspect # 1: Neighbors

Two Scenarios:
Neighbor Doesn’t Form
Neighbor Isn’t Stable

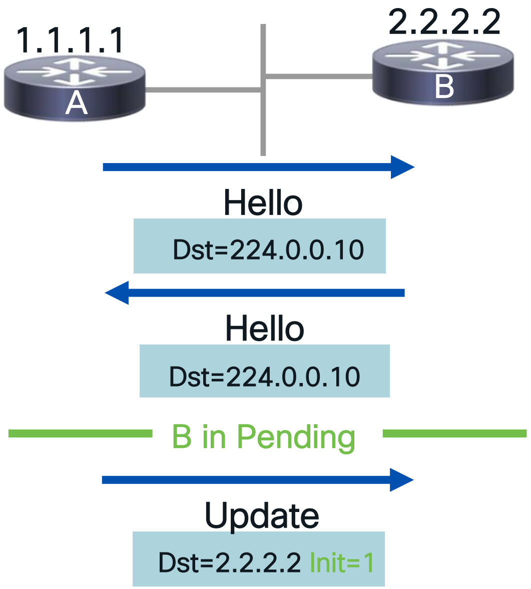
Neighbor Formation

- EIGRP, by default, uses link-local multicast for neighbor communication. (224.0.0.10)
- Neighbors will dynamically form within the same AS, as consistent with other IGP's
- Unicast neighbors are supported but generally not used outside of a few exceptions

How do neighbors get established?

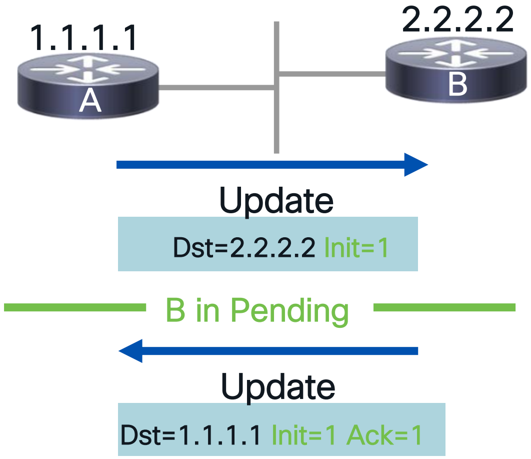
Neighbor Formation

- Router A must receive an initial Hello from B before it will accept reliable packets from it
- When A receives the first Hello from B, it places B in the *pending state*
- A transmits a unicast "NULL" Update which has the initialization (init) bit set, and no routing data
- While A has B is in this state, A will not send it any Queries or other Updates to B



Neighbor Formation

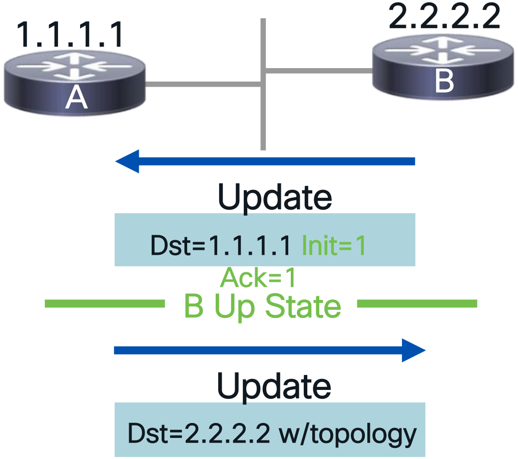
- When B receives this Update with the init bit set, it sends an Update with the init bit set as well
- The acknowledgement (Ack) for A’s initial Update is piggybacked onto this Update packet—it is never transmitted by itself
- Thus, there is no way for A to receive the Ack for its initial Update without also receiving B’s initial unicast Update



Neighbor Formation

- While waiting on the acknowledgement, Query and Update packets from B are ignored
- If the acknowledgement is never received, A will time B out, and the process will start over
- Once the Ack for its initial update is received, A moves B from *pending state* to *up state*
- A then begins sending it’s full topology information to B

✓ This validates bi-directional unicast and multicast communication, completing the 3-Way handshake

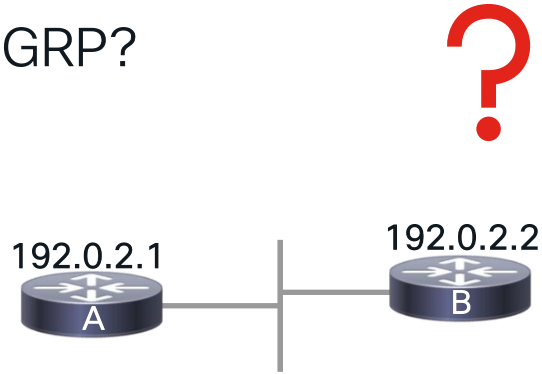


Neighbors

- The most useful command for checking neighbor status is **`show ip eigrp neighbors`**
- Some of the important information provided by this command are
  - Hold time—time left that you'll wait for an EIGRP packet from this peer before declaring him down
  - Uptime—how long it's been since the last time this peer was initialized
  - SRTT (Smooth Round Trip Time)—average amount of time it takes to get an Ack for a reliable packet from this peer
  - RTO (Retransmit Time Out)—how long to wait between retransmissions if Acks are not received from this peer

Where is the Neighbor?

- Is the interface enabled?
  - Should we expect a PEER based on the config?
- Is the interface active in EIGRP?
  - What do we see?



Neighbors – Interface Detail

```
RtrA#show ip eigrp interface
EIGRP-IPv4 VR(ciscolive) Address-Family Interfaces for AS(2022)
                              Xmit Queue   PeerQ        Mean   Pacing Time   Multicast    Pending
Interface              Peers  Un/Reliable  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Gi0/0/0                  1        0/0       0/0           1       0/0           50           0
```

```
RtrB#show ip eigrp interface detail
IP-EIGRP interfaces for process 1
                        Xmit Queue    Mean    Pacing Time   Multicast    Pending
Interface    Peers      Un/Reliable   SRTT    Un/Reliable   Flow Timer   Routes
Et0/0          1            0/0       737        0/10          5376        0
  Hello interval is 5 sec
  Next xmit serial <none>
  Un/reliable mcasts: 0/3  Un/reliable ucasts: 6/3
  Mcast exceptions: 0  CR packets: 0  ACKs suppressed: 0
  Retransmissions sent: 0  Out-of-sequence rcvd: 0
  Authentication mode is not set
Et1/0          1            0/0       885        0/10          6480        0
  Hello interval is 5 sec
  Next xmit serial <none>
  Un/reliable mcasts: 0/2  Un/reliable ucasts: 5/3
  Mcast exceptions: 0  CR packets: 0  ACKs suppressed: 0
  Retransmissions sent: 0  Out-of-sequence rcvd: 0
  Authentication mode is not set
```

Neighbors – Network Statement and Interface

```
router eigrp ciscolive
 !
 address-family ipv4 unicast autonomous-system 2022
  !
  topology base
   eigrp event-log-size 20000
  exit-af-topology
  network 192.0.2.0 255.255.255.0
  eigrp router-id 1.1.1.1
 exit-address-family


interface GigabitEthernet0/0/0
 ip address 192.0.2.1 255.255.255.0
 negotiation auto
End
```

Interface IP must fall within range of EIGRP Network Statement (network/mask)

Neighbors – Passive Interface

```
router eigrp ciscolive
 !
 address-family ipv4 unicast autonomous-system 2022
  !
  af-interface GigabitEthernet0/0/0
   passive-interface
  exit-af-interface
  !
  topology base
   eigrp event-log-size 20000
  exit-af-topology
  network 192.0.2.0
  eigrp router-id 1.1.1.1
 exit-address-family
```

Neighbors – Autonomous System Must Match

```
router eigrp ciscolive
 !
 address-family ipv4 unicast autonomous-system 2022
  !
  topology base
   eigrp event-log-size 20000
  exit-af-topology
  network 192.0.2.0
  eigrp router-id 1.1.1.1
 exit-address-family
```

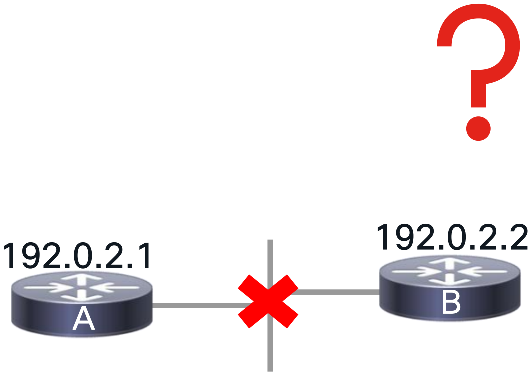
```
router eigrp ciscolive
 !
 address-family ipv4 unicast autonomous-system 2021
  !
  topology base
   eigrp event-log-size 20000
  exit-af-topology
  network 192.0.2.0
  eigrp router-id 2.2.2.2
 exit-address-family
```

Neighbors – Interface Detail

- There is also a show ip eigrp interface which contains a subset of this info; You may want to just use that if you don't need all the detail
- This command supplies a lot of information about how the interfaces are being used and how well they are obeying; some of the interesting information available via this command is:
  - Retransmissions sent—this shows how many times EIGRP has had to retransmit packets on this interface, indicating that it didn't get an ack for a reliable packet; having retransmits is not terrible, but if this number is a large percentage of packets sent on this interface, something is keeping neighbors from receiving (and acking) reliable packets
  - Out-of-sequence rcvd—this shows how often packets are received out of order, which should be a relatively unusual occurrence; again, it's nothing to worry about if you get occasional out-of-order packets since the underlying delivery mechanism is best-effort—if the number is a large percentage of packets sent on the interface, however, then you may want to look into what's happening on the interface—are there errors?
- You can also use this command to see if an interface only contains stub neighbors and if authentication is enabled

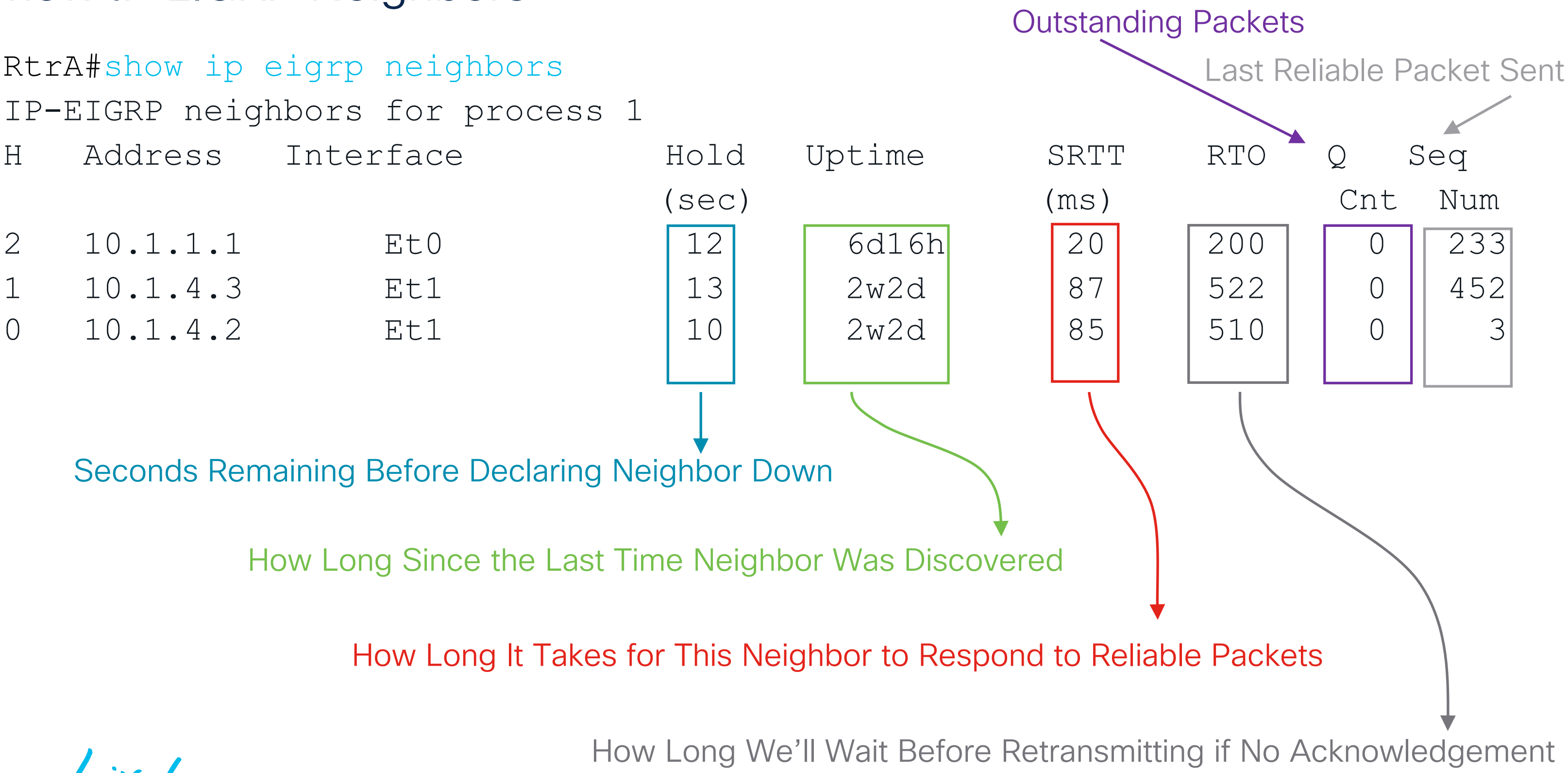
What’s Wrong with the Neighbor?

