

# Chương 4

## Routing Protocol

- ❑ GV : ThS.Nguyễn Duy
- ❑ Email : duyn@uit.edu.vn

# Nội Dung

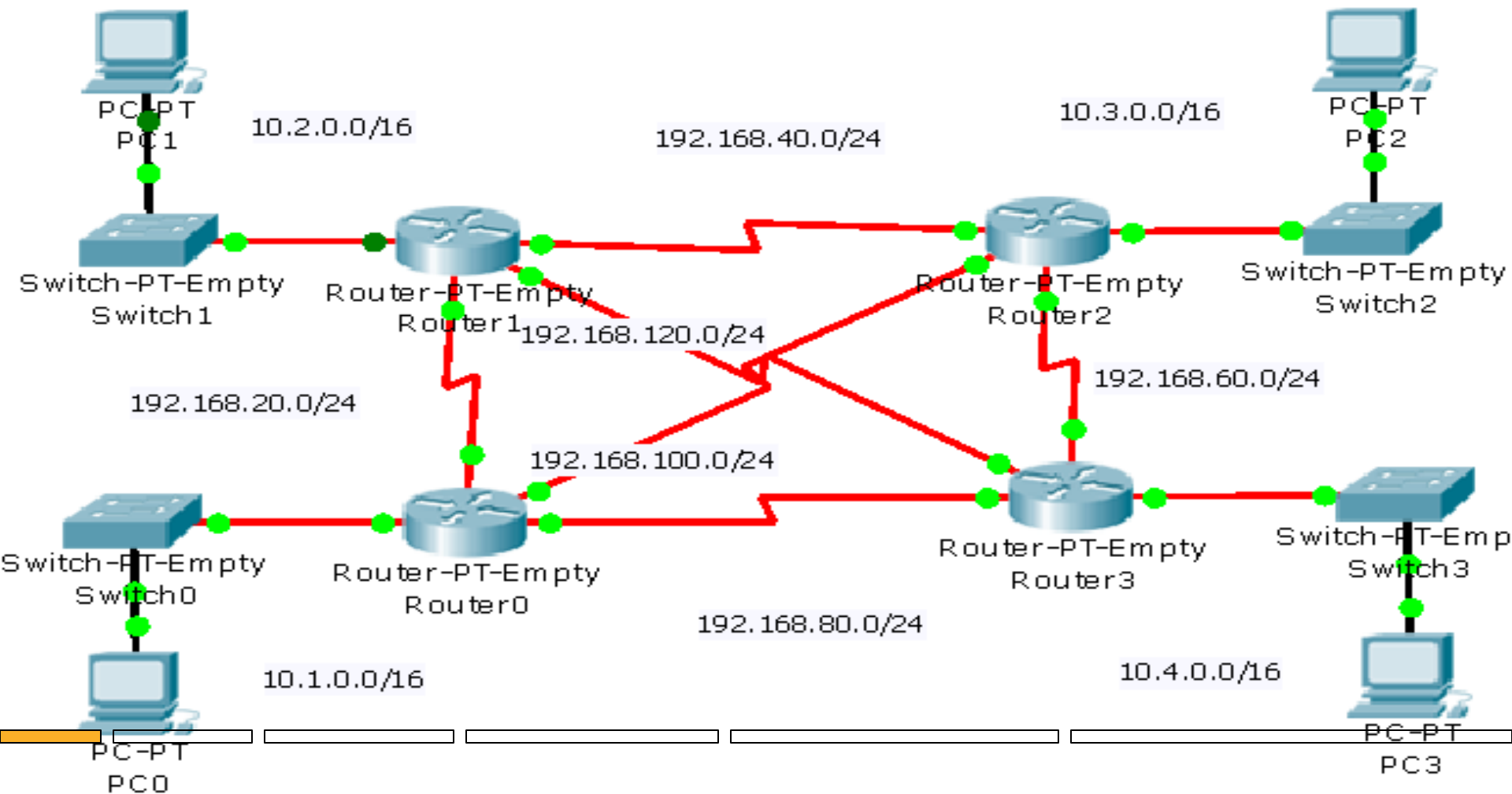
- ❑ Định tuyến là gì ?
- ❑ Phân loại Định Tuyến
- ❑ Static Routing và Default Routing
- ❑ Routing Protocol là gì ?
- ❑ Distance Vector Routing Protocol
- ❑ Link-State Routing Protocol

# Nội Dung

- ❑ Định tuyến là gì ?
- ❑ Phân loại Định Tuyến
- ❑ Static Routing và Defalt Routing
- ❑ Routing Protocol là gì ?
- ❑ Distance Vector Routing Protocol
- ❑ Link-State Routing Protocol

# Định tuyến là gì ?

- ❑ Định tuyến là quy trình Router giúp vận chuyển gói tin từ Mạng Nguồn đến Mạng Đích theo **đường đi tốt nhất**

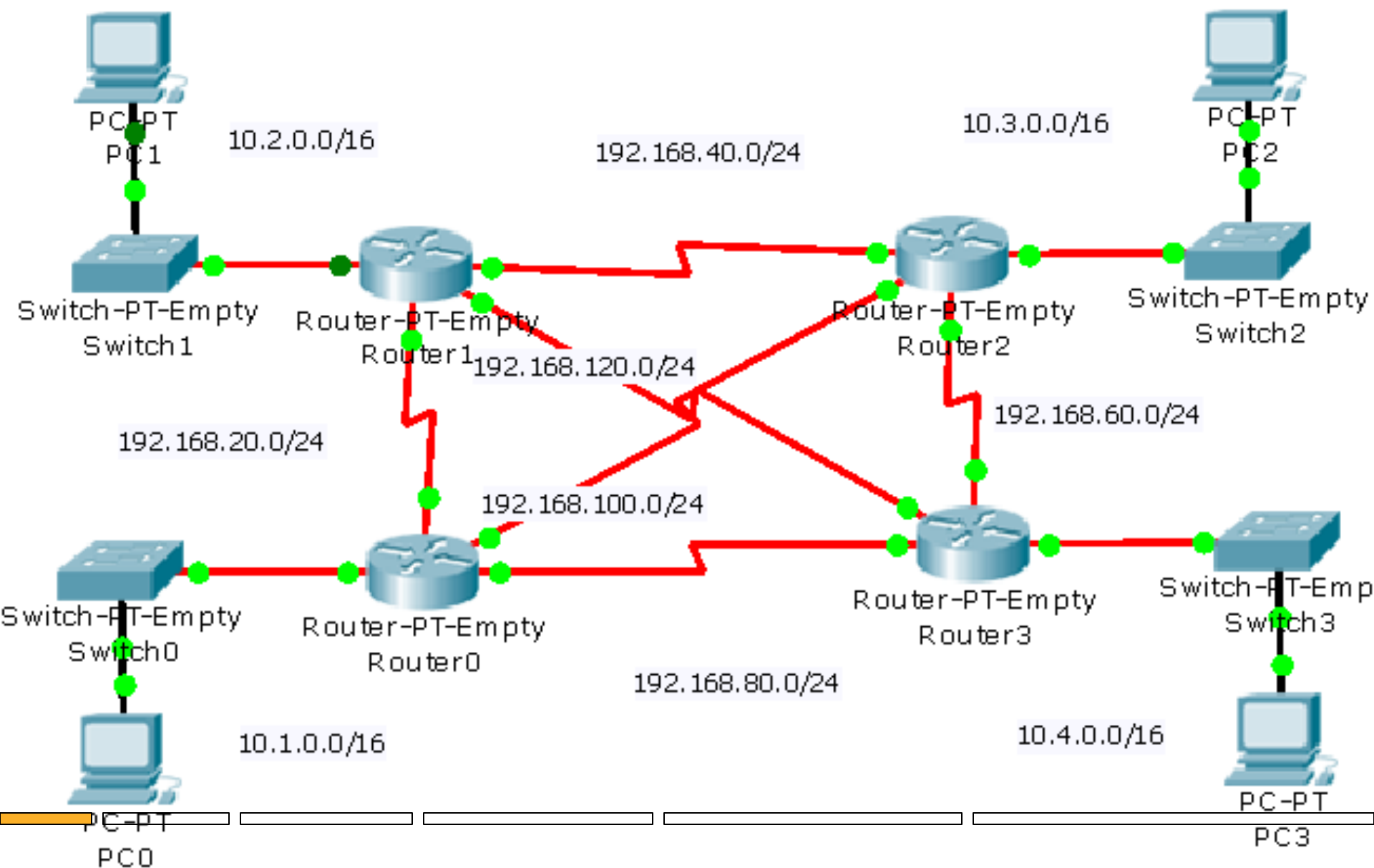


# Định tuyến là gì ?

- ❑ Router dựa vào đâu để chọn đường đi ?
  - ❑ Routing Table
- ❑ Router dựa vào đâu để chọn đường đi tốt nhất ?
  - ❑ Metric
    - Hop Count
    - Bandwidth
    - Load
    - Delay
    - Cost
    - Reliability

# Routing Table và Metric

Định tuyến là gì ?



# Routing Table và Metric

## ❑ R0#show ip route

```
10.0.0.0/16 is subnetted, 4 subnets
C      10.1.0.0 is directly connected, FastEthernet3/0
R      10.2.0.0 [120/1] via 192.168.20.2, 00:00:14, Serial0/0
R      10.3.0.0 [120/1] via 192.168.100.2, 00:00:12, Serial2/0
R      10.4.0.0 [120/1] via 192.168.80.1, 00:00:11, Serial1/0
C      192.168.20.0/24 is directly connected, Serial0/0
R      192.168.40.0/24 [120/1] via 192.168.20.2, 00:00:14, Serial0/0
                        [120/1] via 192.168.100.2, 00:00:12, Serial2/0
R      192.168.60.0/24 [120/1] via 192.168.80.1, 00:00:11, Serial1/0
                        [120/1] via 192.168.100.2, 00:00:12, Serial2/0
C      192.168.80.0/24 is directly connected, Serial1/0
C      192.168.100.0/24 is directly connected, Serial2/0
R      192.168.120.0/24 [120/1] via 192.168.20.2, 00:00:14, Serial0/0
                        [120/1] via 192.168.80.1, 00:00:11, Serial1/0

R0#
```

# Phân tích Routing Table

```
C      10.1.0.0 is directly connected, FastEthernet3/0
R      10.2.0.0 [120/1] via 192.168.20.2, 00:00:14, Serial0/0
```

## ❑ **Code** : local network hay remote network

❑ C : Local network

❑ X : Giao thức định tuyến

○ S – static

○ I – IGRP

○ R – RIP

○ M – mobile

○ B – BGP

○ D – EIGRP

○ EX - EIGRP external

○ O – OSPF

○ IA - OSPF inter area

○ N1 - OSPF NSSA external type 1

○ N2 - OSPF NSSA external type 2

○ E1 - OSPF external type 1

○ E2 - OSPF external type 2

○ E – EGP

○ I - IS-IS

○ L1 - IS-IS level-1

○ L2 - IS-IS level-2

○ Ia - IS-IS inter area

## ❑ **Network** :

❑ Nếu Code là C → Local Network

❑ Nếu Code khác C → Destination Network



# Phân tích Routing Table

- ❑ **Administrative Distance** : Độ tin cậy của giao thức định tuyến

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

# Phân tích Routing Table

- ❑ **Metric** : thông tin được sử dụng để xác định đường đi ngắn nhất.
  - ❑ 1
- ❑ **Gateway** : Địa chỉ của Next-Hop
  - ❑ 192.168.20.2
- ❑ **Interface** : Interface của Router mà gói tin sẽ được chuyển tiếp đến để đi tới mạng đích
  - ❑ Serial0/0

# Nội Dung

- ❑ Định tuyến là gì ?
- ❑ **Phân loại Định Tuyến**
- ❑ Static Routing và Defalt Routing
- ❑ Routing Protocol là gì ?
- ❑ Distance Vector Routing Protocol
- ❑ Link-State Routing Protocol

# Phân loại Định Tuyến

## ❑ Static Routing

- ❑ Người quản trị mạng phải cấu hình trên từng Router để tạo và cập nhật Routing Table

## ❑ Dynamic Routing

- ❑ Routing Table tự động tạo và cập nhật
- ❑ Cách tạo và cập nhật bảng định tuyến như thế nào là tùy thuộc từng giao thức định tuyến.
- ❑ Các giao thức định tuyến thông dụng : RIPv1, RIPv2, IGRP, EIGRP, OSPF, IS-IS, BGP

# Nội Dung

- ❑ Định tuyến là gì ?
- ❑ Phân loại Định Tuyến
- ❑ **Static Routing và Default Routing**
- ❑ Routing Protocol là gì ?
- ❑ Distance Vector Routing Protocol
- ❑ Link-State Routing Protocol

# Static Routing

## Phân loại Định Tuyến

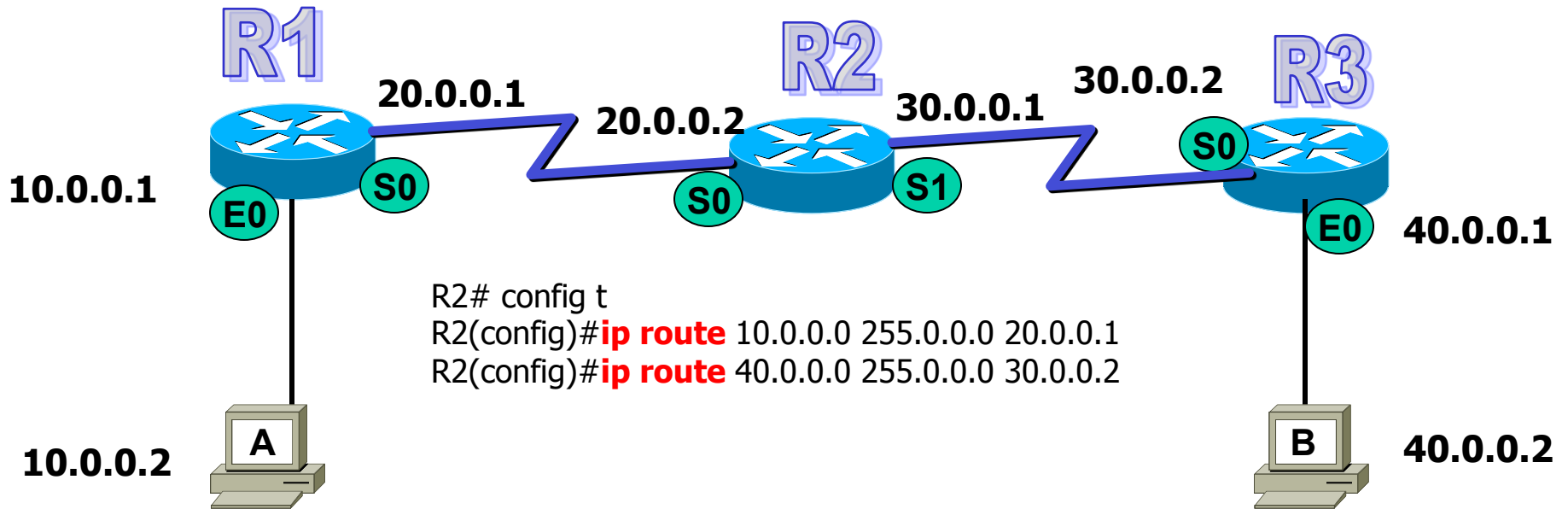
```
ip route [destination_network] [mask] [next-hop_address or exitinterface]  
[administrative_distance] [permanent]
```

- ❑ **ip route** : câu lệnh tạo ra static routing
- ❑ ***Destination\_network*** : *mạng đích*
- ❑ ***Mask*** : *subnet mark của destination network*
- ❑ ***Next-hop\_address*** : *địa chỉ ip của next-hop Router*
- ❑ ***Exitinterface*** : *tên của Interface trên Router mà gói tin sẽ được chuyển ra interface này để tới destination network*
- ❑ ***Administrative\_distance*** : *mặc định*

# Static Routing - LAB

## Phân loại Định Tuyến

### □ Cấu hình IP Route



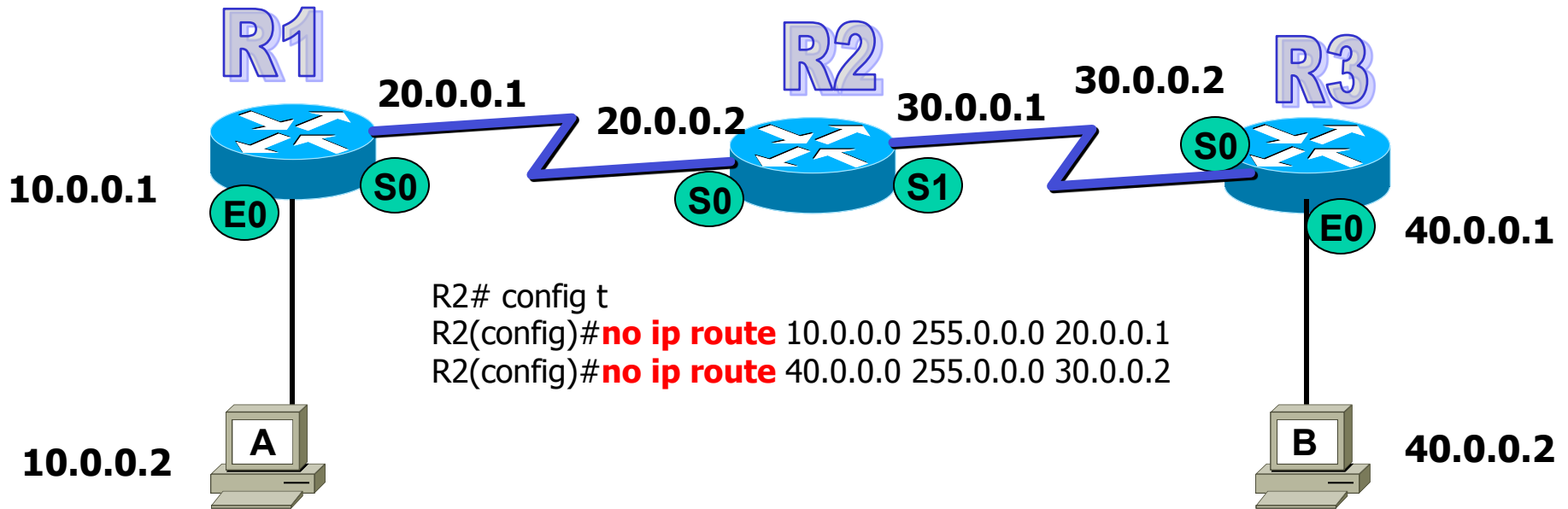
```
R1# config t
R1(config)#ip route 30.0.0.0 255.0.0.0 20.0.0.2
R1(config)#ip route 40.0.0.0 255.0.0.0 20.0.0.2
```

```
R3# config t
R3(config)#ip route 10.0.0.0 255.0.0.0 30.0.0.1
R3(config)#ip route 20.0.0.0 255.0.0.0 30.0.0.1
```

# Static Routing - LAB

## Phân loại Định Tuyến

### ❑ Remove IP Route



```
R2# config t
R2(config)#no ip route 10.0.0.0 255.0.0.0 20.0.0.1
R2(config)#no ip route 40.0.0.0 255.0.0.0 30.0.0.2
```

```
R1# config t
R1(config)#no ip route 30.0.0.0 255.0.0.0 20.0.0.2
R1(config)#no ip route 40.0.0.0 255.0.0.0 20.0.0.2
```

```
R3# config t
R3(config)#no ip route 10.0.0.0 255.0.0.0 30.0.0.1
R3(config)#no ip route 20.0.0.0 255.0.0.0 30.0.0.1
```



# Default Routing

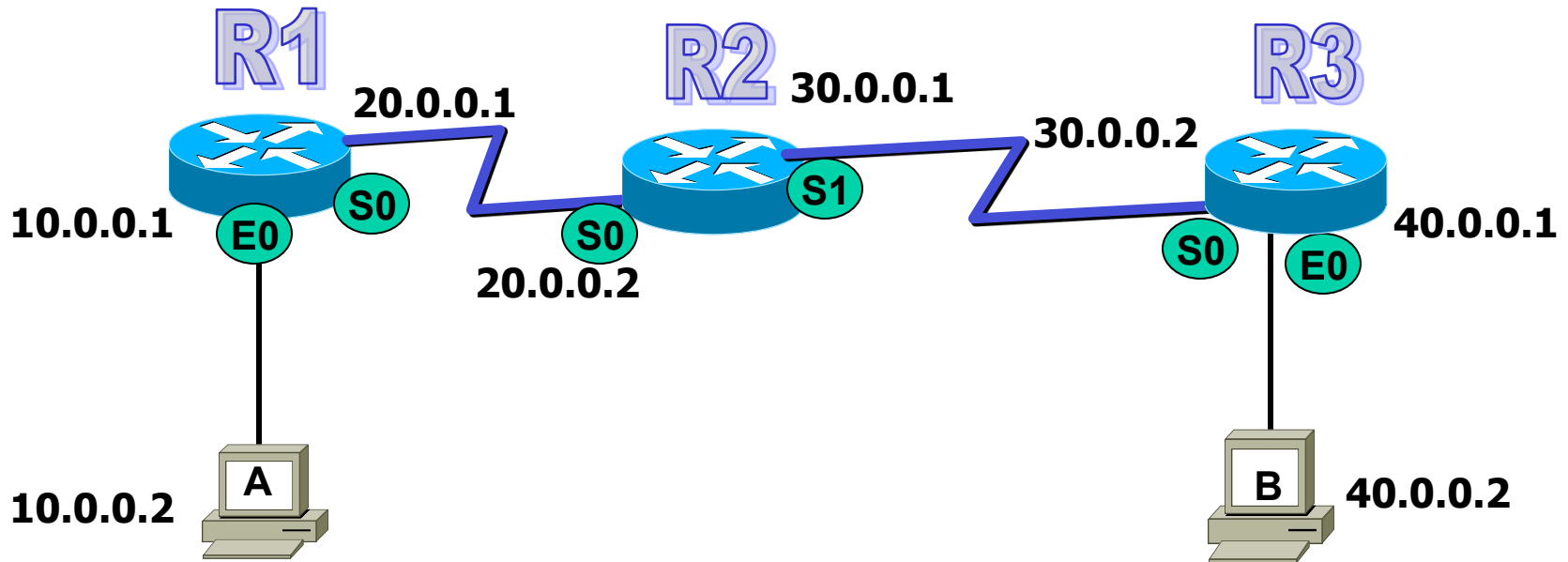
## Phân loại Định Tuyến

- ❑ Được sử dụng để định tuyến gói tin tới Destination Network, khi không có câu lệnh Route nào trong bảng Routing Table phù hợp với Destination Network
- ❑ Cấu trúc câu lệnh default route
  - ❑ **ip route 0.0.0.0 0.0.0.0** [*next-hop-address* | *outgoing interface*]
- ❑ Ví dụ sử dụng next-hop-address để cấu hình :
  - ❑ Router(config)#**ip route 0.0.0.0 0.0.0.0 172.16.4.1**
- ❑ Ví dụ sử dụng outgoing interface để cấu hình :
  - ❑ Router(config)#**ip route 0.0.0.0 0.0.0.0 s0/0**

