Chapter 4

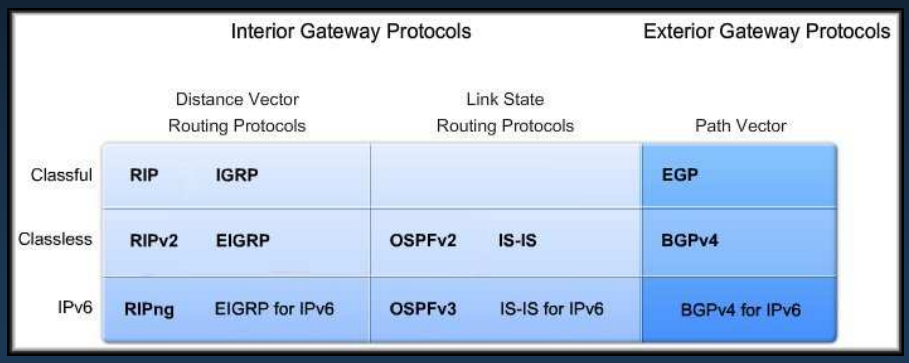
Distance Vector Routing Protocols

Note for Instructors

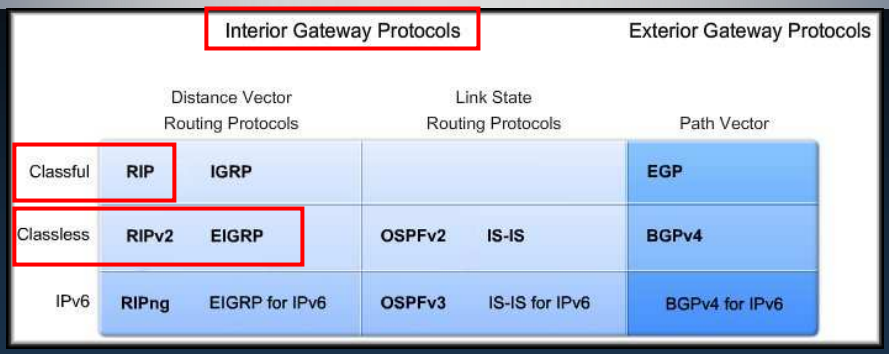
- These presentations are the result of a collaboration among the instructors at St. Clair College in Windsor, Ontario.
- Thanks must go out to Rick Graziani of Cabrillo College. His material and additional information was used as a reference in their creation.
- If anyone finds any errors or omissions, please let me know at:
  - tdame@stclaircollege.ca.

Distance Vector Routing Protocols

Introduction to Distance Vector Routing Protocols



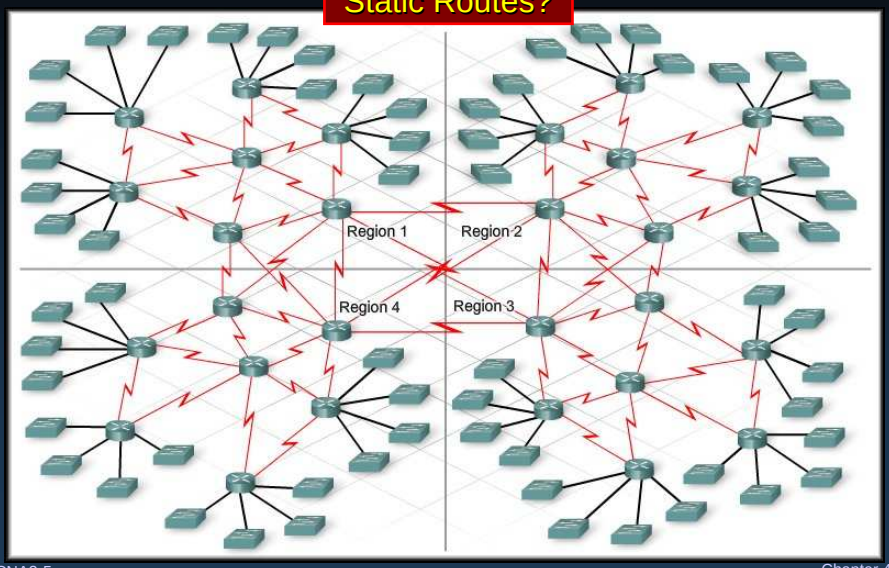
Introduction to Distance Vector



- There are advantages and disadvantages to using any type of routing protocol.
- Understanding the operation of distance vector routing is critical to enabling, verifying, and troubleshooting these protocols.

Introduction to Distance Vector

Static Routes?



Introduction to Distance Vector

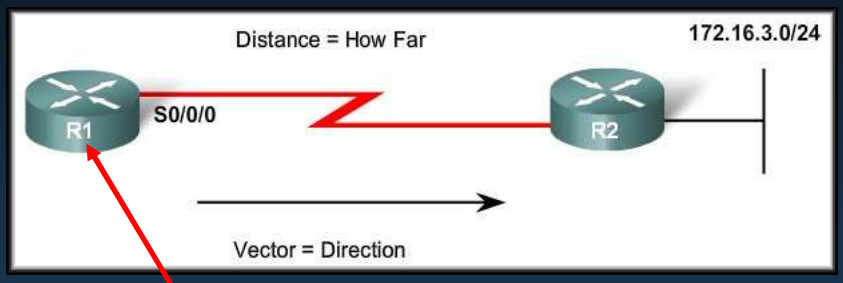
- Routing Information Protocol (RIP):
  - Metric: Hop count.
  - A hop count greater than 15 means that the network is unreachable.
  - Periodic routing updates.
    - Entire routing table is broadcast every 30 seconds.
- Enhanced Interior Gateway Routing Protocol (EIGRP):
  - Cisco proprietary.
  - Composite metric: Bandwidth, delay, reliability and load.
  - It uses Diffusing Update Algorithm (DUAL) to calculate the shortest path.
  - No periodic updates.
    - Multicast updates only on a change in topology.

Meaning of Distance Vector

- The routing protocol does not know the entire topology of a network.
- It only knows the routing information received from its neighbors.
  - *A Distance Vector routing protocol does not have the knowledge of the entire path to a destination network.*

Meaning of Distance Vector

- *A Distance Vector routing protocol does not have the knowledge of the entire path to a destination network.*



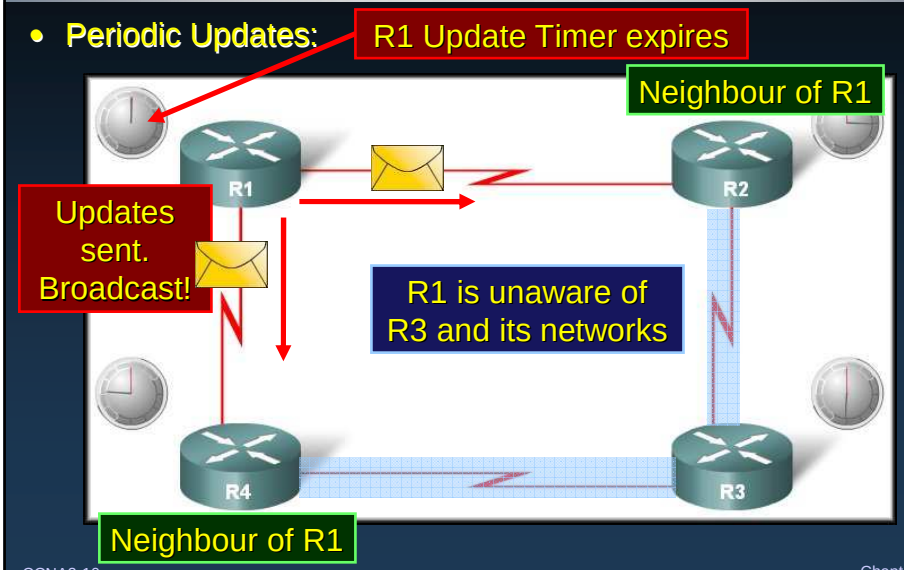
Network 172.16.3.0/24:
- is 1 hop away (Distance)
- through interface s0/0/0 (Vector)

Operation of Distance Vector

- **Periodic Updates:**
  - Some distance vector routing protocols periodically broadcast the entire routing table to each of its neighbors (RIP – every 30 seconds).
    - **Inefficient:** Updates consume bandwidth and router CPU resources.
    - Periodic updates are **always sent** even there have been no changes for weeks or months.
  - Router is only aware of the:
    - Network addresses of its **own interfaces**.
    - Network addresses the **neighbors running the same routing protocol**.