- Check the uptime
- Check the quality of communication
- Who’s fault?



Neighbors

Show IP EIGRP Neighbors



```
RtrA#show ip eigrp neighbors
IP-EIGRP neighbors for process 1
H   Address     Interface            Hold   Uptime      SRTT   RTO    Q    Seq
                                     (sec)              (ms)          Cnt  Num
2   10.1.1.1          Et0             12     6d16h      20     200    0    233
1   10.1.4.3          Et1             13     2w2d       87     522    0    452
0   10.1.4.2          Et1             10     2w2d       85     510    0      3
```

Outstanding Packets

Last Reliable Packet Sent

Seconds Remaining Before Declaring Neighbor Down

How Long Since the Last Time Neighbor Was Discovered

How Long It Takes for This Neighbor to Respond to Reliable Packets

How Long We’ll Wait Before Retransmitting if No Acknowledgement

Neighbors

- EIGRP Log-Neighbor-Changes is on by default since 12.2(12)
- Turn it on and leave it on
- Best to send to buffer log

```
RtrA# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
RtrA(config) # router eigrp 1
RtrA(config-router) # eigrp log-neighbor-changes
RtrA(config-router) # logging buffered 10000
RtrA(config) # service timestamps log datetime msec
```

Neighbors - Detail

- The detailed relative of the show ip eigrp neighbor command; some of the additional information available via the detailed version of this command include
  - Number of retransmissions and retries for each neighbor
  - Version of Cisco IOS and EIGRP
  - Stub information (if configured)

```
rtr302-ce1#show ip eigrp neighbor detail
IP-EIGRP neighbors for process 1
H    Address      Interface        Hold  Uptime    SRTT   RTO    Q    Seq  Type
                                   (sec)           (ms)          Cnt  Num
1    17.17.17.2    Et1/0            14   00:00:03  394    2364   0    124
   Version 12.0/1.2, Retrans: 0, Retries: 0
   Stub Peer Advertising ( CONNECTED SUMMARY ) Routes
0    50.10.10.1    Et0/0            13   04:04:39   55     330   0     13
   Version 12.0/1.2, Retrans: 2, Retries: 0
```

Neighbors – What about when they don’t work?

- Log-Neighbor-Changes Messages
- So this tells us why the neighbor is bouncing—but what do they mean?

```
Neighbor 10.1.1.1 (Ethernet0) is down: peer restarted
Neighbor 10.1.1.1 (Ethernet0) is up: new adjacency
Neighbor 10.1.1.1 (Ethernet0) is down: holding time expired
Neighbor 10.1.1.1 (Ethernet0) is down: retry limit exceeded
```

Sometimes others, but not often

- Hint: peer restarted means you must ask the peer; it’s the one that restarted the session

Neighbors

- EIGRP log-neighbor-changes is the best tool you have to understand why neighbor relationships are not stable. It should be enabled on every router in your network—CSCdx67706 (12.2(12)) made it the default behavior; as explained on the previous slide, the uptime value from show ip eigrp neighbors will tell you the last time a neighbor bounced, but not how often or why—with log-neighbor-changes on and logging buffered, you keep not only a history of when neighbors have been reset, but the reason why… absolutely invaluable
- Logging buffered is also recommended, because logging to a syslog server is not bulletproof; for example, if the neighbor bouncing is between the router losing neighbors and the syslog server, the messages could be lost—it's best to keep these types of messages locally on the router, in addition to the syslog server
- It may also be useful to increase the size of the buffer log in order to capture a greater duration of error messages—you would hate to lose the EIGRP neighbor messages because of flapping links filling the buffer log; if you aren't starved for memory, change the buffer log size using the command logging buffered 10000 in configuration mode
- The service timestamps command above puts more granular timestamps in the log, so it's easier to tell when the neighbor stability problems occurred

Manual Changes

- Some manual configuration changes can also reset EIGRP neighbors, depending on the Cisco IOS version
  - Summary changes (manual and auto)
  - Route filter changes
  - Stub setting changes
- This is normal behavior for much older code
  - CSCdy20284 removed many of these neighbor resets
    - Implemented in 12.2S, 12.3T, and 12.4 (approximately 2005)
- Mismatch of K-values (metric weights) will prevent peers from forming also (best just not to change them at all).

Manual Changes

- Summary changes
  - When a summary changes on an interface, components of the summary may need to be removed from any neighbors reached through that interface; neighbors through that interface are reset to synch up topology entries
- Route filter changes
  - Similar to summary explanation above; neighbors are bounced if a distribute-list is added/removed/changed on an interface in order to synch up topology entries
- In the past, we also bounced neighbors when interface metric info changed (delay, bandwidth), but we no longer do that (CSCdp08764)
- CSCdy20284 was implemented to stop bouncing neighbors when many manual changes occur; in late 12.2S, 12.3T, and 12.4, summary and filter changes no longer bounce neighbors
- Changing Stub setting or router-ids still resets peers! Remember to make these changes during maintenance windows

Suspect #1 – EIGRP Neighbors: Takeaway

TIP

- Is the interface active in EIGRP, and the proper AS?
  - Show ip eigrp interfaces
  - Show run | section router eigrp
  - Is Authentication configured? Keys match?
- Is there a Neighbor over the interface?
  - Show ip eigrp neigh
- Is the Neighbor Healthy?
  - Show ip eigrp neighbor < detail >
  - Show ip eigrp interface < detail >

Neighbor Authentication with EIGRP

```
key chain CL-KEY
 key 2022
  key-string 31415
   accept-lifetime 00:00:01 Jan 1 2000 infinite

router eigrp ciscolive
 !
 address-family ipv4 unicast autonomous-system 2022
  !
  af-interface default
   authentication mode hmac-sha-256 ABCDEFG
   authentication key-chain CL-KEY
  exit-af-interface
```

```
key chain CL-KEY
 key 2022
  key-string 141421
   accept-lifetime 00:00:01 Jan 1 2000 infinite

router eigrp ciscolive
 !
 address-family ipv4 unicast autonomous-system 2022
  !
  af-interface default
   authentication mode hmac-sha-256 ABCDEFG
   authentication key-chain CL-KEY
  exit-af-interface
```

```
May 20 15:06:39.152: EIGRP: Gi0/0/0: ignored packet from 192.0.2.1, opcode = 5 (authentication off or key-chain missing)
May 20 15:06:40.452: EIGRP: Sending HELLO on Gi0/0/0 - paklen 20
May 20 15:06:40.452:   AS 2022, Flags 0x0:(NULL), Seq 0/0 interfaceQ 0/0 iidbQ un/rely 0/0
May 20 15:06:43.609: EIGRP: Gi0/0/0: ignored packet from 192.0.2.1, opcode = 5 (authentication off or key-chain missing)

May 20 15:03:09.498: %DUAL-5-NBRCHANGE: EIGRP-IPv4 2022: Neighbor 192.0.2.2 (GigabitEthernet0/0/0) is down: authentication
HMAC-SHA-256 configured
```

EIGRP Auth Command Ref

Suspect #2: Packets

Two Scenarios: Unreliable Packets Reliable Packets

EIGRP Packets

EIGRP Sends both Reliable and Unreliable Packet types

- Unreliable packets are:
  - Hellos
  - Acknowledgement
- Reliable packets are:
  - Updates
  - Queries
  - Replies
  - SIA-Queries
  - SIA-Replies
- Reliable packets are sequenced, require an acknowledgement, and are retransmitted up to 16 times if not acknowledged

EIGRP Packets

5 Basic Packet Types

EIGRP uses five different packet types to handle session management and pass DUAL Message types:

- HELLO Packets (includes ACK)
- QUERY Packets (includes SIA-Query)
- REPLY Packets (includes SIA-Reply)
- REQUEST Packets
- UPDATE Packets

EIGRP packets are directly encapsulated into a network-layer protocol, such as IPv4 or IPv6. While EIGRP is capable of using additional encapsulation (such as AppleTalk, IPX, etc.) no further encapsulation is specified in this document. (RFC-7868)

Updates & Advertisements

5 Basic Packet Types

- Hello/Acks
  Hellos are used for peer discovery/maintenance. They do not require acknowledgment. A hello packet with a non-zero sequence number is also used as an acknowledgment (ack). Hellos are normally multicast and Acks are always sent using a unicast address

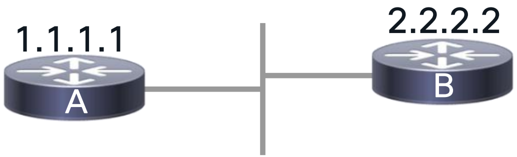
- Updates
  Updates are used to convey reachability of destinations. When a new peer is discovered, update packets are sent so the peer can populate its topology table. In some cases, update packets will be unicast. In other cases, such as a link cost change, updates are multicast. Updates are always transmitted reliably

- Query/Reply
  Queries are sent when destinations go into Active state. Queries are normally multicast to all peers on all interfaces except for the interface to the previous successor. If a receiving peer does not have an alternative path to the destination, it will in turn Query its peers until the *query boundary* is reached. Once the Query is sent, the router must wait for all the Replies from all peers before it can compute a new successor. Replies are sent containing the answer (which may be a valid metric or infinity/not reachable) and are unicast to the originator of the query. Both queries and replies are transmitted reliably

- SIA-Query/SIA-Reply
  If any peer fails to Reply to a Query, the destination is said to be Stuck In Active (SIA), and the peer may be reset. At ½ the SIA time (default 90 seconds) the router will send an SIA-Query to the non-replying peer. The peer must send either an SIA-Reply indicating the destination is still active, or a Reply. Both SIA-Queries and SIA-Replies are transmitted reliably

Unreliable: HELLOs and ACKs

- What can go wrong?
- What would we observe?
- What does EIGRP do?
  - (Hint: Hold time!)
- What should you do?