# Default Routing

## Phân loại Định Tuyến

### □ Cấu hình Default Route



```
R2# config t
R2(config)#ip route 10.0.0.0 255.0.0.0 20.0.0.1
R2(config)#ip route 40.0.0.0 255.0.0.0 30.0.0.2
```

```
R1# config t
R1(config)#ip route 0.0.0.0 0.0.0.0 20.0.0.2
```

```
R3# config t
R3(config)#ip route 0.0.0.0 0.0.0.0 30.0.0.1
```

# Nội Dung

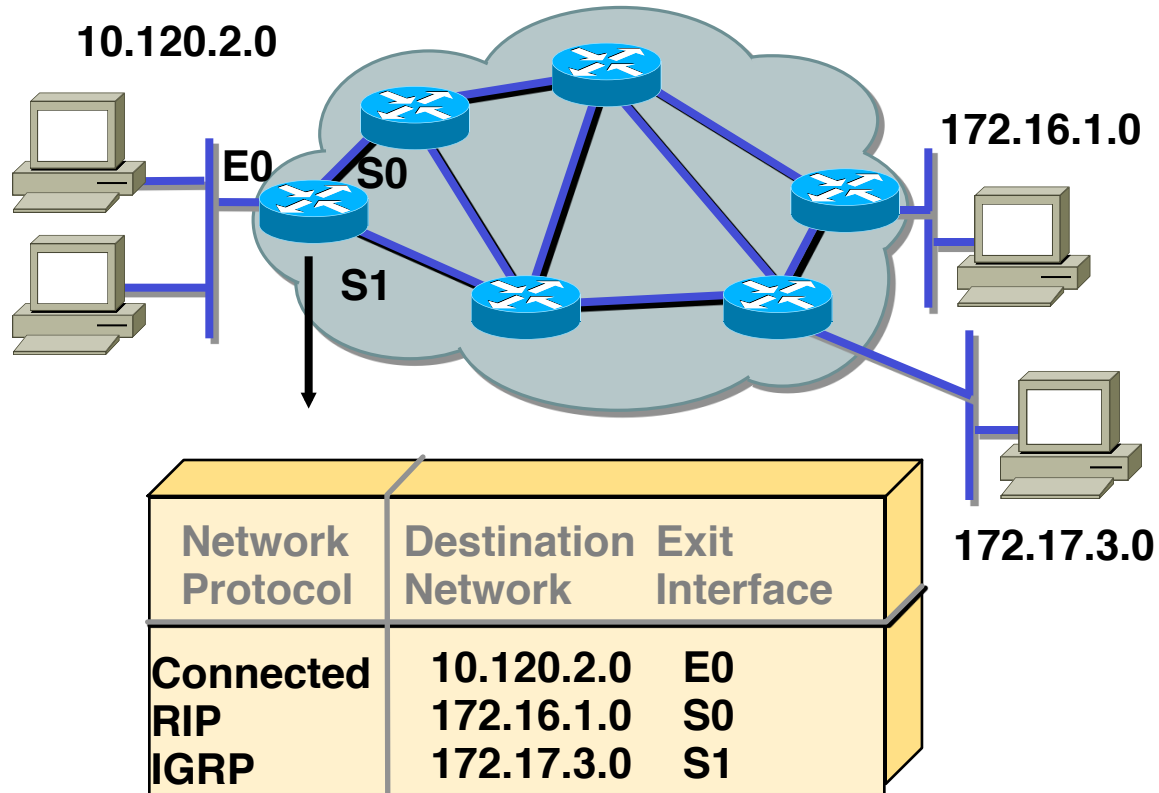
- ❑ Định tuyến là gì ?
- ❑ Phân loại Định Tuyến
- ❑ Static Routing và Defalt Routing
- ❑ **Routing Protocol**
- ❑ Distance Vector Routing Protocol
- ❑ Link-State Routing Protocol

# Routing Protocol

- ❑ Routing Protocol là gì ?
- ❑ Phân loại Routing
- ❑ Phân loại Routing Protocol
- ❑ Classful Routing và Classless Routing

# Routing Protocol là gì ?

- ❑ **Routing Protocol** là giao thức được sử dụng để tạo và cập nhật bảng định tuyến một cách tự động.
- ❑ Hay nói cách khác, Routing Protocol là giao thức được sử dụng để tìm đường đi giữa các Router



# Routing Categories

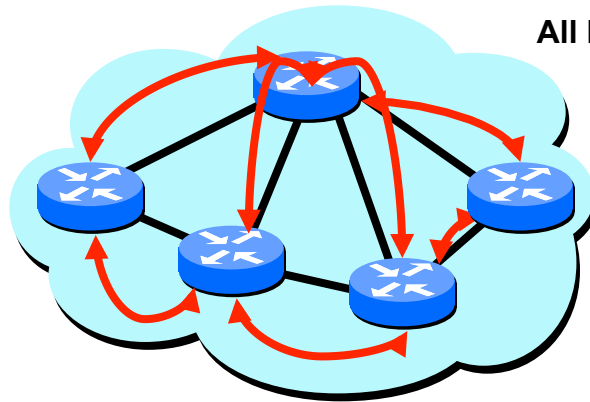
## Autonomous System

- ❑ An Autonomous System (AS) là một nhóm các Router cùng dưới sự quản trị của chính sách Routing
- ❑ Một nhóm các Router trao đổi Routing Table
- ❑ AS được xác định bởi 1 số nguyên 1 - 65535

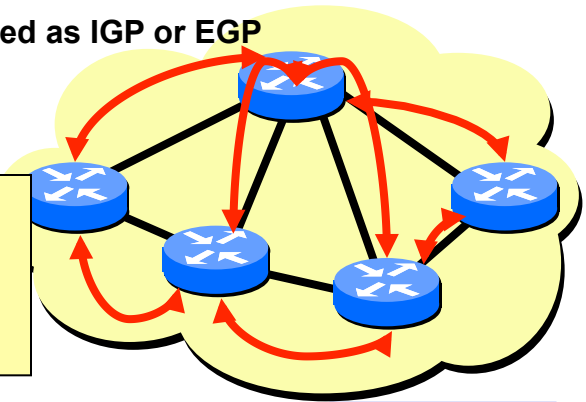
All Routing protocols are categorized as IGP or EGP

### EGP

Exterior Gateway Protocols được sử dụng để định tuyến giữa các Autonomous Systems

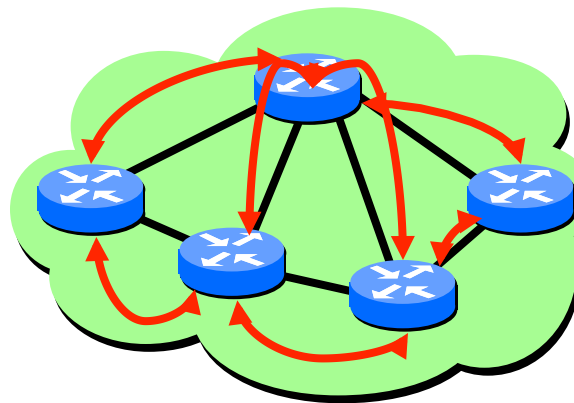


AS 1000



AS 3000

AS 2000



### IGP

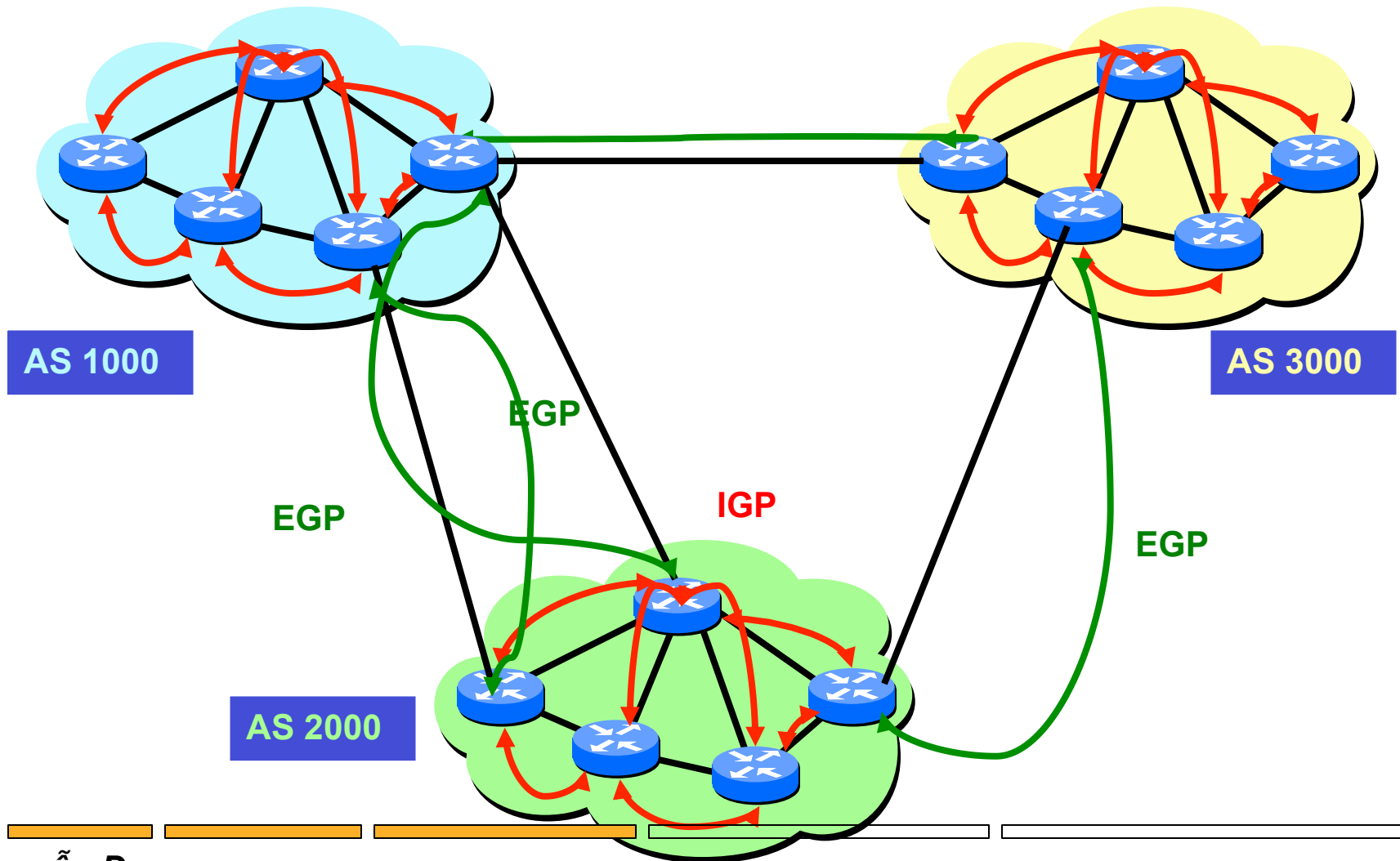
Interior Gateway Protocols được sử dụng để định tuyến bên trong Autonomous System.

# Phân loại Routing

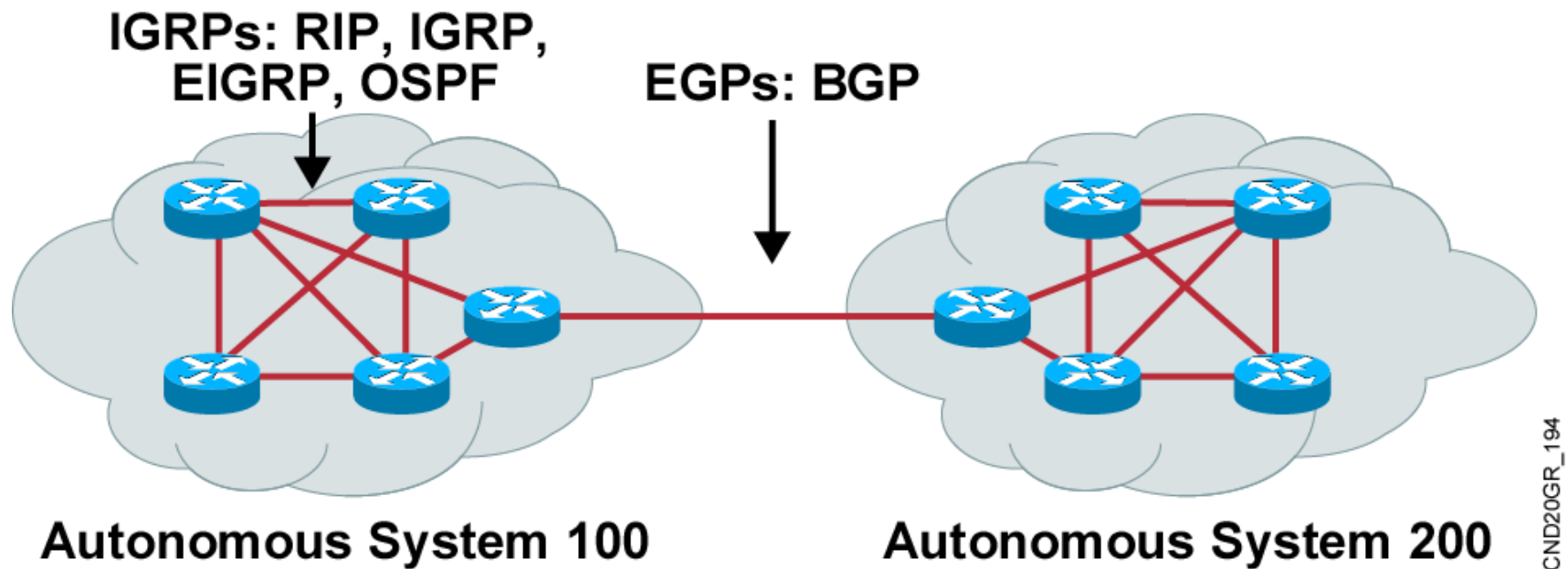
Interior Gateway Protocol  
(IGP)

Exterior Gateway Protocol  
(EGP)

Interior Gateway Protocol  
(IGP)



# AS, IGP và EGP

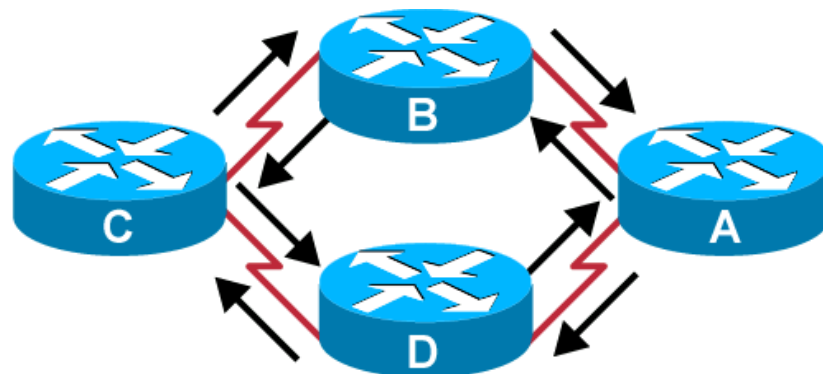


- ❑ IGP hoạt động trong 1 autonomous system.
- ❑ EGP kết nối các autonomous systems.

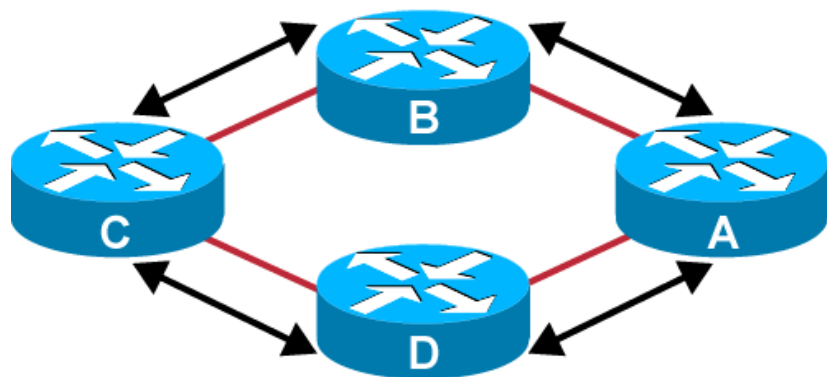


# Phân loại Routing Protocol

**Distance Vector**



**Hybrid Routing**



**Link State**

ICND20GR\_196

# Phân loại Routing Protocol

## ❑ Distance Vector

- ❖ RIP V1

- ❖ IGRP

- ❖ RIP V2

## ❑ Link state

- ❖ OSPF

## ❑ Hybrid

- ❖ EIGRP

# Phân loại Routing Protocol

- ❑ Classful Routing : không chứa subnet mark trong câu lệnh quảng bá định tuyến
  - ❑ RIPv1
  - ❑ IGRP
- ❑ Classless Routing : chứa subnet mark trong câu lệnh quảng bá định tuyến
  - ❑ RIPv2
  - ❑ EIGRP
  - ❑ OSPF
  - ❑ IS-IS
- ❑ Trong cùng 1 network, mặc định subnet mark sẽ bị summary

# Nội Dung

- ❑ Định tuyến là gì ?
- ❑ Phân loại Định Tuyến
- ❑ Static Routing và Defalt Routing
- ❑ Routing Protocol
- ❑ **Distance Vector Routing Protocol**
- ❑ Link-State Routing Protocol

# Distance Vector Routing Protocol

- ❑ Distance vector routing protocol là gì ?
- ❑ Quá trình cập nhật và trao đổi bảng định tuyến ?
- ❑ Routing loop
- ❑ Cách khắc phục loop :
  - ❑ Holddown
  - ❑ Split Horizone
  - ❑ Routing Poisoning
  - ❑ Trigged update

# Distance Vector Routing Protocol

- ❑ Distance Vector Routing Protocol mô tả cách thức các Router :
  - ❑ Tạo bảng định tuyến
  - ❑ Cập nhật bảng định tuyến
  - ❑ Trao đổi **bảng định tuyến** với các Router láng giềng
- ❑ Những đặc điểm chính :
  - ❑ Cập nhật thông tin định tuyến theo định kỳ
  - ❑ Trao đổi thông tin định tuyến với Router láng giềng
  - ❑ Toàn bộ bảng định tuyến được trao đổi với Router láng giềng

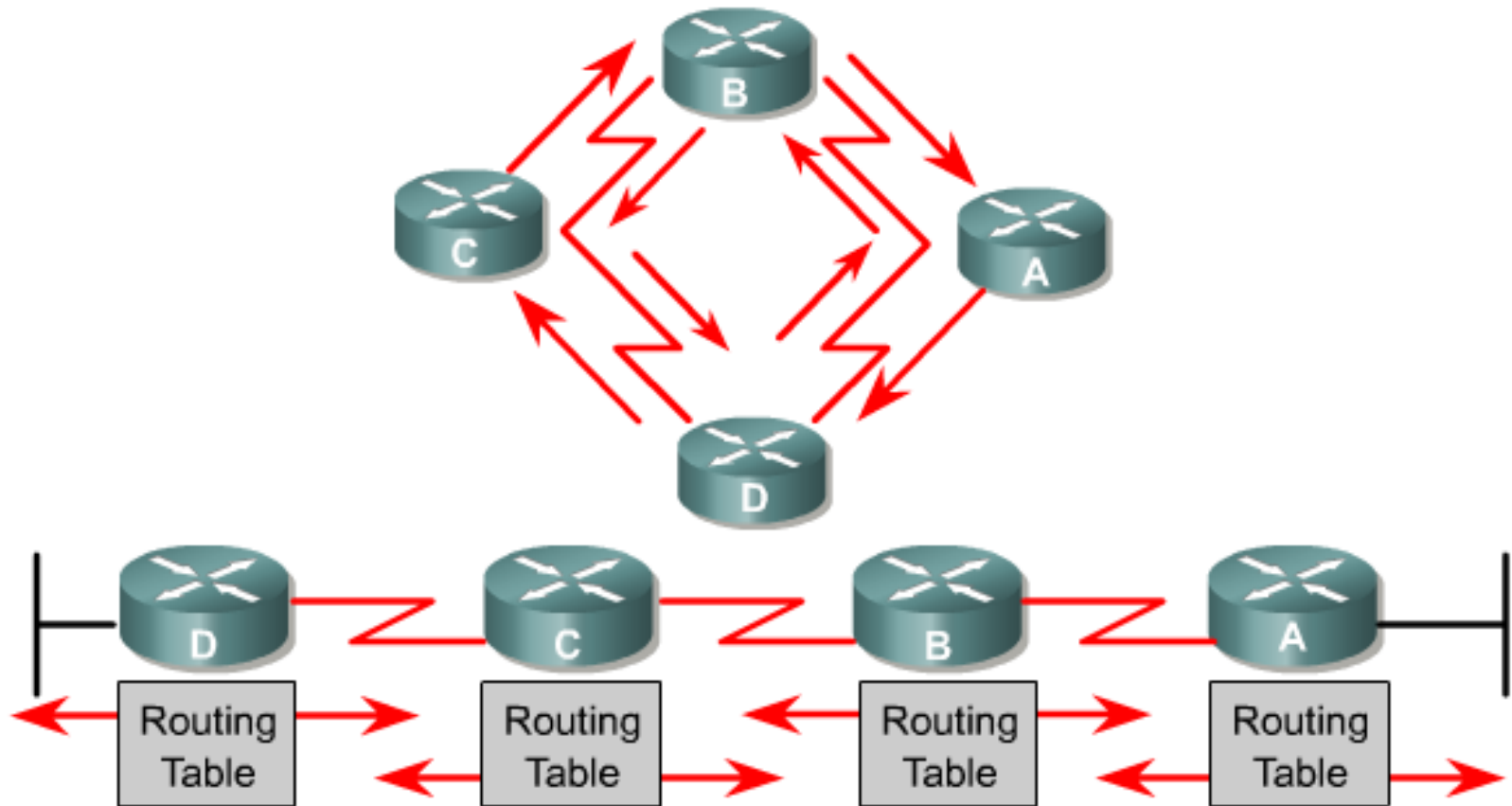
# Quá trình cập nhật bảng định tuyến của Router

## Distance Vector Routing Protocol

- ❑ Router cập nhật bảng định tuyến theo định kỳ hoặc khi cấu trúc mạng có sự thay đổi.
- ❑ Khi cập nhật bảng định tuyến của mình xong, Router sẽ gửi toàn bộ thông tin Bảng Định Tuyến của mình đến các Router lân cận.
- ❑ Các Router lân cận tiếp nhận và cập nhật thông tin Bảng Định Tuyến của mình và tiếp tục gửi cho các Router lân cận.
- ❑ Quá trình được tiếp tục .....