Operation of Distance Vector

- **Periodic Updates:**



Routing Protocol Algorithms

- The algorithm used by a particular routing protocol is responsible for building and maintaining the router's routing table.

Step 0 : Initialize
$d(s) := 0$; $d(v) := +\infty\ \forall v \in V \setminus \{s\}$; $\pi(v) := v\ \forall v \in V$; $Q := V$; $i := 1$
Step 1 : Select the node
If Q=0, then go to step 3, else select the node v from the head of Q
Step 2 :Search the Path (let v be the initial point)
If d(u) > d(v)+l((v, u)) for all path(v,u), then d(u) = d(v) + l((v, u)), $\pi(u) = v$
→ Step 1
Step 3:judgement
i ← i +1
If i < n, then Q ← V and go to step 1,
else check whether triangle inequality* is satisfied or not on all paths.
If any paths "A" not satisfied the triangle inequality, there is the negatively circuit including the path "A".

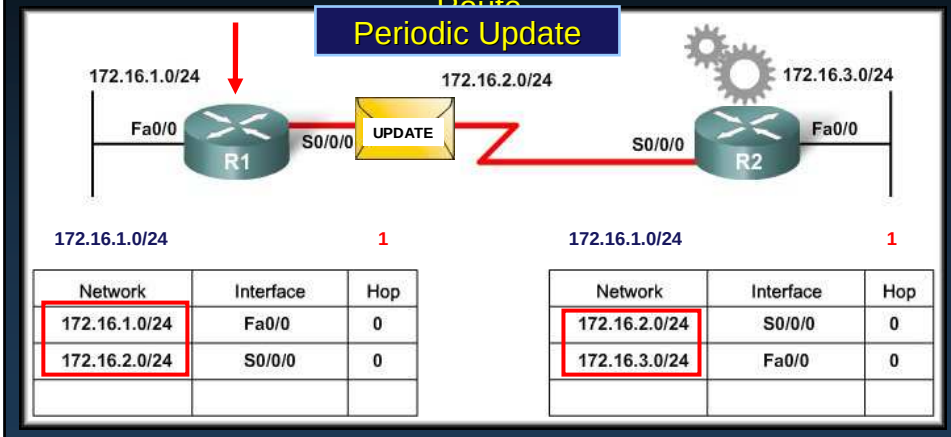
* Triangle inequality
Let X be linear space,
$\| u+v \| \leq \| u \| + \| v \|$ for u,v∈X

- Defines the following processes:
  - Mechanism for sending and receiving routing information.
  - Mechanism for calculating the best paths and installing routes in the routing table.
  - Mechanism for detecting and reacting to topology changes.

Routing Protocol Algorithms

Calculate Best Path and Install Route

Periodic Update



172.16.1.0/24 1

Network	Interface	Hop
172.16.1.0/24	Fa0/0	0
172.16.2.0/24	S0/0/0	0

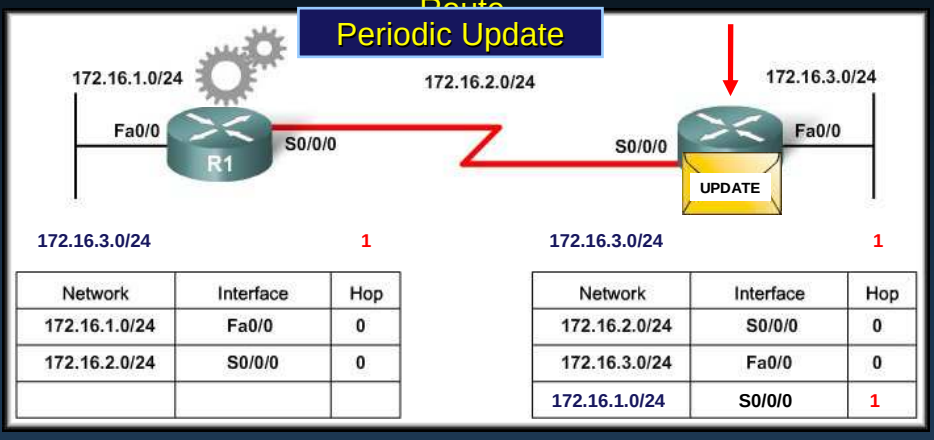
172.16.1.0/24 1

Network	Interface	Hop
172.16.2.0/24	S0/0/0	0
172.16.3.0/24	Fa0/0	0

Routing Protocol Algorithms

Calculate Best Path and Install Route

Periodic Update



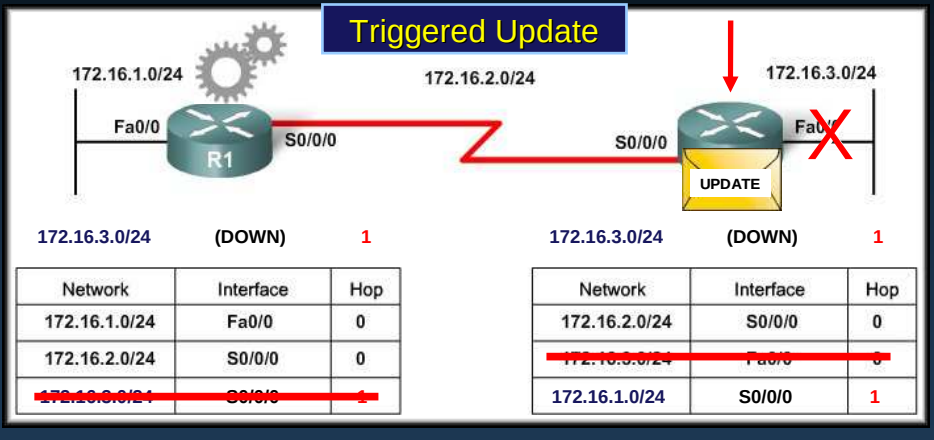
Network	Interface	Hop
172.16.1.0/24	Fa0/0	0
172.16.2.0/24	S0/0/0	0

Network	Interface	Hop
172.16.2.0/24	S0/0/0	0
172.16.3.0/24	Fa0/0	0
172.16.1.0/24	S0/0/0	1

Routing Protocol Algorithms

Detect and React to Topology Changes

Triggered Update



Network	Interface	Hop
172.16.1.0/24	Fa0/0	0
172.16.2.0/24	S0/0/0	0
~~172.16.3.0/24~~	~~S0/0/0~~	~~1~~

Network	Interface	Hop
172.16.2.0/24	S0/0/0	0
~~172.16.3.0/24~~	~~Fa0/0~~	~~0~~
172.16.1.0/24	S0/0/0	1

Routing Protocol Characteristics

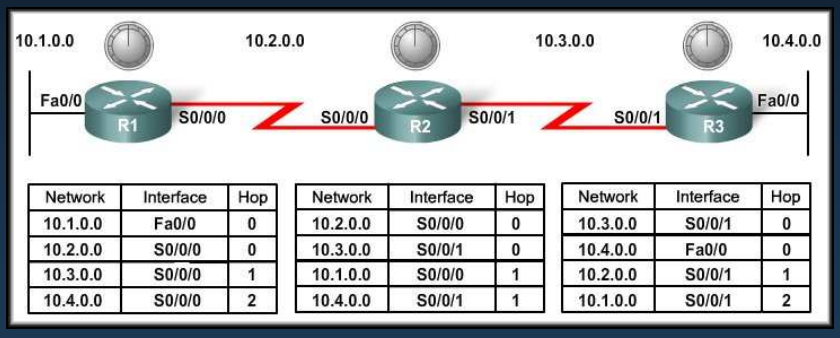
- Other ways to compare routing protocols:
  - Time to convergence:
    - Faster the better.
  - Scalability:
    - How large a network the routing protocol can handle.
  - Classless or Classful:
    - Support VLSM and CIDR.
  - Resource usage:
    - Routing protocol usage of RAM, CPU utilization, and link bandwidth utilization.
  - Implementation and maintenance:
    - Level of knowledge of a network administrator.