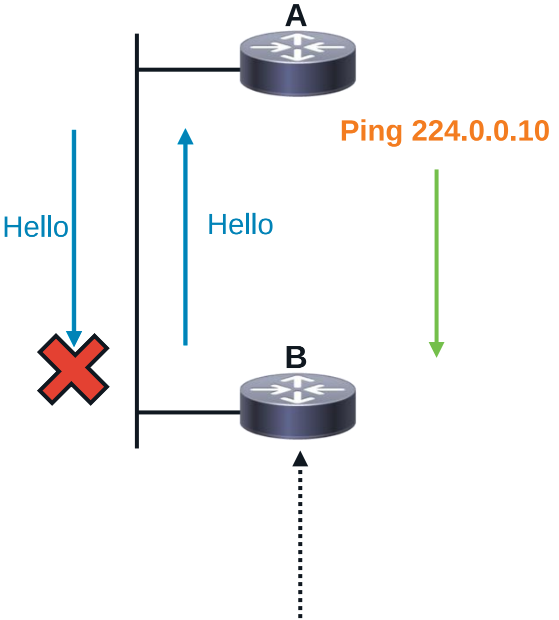


Unreliable: HELLOs and ACKs

Holding Time Expired

- The holding time expires when an EIGRP packet is not received during hold time
  - Typically caused by congestion or physical errors
- Ping the multicast address (224.0.0.10) from the **other** router
  - If there are a lot of interfaces or neighbors, you should use extended ping and specify the source address or interface

Neighbor 10.1.1.2 (Ethernet0) is down:
peer restarted



Neighbor 10.1.1.1 (Ethernet0) is down:
holding time expired

Holding Time Expired

- When an EIGRP packet is received from a neighbor, the hold timer for that neighbor resets to the hold time supplied in that neighbor’s hello packet, then the value begins decrementing
  - The hold timer for each neighbor is reset back to the hold time when each EIGRP packet is received from that neighbor (long ago and far way, it needed to be a hello received, but now any EIGRP packet will reset the timer)
  - Since hellos are sent every five seconds on most networks, the hold time value in a show ip eigrp neighbors is normally between 10 and 15 (resetting to hold time (15), decrementing to hold time minus hello interval or less, then going back to hold time)
- Why would a router not see EIGRP packets from a neighbor?
  - It may be gone (restarted, powered off, disconnected, etc.)
  - It (or we) may be overly congested (input/output queue drops, etc.)
  - Network between us may be dropping packets (CRC errors, frame errors, excessive collisions)

Holding Time Expired

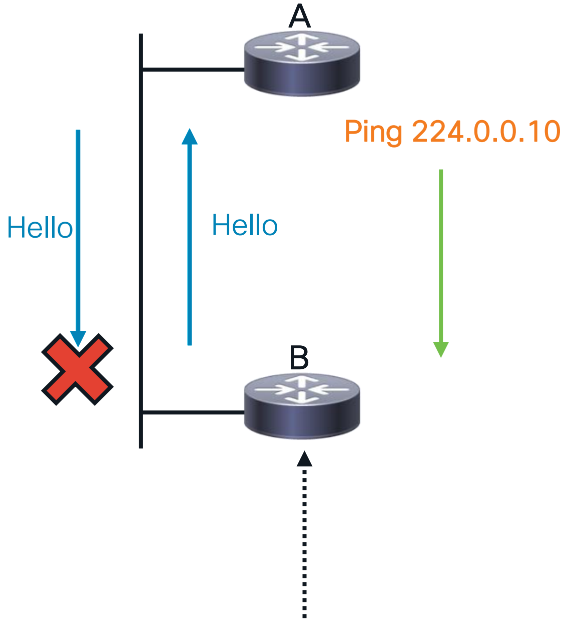
```
RtrA# debug eigrp packet hello
EIGRP Packets debugging is on (HELLO)
19:08:38.521: EIGRP: Sending HELLO on Serial1/1
19:08:38.521:   AS 1, Flags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0
19:08:38.869: EIGRP: Received HELLO on Serial1/1 nbr 10.1.6.2
19:08:38.869:   AS 1, Flags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0
19:08:39.081: EIGRP: Sending HELLO on FastEthernet0/0
19:08:39.081:   AS 1, Fags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0
```

Remember—Any Debug Can Be Hazardous

Holding Time Expired

- The holding time expires when an EIGRP packet is not received during hold time
  - Typically caused by congestion or physical errors
- Next Step:
- Ping the multicast address (224.0.0.10) from the **other** router
  - If there are a lot of interfaces or neighbors, you should use extended ping and specify the source address or interface

Neighbor 10.1.1.2 (Ethernet0) is down: peer restarted



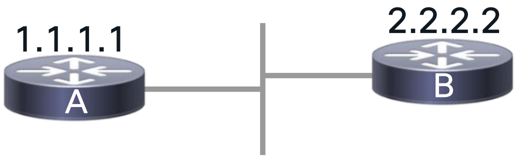
Neighbor 10.1.1.1 (Ethernet0) is down: holding time expired

Holding Time Expired

- Another troubleshooting tool available is to do the command “debug eigrp packet hello”; this will produce debug output to the console or buffer log (depending on how you have it configured) that will show the frequency of hellos sent and received
- You should make sure you have the timestamps for the debugs set to a value that you can actually see the frequency; something like:
  - `service timestamps debug datetime msec`
- Remember that any time you enable a debug on a production router, you are taking a calculated risk; it’s always better to use all of the safer troubleshooting techniques before resorting to debugs—sometimes they’re necessary, however

Reliable: UPDATES, QUERIES, & REPLIES

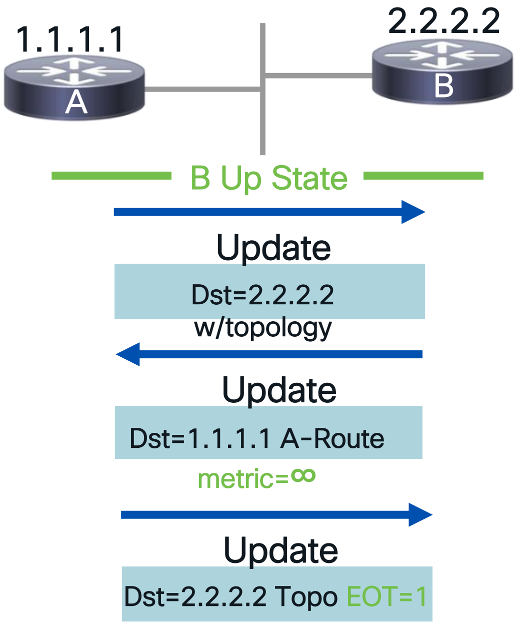
- What can go wrong?
- What would we observe?
- What does EIGRP do?
  - (Hint: Retransmit!)
- What should you do?



Updates & Advertisements

Advertisement

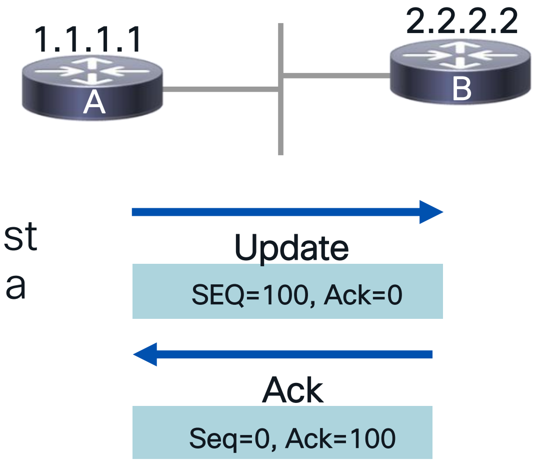
- For each route A sends to B, B sends a poison reverse
- This makes certain the two routers tables are accurate as well as making sure other routers on the interface they share use the right path for each prefix
- When a router finishes sending its table, it sends an end-of-table indicator



Updates & Advertisements

Sequence Numbers and Acks

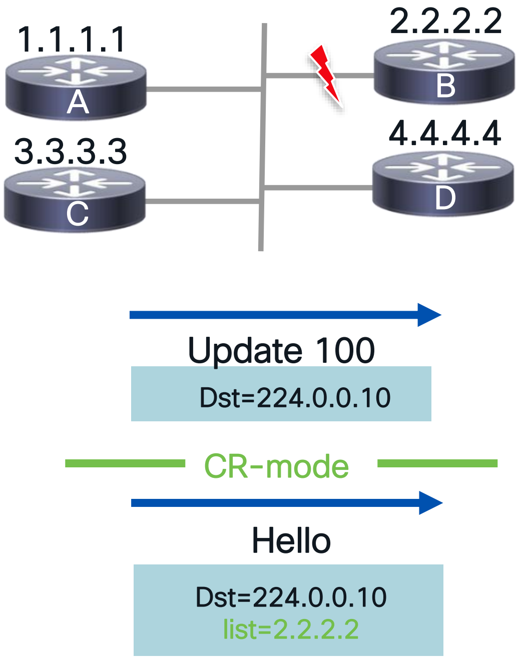
- An Update packet transmitted contains a sequence number that is acknowledged by a receipt of a Ack packet
- If the Update or the Ack packet is lost on the network, the Update packet will be retransmitted
- In the case of the Query packets, the Query packet also must be acknowledged (“I heard the question”, later followed by a Reply packet (“Here is the answer”).
  - Note that both responses are required and perform different functions
- Replies also contain sequence numbers and must be acknowledged



Updates & Advertisements

Conditional Receive (CR-mode)

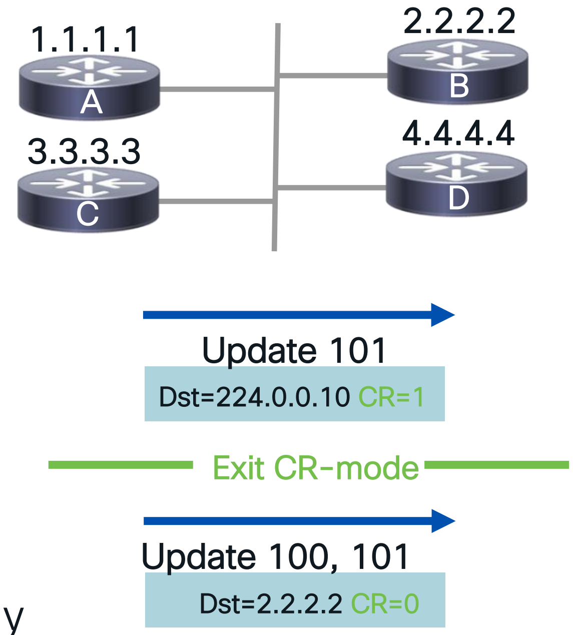
- A sends a multicast Update packet
- B, C and D receive the Update and send an acknowledgment
- B’s acknowledgments is lost on the network
- Before the retransmission timer expires, A has an event that requires it to send a new multicast update on this interface
- A detects that B has not Acked the last packet and enters the Conditional Receive process
- A builds a Hello packet with a SEQUENCE TLV containing B’s address
- This special Hello packet is multicasted unreliably out the interface



Updates & Advertisements

Conditional Receive (CR-mode)

- C and D process the special Hello packet looking for their address in the list. If not found, they put themselves in Conditional Receive (CR-mode) mode
- Any subsequent reliable packets received on C and D with the CR-flag set are accepted and processed
- B does not put itself in CR-mode because it finds its address in the list
- Reliable packets received by B with the CR-flag must be discarded and not acknowledged
- Once A has sent the CR Update(s), it exits CR-mode
- A will unicast the previous, unacknowledged packets directly to B



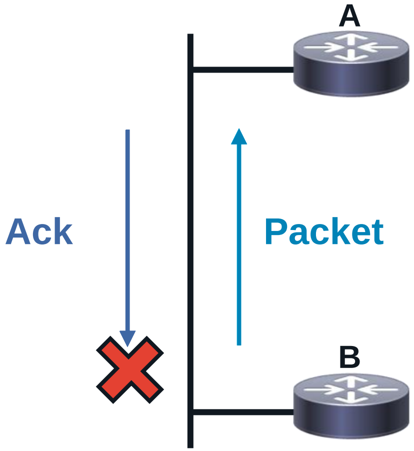
Retry Limit Exceeded

- EIGRP sends both unreliable and reliable packets
  - Hellos and Acks are unreliable
  - Updates, Queries, Replies, SIA-queries and SIA-replies are reliable
- Reliable packets are sequenced and require an acknowledgement
  - Reliable packets are retransmitted up to 16 times if not acknowledged

Retry Limit Exceeded

- Reliable packets are re-sent after Retransmit Time Out (RTO)
  - Typically 6 x Smooth Round Trip Time (SRTT)
  - Minimum 100 ms
  - Maximum 5000 ms (five seconds)
  - 16 retransmits takes between roughly 40 and 80 seconds
- If a reliable packet is not acknowledged before 16 retransmissions and the hold time has not expired, re-initialize the neighbor

Neighbor 10.1.1.2 (Ethernet0) is down: peer restarted



Neighbor 10.1.1.1 (Ethernet0) is down: retry limit exceeded

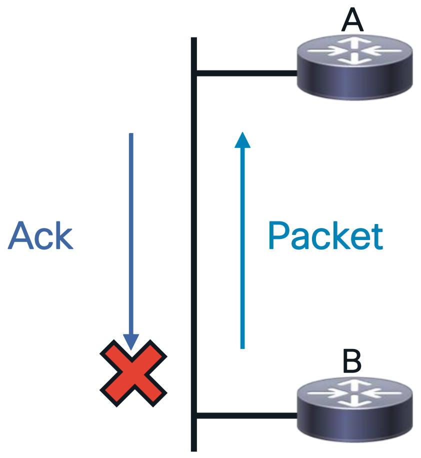
Retry Limit Exceeded

- Exceeding the retry limit means that we're sending reliable packets which are not getting acknowledged by a neighbor—when a reliable packet is sent to a neighbor, it must respond with a unicast acknowledgement; if a router is sending reliable packets and not getting acknowledgements, one of two things are probably happening
  - The reliable packet is not being delivered to the neighbor
  - The acknowledgement from the neighbor is not being delivered to the sender of the reliable packet
- These errors are normally due to problems with delivery of packets, either on the link between the routers or in the routers themselves—congestion, errors, and other problems can all keep unicast packets from being delivered properly; look for queue drops, errors, etc., when the problem occurs, and try to ping the unicast address of the neighbor to see if unicasts in general are broken or whether the problem is specific to EIGRP