# Quá trình cập nhật bảng định tuyến của Router

## Distance Vector Routing Protocol

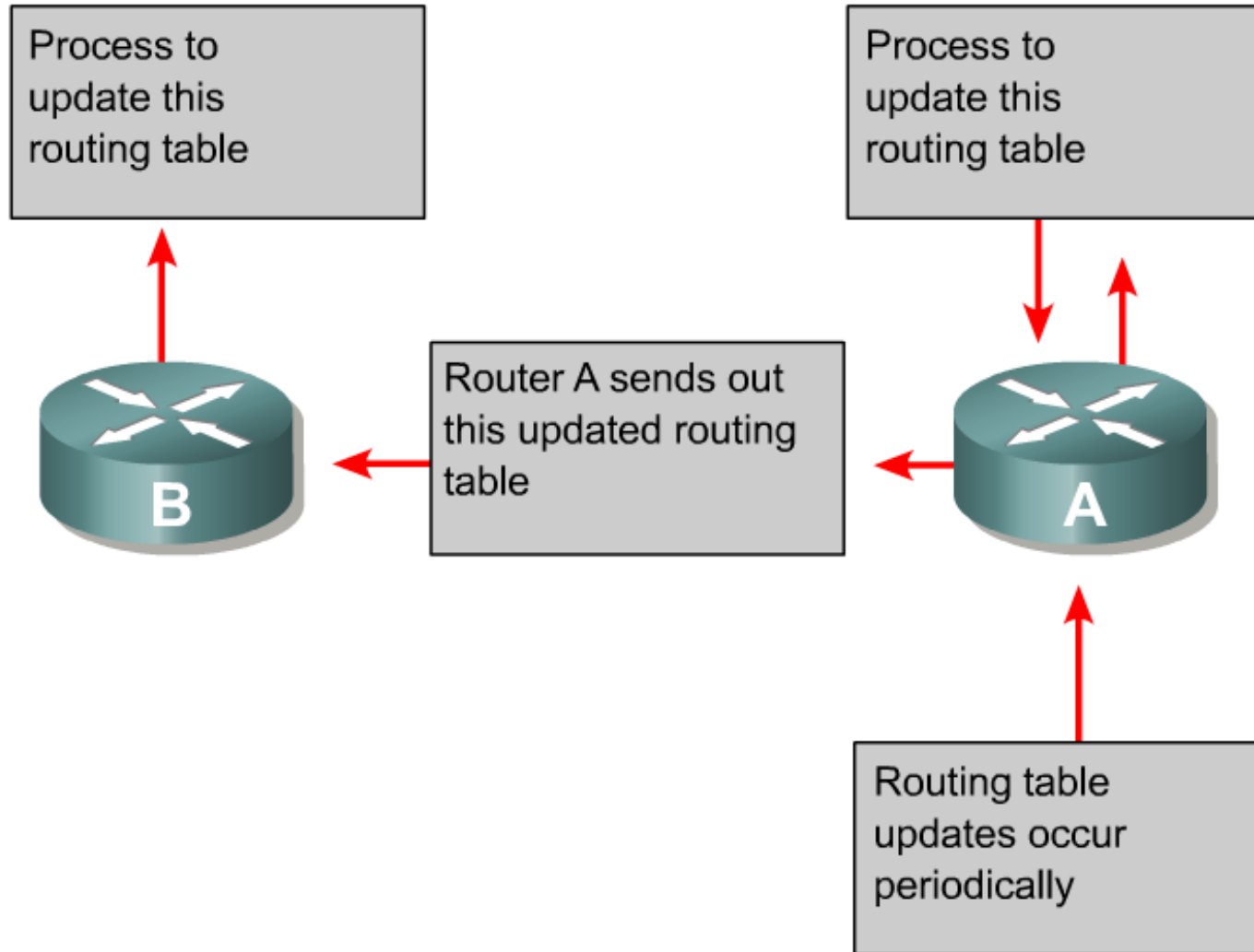


Pass periodic copies of a routing table to neighbor routers and accumulate distance vectors.



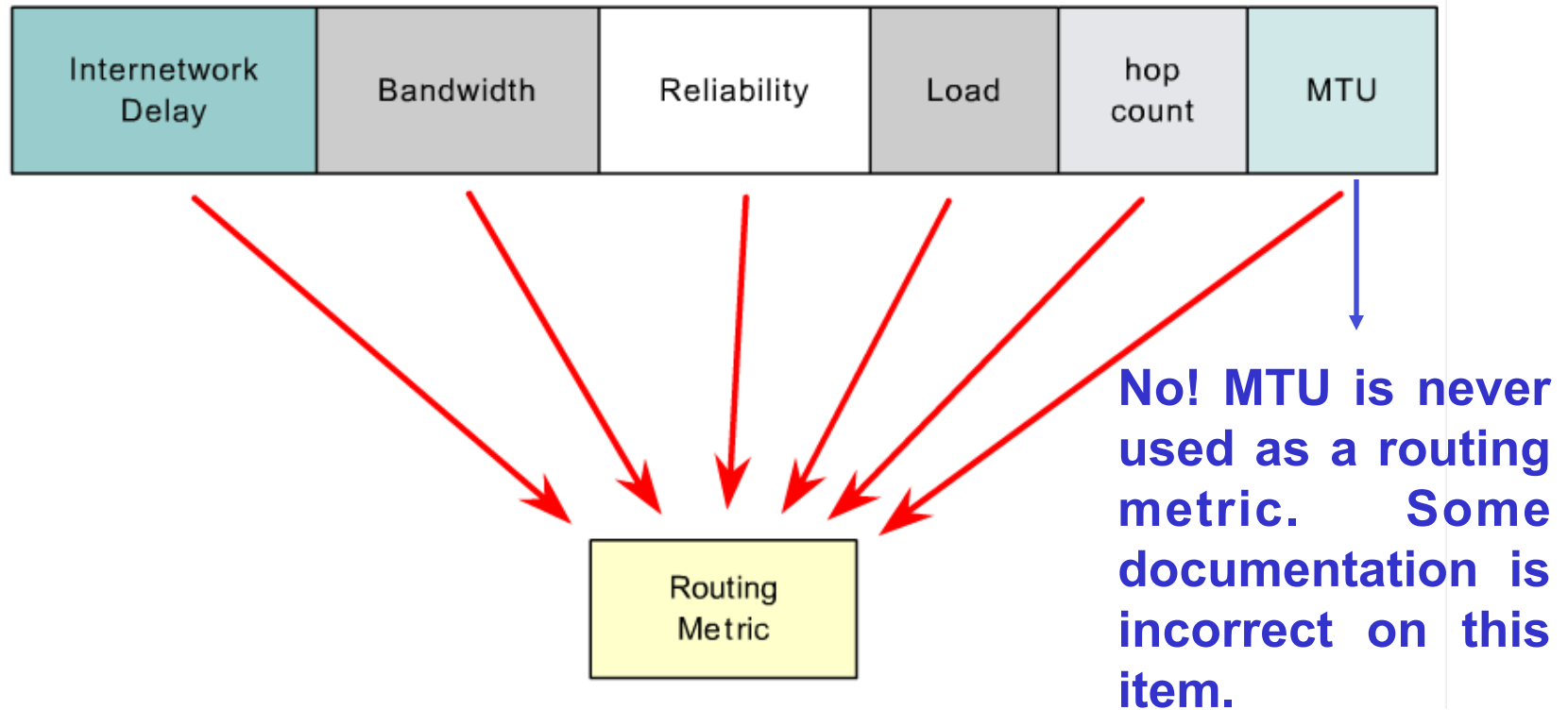
# Quá trình cập nhật bảng định tuyến của Router

## Distance Vector Routing Protocol



# Quá trình cập nhật bảng định tuyến của Router

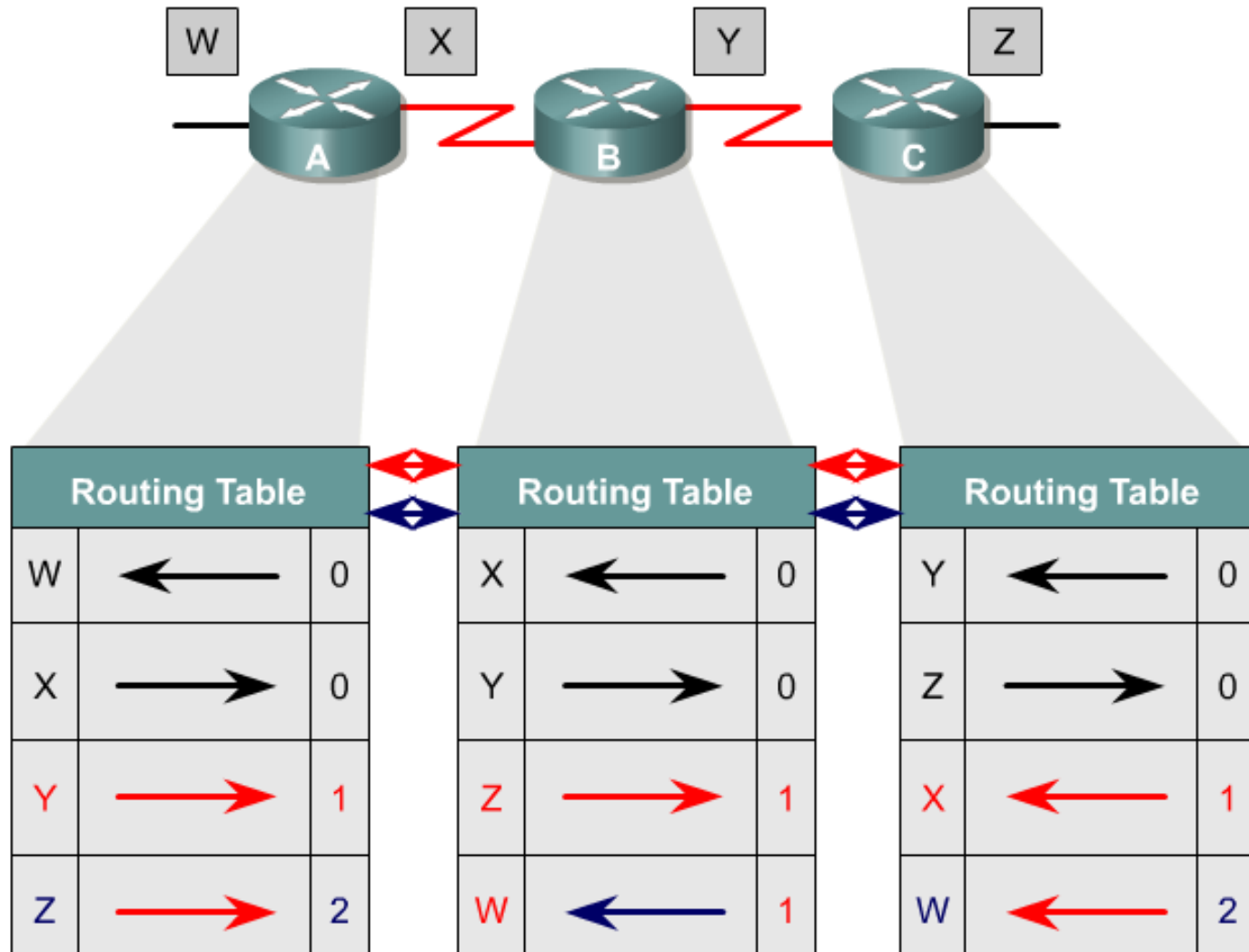
## Distance Vector Routing Protocol



- ❑ **RIP** – Hop Count
- ❑ **IGRP** và **EIGRP** – Bandwidth, Delay, Reliability, Load

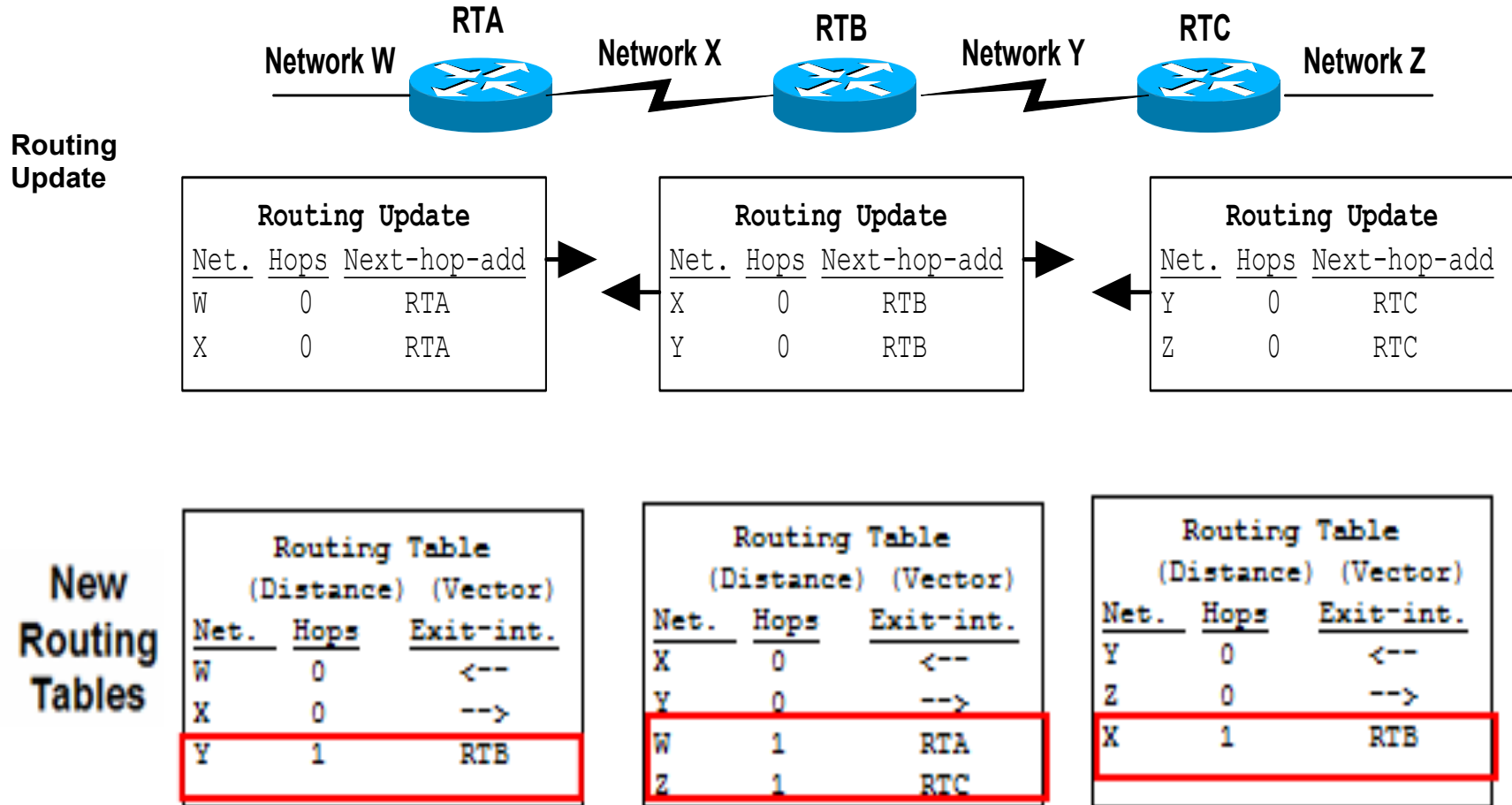
# Quá trình cập nhật bảng định tuyến của Router

## Distance Vector Routing Protocol



# Quá trình cập nhật bảng định tuyến của Router

## Distance Vector Routing Protocol



# Quá trình cập nhật bảng định tuyến của Router

## Distance Vector Routing Protocol



### Existing Routing Tables

Routing Table		
(Distance) (Vector)		
Net.	Hops	Exit-int.
W	0	<--
X	0	-->
Y	1	RTB

Routing Table		
(Distance) (Vector)		
Net.	Hops	Exit-int.
X	0	<--
Y	0	-->
W	1	RTA
Z	1	RTC

Routing Table		
(Distance) (Vector)		
Net.	Hops	Exit-int.
Y	0	<--
Z	0	-->
X	1	RTB

Routing Update		
Net.	Hops	Next-hop-add
W	1	RTA
X	1	RTA
Y	2	RTA

Routing Update		
Net.	Hops	Next-hop-add
X	1	RTB
Y	1	RTB
W	2	RTB
Z	2	RTB

Routing Update		
Net.	Hops	Next-hop-add
Y	1	RTC
Z	1	RTC
X	2	RTC

### New Routing Tables

Routing Table		
(Distance) (Vector)		
Net.	Hops	Exit-int.
W	0	<--
X	0	-->
Y	1	RTB
Z	2	RTB

Routing Table		
(Distance) (Vector)		
Net.	Hops	Exit-int.
X	0	<--
Y	0	-->
W	1	RTA
Z	1	RTC

Routing Table		
(Distance) (Vector)		
Net.	Hops	Exit-int.
Y	0	<--
Z	0	-->
X	1	RTB
W	2	RTB

# Quá trình cập nhật bảng định tuyến của Router

## Distance Vector Routing Protocol



### Routing Tables

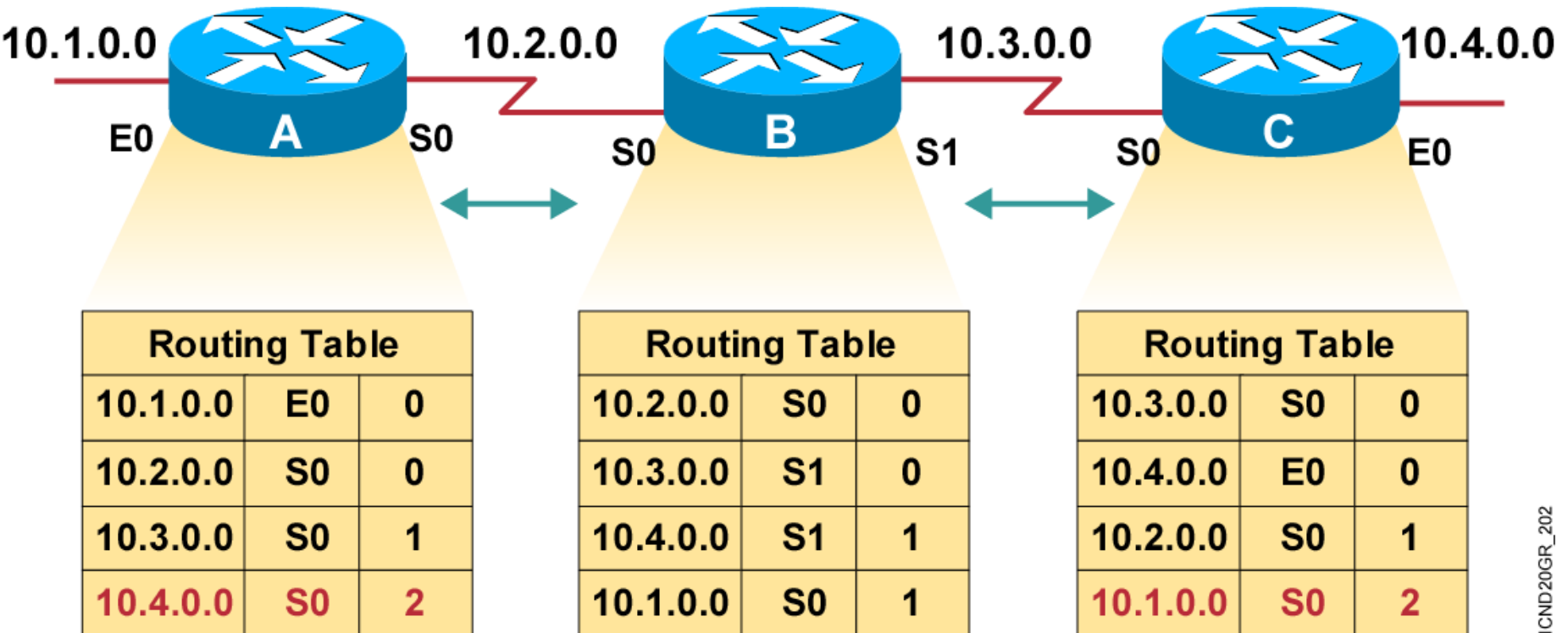
Routing Table		
(Distance)		(Vector)
Net.	Hops	Exit-int.
W	0	<--
X	0	-->
Y	1	RTB
Z	2	RTB

Routing Table		
(Distance)		(Vector)
Net.	Hops	Exit-int.
X	0	<--
Y	0	-->
W	1	RTA
Z	1	RTC

Routing Table		
(Distance)		(Vector)
Net.	Hops	Exit-int.
Y	0	<--
Z	0	-->
X	1	RTB
W	2	RTB