Comparing Routing Protocol Features

Distance Vector Routing Protocols			
Feature	RIPv1	RIPv2	EIGRP
Speed of Convergence	Slow	Slow	Fast
Scalability	Small	Small	Large
Supports VLSM	No	Yes	Yes
Resource Usage	Low	Low	Medium
Implementation	Simple	Simple	Complex

Distance Vector Routing Protocols

Network Discovery



Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

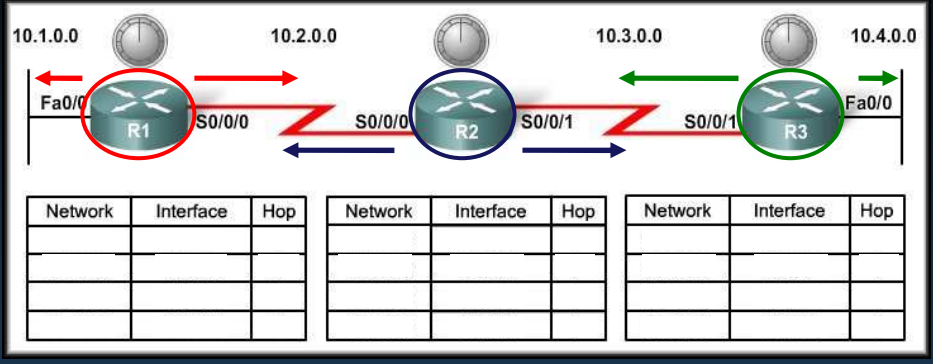
Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

Cold Start

- Network Discovery:
  - Is part of the process of the routing protocol algorithm that enables routers to learn about remote networks for the first time.

Cold Start



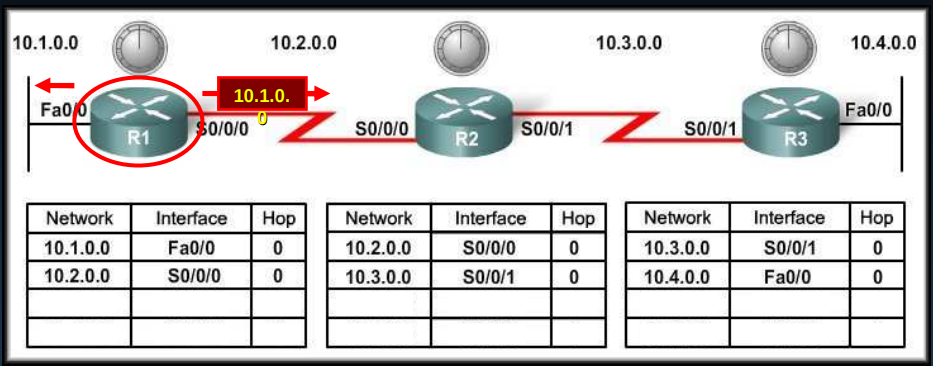
Network	Interface	Hop

Network	Interface	Hop

Network	Interface	Hop

- When a router powers up:
  - Knows nothing about the network topology.
  - Knows only the information saved in NVRAM.
  - Sends updates about its known networks out all ports.

Initial Exchange of Routing Information



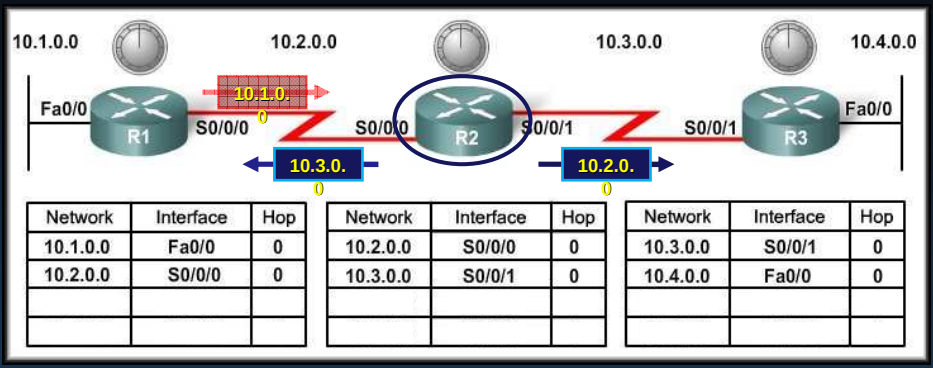
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0

- Sends an update about network 10.1.0.0 out the Serial 0/0/0 interface with a metric of 1.
- Sends an update about network 10.2.0.0 out the Fa0/0 interface with a metric of 1.

Initial Exchange of Routing Information



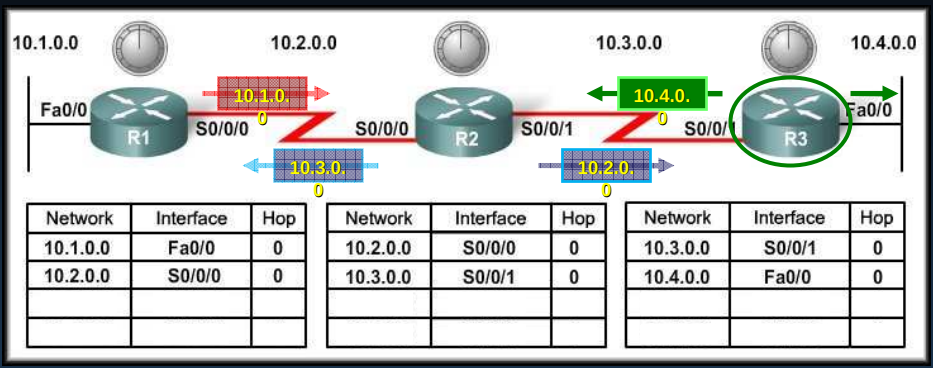
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0

- Sends an update about network 10.3.0.0 out the Serial 0/0/0 interface with a metric of 1.
- Sends an update about network 10.2.0.0 out the Serial 0/0/1 interface with a metric of 1.

Initial Exchange of Routing Information



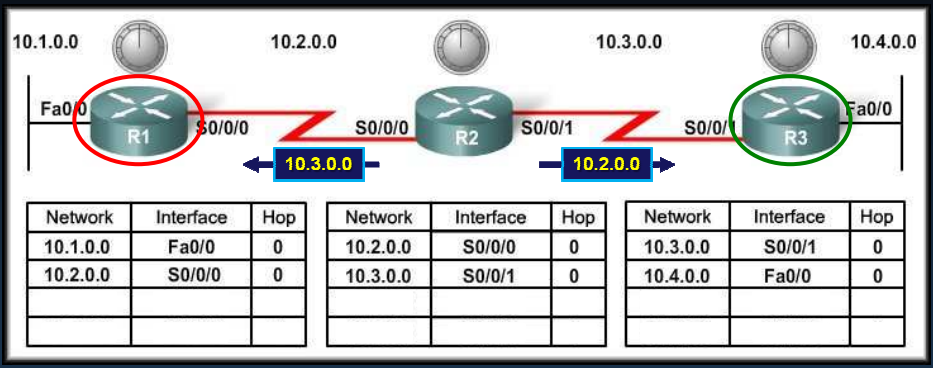
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0

- Sends an update about network 10.4.0.0 out the S0/0/0 interface with a metric of 1.
- Sends an update about network 10.3.0.0 out the Fa0/0 interface with a metric of 1.

Initial Exchange of Routing Information



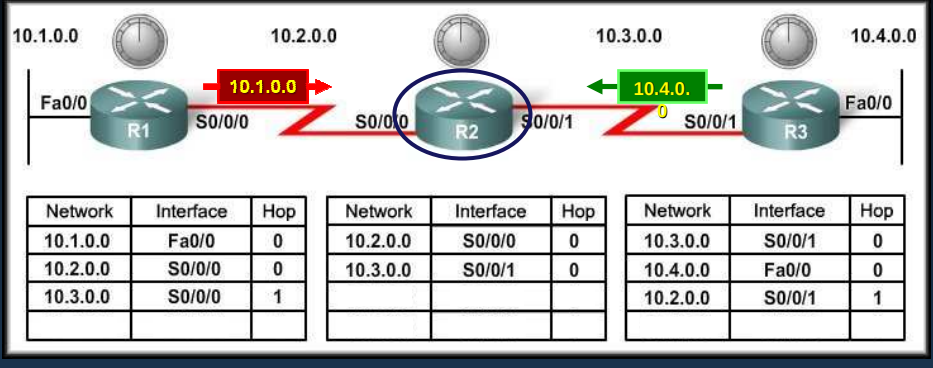
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0

- R1 Receives the update from R2 about network 10.3.0.0 and adds it to its routing table.
- R3 Receives the update from R2 about network 10.2.0.0 and adds it to its routing table.