Retry Limit Exceeded

- Reliable packets are re-sent after Retransmit Time Out (RTO)
  - Typically 6 x Smooth Round Trip Time (SRTT)
  - Minimum 100 ms
  - Maximum 5000 ms (five seconds)
  - 16 retransmits takes between roughly 40 and 80 seconds
- If a reliable packet is not acknowledged before 16 retransmissions and the hold time has not expired, re-initialize the neighbor

Neighbor 10.1.1.2 (Ethernet0) is down: peer restarted



Neighbor 10.1.1.1 (Ethernet0) is down: retry limit exceeded

Retry Limit Exceeded

- The Retransmit Timeout (RTO) is used to determine when to retry sending a packet when an Ack has not been received, and is (generally) based on 6 X Smooth Round Trip Time (SRTT); the SRTT is derived from previous measurements of how long it took to get an Ack from this neighbor—the minimum RTO is 100 Msec and the maximum is 5000 Msec; each retry backs off 1.5 times the last interval
- The minimum time required for 16 retransmits is approximately 40 seconds (minimum interval of 100 ms with a max interval of 5000 ms); for example, If there isn't an acknowledgement after 100 ms, the packet is retransmitted and we set a timer for 150 ms—if it expires, we send it again and set the timer for 225 ms, then 337 ms, etc., until 5000 ms is reached; 5000 ms is then repeated until a total of 16 retransmissions have been sent
- The maximum time for 16 retransmits is approximately 80 seconds, if the initial retry is 5000 ms and all subsequent retries are also 5000 ms

Retry Limit Exceeded

- If a reliable packet is retransmitted 16 times without an acknowledgement, EIGRP checks to see if the duration of the retries has reached the hold time, as well
- Since the hold time is typically 15 sec on anything but low-speed NBMA, it normally isn’t a factor in the retry limit; NBMA links that are T1 or less, however, wait an additional period of time after re-trying 16 times, until the hold-time period (180 seconds) has been reached before declaring a neighbor down due to retry limit exceeded
- This was done to give the low-speed NBMA networks every possible chance to get the Acks across before downing the neighbor
- Remember this if you modify the hold times!

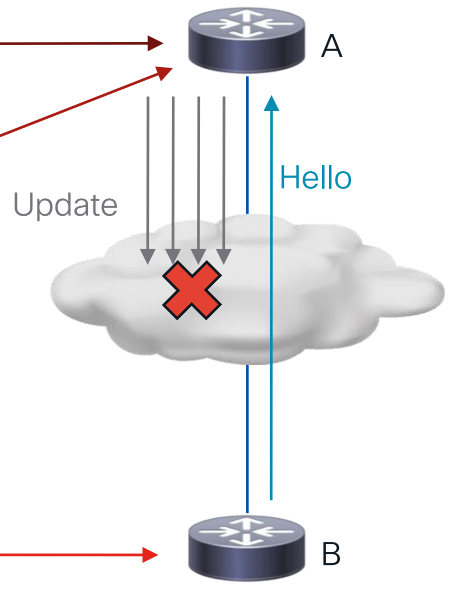
Unidirectional Links

```
RtrB#show ip eigrp neighbors
IP-EIGRP neighbors for process 1
H Address      Interface  Hold   Q     Seq
                                Cnt    Num

1 10.1.102.2      Et0        14    4       0
```

```
%DUAL-5-NBRCHANGE: IP-EIGRP 1: Neighbor 10.1.5.4 (Serial1) is down:
retry limit exceeded
```

```
RtrA#show ip eigrp neighbors
IP-EIGRP neighbors for process 1
RtrA#
```



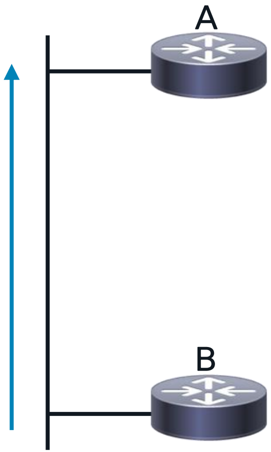
Unidirectional Links

- In this example, we see what happens when a link is only working in one direction; unidirectional links can occur because of a duplicate IP address, a wedged input queue, link errors, or any other reason you can think of that would allow packets to be delivered only in one direction on a link
- RtrB doesn’t even realize that RtrA exists—RtrA is sending out his hellos, waiting for a neighbor to show up on the network; what it doesn’t realize is that the RtrB is already out there and trying to bring up the neighbor relationship
- RtrB, on the other hand, sees the hellos from RtrA, sends his own hellos and then sends an update to RtrA to try to get their topology tables/routing tables populated—unfortunately, since the updates are also not being received by RtrA, it of course isn’t sending acknowledgements; RtrB tries it 16 times and then resets his relationship with RtrA and starts over
- You’ll spot this symptom by the retry limit exceeded messages on RtrB, RtrB having RtrA in his neighbor table with a continual Q count, and RtrA not seeing RtrB, at all
- CSCdy45118 has been implemented to create a reliable neighbor establishment process (three-way handshake) and reliable neighbor maintenance (neighbor taken down more quickly when unidirectional link encountered). 12.2T, 12.3 and up

Retry Limit Exceeded

- Ping the neighbor’s unicast address
  - Vary the packet size
  - Try large numbers of packets
- This ping can be issued from either neighbor; the results should be the same
- Common causes
  - Mismatched MTU (check for giants)
  - Unidirectional link
  - Dirty link (check show interface for errors)

```
RtrB# ping
Protocol[ip]:
Target IP address: 10.1.1.1
Repeat count [5]: 100
Datagram Size: 1500
Timeout in seconds[2]:
Extended commands[n]: y
....
```



Log-Neighbor-Changes Messages

- Peer restarted—the other router reset our neighbor relationship; you need to go to that device to see why it thought our relationship had to be reset
- New adjacency—established a new neighbor relationship with this neighbor; happens at initial startup and after recovering from a neighbor going down
- Holding time expired—we didn’t hear any EIGRP packets from this neighbor for the duration of the hold time; this is typically 15 seconds for most media (180 seconds for low-speed NBMA)
- Retry limit exceeded—this neighbor didn’t acknowledge a reliable packet after at least 16 retransmissions (actual duration of retransmissions is also based on the hold time, but there were at least 16 attempts)

Suspect #2 – Packets: Takeaway

TIP

- Unicast
  - Can you ping the other side?
- Multicast
  - Check the Reliable Transport, retries, and ability to do a multicast ping to the EIGRP link-local mcast address.
- Regardless, EIGRP rides on an IP based transport (v4, v6). If the traffic is not getting through, check the underlying path and interface health.

Suspect #3: Computations

Two Scenarios: Feasibility Active Process

Understanding Convergence

- The one constant in life is that nothing is constant. Not even our networks. ☺
- There are three network change scenarios to consider:
  - Convergence with Feasible Successors
  - Convergence with Non-Feasible Successors
  - When things don't converge as you had planned!
- Understanding your topology and the appropriate scenario is essential to know where to look and what to look for.
- Get up close and personal with your topology table and event-logs!

Topology Table & DUAL

Topology Table

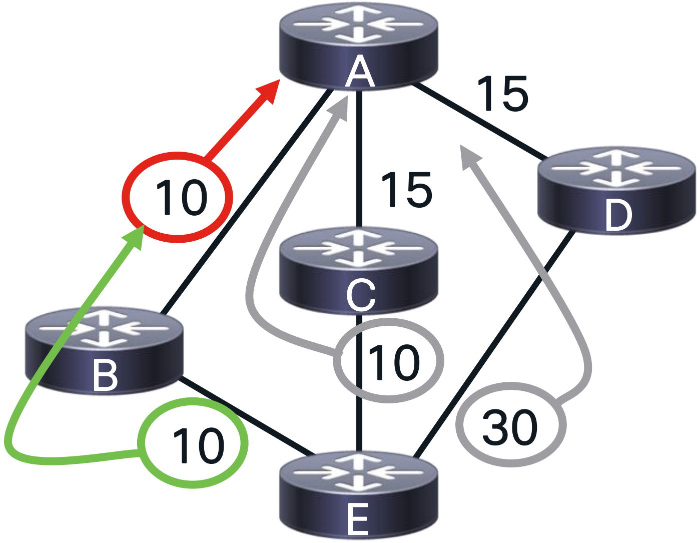
- The topology table is probably the most critical structure in EIGRP
  - Contains all known paths, local, learned, and external (redistributed)
  - Contains building blocks used by DUAL
  - Used to create updates for neighbors
  - Used to populate the routing table
- Understanding the topology table contents is extremely important for understanding your network and troubleshooting EIGRP

Topology Table

- Remember our order of operation!
  - Form Neighbors
  - Exchange Updates
  - Use Metrics to determine which path is best, through which the destination is “closest”.
  - Best path(s) are installed into the routing table (rib)
  - Neighbors are updated
- Topology Table is like the local database of all known paths, from which we determine what is best.

Topology Table & DUAL - Key Terms

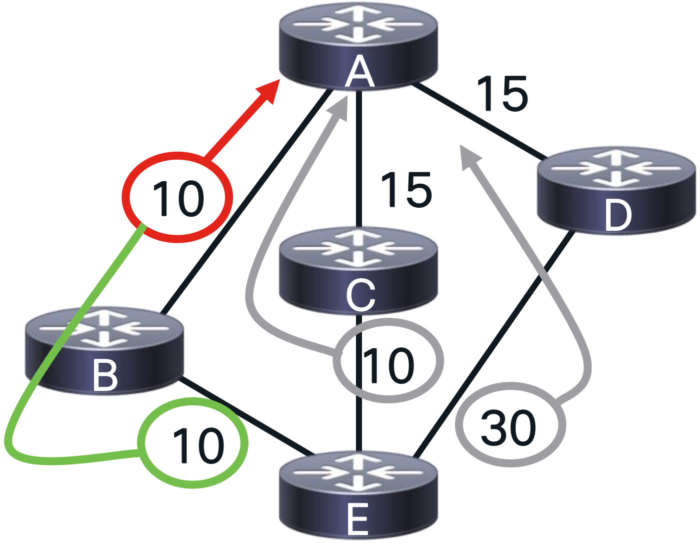
- From Router A's perspective, looking at updates coming inbound:
- Reported Distance (RD)
  - Reported Distance is the total distance along a path to a destination network as advertised by an upstream peer
  - Example: 10, 10, and 30
- Computed Distance (CD)
  - The Reported Distance plus link cost to reach the upstream peer
  - Example: 10 + 10 = 20, 10+15, 30+15
- Feasible Distance (FD)
  - Feasible Distance is the *lowest* Computed Distance to a particular destination
  - Example: 20 is less than 25, 20 is less than 45, thus 20 is the FD for this particular destination



"Particular Destination"