# Routing Loop - 1

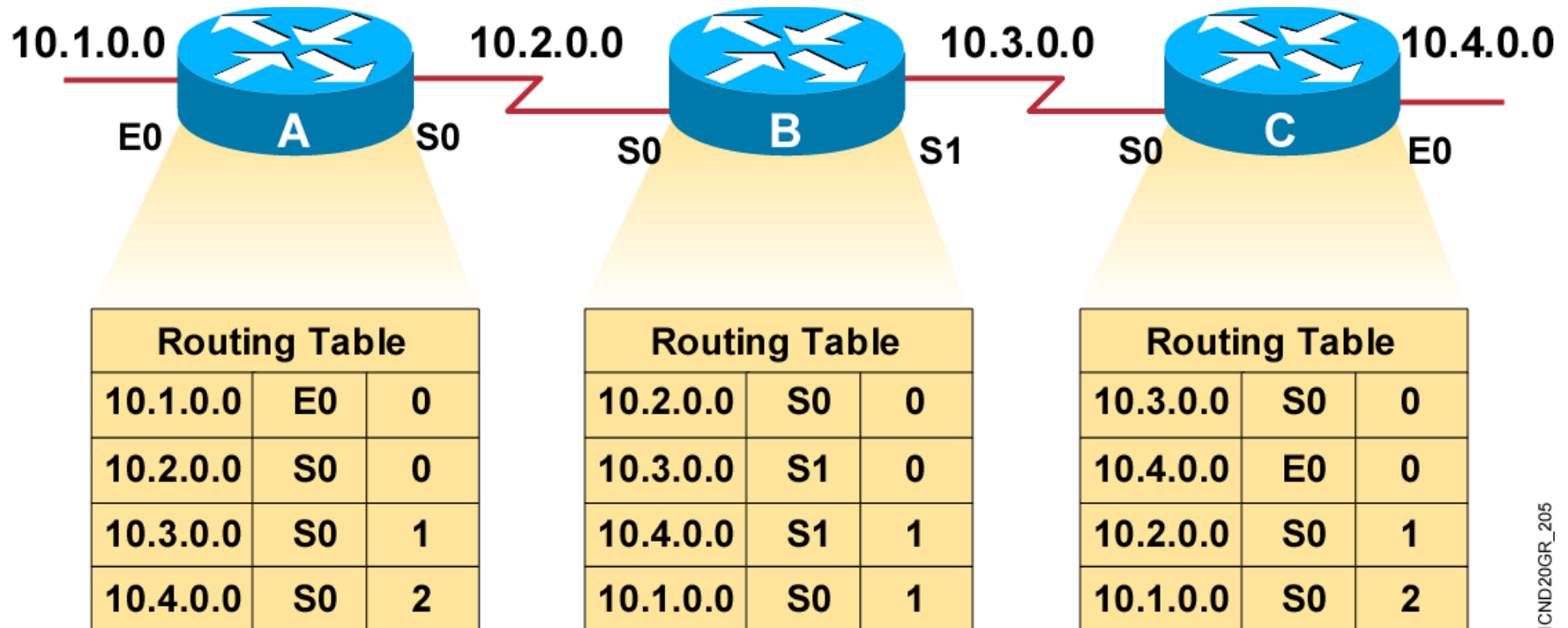
## Distance Vector Routing Protocol



- ❑ Routers cập nhật đường đi tốt nhất thông qua các Router láng giềng

# Routing Loop - 1

## Distance Vector Routing Protocol



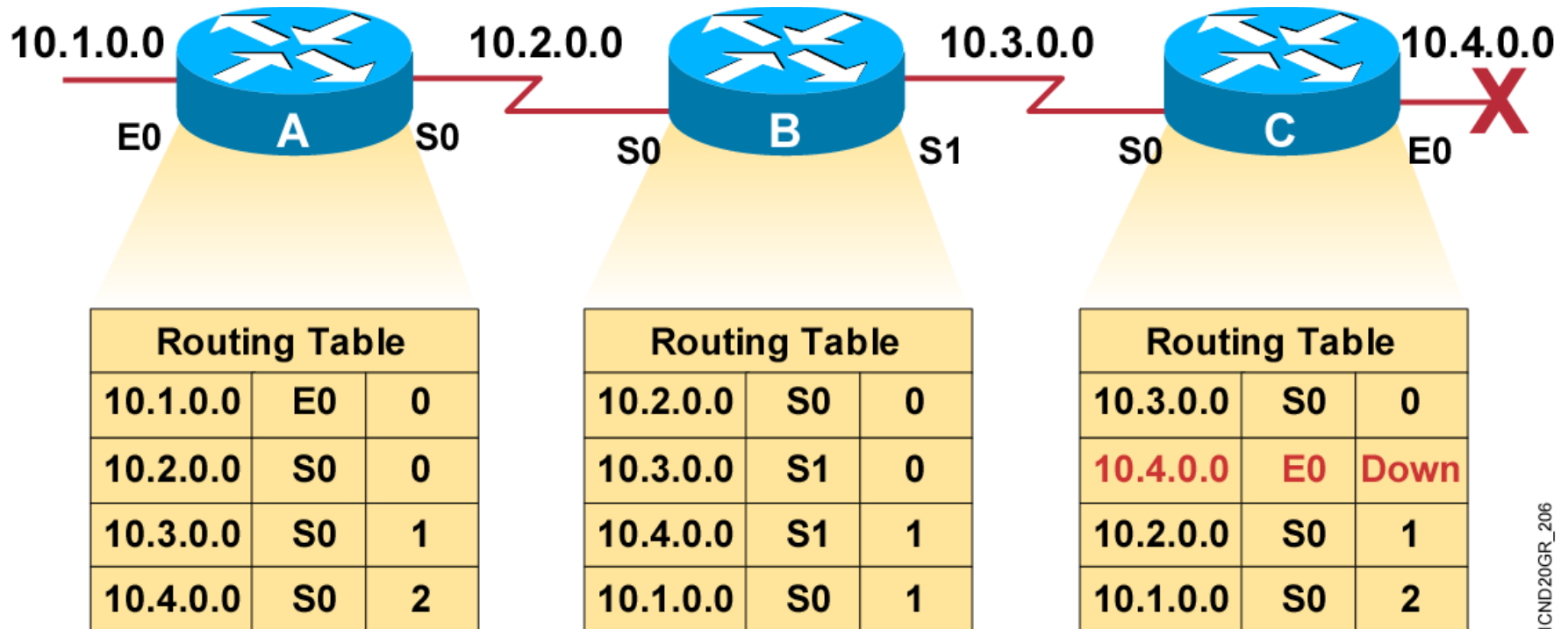
ICND20GR\_205

- ❑ Mỗi Router có được thông tin đến mạng đích và khoảng cách là bao xa....



# Routing Loop - 1

## Distance Vector Routing Protocol

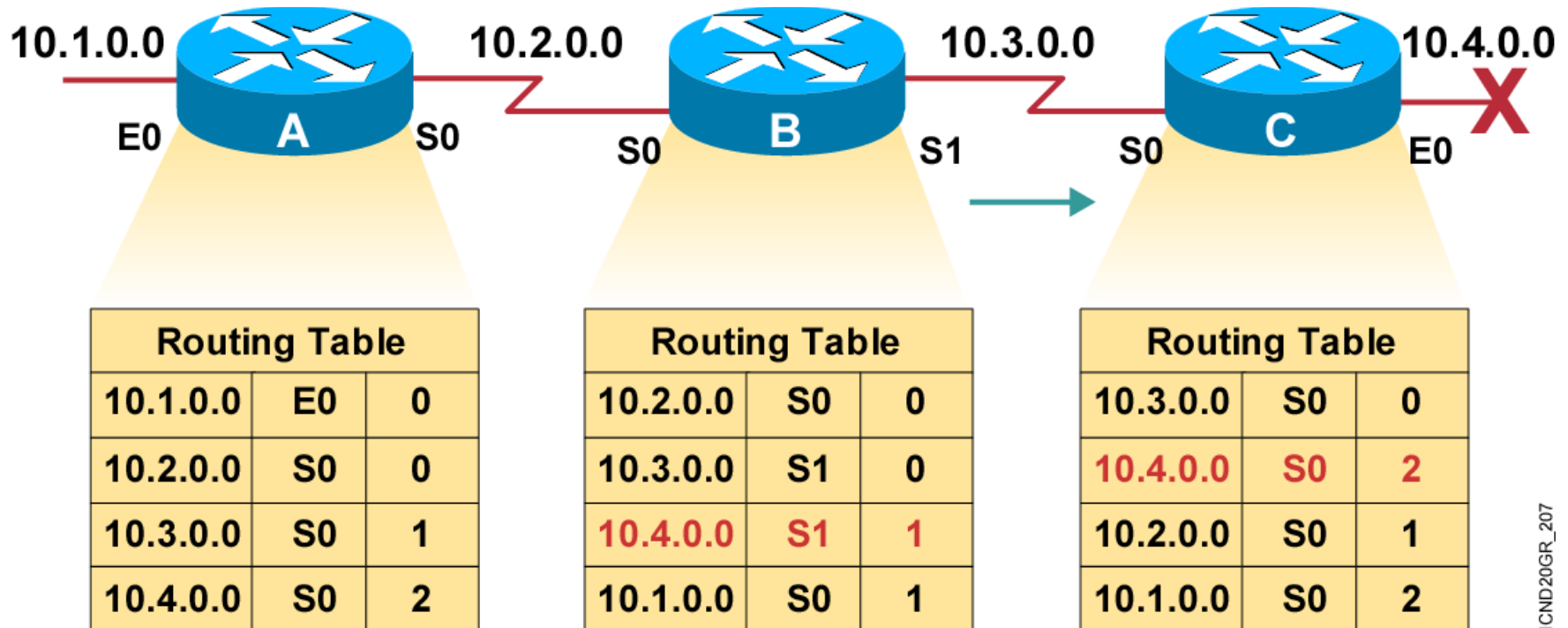


ICND20GR\_206

- ❑ Đường mạng 10.4.0.0 của Router C bị down

# Routing Loop - 1

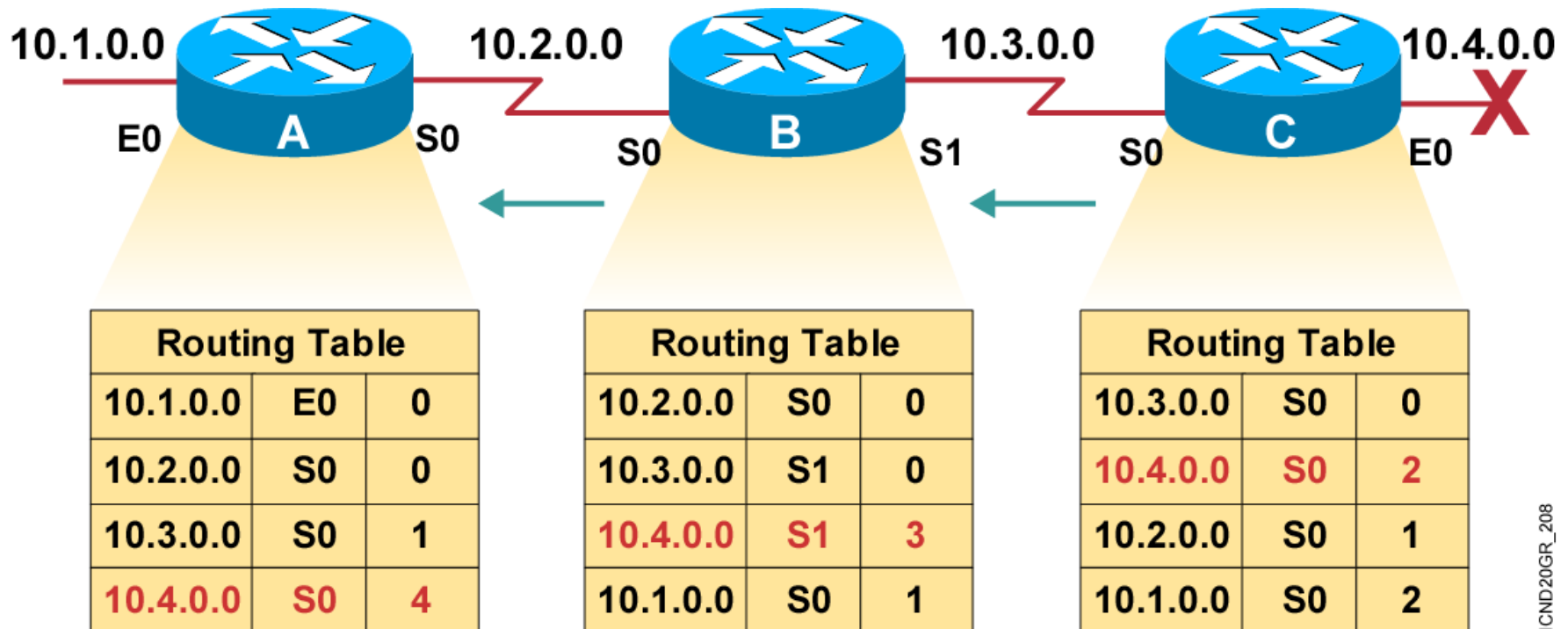
## Distance Vector Routing Protocol



- ❑ Router C cập nhật đường đi tới đường mạng 10.4.0.0 thông qua Router B

# Routing Loop - 1

## Distance Vector Routing Protocol

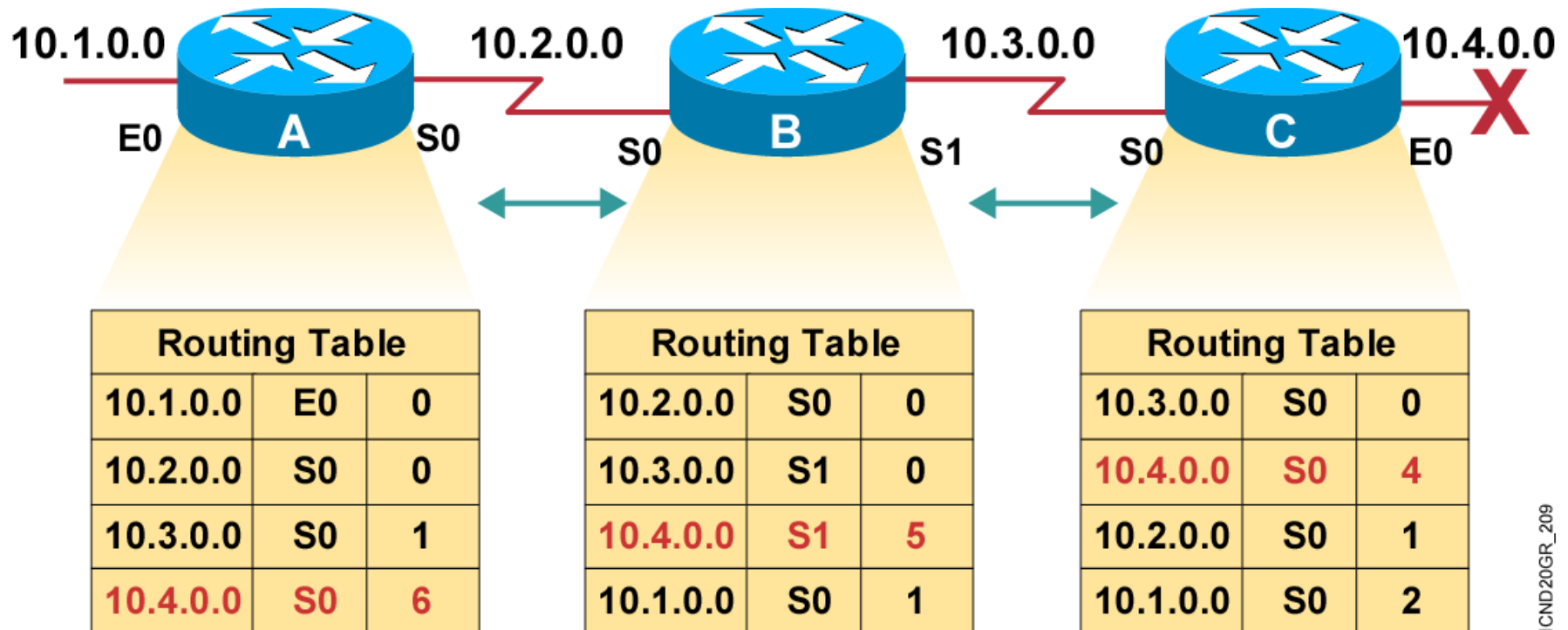


ICND20GR\_208

- ❑ Router C quảng bá bảng định tuyến cho Router B
- ❑ Router B quảng bá bảng định tuyến cho Router A

# Routing Loop - 1

## Distance Vector Routing Protocol

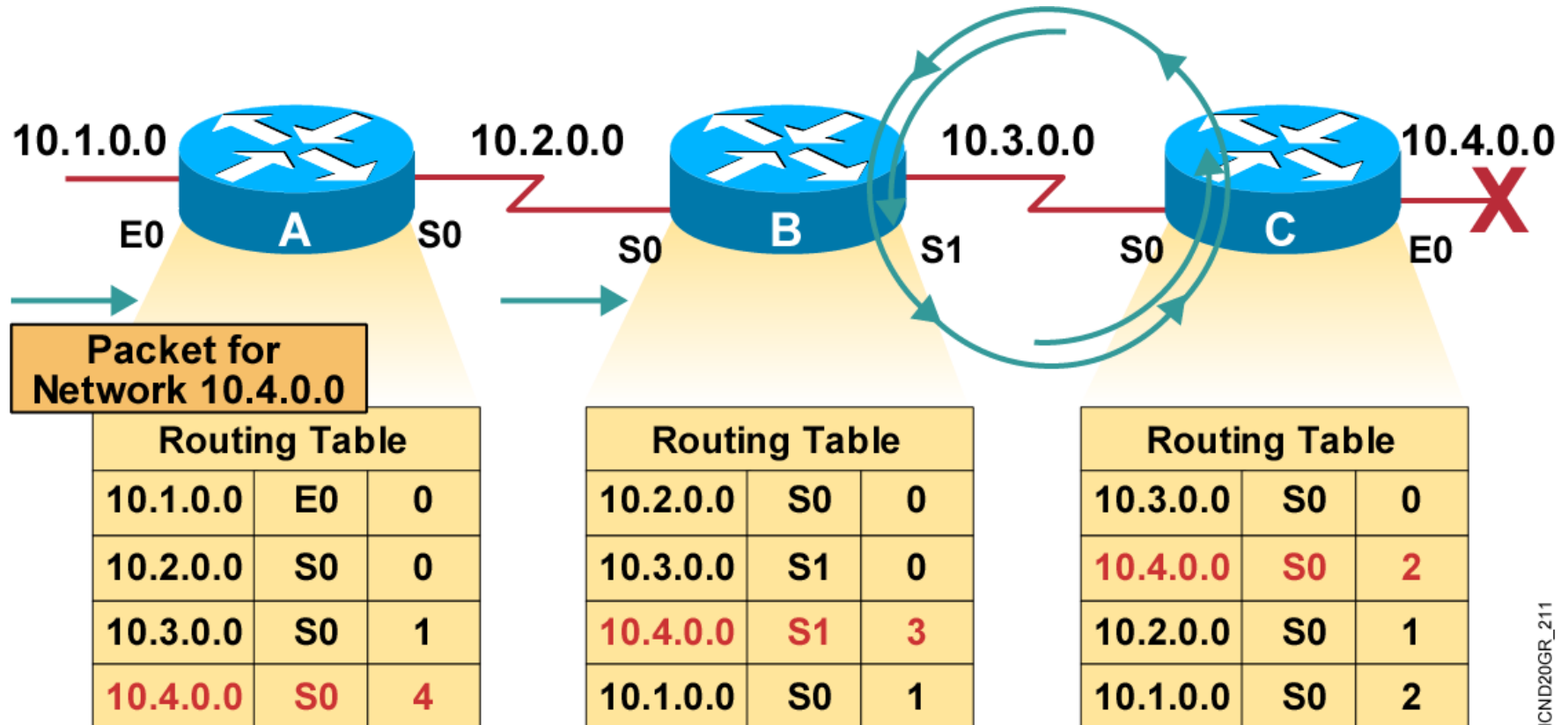


ICND20GR\_209

- ❑ Quá trình trao đổi diễn ra liên tục và Router cập nhật thông tin định tuyến sai lệch....Hop Count sẽ tiến đến vô hạn

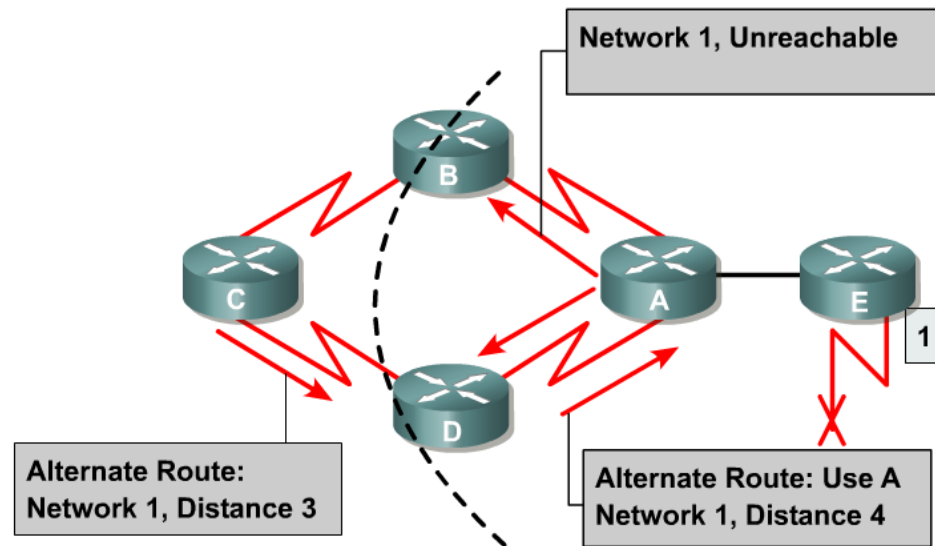
# Routing Loop - 1

## Distance Vector Routing Protocol



ICND20GR\_211

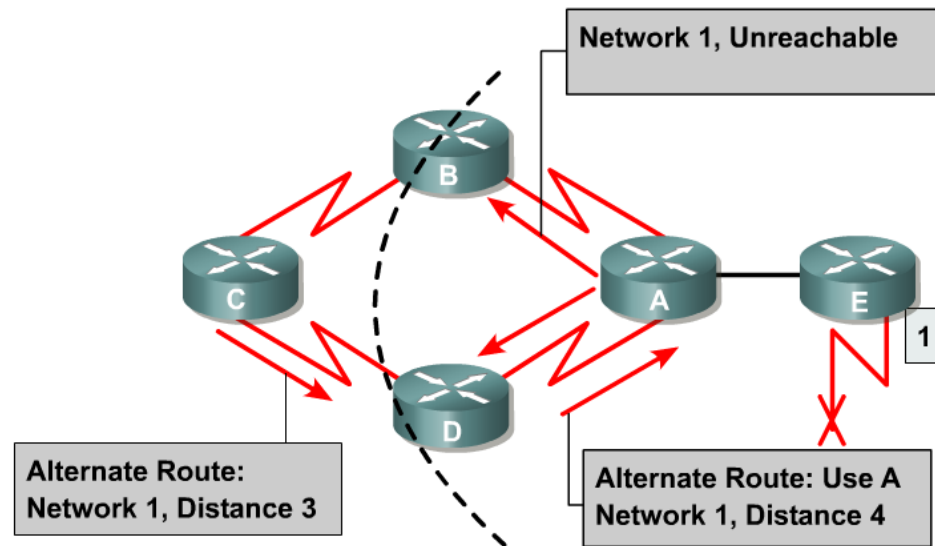
# Routing Loop - 2



## Network 1 : **fail**

- ❑ **Router E** cập nhật thông tin định tuyến và gửi bảng định tuyến cho **Router A** (thông báo Network 1 đã bị fail)
- ❑ **Router A** cập nhật thông tin, nhưng **Router B** và **D** không biết Network 1 đã fail vì chưa được thông báo
- ❑ **Router A** gửi thông tin bảng định tuyến đến **Router B** và **D**. **Router B** và **D** cập nhật thông tin