Initial Exchange of Routing Information



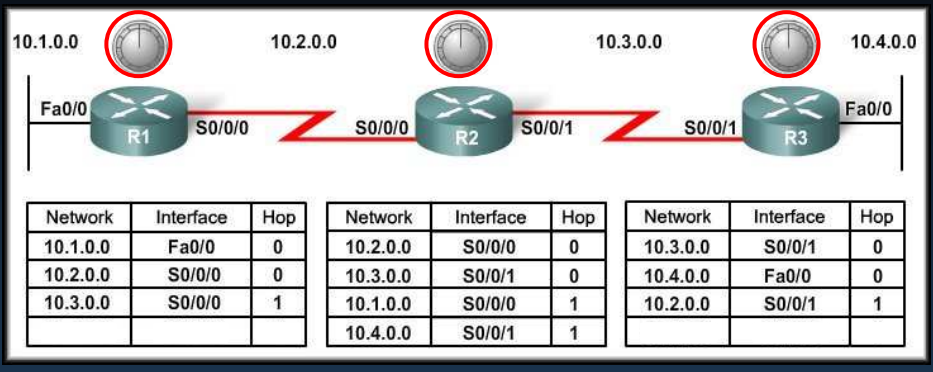
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1

- R2 Receives the update from R1 about network 10.1.0.0 and adds it to its routing table.
- R2 Receives the update from R3 about network 10.4.0.0 and adds it to its routing table.

Initial Exchange of Routing Information



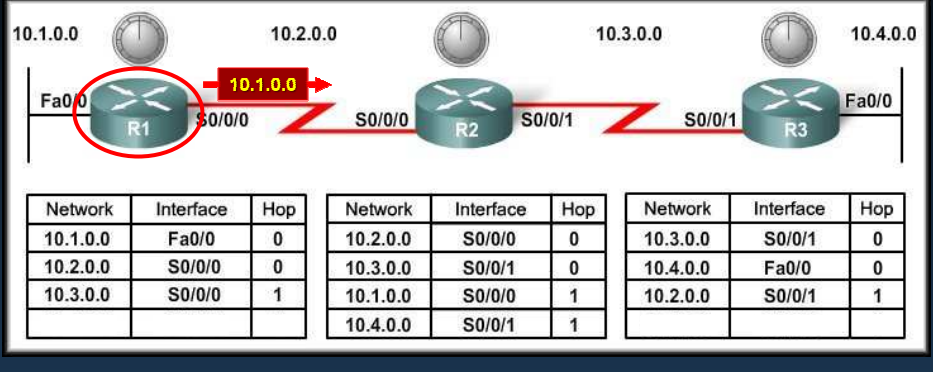
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1

- **First round of update exchanges**: Each router knows about the connected networks of its **directly connected neighbors**.
- When the update timers expire **(Periodic Update)**, the routers begin the next exchange of information.

Next Exchange of Routing Information



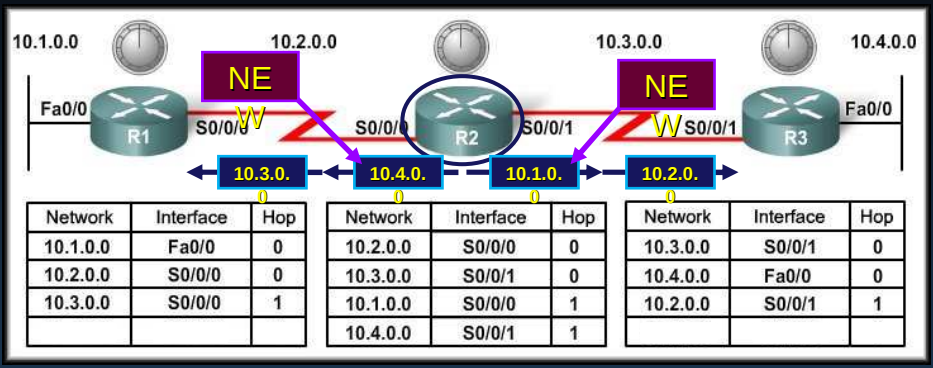
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1

- **Sends** an update about network **10.1.0.0** out the **S0/0/0** interface with a metric of 1 - **AGAIN!**
- ***When R2 receives the update***, there is **no change** in information so the update is ignored.

Next Exchange of Routing Information



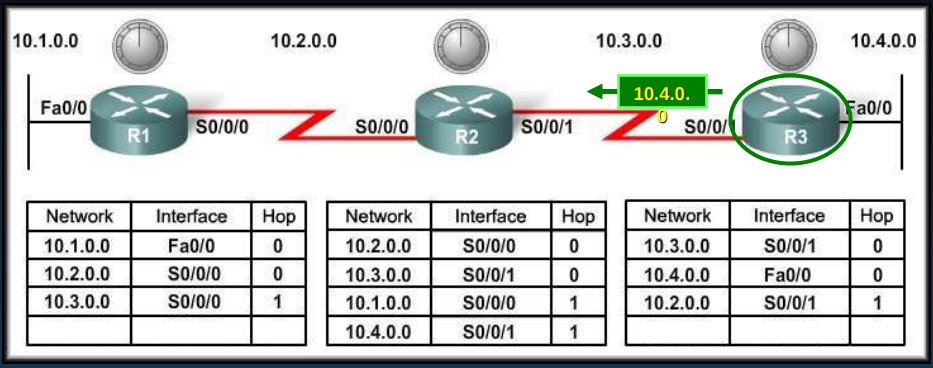
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1

- Sends an update about networks 10.3.0.0 with a metric of 1 and 10.4.0.0 with a metric of 2 out the Serial 0/0/0 interface.
- Sends an update about networks 10.1.0.0 with a metric of 2 and 10.2.0.0 with a metric of 1 out the Serial 0/0/1 interface.

Initial Exchange of Routing Information



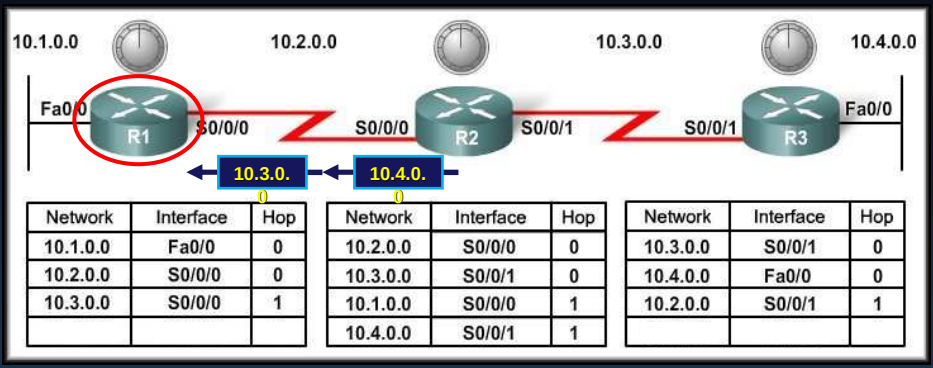
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1

- Sends an update about network 10.4.0.0 out the S0/0/0 interface with a metric of 1 - AGAIN!
- *When R2 receives the update*, there is no change in information so the update is ignored.