Topology Table & DUAL - Key Terms

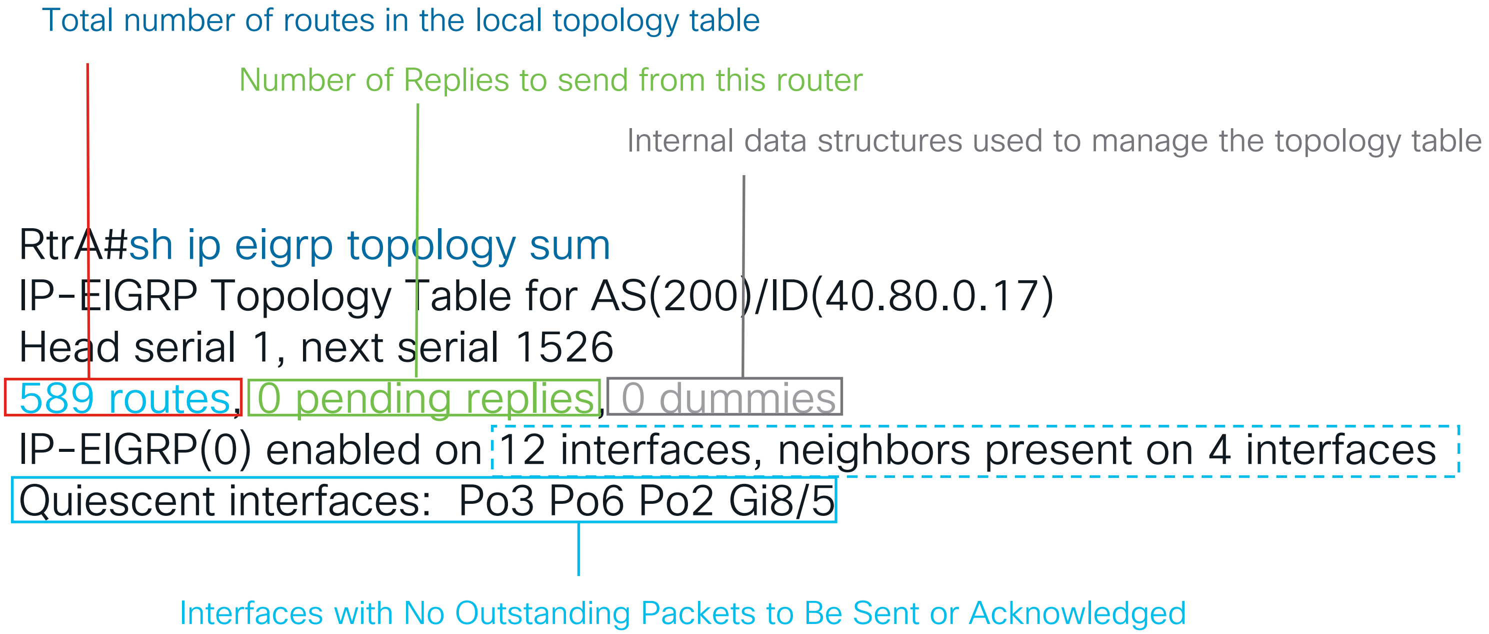
- Feasibility Condition
  - If a router’s Reported Distance is less than our Feasible Distance, then this router is a loop-free path to this destination and meets the Feasibility Condition (RD < FD) (RD of 10 is less than FD of 20)
- Feasible Successor
  - A router is a Feasible Successor if it satisfies the Feasibility Condition for a particular destination
  - C-RD:10 < 20, thus **C is a FS**. D-RD:30 is not < 20, **D is NOT a FS**.
- Successor
  - A router is a Successor if it satisfies the Feasibility Condition AND it provides the lowest distance (metric) to that destination
- Active and Passive State
  - A route is in the Passive state when it has a successor for the destination. The router goes to Active state when current successor no longer satisfies the Feasibility Condition and there are no Feasible Successors identified for that destination



“Particular Destination”

Topology Table

Show IP EIGRP Topology Summary



Total number of routes in the local topology table

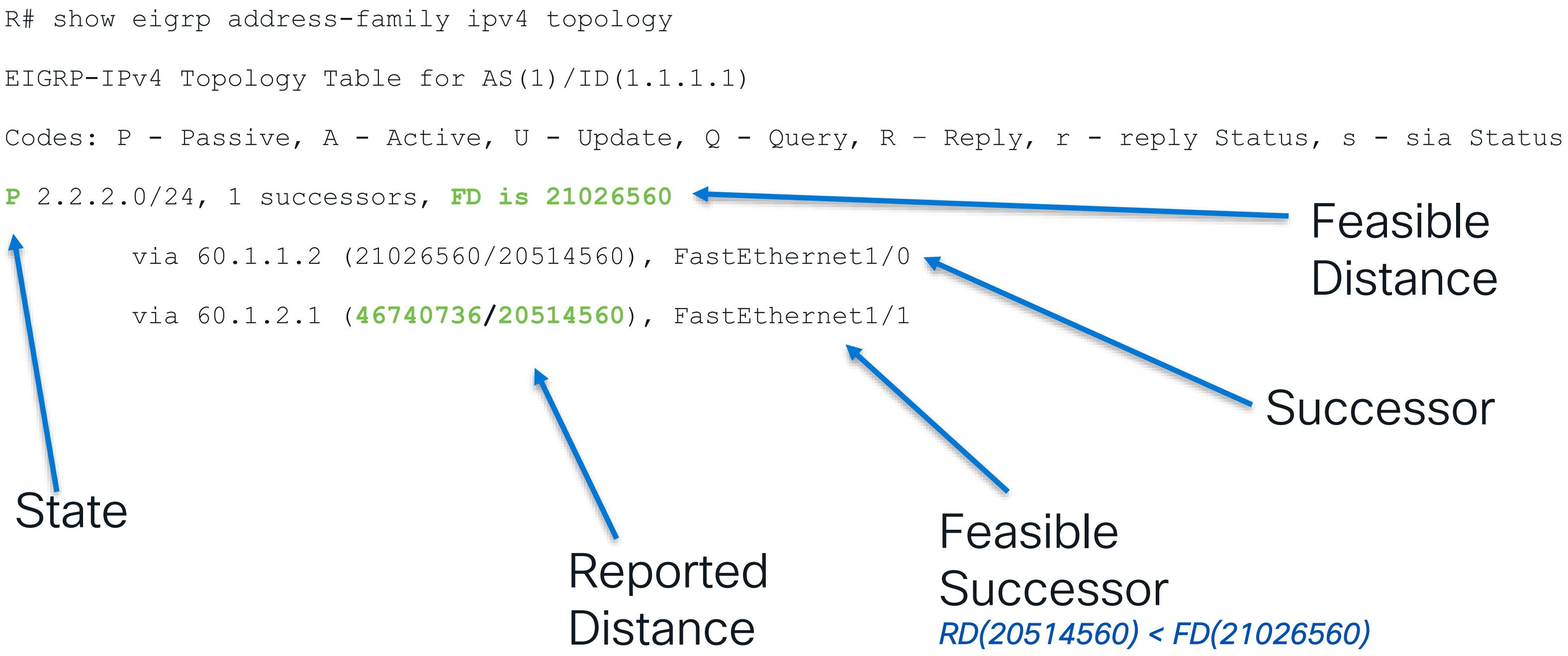
Number of Replies to send from this router

Internal data structures used to manage the topology table

```
RtrA#sh ip eigrp topology sum
IP-EIGRP Topology Table for AS(200)/ID(40.80.0.17)
Head serial 1, next serial 1526
589 routes, 0 pending replies, 0 dummies
IP-EIGRP(0) enabled on 12 interfaces, neighbors present on 4 interfaces
Quiescent interfaces:  Po3 Po6 Po2 Gi8/5
```

Interfaces with No Outstanding Packets to Be Sent or Acknowledged

Topology Table – Where?



```
R# show eigrp address-family ipv4 topology

EIGRP-IPv4 Topology Table for AS(1)/ID(1.1.1.1)

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

P 2.2.2.0/24, 1 successors, FD is 21026560

        via 60.1.1.2 (21026560/20514560), FastEthernet1/0

        via 60.1.2.1 (46740736/20514560), FastEthernet1/1
```

Feasible Distance

Successor

State

Reported Distance

Feasible Successor

RD(20514560) < FD(21026560)

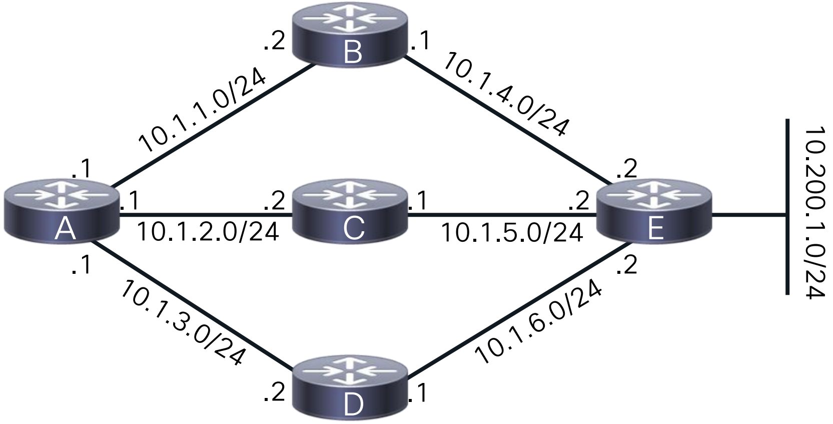
Topology Table

- One of the reasons that EIGRP is called an advanced distance vector protocol is that it retains more information than just the best path for each route it receives—this means that it can potentially make decisions more quickly when changes occur, because it has a more complete view of the network than RIP, for example; the place this additional information is stored is in the topology table
- The topology table contains an entry for every route EIGRP is aware of, and includes information about the paths through all neighbors that have reported this route to him—when a route is withdrawn by a neighbor, EIGRP will look in the topology table to see if there is a feasible successor, which is another downstream neighbor that is guaranteed
  to be loop-free; if so, EIGRP will use that neighbor and never have to go looking farther
- Contrary to popular belief, the topology table also contains routes which are not feasible; these are called possible successors and may be promoted to feasible successors, or even successors if the topology of the network were to change
- The following slides show a few different ways to look at the topology table and give hints on how to evaluate it

Topology Table

Show IP EIGRP Topology

- Displays a list of successors and feasible successors for all destinations known by EIGRP



```
RtrA#show ip eigrp topology
IP-EIGRP Topology Table for AS(1)/ID(10.1.6.1)
..snip…..
P 10.200.1.0/24, 1 successors, FD is 21026560
         via 10.1.1.2 (21026560/20514560), Serial1/0
         via 10.1.2.2 (46740736/20514560), Serial1/1
```

Feasible distance

Successor

Feasible successor

Computed distance

Reported distance

Topology Table

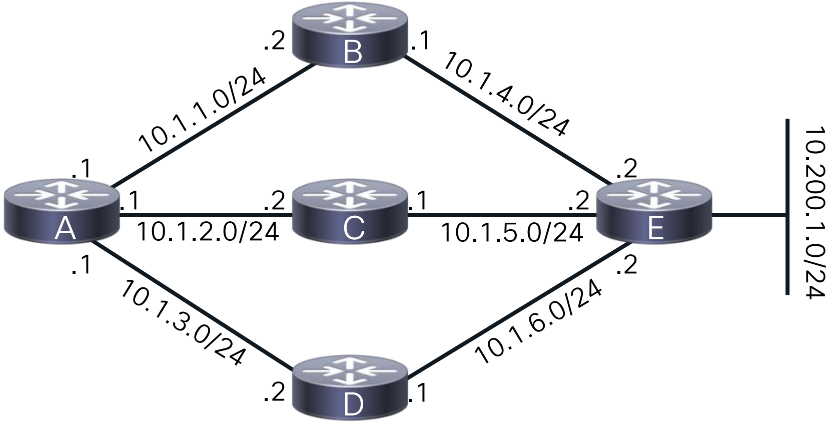
Show IP EIGRP Topology

- The most common way to look at the topology table is with the generic show ip eigrp topology command; this command displays all of the routes in the EIGRP topology table, along with their successors and feasible successors
- In the above example, the P on the left side of the topology entry displayed means the route is Passive–if it has an A, it means the route is Active; the destination being described by this topology entry is for 10.200.1.0 255.255.255.0–this route has one successor, and the feasible distance is 21026560; the feasible distance is normally the metric that would appear in the routing table if you did the command show ip route 10.200.1.0 255.255.255.0 (but not always)
- Following the information on the destination network, the successors and feasible successors are listed–the successors (one or more) are listed first, then the feasible successors are listed; the entry for each next-hop includes the IP address, the computed distance through this neighbor, the reported distance this neighbor told us, and which interface is used to reach him
- 10.1.2.2 is a feasible successor because his reported distance (21514560) is less than our current feasible distance (21026560) (It’s a smaller distance, and that means it is closer.)