# Routing Loop - 2

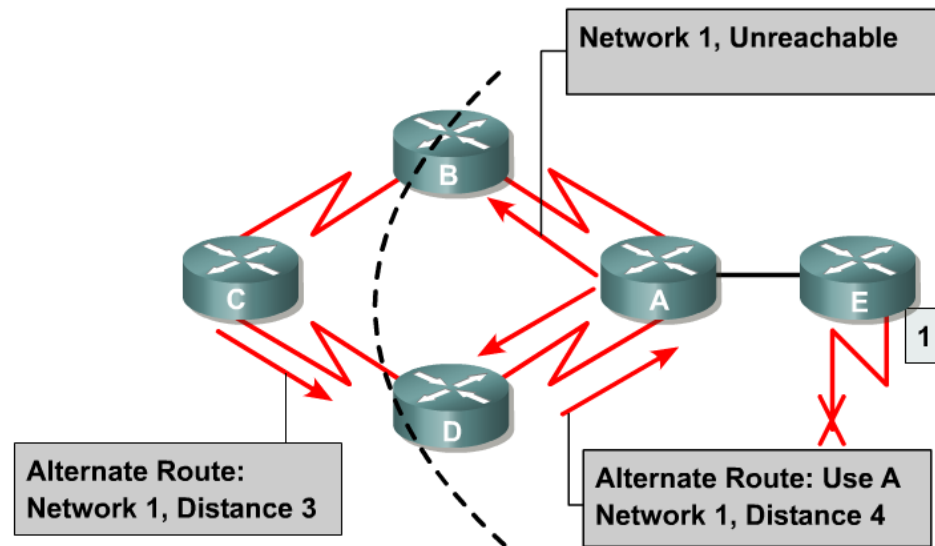


- ❑ **Router C** không biết Network 1 đã bị fail vì chưa được thông báo. Theo Router C, để đến Network 1 thì đi qua Router B và Hop-Count là 3

**Theo định kỳ, Router C gửi bảng định tuyến đến Router D**

- ❑ **Router C** muốn thông báo với **Router D**, muốn đi tới Network 1 thì đi qua **Router C** và Hop-Count là 4

# Routing Loop - 2

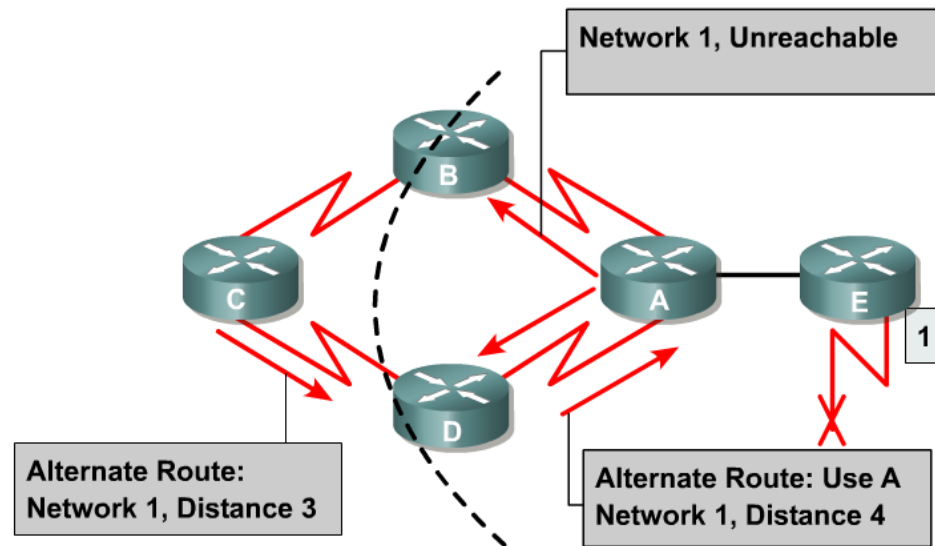


Thông tin định tuyến tại **Router D**

- ❑ **Current Path** đến Network 1 = unreachable (down)
- ❑ Sau khi nhận thông tin từ **Router C**, **current path** đến Network 1 = đi qua **Router C** và **Hop-Count** là 4
- ❑ **Router D** gửi thông tin bảng định tuyến đến **Router A**



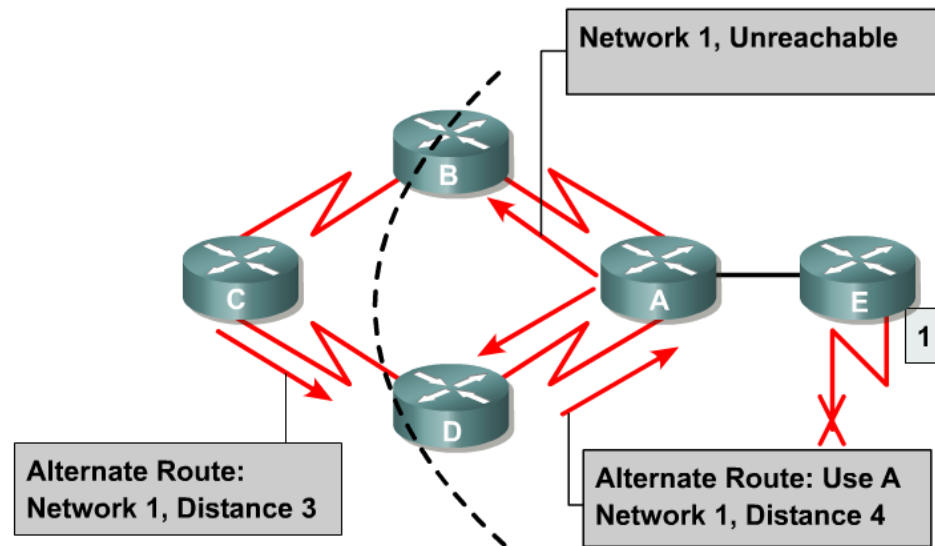
# Routing Loop - 2



Thông tin định tuyến tại **Router A**

- ❑ **Current Path** đến Network 1 = unreachable (down)
- ❑ Sau khi nhận thông tin từ **Router D**, **current path** đến Network 1 = đi qua **Router D** và **Hop-Count** là 5
- ❑ **Router A** gửi thông tin bảng định tuyến đến **Router B** và **E**

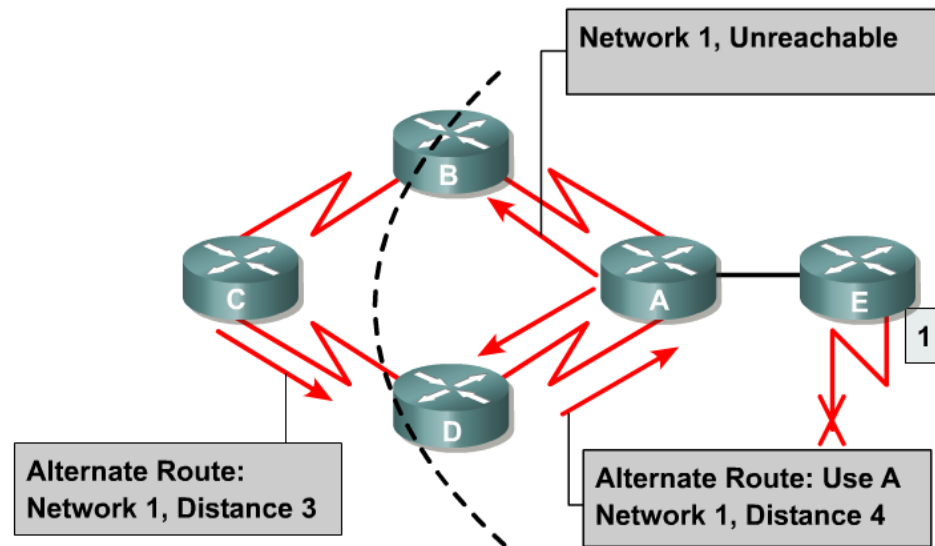
# Routing Loop - 2



Thông tin định tuyến tại **Router B**

- ❑ **Current Path** đến Network 1 = unreachable (down)
- ❑ Sau khi nhận thông tin từ **Router A**, **current path** đến Network 1 = đi qua **Router A** và **Hop-Count** là 6
- ❑ **Router B** gửi thông tin bảng định tuyến đến **Router C**

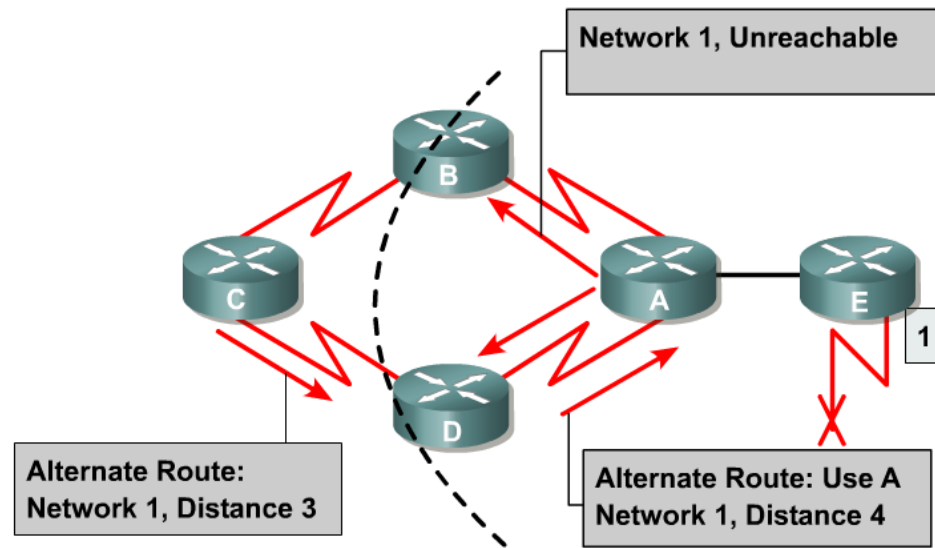
# Routing Loop - 2



Thông tin định tuyến tại **Router C**

- ❑ **Current Path** đến Network 1 đi qua **Router B** và **Hop-Count** là 3
- ❑ Sau khi nhận thông tin từ **Router A**, **current path** đến Network 1 = đi qua **Router B** và **Hop-Count** là 7
- ❑ **Router C** gửi thông tin bảng định tuyến đến **Router D**

# Routing Loop - 2



- ❑ Quá trình cập nhật thông tin về Network 1 sẽ được cập nhật theo chu trình :
  - ❑ Router C → Router D → Router A → Router B → Router C
- ❑ Quá trình Loop xảy ra ở đây

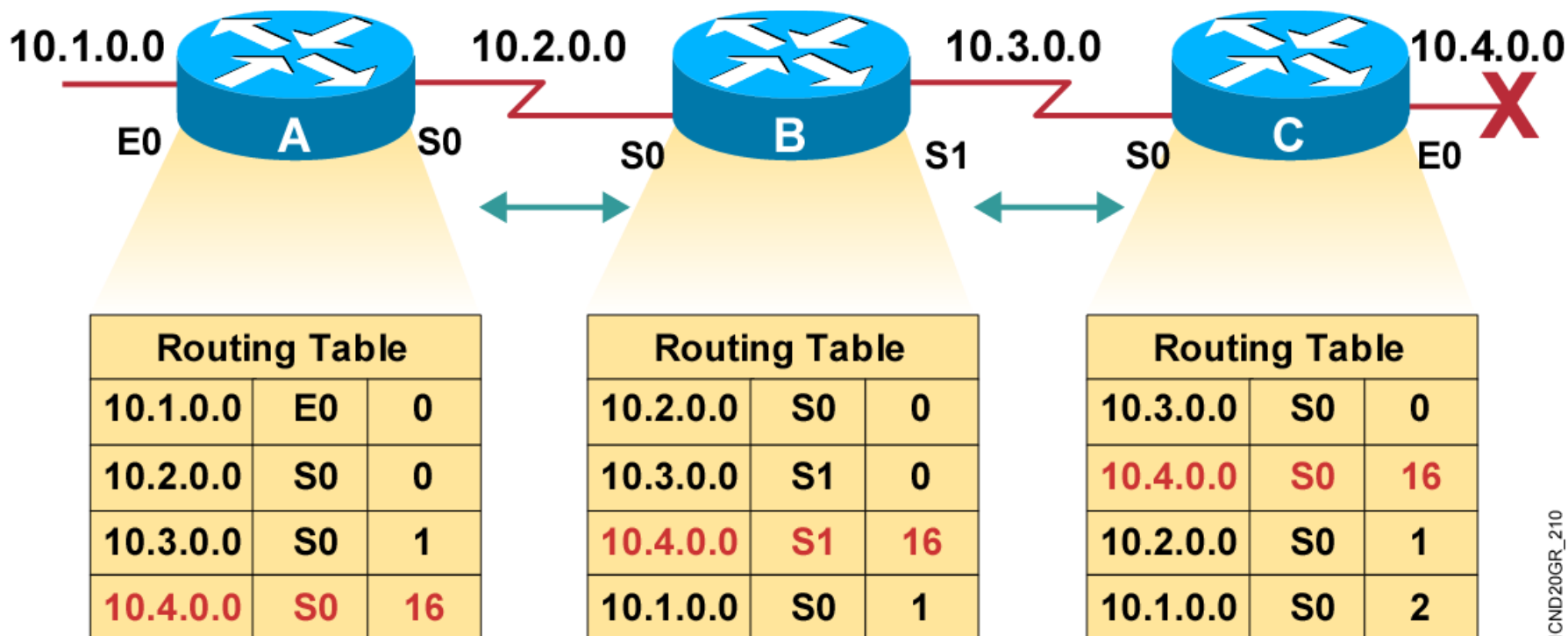
# Routing Loop

## Distance Vector Routing Protocol

- ❑ Các phương pháp khắc phục Loop
  - ❑ Định nghĩa giá trị tối đa cho Hop Count
  - ❑ Hold-down
  - ❑ Split Horizon
  - ❑ Routing Poisoning
  - ❑ Triggered update

# Định nghĩa giá trị tối đa cho Hop Count

Distance Vector Routing Protocol / Routing Loop



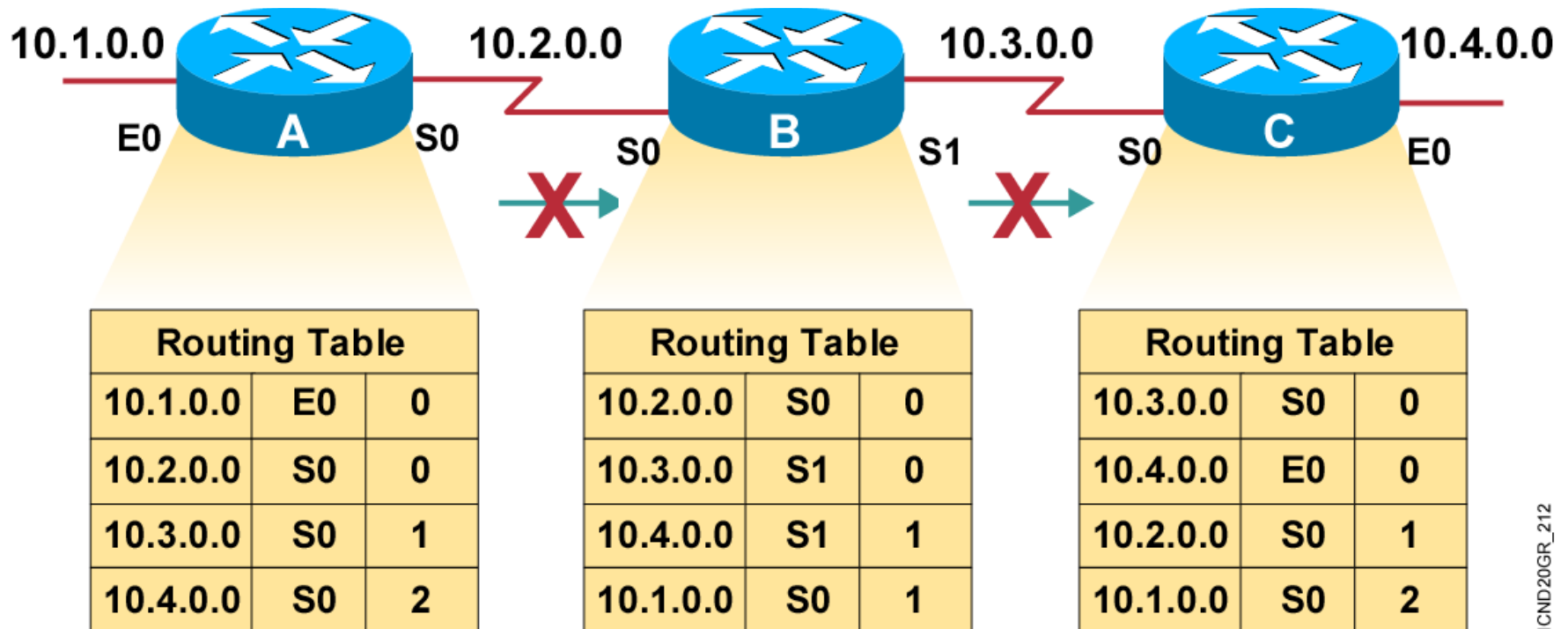
ICND20GR\_210

# Holddown Timer

- ❑ Khi có sự thay đổi về các Network mà Router quản lý, Router sẽ cập nhật thông tin này và gửi thông tin bảng định tuyến của nó cho Router lân cận, đồng thời bật **Timer**
- ❑ **Timer** là khoảng thời gian mà Router sẽ không nhận thông tin định tuyến của các Router láng giềng của nó về Network mà nó vừa thông báo bị thay đổi

# Split Horizon

## Distance Vector Routing Protocol / Routing Loop



- ❑ Router không được gửi thông tin định tuyến ra interface mà nó học thông tin định tuyến từ interface đó





**Routing Table**

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.1.0/24	0	e0
10.1.2.0/24	0	s0

**Routing Table**

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.2.0/24	0	s0
10.1.3.0/24	0	e0

Thông tin bảng định tuyến cũ

**Routing Update**

		Next-hop
<u>Net.</u>	<u>Hops</u>	<u>Address</u>
10.1.1.0/24	1	10.1.1.1
10.1.2.0/24	1	10.1.1.1

**Routing Update**

		Next-hop
<u>Net.</u>	<u>Hops</u>	<u>Address</u>
10.1.2.0/24	1	10.1.2.2
10.1.3.0/24	1	10.1.2.2

Quá trình cập nhật thông tin định tuyến

**Routing Table**

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.1.0/24	0	e0
10.1.2.0/24	0	s0
10.1.3.0/24	1	10.1.2.2

**Routing Table**

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.2.0/24	0	s0
10.1.3.0/24	0	e0
10.1.1.0/24	1	10.1.2.1

Thông tin bảng định tuyến mới



**Routing Table**

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.1.0/24	0	e0
10.1.2.0/24	0	s0
10.1.3.0/24	1	10.1.2.2

**Routing Table**

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.2.0/24	0	s0
10.1.3.0/24	0	e0
10.1.1.0/24	1	10.1.2.1

**Split Horizon  
Disable**

**Routing Update**

<u>Net.</u>	<u>Hops</u>	<u>Next-hop Address</u>
10.1.1.0/24	1	10.1.1.1
10.1.2.0/24	1	10.1.1.1
10.1.3.0/24	2	10.1.1.1

**Routing Update**

<u>Net.</u>	<u>Hops</u>	<u>Next-hop Address</u>
10.1.2.0/24	1	10.1.2.2
10.1.3.0/24	1	10.1.2.2
10.1.1.0/24	2	10.1.2.2

**Routing Table**

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.1.0/24	0	e0
10.1.2.0/24	0	s0
10.1.3.0/24	1	10.1.2.2

**Routing Table**

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.2.0/24	0	s0
10.1.3.0/24	0	e0
10.1.1.0/24	1	10.1.2.1



**Routing Table**

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.1.0/24	0	e0
10.1.2.0/24	0	s0
10.1.3.0/24	1	10.1.2.2

**Routing Table**

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.2.0/24	0	s0
10.1.3.0/24	0	e0
10.1.1.0/24	1	10.1.2.1

**Split Horizon  
Disable**

**Routing Update**

<u>Net.</u>	<u>Hops</u>	<u>Next-hop Address</u>
10.1.1.0/24	1	10.1.1.1
10.1.2.0/24	1	10.1.1.1
10.1.3.0/24	2	10.1.1.1

**Routing Table**

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.1.0/24	0	e0
10.1.2.0/24	0	s0
10.1.3.0/24	1	10.1.2.2

**Routing Table**

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.2.0/24	0	s0
10.1.3.0/24	2	10.1.2.1
10.1.1.0/24	1	10.1.2.1



**Routing Table**

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.1.0/24	0	e0
10.1.2.0/24	0	s0
10.1.3.0/24	1	10.1.2.2

**Routing Table**

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.2.0/24	0	s0
10.1.3.0/24	2	10.1.2.1
10.1.1.0/24	1	10.1.2.1

**Split Horizon  
Disable**

**Routing Update**

Next-hop		
<u>Net.</u>	<u>Hops</u>	<u>Address</u>
10.1.2.0/24	1	10.1.2.2
10.1.3.0/24	3	10.1.2.2
10.1.1.0/24	2	10.1.2.2

**Routing Table**

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.1.0/24	0	e0
10.1.2.0/24	0	s0
10.1.3.0/24	3	10.1.2.2

**Routing Table**

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.2.0/24	0	s0
10.1.3.0/24	2	10.1.2.1
10.1.1.0/24	1	10.1.2.1



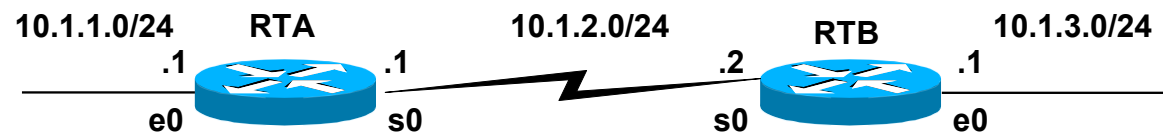
Routing Table

Net.	Hops	Ex-Int
10.1.1.0/24	0	e0
10.1.2.0/24	0	s0
10.1.3.0/24	16	10.1.2.2

Routing Table

Net.	Hops	Ex-Int
10.1.2.0/24	0	s0
10.1.3.0/24	16	10.1.2.1
10.1.1.0/24	1	10.1.2.1

**Split Horizon  
Disable**



Routing Table		
<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.1.0/24	0	e0
10.1.2.0/24	0	s0

Routing Table		
<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.2.0/24	0	s0
10.1.3.0/24	0	e0

**Split Horizon  
Enable**

Routing Update		
<u>Net.</u>	<u>Hops</u>	<u>Next-hop Address</u>
10.1.1.0/24	1	10.1.1.1