Next Exchange of Routing Information



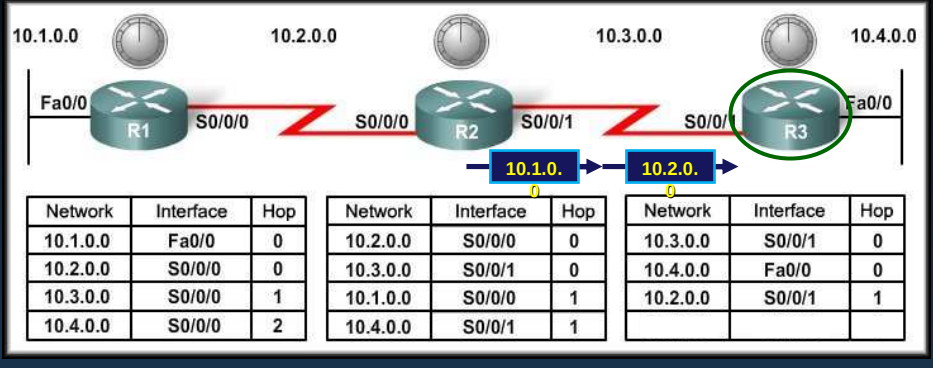
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1

- R1 receives an update from R2 about network 10.3.0.0 and there is no change – update ignored.
- R1 receives an update from R2 about network 10.4.0.0 (new) and adds it to its routing table.

Next Exchange of Routing Information



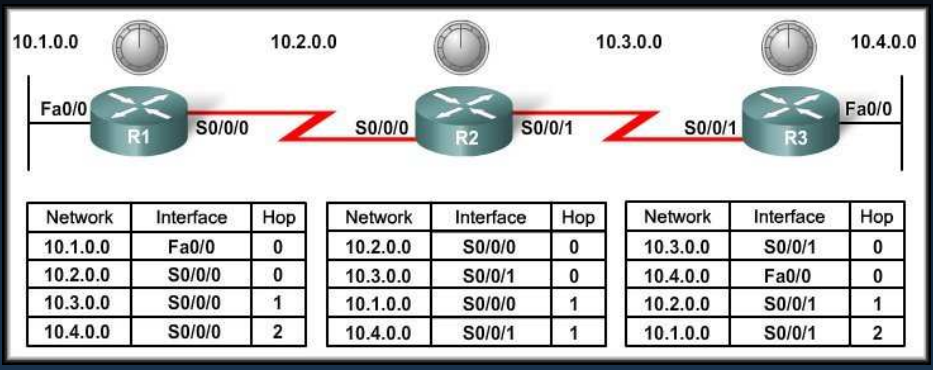
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1

- R3 receives an update from R2 about network 10.2.0.0 and there is no change – update ignored.
- R3 receives an update from R2 about network 10.1.0.0 (new) and adds it to its routing table.

Next Exchange of Routing Information



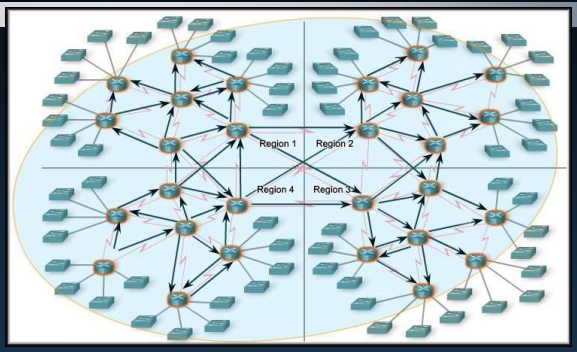
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

- The network has CONVERGED!
  - All routers now know about all of the networks attached to all of their neighbouring routers.

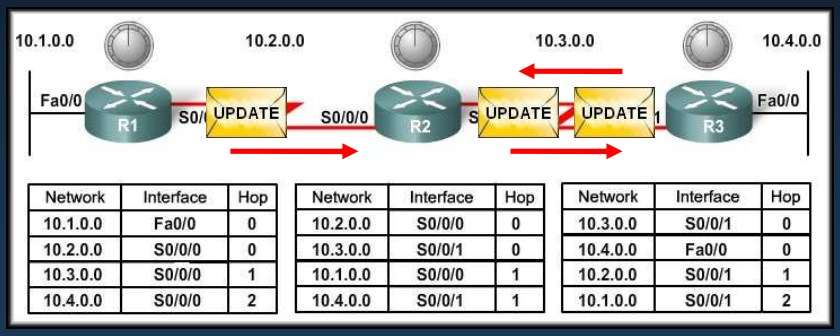
Convergence



- The amount of time it takes for a network to converge is directly proportional to the size of that network.
- Routing protocols are compared based on how fast they can propagate this information - their speed to convergence.
- A network is not completely operable until it has converged.
  - Network administrators prefer routing protocols with shorter convergence times.

Distance Vector Routing Protocols

Routing Table Maintenance

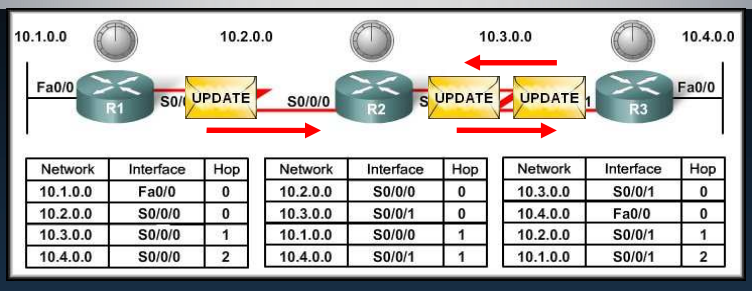


Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

Routing Table Maintenance



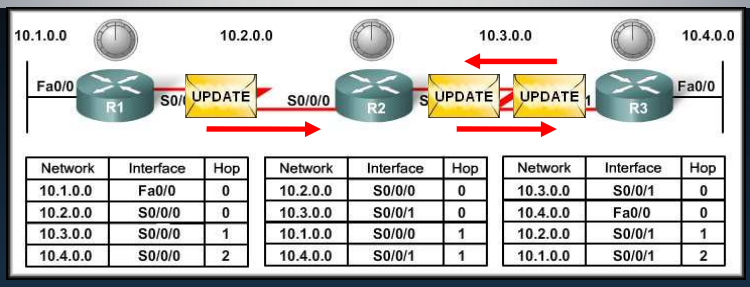
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

- Routing protocols must maintain the routing tables so that they have the most current information.
- How it is maintained depends upon:
  - The type of routing protocol
    (distance vector, link state, path state)
  - The routing protocol itself (RIP, EIGRP, OSPF)

Routing Table Maintenance



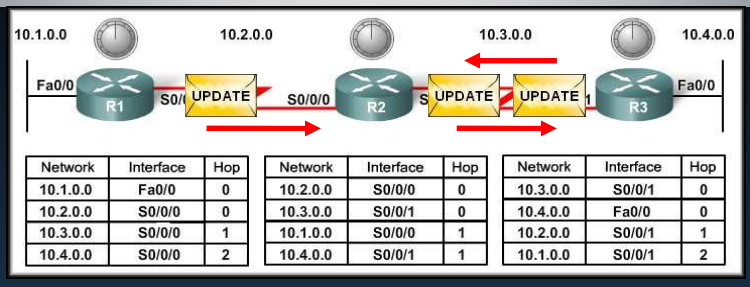
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

- Distance Vector Updates:
  - Periodic
  - Bounded
  - Triggered

Periodic Updates



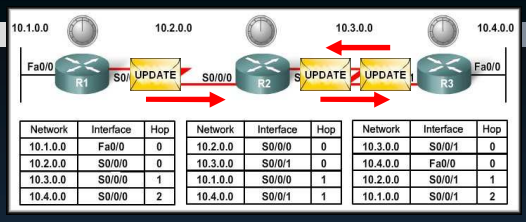
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

- A router sends the complete routing table to its neighbors at a predefined interval.
- RIP – every 30 seconds.
  - Link failure, New Link, Router Failure, Link parameter change.

Periodic Updates - RIP Timers



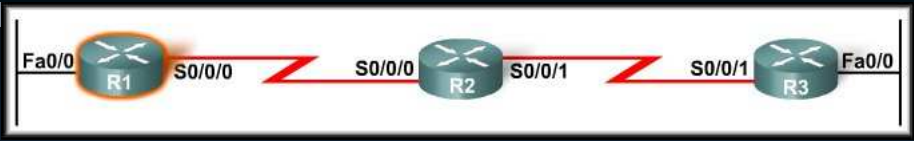
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

- **Invalid Timer:**
  - If an update has not been received in 180 seconds (the default), the route is marked as invalid by setting the metric to 16.
  - Route still is in routing table.
- **Flush Timer:** 240 seconds (default)
  - When the flush timer expires, the route is removed from the routing table.
- **Hold-down Timer:**
  - Helps stabilize routing information and helps prevent routing loops. (Much more later!)