Topology Table

Show IP EIGRP Topology All-links

- Displays a list of all neighbors who are providing EIGRP with an alternative path to each destination



```
RtrA#show ip eigrp topology all-links
IP-EIGRP Topology Table for AS(1)/ID(10.1.6.1)
…..snip…..
P 10.200.1.0/24, 1 successors, FD is 21026560
        via 10.1.1.2 (21026560/20514560), Serial1/0
        via 10.1.2.2 (46740736/20514560), Serial1/1
        via 10.1.3.2 (46740736/46228736), Serial1/2
```

Feasible distance

Successor

Feasible successor

Possible successor

Topology Table

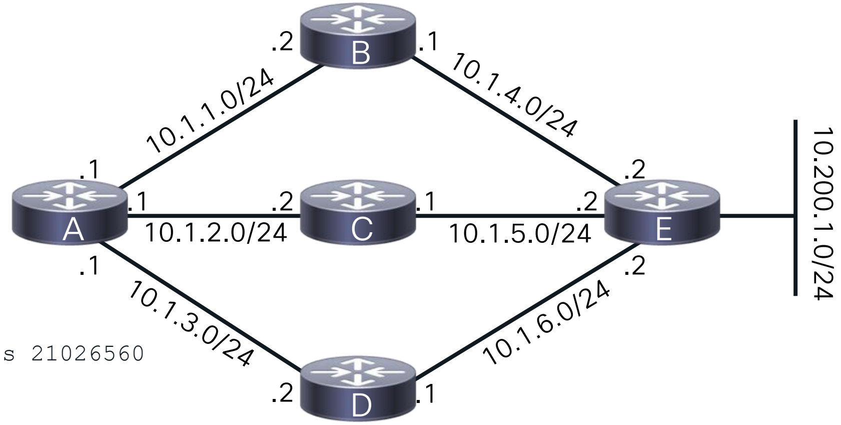
Show IP EIGRP Topology All-links

- If you want to display all of the paths which EIGRP contains in its topology table, use the show ip eigrp topology all-links command
- You'll notice in the above output that not only are the successor (10.1.1.2) and feasible successor (10.1.2.2) shown, but another router that doesn't qualify as either is also displayed; the reported distance from 10.1.3.2 (46228736) is far worse than the current feasible distance (21026560), so it isn't feasible
- This command is often useful to understand the true complexity of network convergence—I've been on networks with pages of non-feasible alternative paths in the topology table because of a lack of summarization/distribution lists; these large numbers of alternative paths can cause EIGRP to work extremely hard when transitions occur and can actually keep EIGRP from successfully converging

Topology Table

Show IP EIGRP Topology <net><mask>

- Displays detailed information for all paths received for a particular destination



```
RtrA#show ip eigrp topology 10.200.1.0/24
IP-EIGRP topology entry for 10.200.1.0/24
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 21026560
  Routing Descriptor Blocks:
  10.1.1.2 (Serial1/0), from 10.1.1.2, Send flag is 0x0
      Composite metric is (21026560/20514560), Route is Internal
      Vector metric:
        ....
  10.1.2.2 (Serial1/1), from 10.1.2.2, Send flag is 0x0
      Composite metric is (46740736/20514560), Route is Internal
      Vector metric:
        ....
  10.1.3.2 (Serial1/2), from 10.1.3.2, Send flag is 0x0
      Composite metric is (46740736/46228736), Route is Internal
      Vector metric:
        ....
```

Topology Table

Show IP EIGRP Topology <network><mask>

- If you really want to know all of the information EIGRP stores about a particular route, use the command show ip eigrp topology <network><mask>
  - Note that the mask can be supplied in dotted decimal or /xx form
- In the above display, you'll see that EIGRP not only stores which next-hops have reported a path to the target network, it stores the metric components used to reach the total (composite) metric
- You also may notice that EIGRP contains a hop count in the vector metrics—the hop count isn't actually used in calculating the metric, but instead was included to limit the apparent maximum diameter of the network; in EIGRP's early days, developers wanted to ensure that routes wouldn't loop forever and put this safety net in place—in today's EIGRP, it actually isn't necessary any longer, but is retained for compatibility

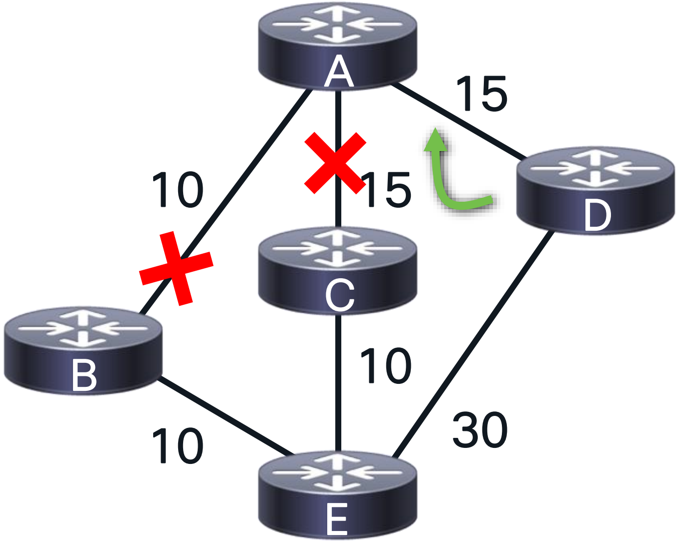
EIGRP Convergence – With Feasible Successor

- EIGRP selects Successor and Feasible Successor (FS)
- Successor is the best route
- FS is 2nd best route
- Must be mathematically loop-free (meets feasibility condition)
- FS acts as a “backup route”
- Kept in topology table (not routing table)
- Up to 6 Feasible Successors
- Built into the protocol, nothing to enable

DUAL – Failover to a Non-Feasible Successor

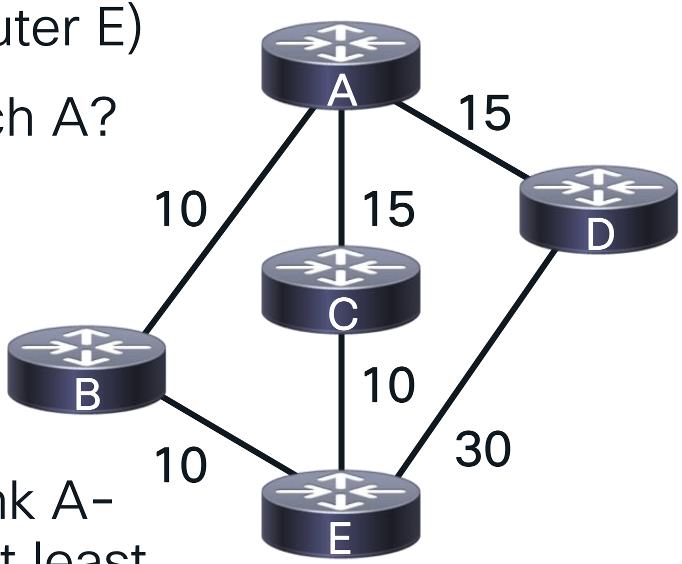
- D receives a query from Router-A and examines its topology information
- Since its best path is not through A, the path it has to E is still valid
- D sends a reply to this query, indicating it still has a valid loop free path to E

✓ Once A receives this reply, it begins using the path through D



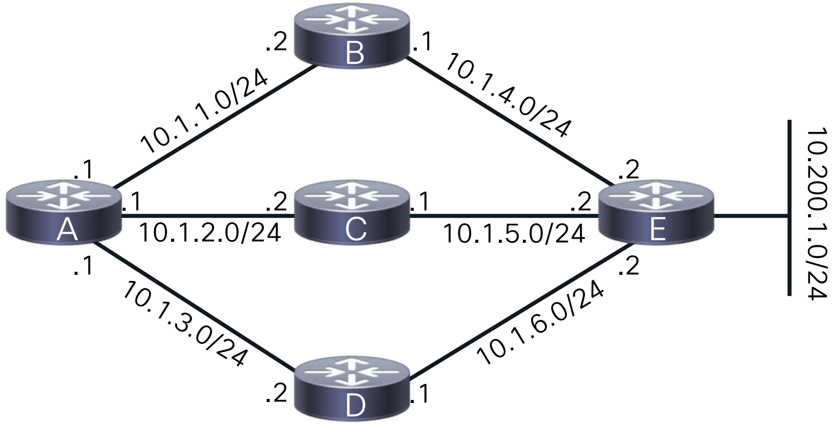
DUAL – Failover to Feasible Successor

- What about the *other* direction? (From Router E)
- Are there any Feasible Successors to reach A?
  - FD is 20 through B
  - RD from C is 15
  - RD from D is 15
  - RD < FD, so both satisfy the Feasibility Condition (FC)
  - We have two FS!
- In order for there to be only one FS, the link A-D or A-C would need to be increased to at least 20



Convergence of a Feasible Successor

- Near immediate rewrite of the next hop in the rib/fib.
- Extremely fast, and linear convergence based on prefix count.



```
RtrA#show ip eigrp topology
IP-EIGRP Topology Table for AS(1)/ID(10.1.6.1)
..snip…..
P 10.200.1.0/24, 1 successors, FD is 21026560
         via 10.1.1.2 (21026560/20514560), Serial1/0
         via 10.1.2.2 (46740736/20514560), Serial1/1
```

Feasible distance

Successor

Feasible successor

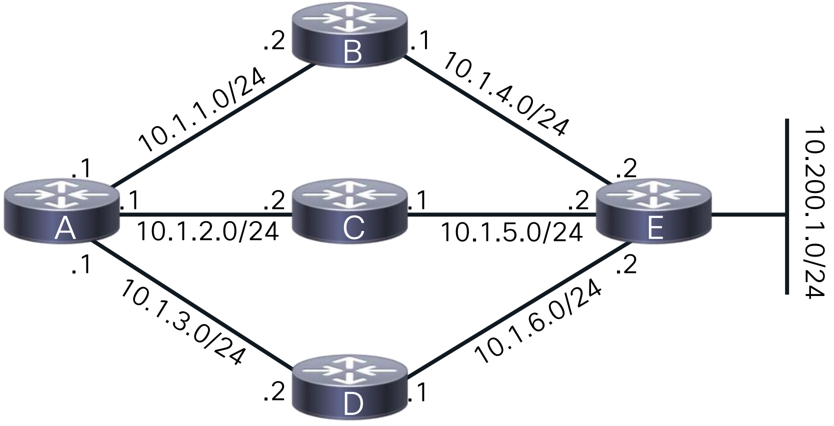
Computed distance

Reported distance

RD (20514560) < FD (21026560) = FS!

Convergence of a Feasible Successor

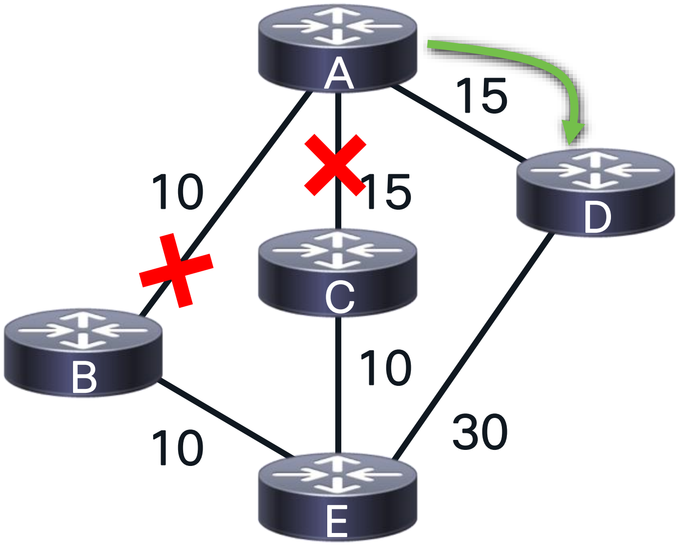
- Indicated in the Event Log, local computation
- Extremely fast, and linear convergence based on prefix count.



```
RtrA#show ip eigrp event
…
97   11:12:06.124 Metric set: 10.1.4.0/24 metric(20480)
98   11:12:06.124 Route installing: 10.1.4.0/24 10.1.2.2
99   11:12:06.124 Route installed: 10.1.4.0/24 10.1.1.2
100  11:12:06.124 Route installing: 10.1.4.0/24 10.1.1.2
101  11:12:06.124 RDB delete: 10.1.4.0/24 10.1.3.2
102  11:12:06.124 FC sat rdbmet/succmet: metric(20480) metric(20224)
103  11:12:06.124 FC sat nh/ndbmet: 10.1.1.2 metric(20480)
104  11:12:06.124 Find FS: 10.1.4.0/24 metric(20480)
105  11:12:06.124 Rcv update met/succmet: metric(Infinity) metric(Infinity)
106  11:12:06.124 Rcv update dest/nh: 10.1.4.0/24 10.1.3.2
107  11:12:06.123 Send reply: 10.1.4.0/24 10.1.2.2
108  11:12:06.123 Rcv query met/succ met: metric(Infinity) metric(Infinity)
109  11:12:06.123 Rcv query dest/nh: 10.1.4.0/24 10.1.2.2
110  11:12:06.123 Ignored route, hopcount: 10.1.4.0 255
```

Failover to a Non-Feasible Successor

- What if the path through C now fails?
- A examines its topology information, and finds it has no loop free path to E
- However, it does have a peer, and that peer might have a loop free path
- So, it places E in active state and Queries D