Routing Update		
<u>Net.</u>	<u>Hops</u>	<u>Next-hop Address</u>
10.1.3.0/24	1	10.1.2.2

Routing Table		
<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.1.0/24	0	e0
10.1.2.0/24	0	s0
10.1.3.0/24	1	10.1.2.2

Routing Table		
<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.2.0/24	0	s0
10.1.3.0/24	0	e0
10.1.1.0/24	1	10.1.2.1

Routing Update		
<u>Net.</u>	<u>Hops</u>	<u>Next-hop Address</u>
10.1.1.0/24	1	10.1.1.1

Routing Update		
<u>Net.</u>	<u>Hops</u>	<u>Next-hop Address</u>
10.1.3.0/24	1	10.1.2.2



Routing Table

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.1.0/24	0	e0
10.1.2.0/24	0	s0
10.1.3.0/24	1	10.1.2.2

Routing Table

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.2.0/24	0	s0
10.1.3.0/24 (down)	e0	
10.1.1.0/24	1	10.1.2.1

**Split Horizon  
Enable**

Routing Update

Next-hop		
<u>Net.</u>	<u>Hops</u>	<u>Address</u>
10.1.3.0/24	16	10.1.2.2

Routing Table

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.1.0/24	0	e0
10.1.2.0/24	0	s0
10.1.3.0/24 (down)	10.1.2.2	

Routing Table

<u>Net.</u>	<u>Hops</u>	<u>Ex-Int</u>
10.1.2.0/24	0	s0
10.1.3.0/24 (down)	e0	
10.1.1.0/24	1	10.1.2.1

# Split Horizon with Poison Reverse

## Distance Vector Routing Protocol / Routing Loop



Routing Table

Net.	Hops	Ex-Int
10.1.1.0/24	0	e0
10.1.2.0/24	0	s0
10.1.3.0/24	1	10.1.2.2

Routing Table

Net.	Hops	Ex-Int
10.1.2.0/24	0	s0
10.1.3.0/24	0	e0
10.1.1.0/24	1	10.1.2.1

Routing Update

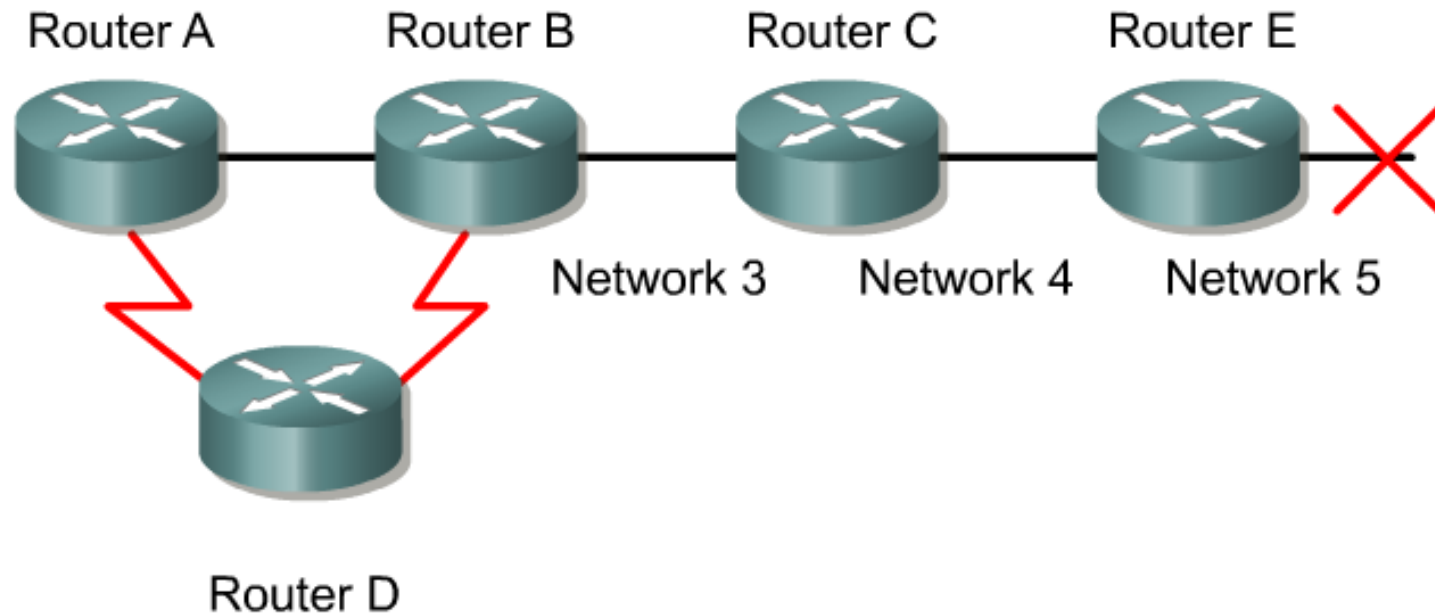
Net.	Hops	Next-hop Address
10.1.1.0/24	1	10.1.1.1
10.1.2.0/24	16	10.1.2.1
10.1.3.0/24	16	10.1.2.1

Routing Update

Net.	Hops	Next-hop Address
10.1.3.0/24	1	10.1.2.2
10.1.2.0/24	16	10.1.2.2
10.1.1.0/24	16	10.1.2.2

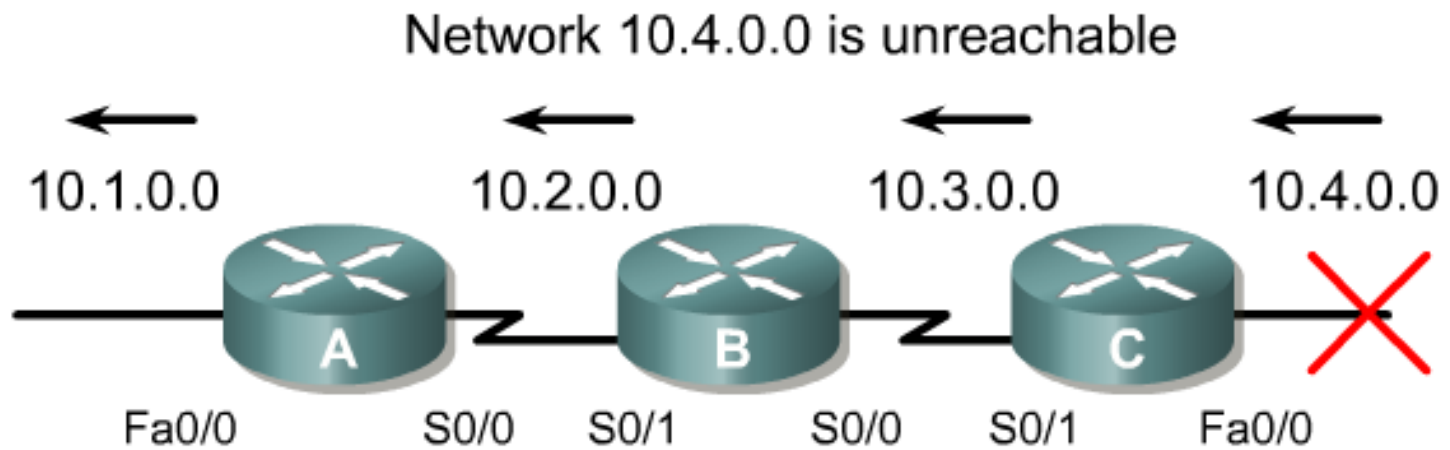


# Routing Poisoning



When Network 5 goes down, Router E initiates route poisoning by entering a table entry metric of 16 (unreachable).

# Triggered Update



# Nội Dung

- ❑ Định tuyến là gì ?
- ❑ Phân loại Định Tuyến
- ❑ Static Routing và Defalt Routing
- ❑ Routing Protocol
- ❑ Distance Vector Routing Protocol
- ❑ **Link-State Routing Protocol**

# Link-State Routing Protocol

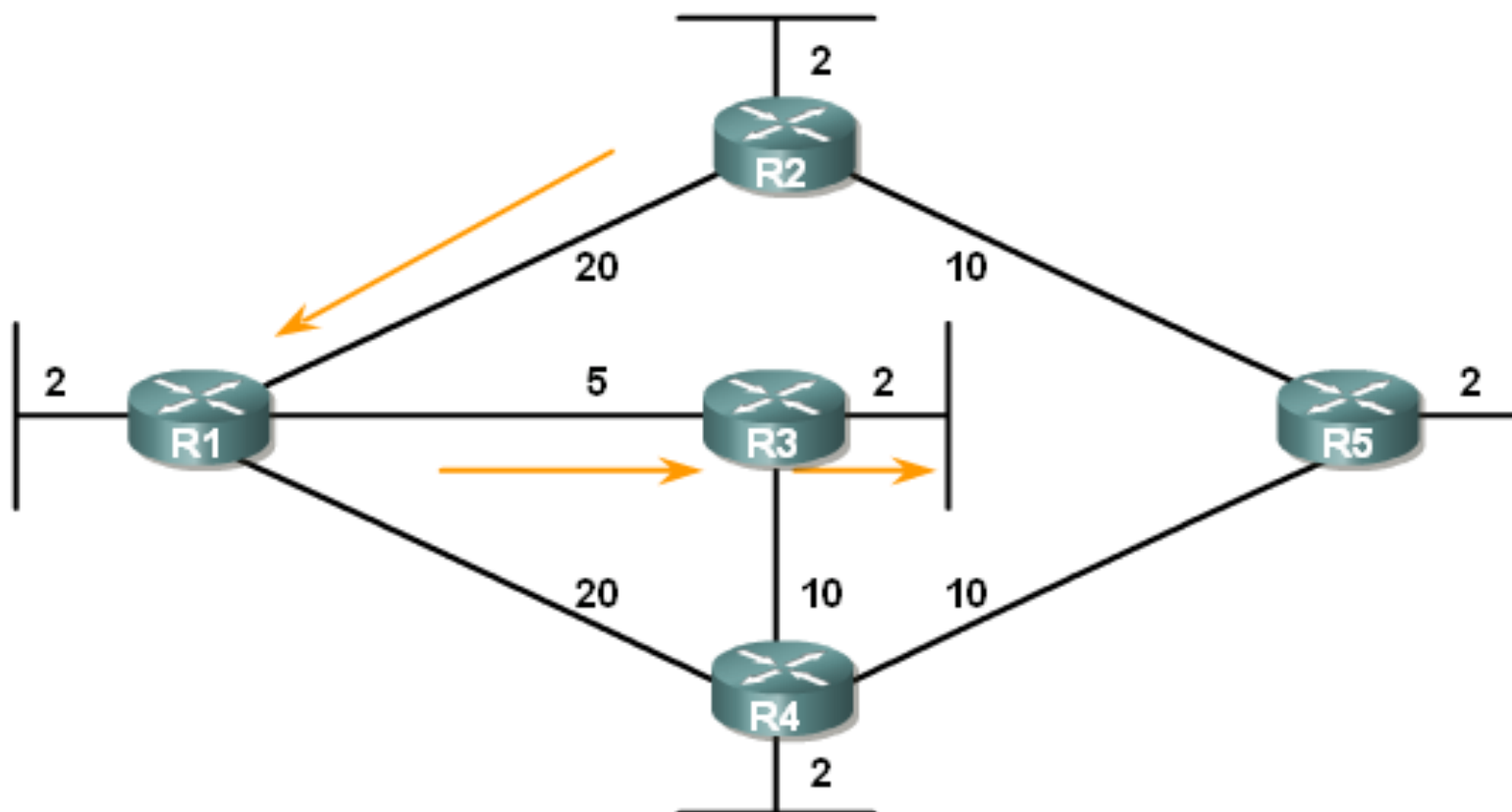
## ❑ Link-State Routing Protocol mô tả cách thức các Router :

- ❑ Tạo bảng định tuyến
- ❑ Cập nhật bảng định tuyến
- ❑ Trao đổi **thông tin định tuyến** với tất cả Router

## ❑ Những đặc điểm chính :

- ❑ Cập nhật thông tin định tuyến khi có sự thay đổi về cấu trúc mạng
- ❑ Trao đổi thông tin định tuyến với tất cả Router
- ❑ Dựa vào Thuật toán Dijkstra và thông tin định tuyến để xác định đường đi ngắn nhất

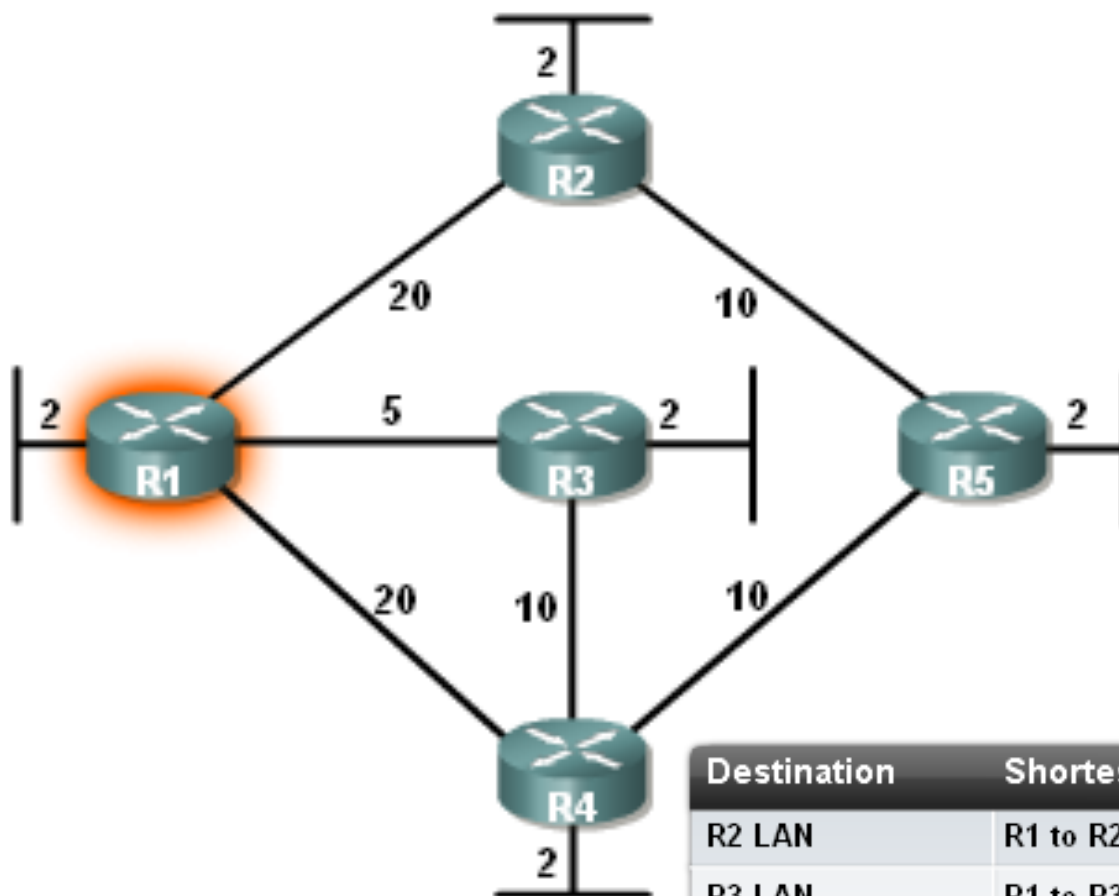
# Link-State Routing Protocol



Shortest Path for host on R2 LAN to reach host on R3 LAN:  
 $R2 \text{ to } R1 (20) + R1 \text{ to } R3 (5) + R3 \text{ to LAN } (2) = 27$

# Shortest Path First (SPF)

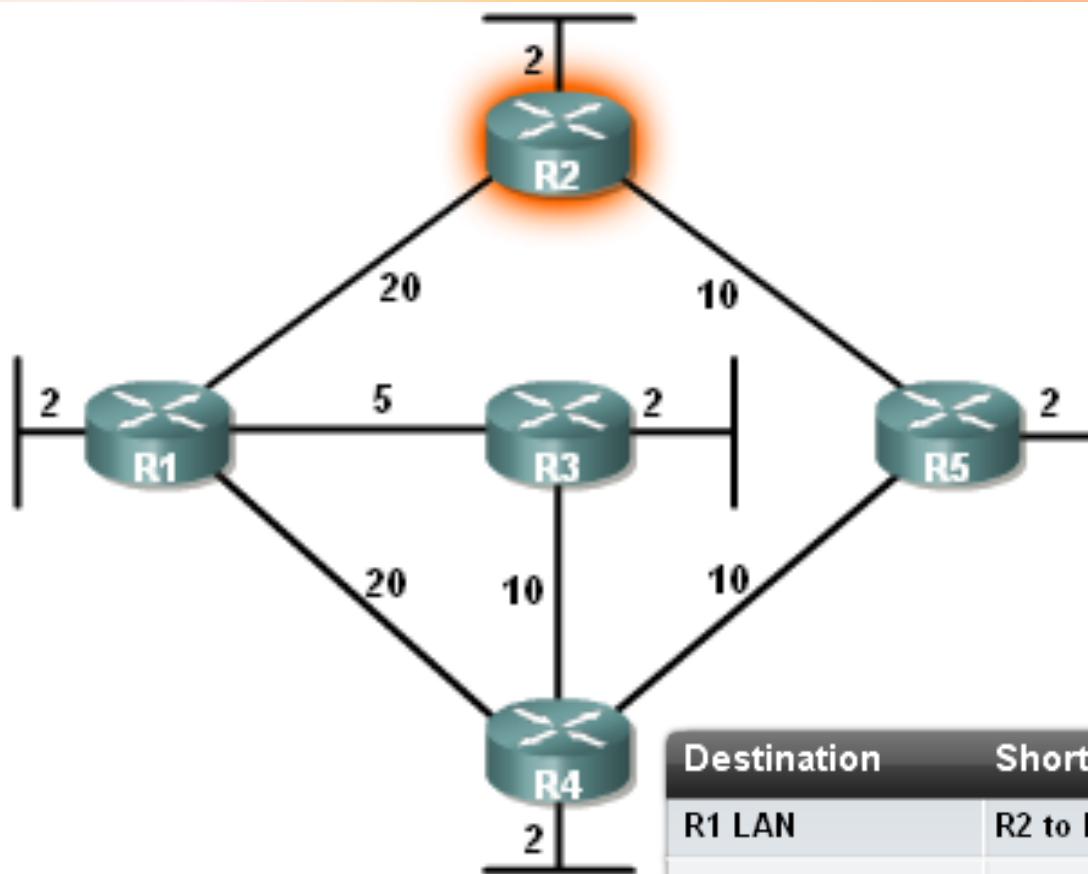
## Link-State Routing Protocol



Destination	Shortest Path	Cost
R2 LAN	R1 to R2	22
R3 LAN	R1 to R3	7
R4 LAN	R1 to R3 to R4	17
R5 LAN	R1 to R3 to R4 to R5	27

# Shortest Path First (SPF)

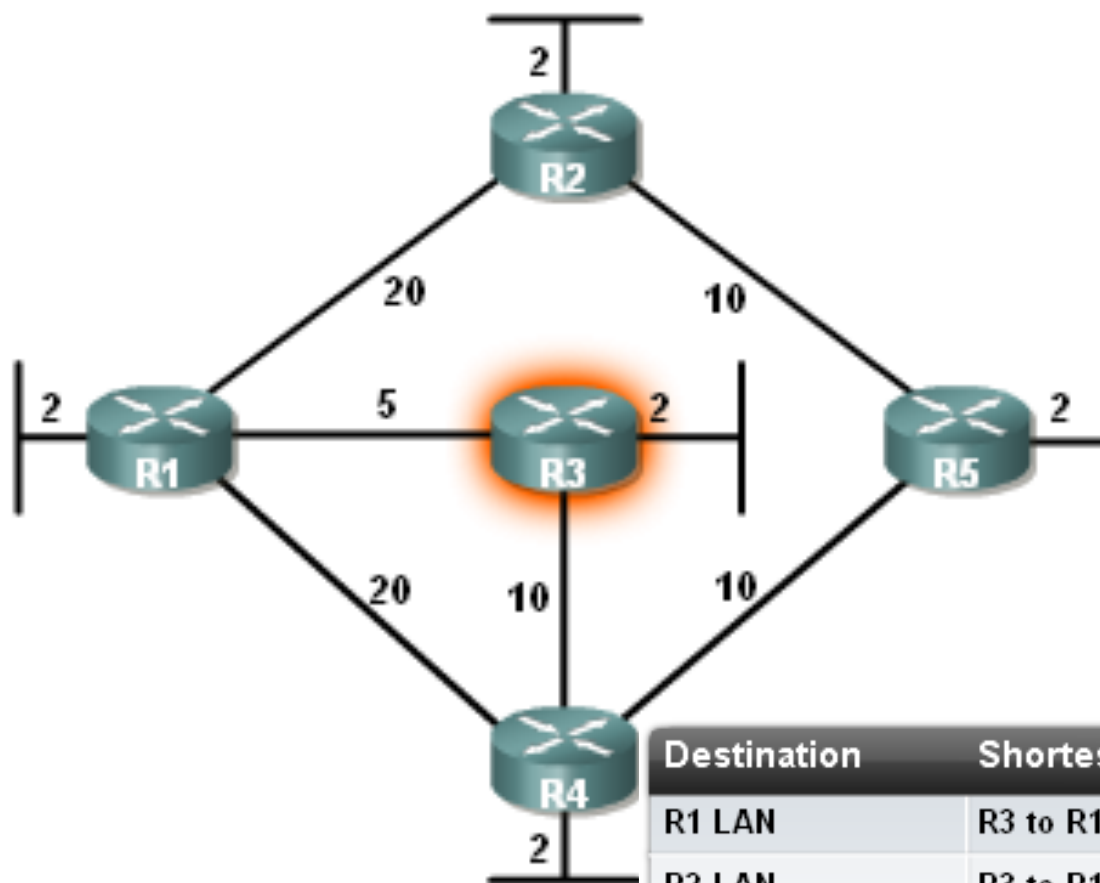
## Link-State Routing Protocol



Destination	Shortest Path	Cost
R1 LAN	R2 to R1	22
R3 LAN	R2 to R1 to R3	27
R4 LAN	R2 to R5 to R4	22
R5 LAN	R2 to R5	12

# Shortest Path First (SPF)

## Link-State Routing Protocol



Destination	Shortest Path	Cost
R1 LAN	R3 to R1	7
R2 LAN	R3 to R1 to R2	27
R4 LAN	R3 to R4	12
R5 LAN	R3 to R4 to R5	22