Periodic Updates – Verifying RIP Timers



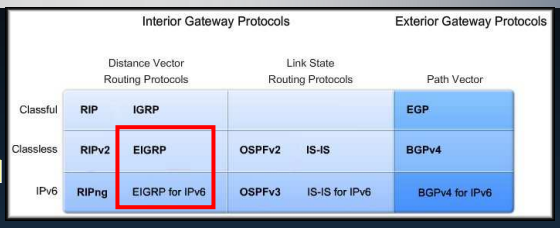

```
R1# show ip route
     10.0.0.0/16 is subnetted, 4 subnets
C       10.2.0.0 is directly connected, Serial0/0/0
R       10.3.0.0 [120/1] via 10.2.0.2, 00:00:04, Serial0/0/0
C       10.1.0.0 is directly connected, FastEthernet0/0
R       10.4.0.0 [120/2] via 10.2.0.2, 00:00:04, Serial0/0/0
```

Elapsed time since last update.

```
R1# show ip protocols
Routing Protocol is "rip"
  Sending updates every 30 seconds, next due in 13 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
<output omitted>
Routing Information Sources:
  Gateway         Distance      Last Update
  10.3.0.1        120           00:00:27
```

Bounded Updates

- *EIGRP does not send periodic updates.*
- EIGRP sends bounded updates about a route when a path changes or the metric for that route changes.
  - Nonperiodic: Because they are not sent out on a regular basis.
  - Partial: Because they are sent only when there is a change in topology.
  - Bounded: Because they are sent to only those routers that need the information.
  - (More in chapter 9)



Triggered Updates

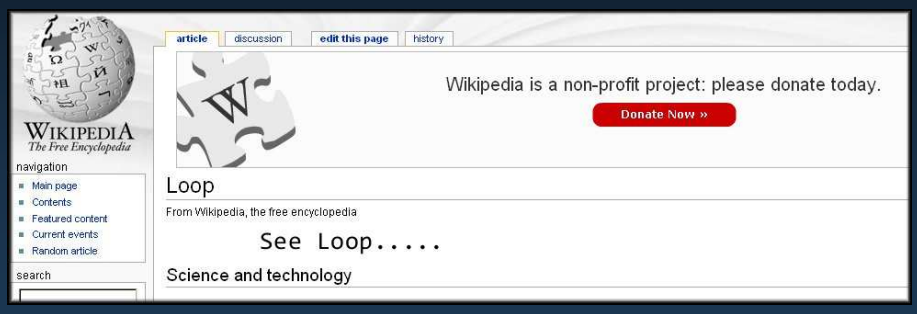
- A triggered update is a routing table update that is sent immediately in response to a routing change.
  - Do not wait for update timers to expire.
  - The detecting router immediately sends an update message to adjacent routers.
  - The receiving routers generate triggered updates that notify their neighbors of the change.

Triggered Updates

- Speeds up convergence.
- Sent when one of the following events occurs:
  - An interface changes state (up or down).
  - A route has entered (or exited) the unreachable state.
  - A route is installed in the routing table.

Distance Vector Routing Protocols

Routing Loops



Defining a Routing Loop

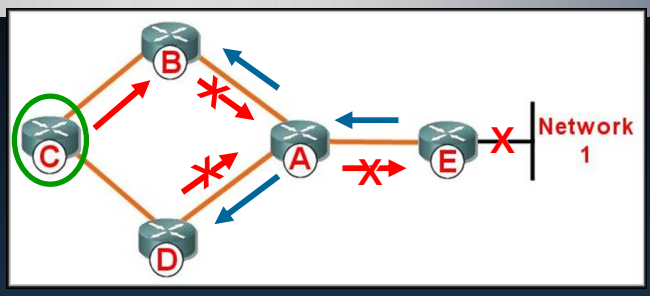
- A routing loop is a condition in which a packet is **continuously transmitted** within a series of routers **without ever reaching its intended destination** network.
- The loop can be a result of:
  - Incorrectly configured static routes.
  - Incorrectly configured route redistribution.
  - Inconsistent routing tables not being updated because of slow convergence in a changing network.
- Distance vector routing protocols are simple in their implementation and configuration, but this comes at a price.
- Pure distance vector routing protocols suffer from possible routing loops.

Implications of Routing Loops

- A routing loop can have a **devastating effect on a network**, resulting in degraded network performance or even network downtime.
  - Link bandwidth will be used for traffic looping back and forth between the routers.
  - A router's CPU will be burdened with useless packet forwarding.
  - Routing updates might get lost or not be processed in a timely manner, making the situation even worse.
  - Packets might get lost in **black holes**, never reaching their intended destinations.

Routing Loop - Example

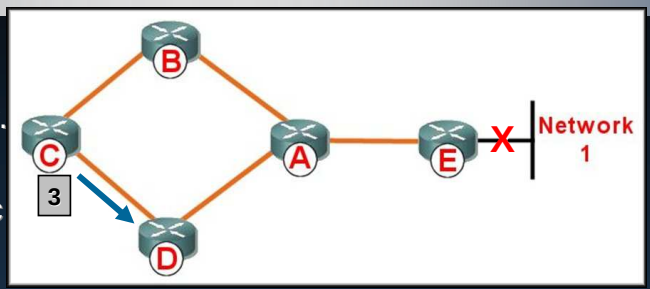
- Network 1 Fails.
- Router E sends an update to Router A.
- Router A stops routing packets to Network 1.
  - But Routers B, C, and D continue to do so because they have not yet been informed about the failure.
  - Router A sends out its update.
  - Routers B and D stop routing to network1, (via Router A).
  - However, Router C is still not updated. To router C, network 1 is still reachable via router B.



Routing Loop - Example

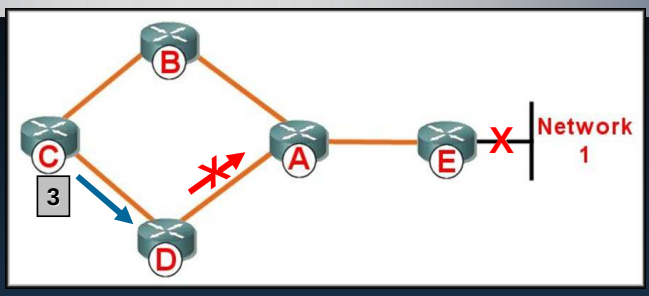
- Router C thinks network 1 is still 3 hops away.
- Sends a periodic update to Router D.
- This update says:

A path to network 1 exists by way of Router B and network 1 is 4 hops away.



Routing Loop - Example

- *Router D routing table information for Network 1.*
- Current path to Network 1 = Unreachable



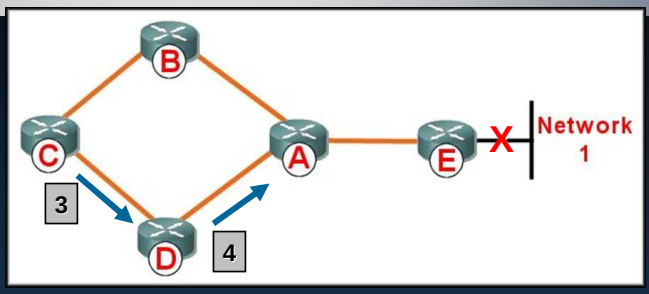
- Update from Router C:

  Network 1 is 4 hops by way of Router C

- Normally, Router D ignores this routing information because it usually has a better route (2 hops via Router A) but this route is now down.