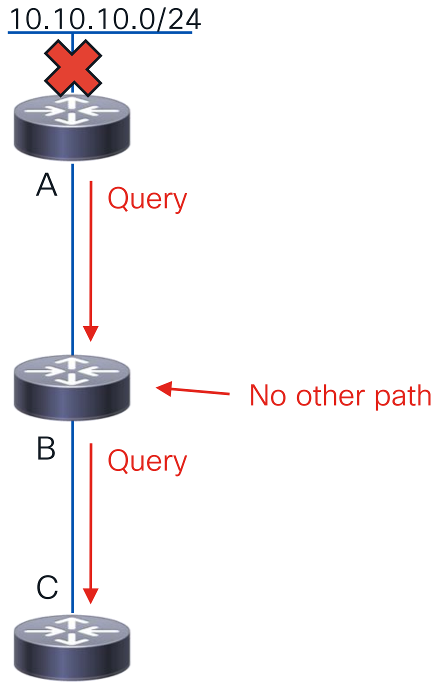


EIGRP Convergence – Without Feasible Successor

- Distance Vector Protocol
  - Doesn’t see the entire network like OSPF
- Based on QUERY and ACK messages for convergence
  - QUERY sent to determine best path for failed route
  - ACK sent when alternative path found or no other paths
- DUAL algorithm determines best path
  - Runs as soon as all outstanding QUERIES are received
- Query domain size can effect convergence time

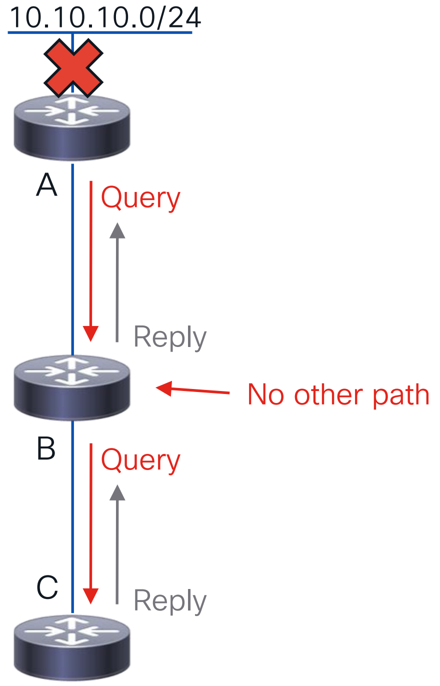
The Active Process

- RtrA loses its route to 10.10.10.0/24
- RtrA has no other path to this destination, so it marks the route as Active and sends a Query to RtrB
- RtrB receives this Query from its successor and has no other paths to reach the destination
- RtrB marks 10.10.10.0/24 as Active, and sends a Query to RtrC



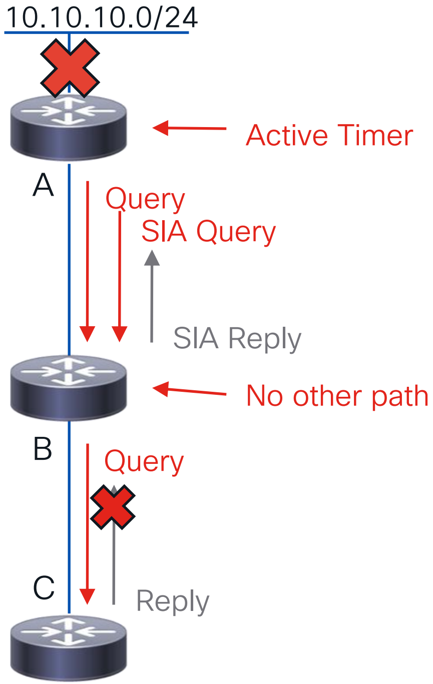
The Active Process

- RtrC receives the Query and has no more neighbors to Query and no alternate paths to 10.10.10.0/24
- RtrC marks the route as unreachable, and sends a Reply to RtrB
- RtrB receives the Reply, marks 10.10.10.0/24 as unreachable, and sends a Reply to RtrA
- RtrA receives the Reply and since it didn't learn any viable paths to reach 10.10.10.0/24, it deletes the route from the topology and routing tables



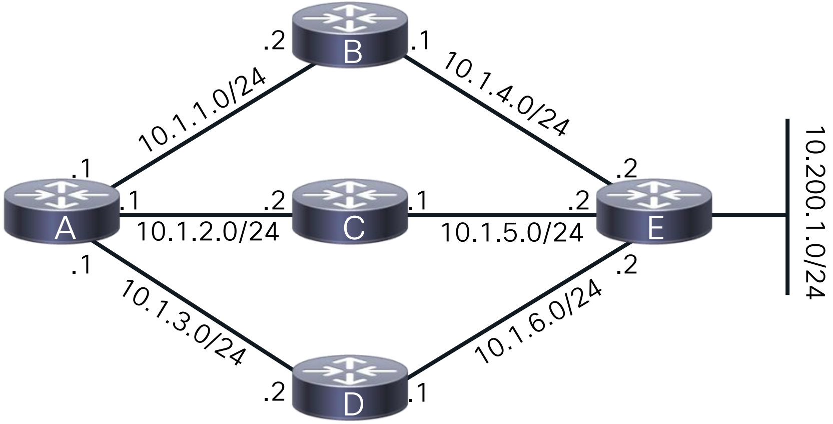
The Active Process

- What happens if RtrC ’s Reply isn’t sent, or doesn’t make it toRtrB?
- While RtrC is trying to send the Reply, RtrA ’s Active timer is running
- After 90 seconds, RtrA sends an SIA query to RtrB
- If RtrB is still waiting on RtrC , it sends an SIA reply to RtrA



Convergence of a Non-Feasible Successor

- Not as fast as Feasible Successor
- Requires co-operative processing with peers: active, query, reply.



```
RtrA#show ip eigrp topology all-links
IP-EIGRP Topology Table for AS(1)/ID(10.1.6.1)
…..snip…..
P 10.200.1.0/24, 1 successors, FD is 21026560
         via 10.1.1.2 (21026560/20514560), Serial1/0
         via 10.1.2.2 (46740736/20514560), Serial1/1
         via 10.1.3.2 (46740736/46228736), Serial1/2
```

Feasible distance

Successor

Feasible successor

Possible successor

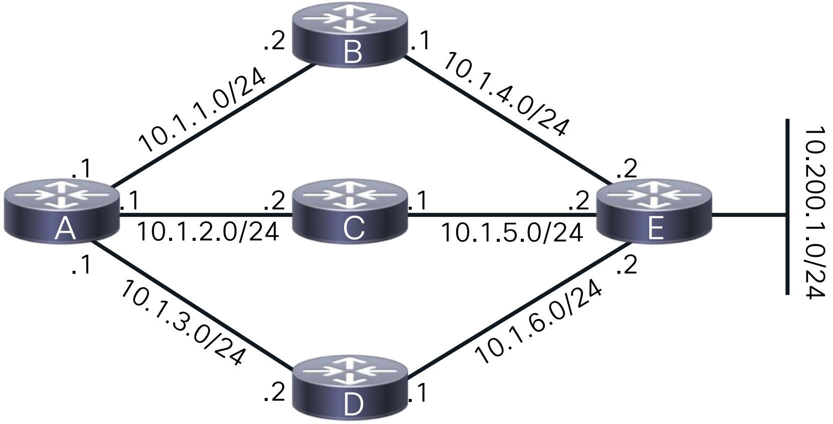
Computed distance

Reported distance

RD (46228736) is not < FD (21026560); NO FS

Convergence of a Non-Feasible Successor

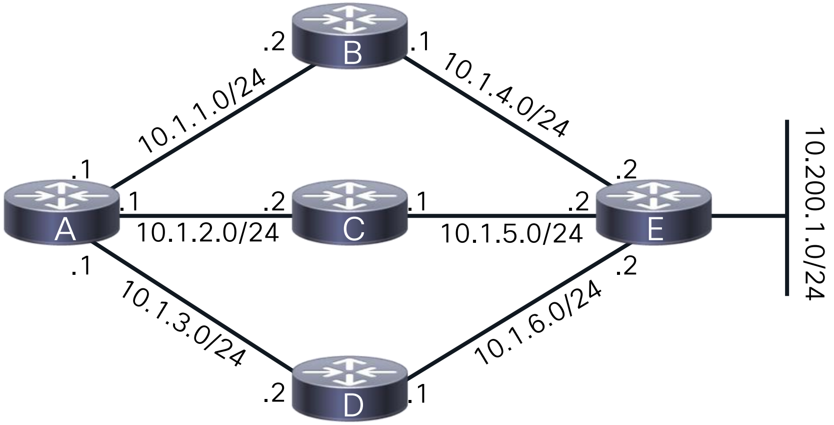
- Show ip eigrp event
- Look for FC not sat, transition to Active



```
RtrA#show ip eigrp event
169  12:04:09.627 State change: Local origin Successor Origin
170  12:04:09.627 Metric set: 10.1.4.0/24 metric(Infinity)
171  12:04:09.627 Active net/peers: 10.1.4.0/24 2
172  12:04:09.627 FC not sat Dmin/met: metric(47360) metric(20480)
173  12:04:09.627 Find FS: 10.1.4.0/24 metric(20480)
174  12:04:09.627 Rcv query met/succ met: metric(Infinity) metric(Infinity)
175  12:04:09.627 Rcv query dest/nh: 10.1.4.0/24 10.1.1.2
```

Convergence of a Non-Feasible Successor

- Show ip eigrp event, cont’d.
- Look for Rcv reply, and Route installed

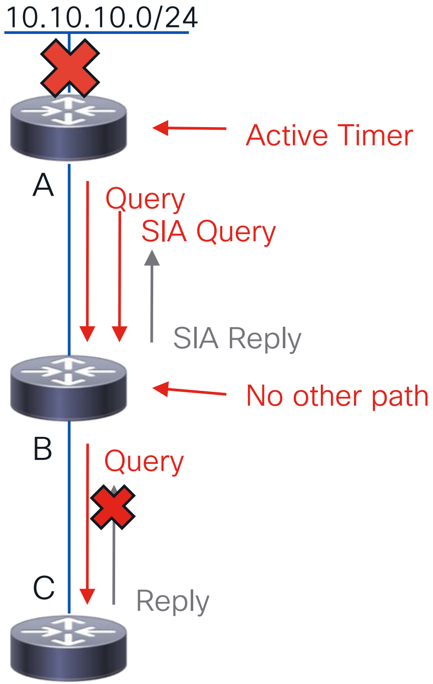


```
RtrA#show ip eigrp event
…
110  12:04:09.659 Route installing: 10.1.4.0/24 10.1.2.2
111  12:04:09.659 Route installed: 10.1.4.0/24 10.1.3.2
112  12:04:09.659 Route installing: 10.1.4.0/24 10.1.3.2
113  12:04:09.659 Route installing: 10.1.4.0/24 10.1.1.2
114  12:04:09.659 Send reply: 10.1.4.0/24 10.1.1.2
115  12:04:09.659 Find FS: 10.1.4.0/24 metric(Infinity)
116  12:04:09.659 Free reply status: 10.1.4.0/24
117  12:04:09.659 Clr handle num/bits: 2 0x0
118  12:04:09.659 Clr handle dest/cnt: 10.1.4.0/24 0
119  12:04:09.659 Rcv reply met/succ met: metric(48640) metric(22784)
120  12:04:09.659 Rcv reply dest/nh: 10.1.4.0/24 10.1.2.2
```

Active Process – When things don’t go as planned

```
%DUAL-3-SIA: Route 10.1.1.0 255.255.255.0 stuck-in-active state in IP-EIGRP 100. Cleaning up
```

- If you reach this point, at least two problems have occurred:
  - A route went active – there’s a network that went missing somewhere
  - It got stuck ☹
- Both the ‘stuck’ and the ‘active’ have already occurred prior to the message being logged!
  - Event-logs are your friend



The SIA Query Process

- SIA-queries are sent to a neighbor up to three times
  - May attempt to get a reply from a neighbor for a total of six minutes
  - If a Reply is not received by the end of this process, the route is considered stuck through this neighbor
- On the router that doesn't get a reply after three SIA-queries
  - Reinitializes neighbor(s) who didn't answer
  - Goes active on all routes known through bounced neighbor(s)
  - Re-advertises to bounced neighbor all routes that were previously advertised

The SIA Query Process

- Sometimes the active process doesn't complete normally; this can be due to a number of different problems which are covered later in this presentation… what happens when things go wrong?
- If RtrB doesn't respond to RtrA within 1.5 minutes because it's still waiting for a Reply from RtrC, RtrA will send an SIA-query to RtrB checking the status—if RtrB is still waiting for a Reply itself, it will respond to RtrA with an SIA-reply; this resets the SIA timer on RtrA so it will wait another 1.5 minutes
- Eventually, the problem keeping RtrC from responding to RtrB will take the neighbor relationship down between RtrB and RtrC, which will cause RtrB to reply to A, ending the Query process