# Quy trình xử lý của Link-State Routing Protocol

## Link-State Routing Protocol

- 1) Mỗi Router sẽ “học” những Network kết nối trực tiếp với nó
- 2) Mỗi Router sẽ gửi gói tin “Hello” đến những Router kết nối trực tiếp với nó
- 3) Mỗi Router sẽ xây dựng Link-State Packet (LSP) chứa đựng trạng thái của mỗi Link kết nối trực tiếp với nó.
  - 1) Neighbor ID, Link Type và Bandwidth
- 4) Mỗi Router sẽ gửi LSP đến tất cả các Router
- 5) Mỗi Router dựa vào cơ sở dữ liệu nhận được từ Router lân cận để xây dựng cấu trúc của mạng và tìm đường đi ngắn nhất dựa trên cấu trúc mạng vừa xây dựng

# Quy trình xử lý của Link-State Routing Protocol

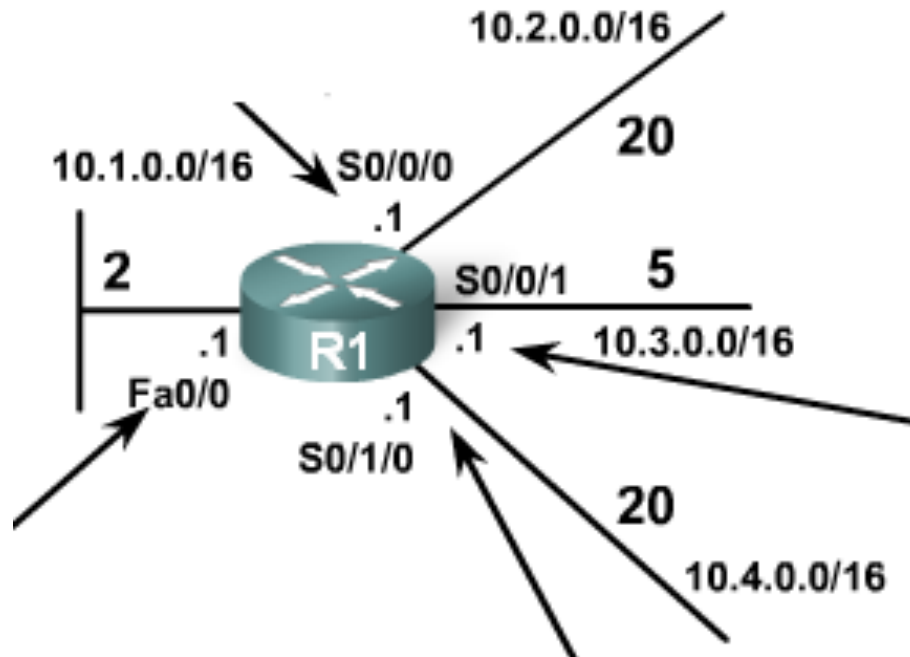
## Link-State Routing Protocol

- 1) **Mỗi Router sẽ “học” những Network kết nối trực tiếp với nó**
- 2) Mỗi Router sẽ gửi gói tin “Hello” đến những Router kết nối trực tiếp với nó
- 3) Mỗi Router sẽ xây dựng Link-State Packet (LSP) chứa đựng trạng thái của mỗi Link kết nối trực tiếp với nó.
  - 1) Neighbor ID, Link Type và Bandwidth
- 4) Mỗi Router sẽ gửi LSP đến tất cả các Router lân cận
- 5) Mỗi Router dựa vào cơ sở dữ liệu nhận được từ Router lân cận để xây dựng cấu trúc của mạng và tìm đường đi ngắn nhất dựa trên cấu trúc mạng vừa xây dựng

# Link và Link-State

❑ **Link** : là interface của Router. Interface này cần phải được cấu hình :

- ❑ Tên của Interface
- ❑ Network
- ❑ Up/Down



❑ Thông tin Link trên R1

- ❑ FastEthernet 0/0 interface on the 10.1.0.0/16 network
- ❑ Serial 0/0/0 network on the 10.2.0.0/16 network
- ❑ Serial 0/0/1 network on the 10.3.0.0/16 network
- ❑ Serial 0/0/2 network on the 10.4.0.0/16 network.

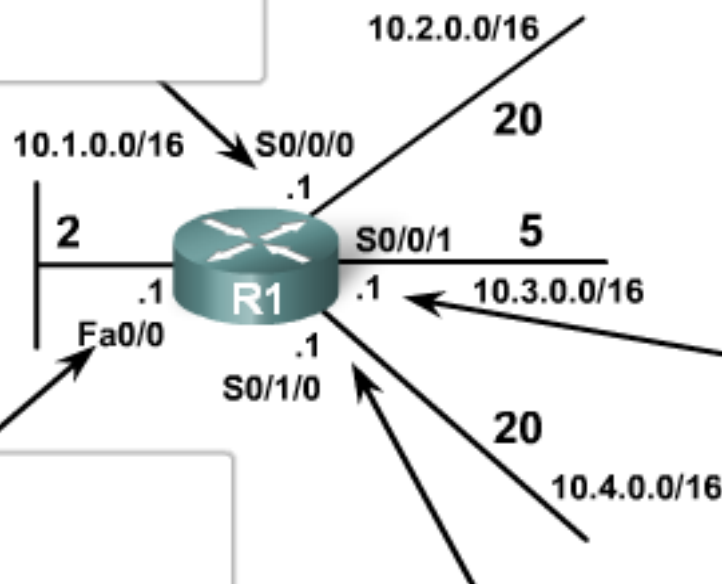
# Link và Link-State

- ❑ **Link-State** : những thông tin trạng thái của Link
  - ❑ IP Address của Interface và Subnet mark
  - ❑ Loại Network : Ethernet , Serial
  - ❑ Cost của Link
  - ❑ Tên Router láng giềng kết nối thông qua Link này

# Link và Link-State

## Link 2:

- Network 10.2.0.0/16
- IP address 10.2.0.1
- Type of network: Serial
- Cost of that link: 20
- Neighbors: R2



## Link 3:

- Network 10.3.0.0/16
- IP address 10.3.0.1
- Type of network: Serial
- Cost of that link: 5
- Neighbors: R3

## Link 1:

- Network 10.1.0.0/16
- IP address 10.1.0.1
- Type of network: Ethernet
- Cost of that link: 2
- Neighbors: none

## Link 4:

- Network 10.4.0.0/16
- IP address 10.4.0.1
- Type of network: Serial
- Cost of that link: 20
- Neighbors: R4

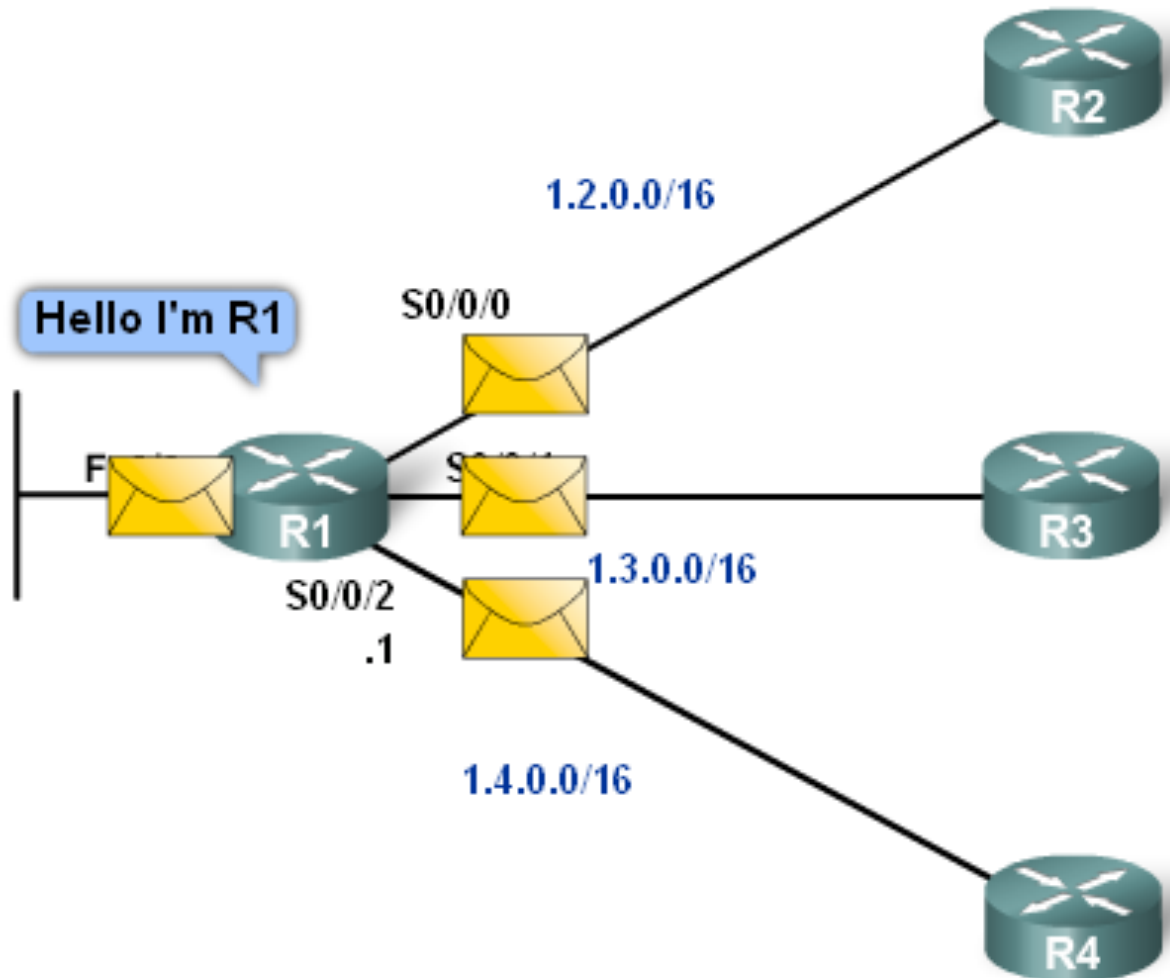
# Quy trình xử lý của Link-State Routing Protocol

## Link-State Routing Protocol

- 1) Mỗi Router sẽ “học” những Network kết nối trực tiếp với nó
- 2) **Mỗi Router sẽ gửi gói tin “Hello” đến những Router kết nối trực tiếp với nó**
- 3) Mỗi Router sẽ xây dựng Link-State Packet (LSP) chứa đựng trạng thái của mỗi Link kết nối trực tiếp với nó.
  - 1) Neighbor ID, Link Type và Bandwidth
- 4) Mỗi Router sẽ gửi LSP đến tất cả các Router lân cận
- 5) Mỗi Router dựa vào cơ sở dữ liệu nhận được từ Router lân cận để xây dựng cấu trúc của mạng và tìm đường đi ngắn nhất dựa trên cấu trúc mạng vừa xây dựng

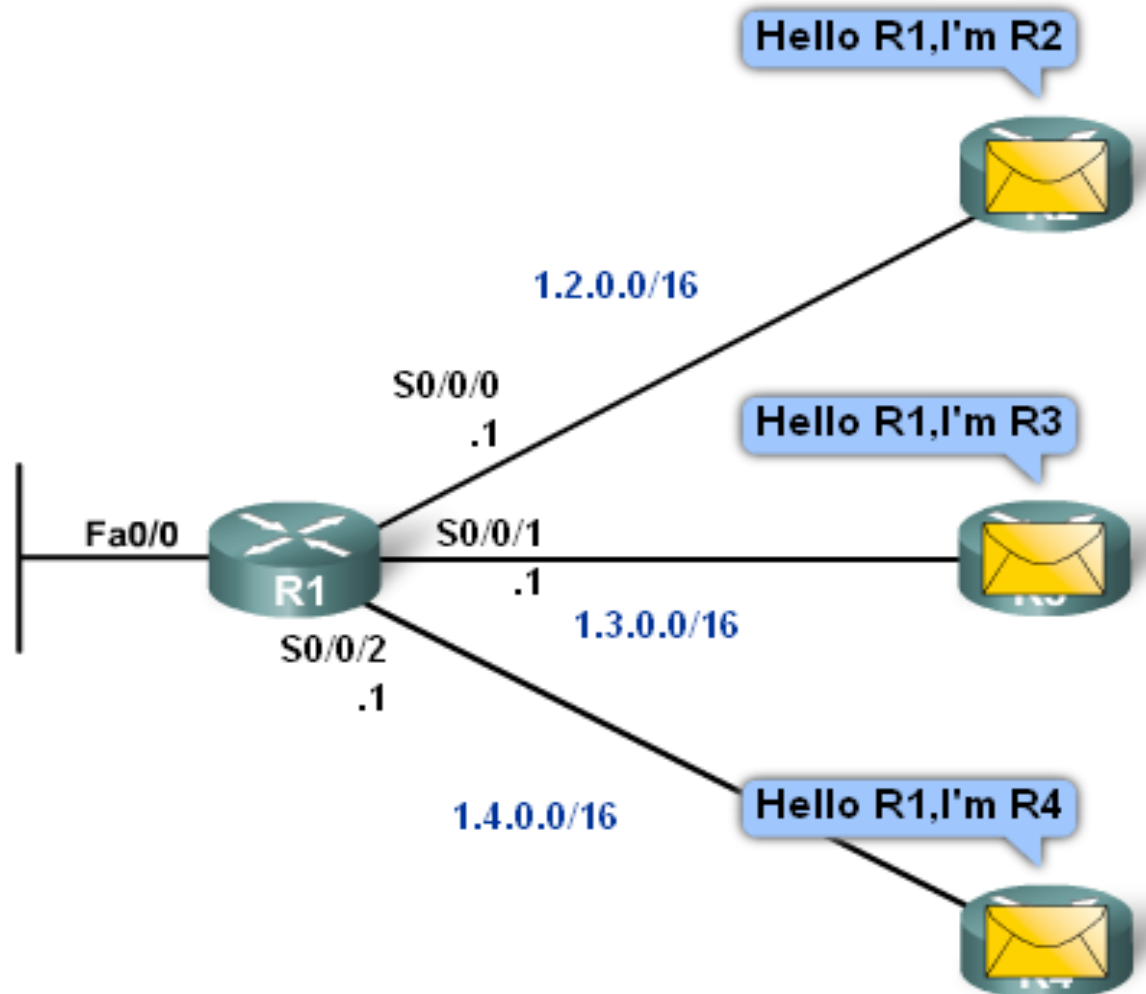
# Hello Packet - Request

- Gửi gói tin “Hello” đến các Router lân cận để thu thập thông



# Hello Packet - Reply

- ❑ Router láng giềng trả lời cho Router 1





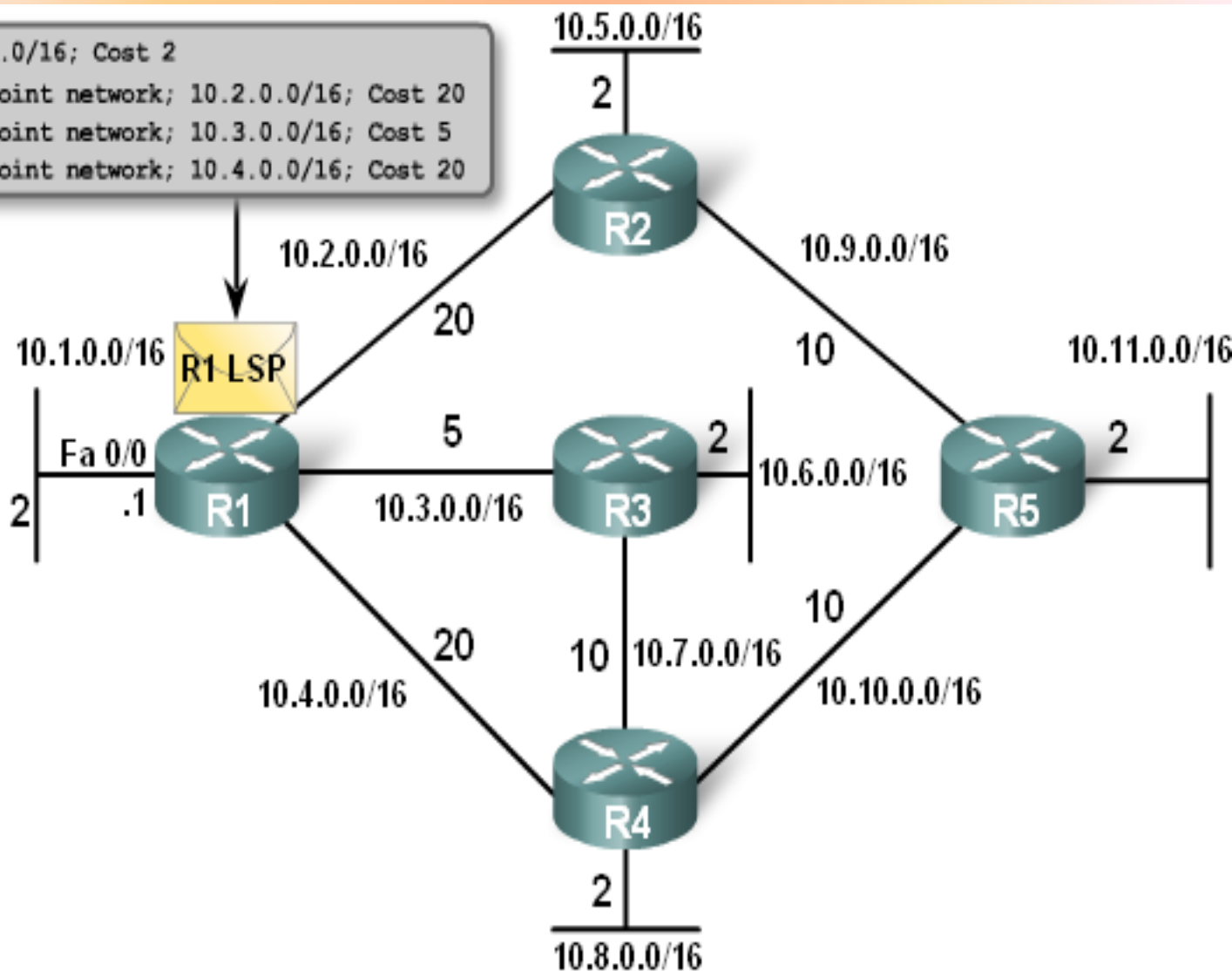
# Quy trình xử lý của Link-State Routing Protocol

## Link-State Routing Protocol

- 1) Mỗi Router sẽ “học” những Network kết nối trực tiếp với nó
- 2) Mỗi Router sẽ gửi gói tin “Hello” đến những Router kết nối trực tiếp với nó
- 3) **Mỗi Router sẽ xây dựng Link-State Packet (LSP) chứa đựng trạng thái của mỗi Link kết nối trực tiếp với nó.**
  - 1) Neighbor ID, Link Type và Bandwidth
- 4) Mỗi Router sẽ gửi LSP đến tất cả các Router lân cận
- 5) Mỗi Router dựa vào cơ sở dữ liệu nhận được từ Router lân cận để xây dựng cấu trúc của mạng và tìm đường đi ngắn nhất dựa trên cấu trúc mạng vừa xây dựng

# Xây dựng LSP

1. R1; Ethernet network 10.1.0.0/16; Cost 2
2. R1 -> R2; Serial point-to-point network; 10.2.0.0/16; Cost 20
3. R1 -> R3; Serial point-to-point network; 10.3.0.0/16; Cost 5
4. R1 -> R4; Serial point-to-point network; 10.4.0.0/16; Cost 20



# Quy trình xử lý của Link-State Routing Protocol

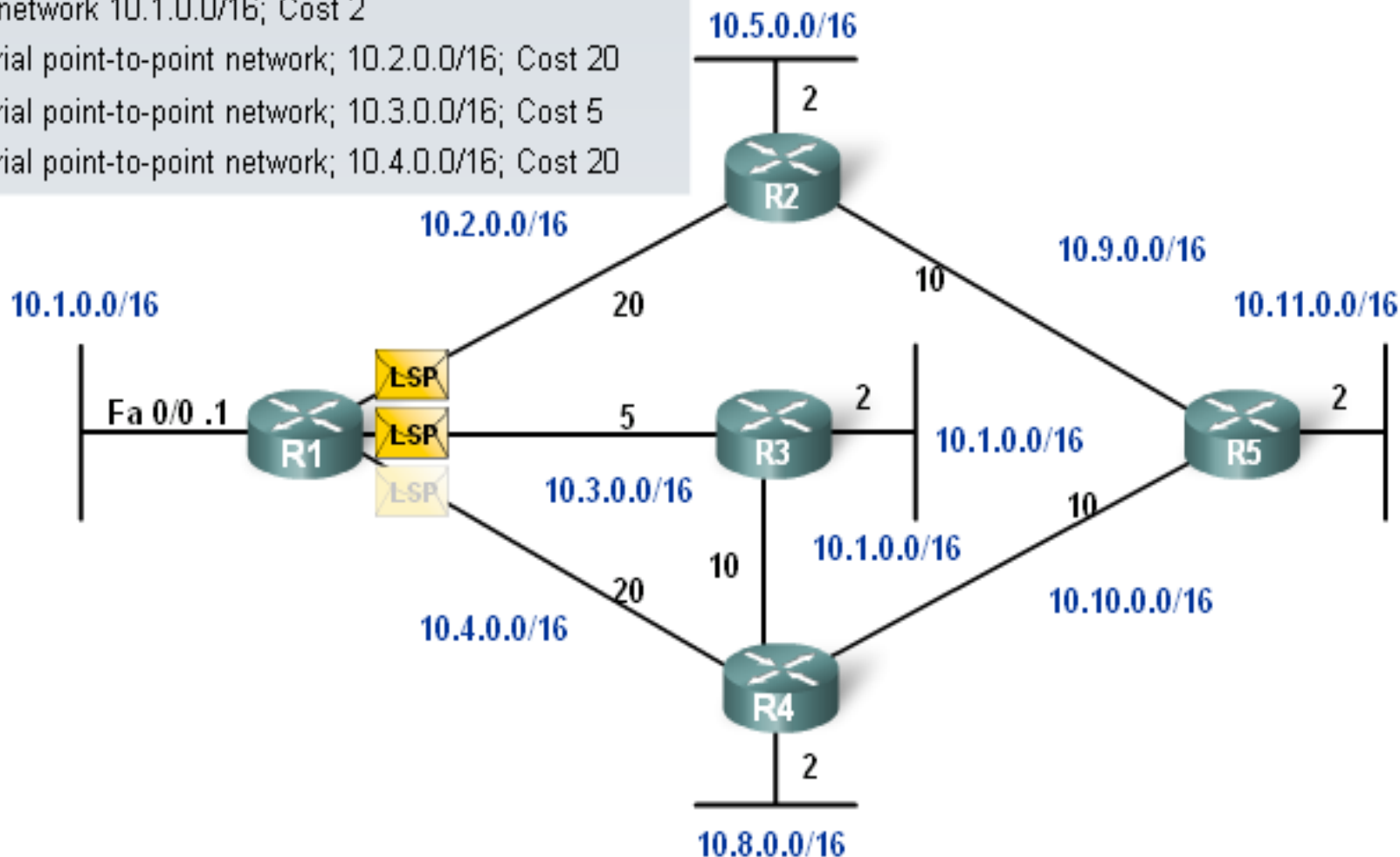
## Link-State Routing Protocol

- 1) Mỗi Router sẽ “học” những Network kết nối trực tiếp với nó
- 2) Mỗi Router sẽ gửi gói tin “Hello” đến những Router kết nối trực tiếp với nó
- 3) Mỗi Router sẽ xây dựng Link-State Packet (LSP) chứa đựng trạng thái của mỗi Link kết nối trực tiếp với nó.
  - 1) Neighbor ID, Link Type và Bandwidth
- 4) Mỗi Router sẽ gửi LSP đến tất cả các Router**
- 5) Mỗi Router dựa vào cơ sở dữ liệu nhận được từ Router lân cận để xây dựng cấu trúc của mạng và tìm đường đi ngắn nhất dựa trên cấu trúc mạng vừa xây dựng