Routing Loop - Example

- Router D changes its routing table to reflect this *better, but incorrect information.*

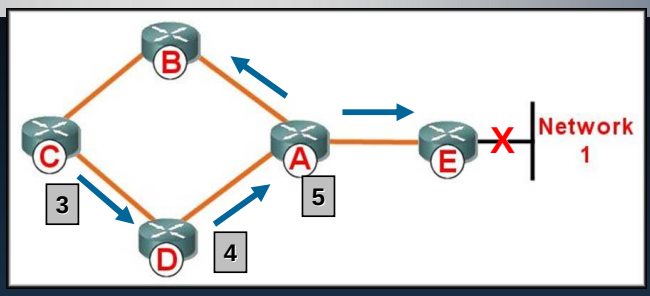


Network 1 is available by way of Router C (4 hops)

- Router D propagates the information to Router A.

Routing Loop - Example

- **Router A** changes its routing table.



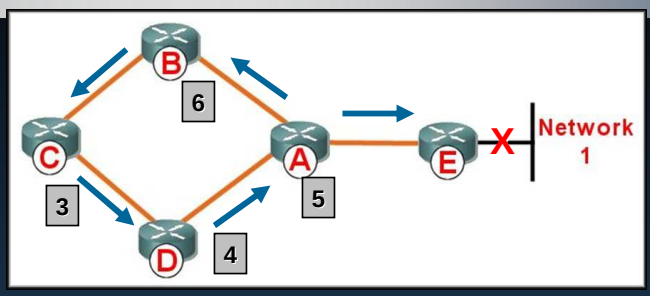
- Router A adds a new route to its routing table:

  Network 1 is available by way of Router D (5 hops).

- Propagates the information to Routers B and E.

Routing Loop - Example

- **Router B and Router E** change their routing tables.

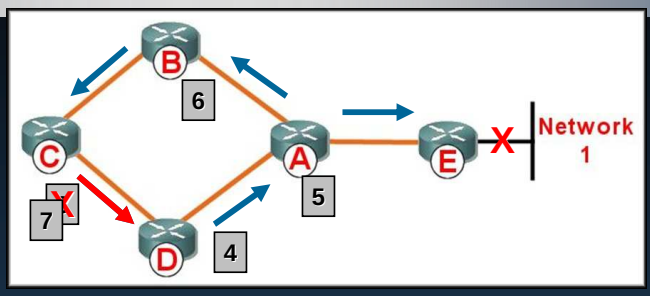


- Router B now believes:
  - *Network 1 is available by way of Router A (6 hops).*

    "Wow! I was about to tell Router C that Network 1 was down, but now I have new information!"
  - Router B sends the incorrect information to Router C.

Routing Loop - Example

- **Router C** changes its routing table.



- Router C still believes:

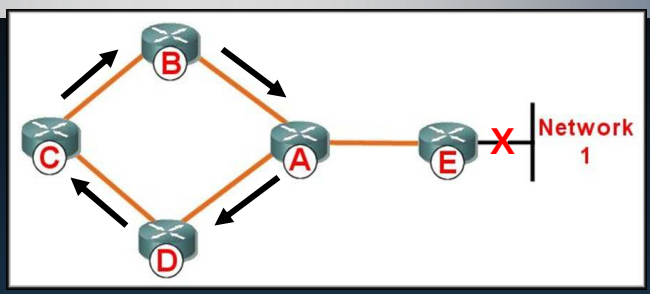
  Network 1 is available by way of Router B

  But now it believes its 7 hops instead of 3!

- Propagates the incorrect information to Router D.

Routing Loop - Example

- Here we go again!
- The routers keep sending data packets and updates!
- ***BUT……***



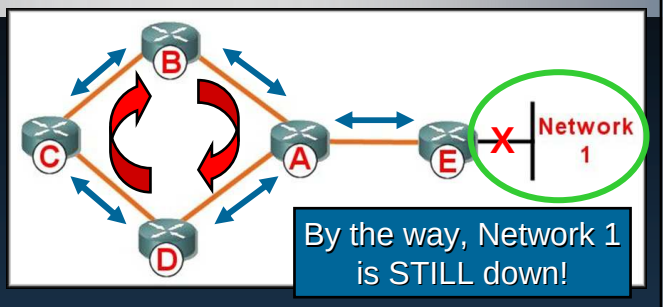
Router A thinks Network 1 is available via Router D.

Router D thinks Network 1 is available via Router C.

Router C thinks Network 1 is available via Router B.

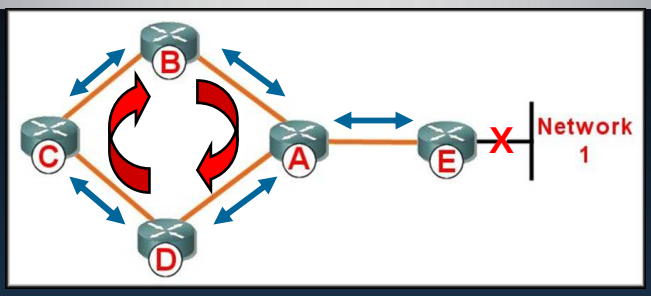
Router B thinks Network 1 is available via Router A.

Routing Loop - Example



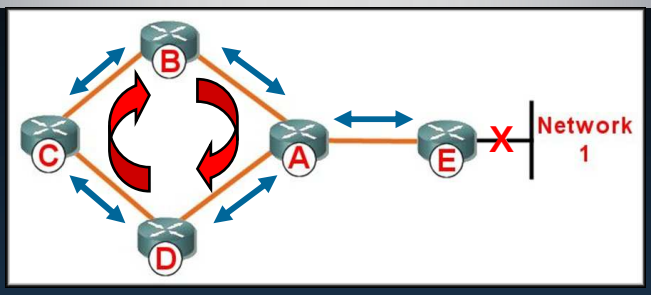
- Data packets destined for Network 1 get caught in a routing loop, from Routers A to D to C to B to A to D etc.
- As routing updates continue between the routers, the hop count gets greater – to infinity? (Not quite – we will see in a moment.)

Count-to-Infinity Condition



- Count to infinity is a condition that exists when inaccurate routing updates increase the metric value to infinity for a network that is no longer reachable.
  - Each protocol defines infinity at a different value.
  - When the metric value exceeds the maximum value, and as each router receives this maximum metric, the network is then considered *unreachable*.

Count-to-Infinity Condition

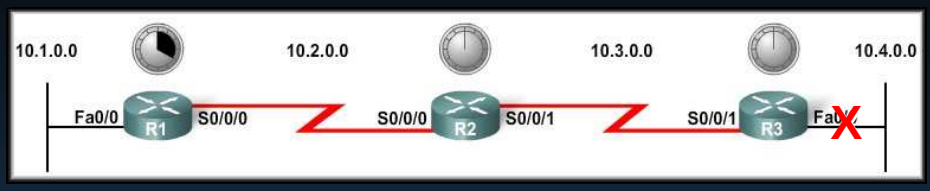


- RIP defines infinity as 16 hops.
  - When the routers "count to infinity," they mark the route as unreachable.

Hold-Down Timers

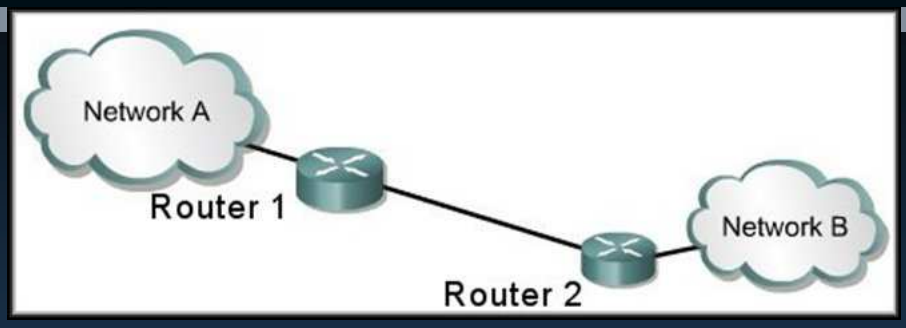
- Hold-Down timers are used to prevent regular update messages from inappropriately reinstating a route that may have gone bad.
  - They instruct routers to hold any changes that might affect routes for a specified period of time.