Troubleshooting SIAs

- Two (probably) unrelated causes of the problem – stuck and active
- Need to troubleshoot both parts
  - Cause of active often easier to find
  - Cause of stuck more important to find

Troubleshooting SIAs

- If routes never went active in the network, we would never have to worry about any getting stuck; unfortunately, in a real network there are often link failures and other situations that will cause routes to go active—one of our jobs is to minimize them, however;
- If there are routes that regularly go active in the network, you should absolutely try to understand why they are not stable; while you cannot ensure that routes will never go active on the network, a network manager should work to minimize the number of routes going active by finding and resolving the causes
- Even if you reduce the number of routes going active to the minimum possible, if you don't eliminate the reasons that they get stuck you haven't fixed the most important part of the problem; the next time you get an active route, you could again get stuck
- The direct impact of an active route is small; the possible impact of a stuck-in-active route can be far greater

Troubleshooting the Active Part of SIAs

- Determine what is common to routes going active
  - Known network problems?
  - Flapping link(s)?
  - From the same region of the network?
- Resolve whatever is causing them to go active (if possible)

Dealing with SIA Conditions

```
raven(config-router-af)#?
Address Family configuration commands:
  af-interface         Enter Address Family interface configuration
  default              Set a command to its defaults
  eigrp                EIGRP Address Family specific commands
  …
  soft-sia             Enable graceful restart for stuck-in-active neighbors
  timers               Adjust peering based timers
  topology             Topology configuration mode

raven(config-router-af)#eigrp ?
  default-route-tag      Default Route Tag for the Internal Routes
  graceful-restart       Peer resync without adjacency reset
  kill-everyone          Kill all adjacencies on SIA
  log-neighbor-changes   Enable/Disable EIGRP neighbor logging
  log-neighbor-warnings  Enable/Disable EIGRP neighbor warnings
  router-id              router id for this EIGRP process
  stub                   Set address-family in stubbed mode
  stub-site              Set address-family in stub-site mode
```

Troubleshooting the Active Part of SIAs

- The syslog may tell you which routes are going active, causing you to get stuck. Since the SIA message reports the route that was stuck, it seems rather straight forward to determine which routes are going active. This is only partially true—once SIAs are occurring in the network, many routes will go active due to the reaction to the SIA; you need to determine which routes went active early in the process in order to determine the trigger
- Additionally, you can do show ip eigrp topology active on the network when SIAs are not occurring and see if you regularly catch the same set of routes going active
- If you are able to determine which routes are regularly going active, determine what is common to those routes—are links flapping (bouncing up and down) causing the routes (and everything behind it) to regularly go active?
- Are most or all of the routes coming from the same area of the network? If so, you need to determine what is common in the topology to them so that you can determine why they are not stable

Troubleshooting the Stuck Part of SIAs

- **Show ip eigrp topology active**
- Useful only while the problem is occurring
- If the problem isn’t occurring at the time, it is very difficult to find the reason the routes are getting stuck

Troubleshooting the Stuck Part of SIAs

- Our best weapon to use to find the cause of routes getting stuck-in-active is the command show ip eigrp topology active; it provides invaluable information about routes that are in transition—examples of the output of this command and how to evaluate it will be in the next several slides
- Unfortunately, this command only shows routes that are currently in transition; it isn't useful after the fact when you are trying to determine what happened earlier—if you aren't chasing it while the problem is occurring, there aren't really any tools that will help you find the cause

Troubleshooting the Stuck Part of SIAs

- It’s not always this easy to find the cause of an SIA
- Sometimes you chase the waiting neighbors in a circle
  - If so, summarize and simplify
- Easier after CSCdp33034 (circa 2000)
  - SIA should happen closer to the location of the cause of the problem
- CSCul80747 – introduces a new ‘soft reset’ for the SIA condition. Graceful Resync of the peer can be enabled by the soft-sia cli command.

Troubleshooting the Stuck Part of SIAs

- Our example of chasing SIA routes was intentionally made very easy in order to demonstrate the tools and techniques—in a real event on a network, there would probably be many more routes active, and many more neighbors replying; this can make chasing the waiting neighbors significantly more challenging
- Usually, you will be able to succeed at tracking the waiting neighbors back to the source of the problem—occasionally, you can't—on highly redundant networks, in particular, you can find yourself chasing neighbors in circles without reaching an endpoint cause of the waiting; if you run into this case, you may need to temporarily reduce the redundancy in order to simplify the network for troubleshooting and convergence

Likely Causes for Stuck-in-Active

- Bad or congested links
- Query range is “too long”
- Excessive redundancy
- Overloaded router (high CPU)
- Router memory shortage
- Software defects (seldom)

Likely Causes for SIAs

- Remember that the cause of the SIA route could be a different location than where the SIA message and bounced neighbors happened; this is particularly true with code older than CSCdp33034
- Some of the possible causes of SIAs are:
  - Links that are either experiencing high CRC or other physical errors or are congested to the point of dropping a significant number of frames—queries, replies, or acknowledgements could be lost
  - The time it takes for a query to go from one end of the network to the other is too long and the active timer expires before the query process completes; I don't think I've ever seen a network where this is true, by the way
  - The complexity in the network is so great due to excessive redundancy that EIGRP is required to work so hard at sending and replying to queries that it cannot complete them in time
  - A router is low on memory so that it is able to send hellos, which are very small, but be unable to send queries or replies
- There have occasionally been software defects that caused SIAs (CSCdi83660, CSCdv85419, CSCtc31545)

Event Log

Event Log

- EIGRP keeps a log of common events for each AS, 500 lines by default, rotating.
- 500 lines are not very much; on a network where there is significant instability or activity, 500 lines may only be a second or two (or less) – you can change the size of the event log (if needed) by the command
  - eigrp event-log-size <number of lines>
    - IOS limits to half of available memory
  - If number of lines set to 0, it disables the log
- You can clear the event log by typing
  - clear ip eigrp event
- Most recent events are at the top of the log by default, so time flows from bottom to top. The <reverse> keyword lets you display it from oldest to newest.

Event Log

- Three different event types can be logged
  - EIGRP log-event-type [dual][xmit][transport]
- Default is DUAL—normally most useful
  - DUAL is the EIGRP FSM (decisions in finite state machine)
  - xmit and transport are different aspects of actually sending packets to peers
- Any combination of the three can be on at the same time
- Work is in progress to add additional debug information to event log

EIGRP Event Log

- The two primary weapons at your disposal are debugs and the event log; realize that the output of both debugs and the event log are cryptic and probably not tremendously useful to you (so why am I telling you about them?)
- There are times when the output of debugs or the event log is enough to lead you in a direction, even if you don't really understand all that it is telling you; don't expect to be an expert at EIGRP through the use of debugs or the event log, but they can help
- Don't forget, debugs can kill your router—don't do a debug if you don't know how heavy the overhead is; I may tell you below about some debugs, but don't consider this approval from Cisco to run them on your production network
- The event log is non-disruptive, so it is much safer; just display it and see what's been happening lately

Event Log

- New parameters available for showing the event log

```
Rtr2#show ip eigrp event ?
  <1-4294967295>  Starting event number
  errmsg          Show Events being logged
  reverse         Show most recent event last
  sia             Show Events being logged
  type            Show Events being logged
  |               Output modifiers
  <cr>
```

Event Log

```
RtrA#show ip eigrp events (reverse)
Event information for AS 1:
1    01:52:51.223 NDB delete: 30.1.1.0/24 1
2    01:52:51.223 RDB delete: 30.1.1.0/24 10.1.3.2
3    01:52:51.191 Metric set: 30.1.1.0/24 4294967295
4    01:52:51.191 Poison squashed: 30.1.1.0/24 lost if
5    01:52:51.191 Poison squashed: 30.1.1.0/24 metric chg
6    01:52:51.191 Send reply: 30.1.1.0/24 10.1.3.2
7    01:52:51.187 Not active net/1=SH: 30.1.1.0/24 1
8    01:52:51.187 FC not sat Dmin/met: 4294967295 46738176
9    01:52:51.187 Find FS: 30.1.1.0/24 46738176
10   01:52:51.187 Rcv query met/succ met: 4294967295 4294967295
11   01:52:51.187 Rcv query dest/nh: 30.1.1.0/24 10.1.3.2
12   01:52:36.771 Change queue emptied, entries: 1
13   01:52:36.771 Metric set: 30.1.1.0/24 46738176
```

Suspect #3 – Computations: Takeaway

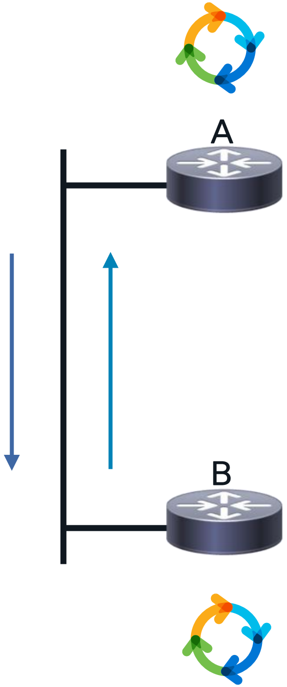
- Check your Topology table
  - Understand if you have a Feasible Successor and how to read
- Are you Active?
  - How far will Queries need to go?
  - Know how to bound the Query domain. (STUB, Summary)
- Know how to utilize the Event Log
  - Valuable information about what your suspects were up to when you weren’t looking. Forensic analysis of the crime scene!

Review

Troubleshooting Methodology

TIP

- Knowledge of the System
  - Expected Behaviors / Baseline
  - Relationships
  - Where to look for information!
- Logical Sequence of Events
- EIGRP:
  - Peers Form
  - Routes Exchanged
  - Path Computation (DUAL)
  - Routing Table Updated (if necessary)
  - Peers Updated (if necessary)



Troubleshooting Methodology

Where to Look for Information – Cheat Sheet

- Peer Issues
  - Show ip eigrp neighbor
  - Show log (with log-neighbor-changes enabled)
  - Show run | s router eigrp
- Packet Issues
  - Show ip eigrp interface
  - Show ip eigrp neighbor
  - Check Unicast, Multicast delivery
  - Debug ip eigrp packet (with care)
- Path Computation and Propagation Issues
  - Show ip eigrp event
  - Show ip eigrp topology
- Info for TAC
  - Show eigrp plugin detail
  - Show eigrp tech detail
- CiscoLive Online Library!

Cornered: The Usual Suspects

Who Are They?

- Neighbors
- Packets
- Computations

What Are Their Motives?

- Establish Communication
- Exchange Information
- Make Decisions

What Is Your Strategy?

- Neighbor Table, Logs
- Topology, Event Log
- DUAL

Technical session surveys

- Attendees who fill out a minimum of four session surveys and the overall event survey will get Cisco Live branded socks!
- Attendees will also earn 100 points in the Cisco Live Game for every survey completed.
- These points help you get on the leaderboard and increase your chances of winning daily and grand prizes.

Cisco learning and certifications

From technology training and team development to Cisco certifications and learning plans, let us help you empower your business and career. www.cisco.com/go/certs

Pay for Learning with Cisco Learning Credits

(CLCs) are prepaid training vouchers redeemed directly with Cisco.

Learn

Cisco U.
IT learning hub that guides teams and learners toward their goals

Cisco Digital Learning
Subscription-based product, technology, and certification training

Cisco Modeling Labs
Network simulation platform for design, testing, and troubleshooting

Cisco Learning Network
Resource community portal for certifications and learning

Train

Cisco Training Bootcamps
Intensive team & individual automation and technology training programs

Cisco Learning Partner Program
Authorized training partners supporting Cisco technology and career certifications

Cisco Instructor-led and Virtual Instructor-led training
Accelerated curriculum of product, technology, and certification courses

Certify

Cisco Certifications and Specialist Certifications
Award-winning certification program empowers students and IT Professionals to advance their technical careers

Cisco Guided Study Groups
180-day certification prep program with learning and support

Cisco Continuing Education Program
Recertification training options for Cisco certified individuals

Here at the event? Visit us at **The Learning and Certifications lounge at the World of Solutions**

Continue your education

- Visit the Cisco Showcase for related demos
- Book your one-on-one Meet the Engineer meeting
- Attend the interactive education with DevNet, Capture the Flag, and Walk-in Labs
- Visit the On-Demand Library for more sessions at www.CiscoLive.com/on-demand

Thank you

CISCO Live!

#CiscoLive