Hold-Down Timers



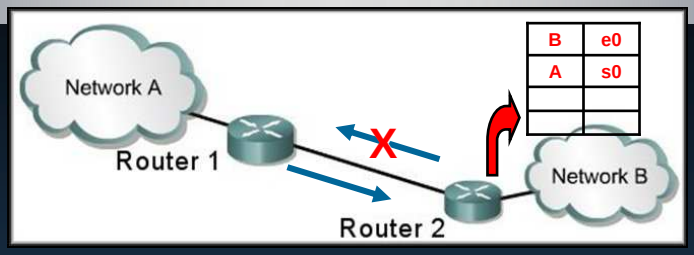
- If a route is identified as down or possibly down, any other information for that route containing the same status, or worse, is ignored for a predetermined amount of time (the hold-down period).
- This means that routers will leave a route marked as unreachable in that state for a period of time that is long enough for updates to propagate the routing tables with the most current information.

Split Horizon Rule



- Split Horizon:
  - Is another method used to prevent routing loops.
  - *The split horizon rule says that a router should not advertise a network through the interface from which the update came.*

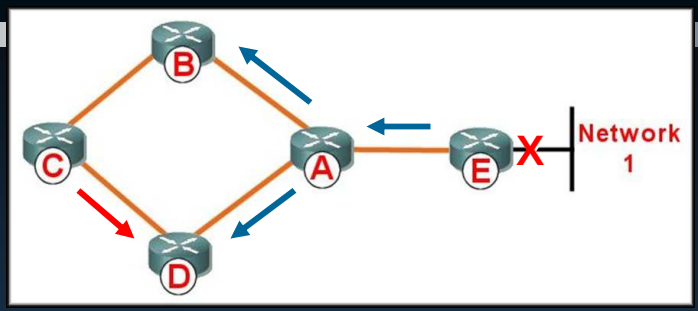
Split Horizon Rule



- Router 1 sends an update to Router 2 that Network A is available and Router 2 updates its routing table with the information.
- Router 2 recognizes a change in topology.
- This would normally trigger an update to neighbouring routers and cause a routing loop.
- With split horizon enabled, Router 2 realizes it received the information from Router 1 and does not send the update.

Split Horizon Rule

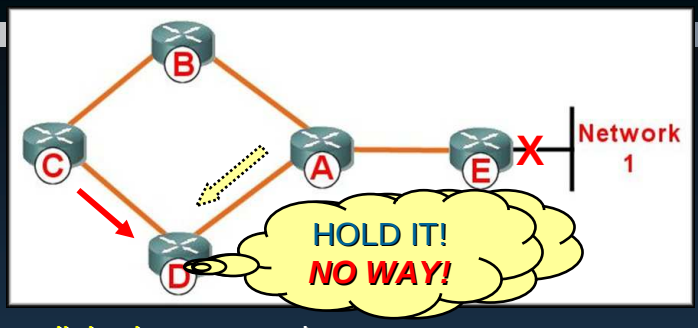
Would Split Horizon avoid the routing loop in our example?



- When Network 1 went down, Router E sent an update to Router A.
- Router A then sent an update to Routers B and D that Network 1 was no longer available.
- Router C then sent an update to Router D that Network 1 IS available.

Split Horizon Rule

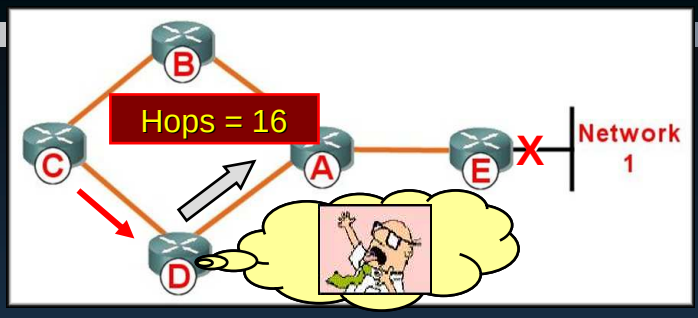
Would Split Horizon avoid the routing loop in our example?



- Here is where *split horizon* comes in……
  - With Split Horizon disabled, Router D would send an update to Router A about the status of Network 1 and set the routing loop in motion.
  - With Split Horizon enabled, Router D *does not send* the update to Router A because it *already received an update about the status of Network 1 from Router A*.

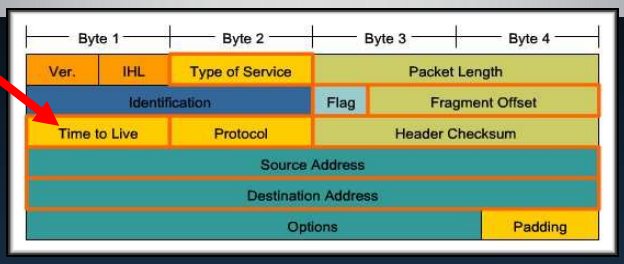
Split Horizon

Actually, that's not quite true!



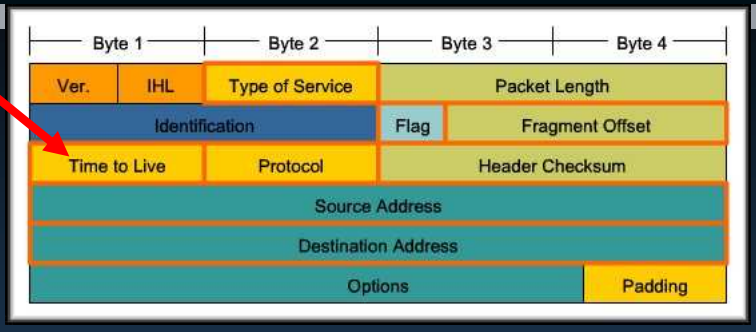
- Router D actually does send an update to Router A.
- This update has a metric of 16 which means that the route is unreachable and Router A ignores the update.
- When applied with split horizon, this deliberate "poisoning" of the route is called *poison reverse*.
  - *Split Horizon with Poison Reverse is enabled by default.*

IP and Time-To-Live (TTL)



- The Time to Live (TTL) is an 8-bit field in the IP header that limits the number of hops a packet can traverse through the network before it is discarded.
- The purpose of the TTL field is to avoid a situation in which an undeliverable packet keeps circulating on the network endlessly.
- With TTL, the 8-bit field is set with a value by the source device of the packet.

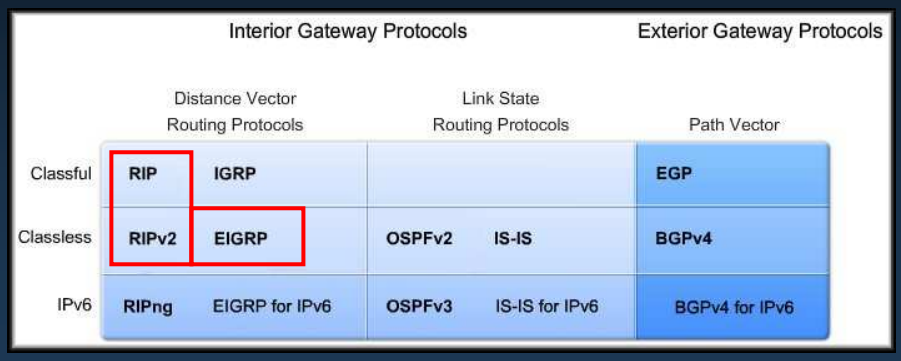
IP and Time-To-Live (TTL)



- The TTL is decreased by 1 by every router on the route to its destination.
- If the TTL field reaches 0 before the packet arrives at its destination, the packet is discarded and the router sends an Internet Control Message Protocol (ICMP) error message back to the source of the IP packet.

Distance Vector Routing Protocols

Distance Vector Routing Protocols Today



RIP and EIGRP

- For distance vector routing protocols, there are really only two choices: **RIP or EIGRP**.

Distance Vector Routing Protocols			
Feature	**RIP**	**RIPv2**	**EIGRP**
Speed of Convergence	**Slow**	**Slow**	**Fast**
Scalability	**Small**	**Small**	**Large**
Supports VLSM	**No**	**Yes**	**Yes**
Resource Usage	**Low**	**Low**	**Medium**
Implementation	**Simple**	**Simple**	**Complex**

- The decision about which routing protocol to use in a given situation is influenced by a number of factors, including
  - Size of the network.
  - Compatibility between models of routers.
  - Administrative knowledge required.