# Quảng bá LSP

## R1 Link State Contents

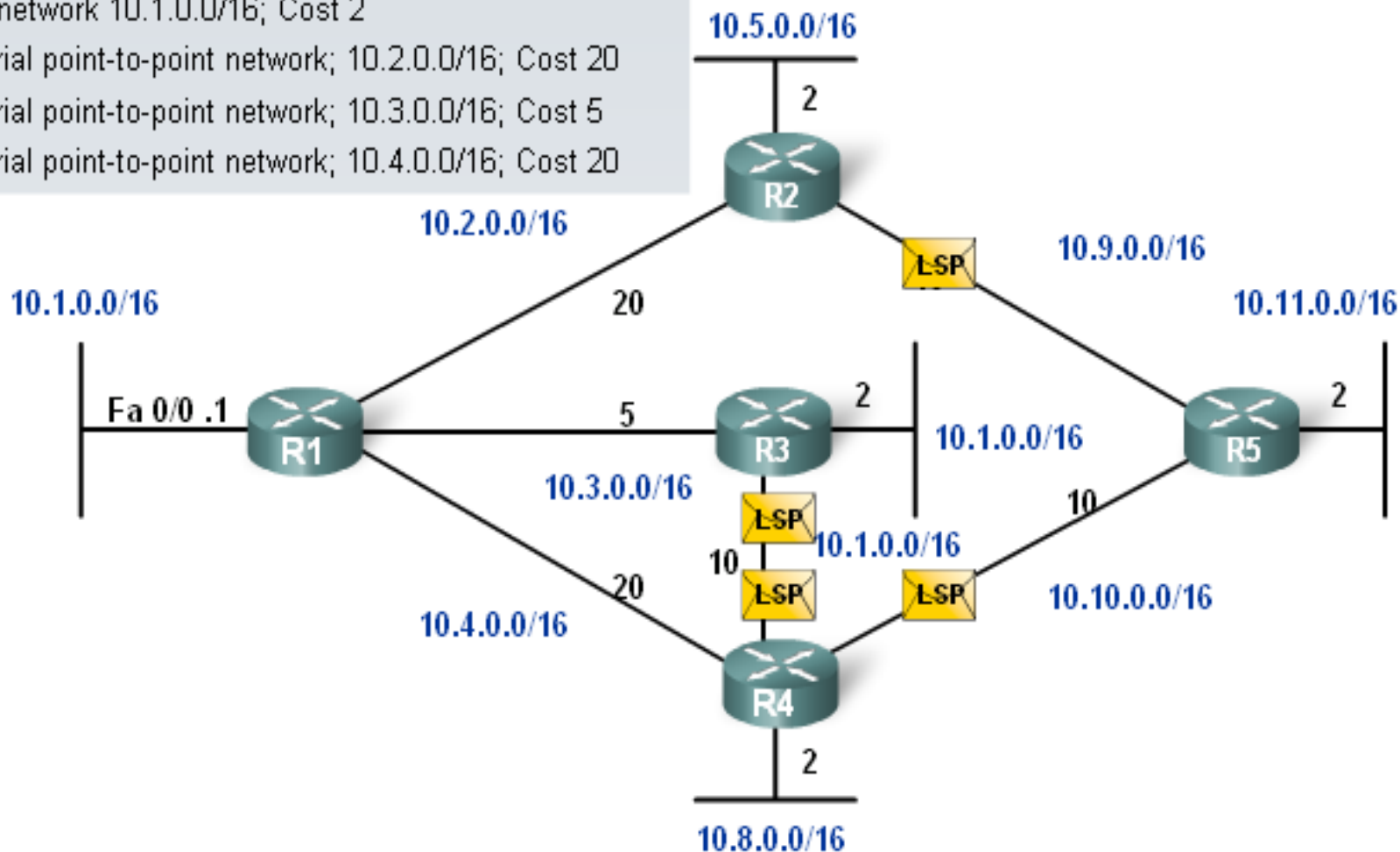
- R1; Ethernet network 10.1.0.0/16; Cost 2
- R1 -> R2; Serial point-to-point network; 10.2.0.0/16; Cost 20
- R1 -> R3; Serial point-to-point network; 10.3.0.0/16; Cost 5
- R1 -> R4; Serial point-to-point network; 10.4.0.0/16; Cost 20



# Quảng bá LSP - tt

## R1 Link State Contents

- R1; Ethernet network 10.1.0.0/16; Cost 2
- R1 -> R2; Serial point-to-point network; 10.2.0.0/16; Cost 20
- R1 -> R3; Serial point-to-point network; 10.3.0.0/16; Cost 5
- R1 -> R4; Serial point-to-point network; 10.4.0.0/16; Cost 20

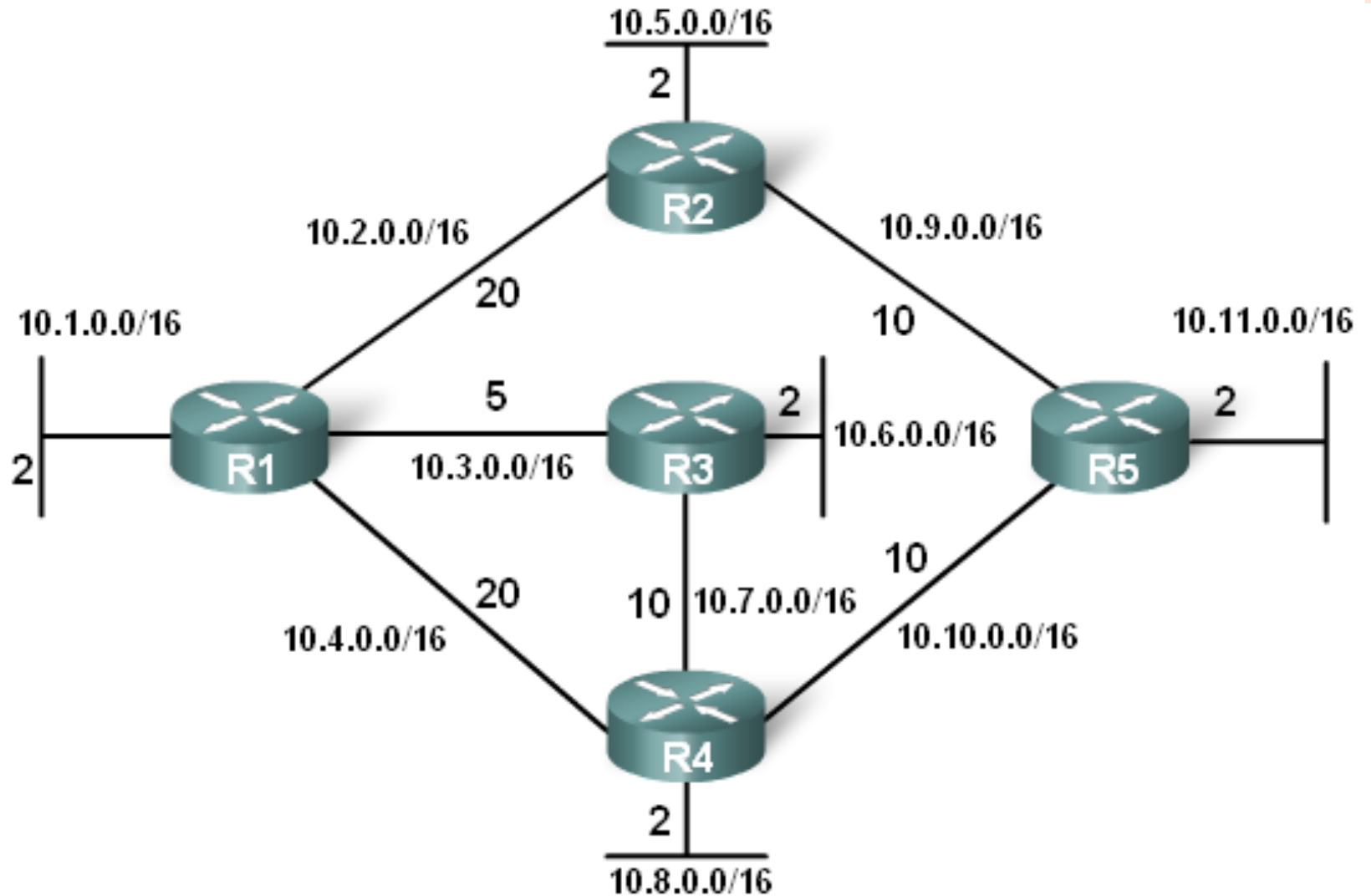


# Quy trình xử lý của Link-State Routing Protocol

## Link-State Routing Protocol

- 1) Mỗi Router sẽ “học” những Network kết nối trực tiếp với nó
- 2) Mỗi Router sẽ gửi gói tin “Hello” đến những Router kết nối trực tiếp với nó
- 3) Mỗi Router sẽ xây dựng Link-State Packet (LSP) chứa đựng trạng thái của mỗi Link kết nối trực tiếp với nó.
  - 1) Neighbor ID, Link Type và Bandwidth
- 4) Mỗi Router sẽ gửi LSP đến tất cả các Router lân cận
- 5) **Mỗi Router dựa vào cơ sở dữ liệu nhận được từ Router lân cận để xây dựng cấu trúc của mạng và tìm đường đi ngắn nhất dựa trên cấu trúc mạng vừa xây dựng**

# Sơ đồ mạng

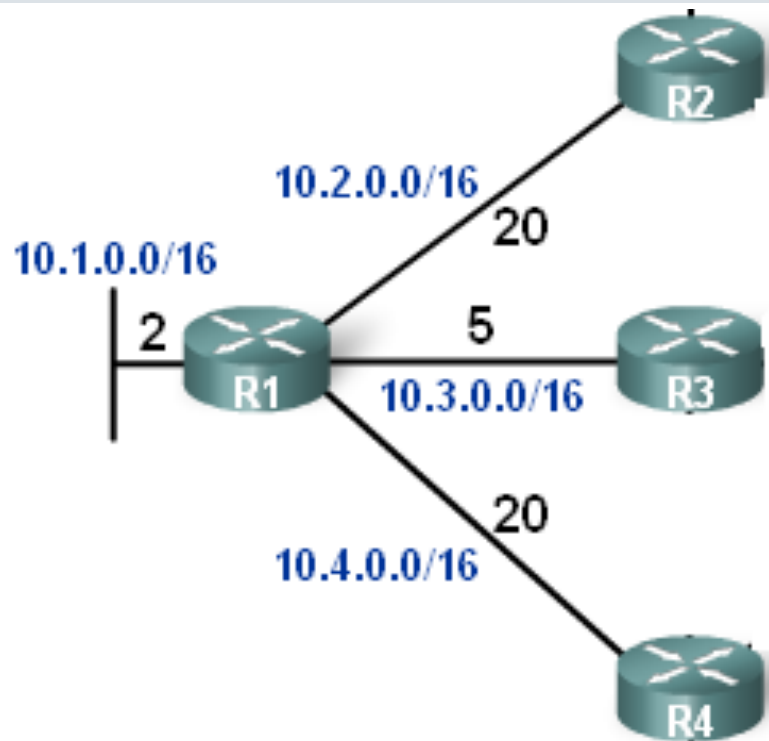


# Quá trình xây dựng cấu trúc mạng

## R1s Link State Database

### R1 Links-states:

- Connected to neighbor R2 on network 10.2.0.0/16, cost of 20
- Connected to neighbor R3 on network 10.3.0.0/16, cost of 5
- Connected to neighbor R4 on network 10.4.0.0/16, cost of 20
- Has a network 10.1.0.0/16, cost of 2

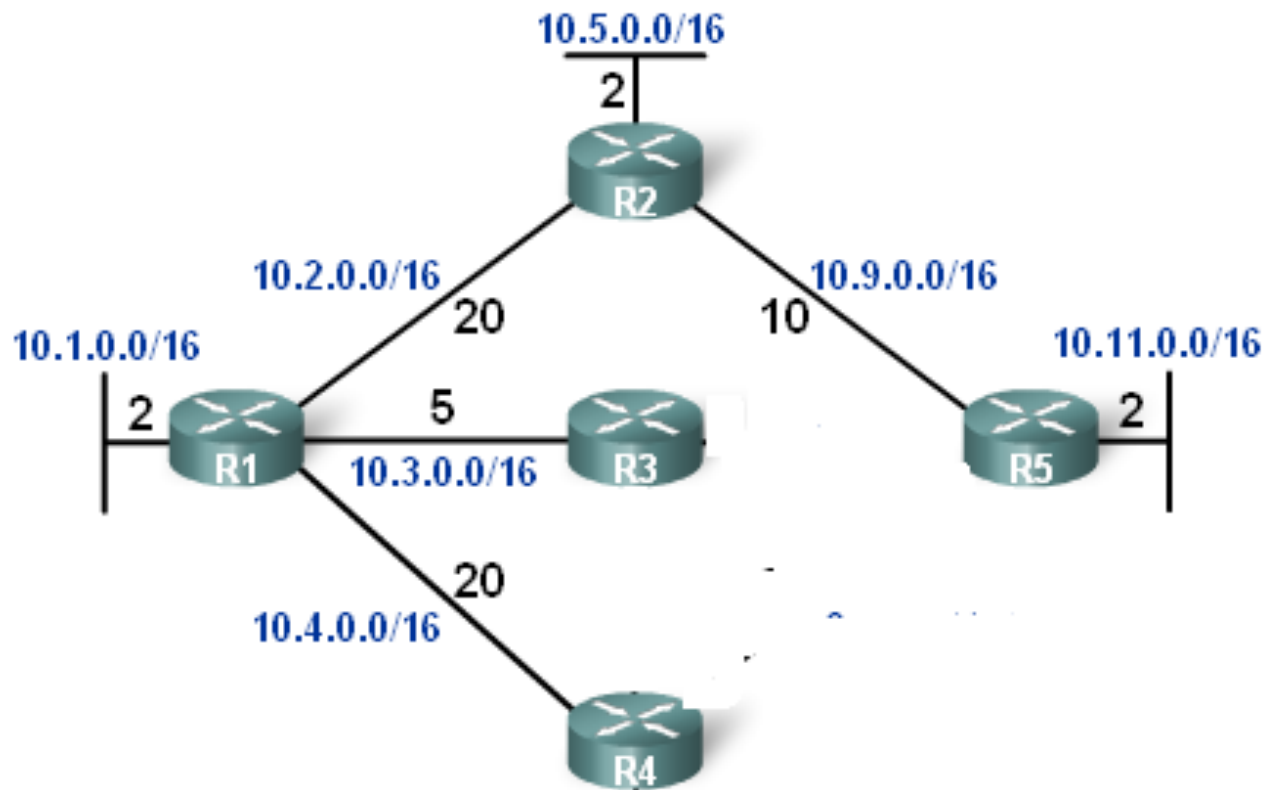




# Quá trình xây dựng cấu trúc mạng

LSPs from R2:

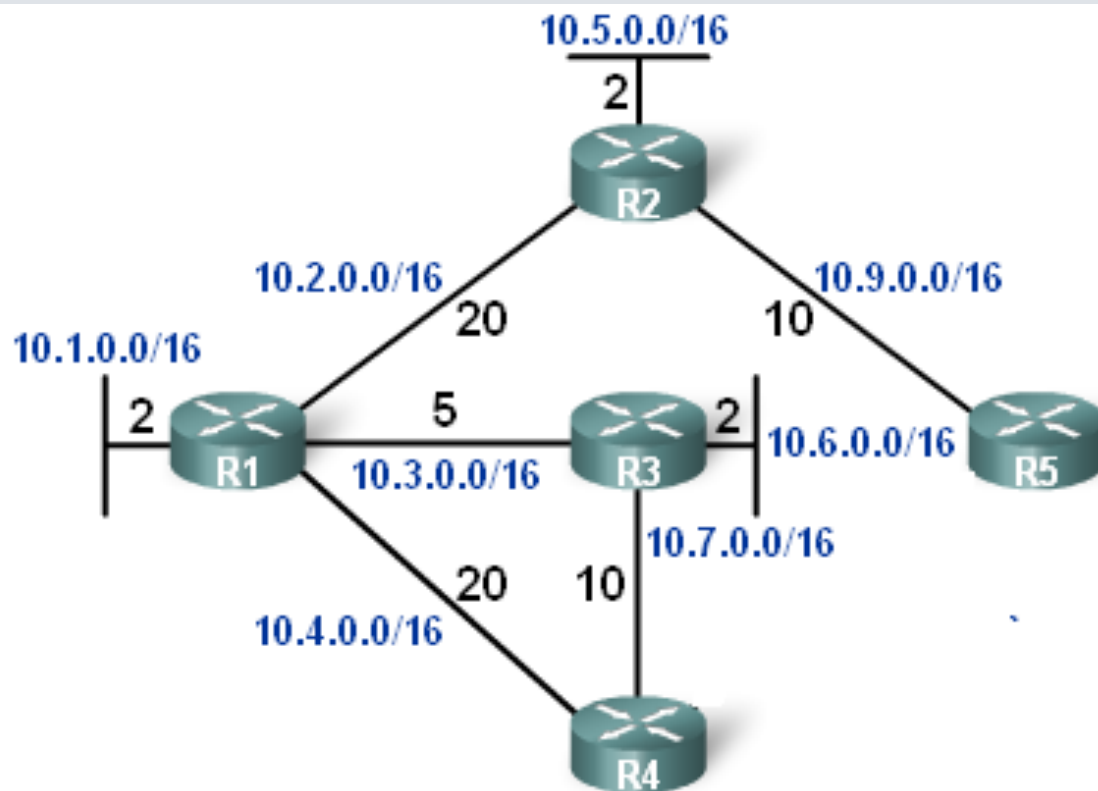
- Connected to neighbor R1 on network 10.2.0.0/16, cost of 20
- Connected to neighbor R5 on network 10.9.0.0/16, cost of 10
- Has a network 10.5.0.0/16, cost of 2



# Quá trình xây dựng cấu trúc mạng

LSPs from R3:

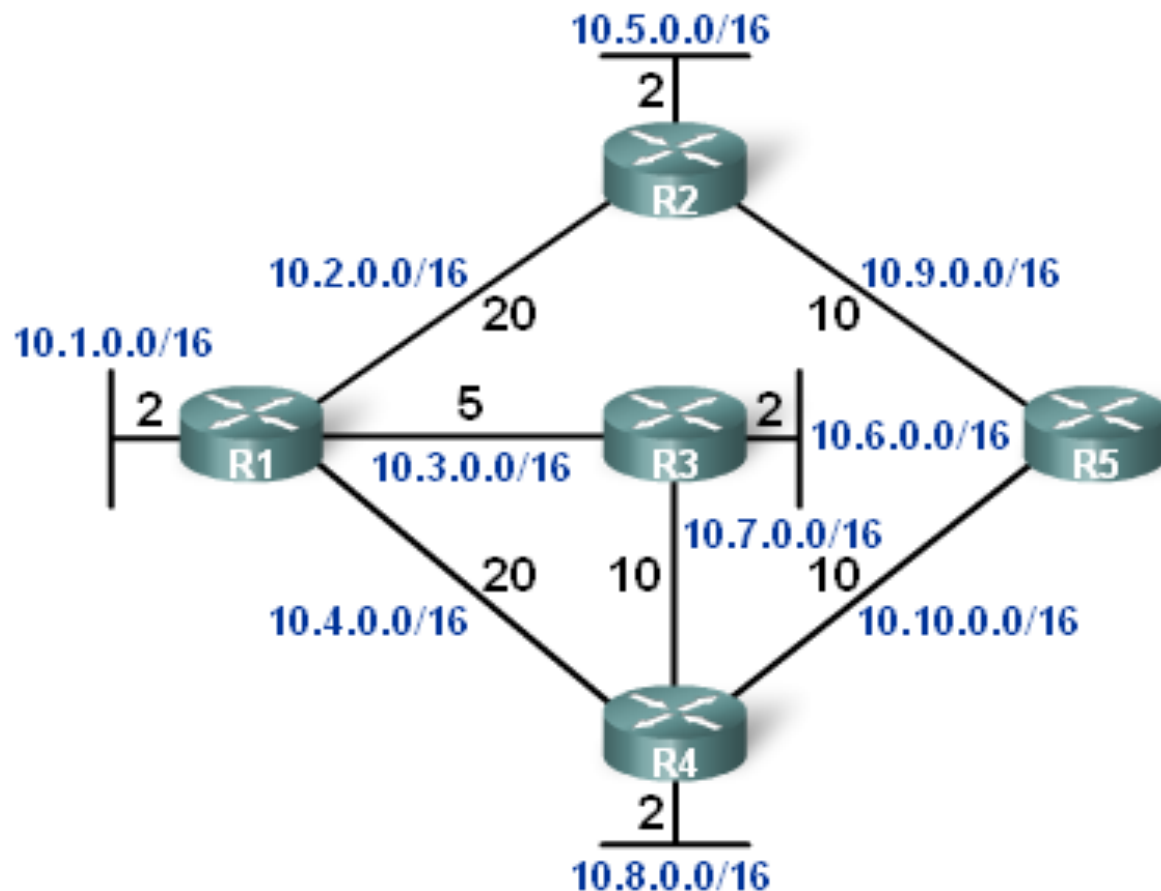
- Connected to neighbor R1 on network 10.3.0.0/16, cost of 5
- Connected to neighbor R4 on network 10.7.0.0/16, cost of 10
- Has a network 10.6.0.0/16, cost of 2



# Quá trình xây dựng cấu trúc mạng

LSPs from R4:

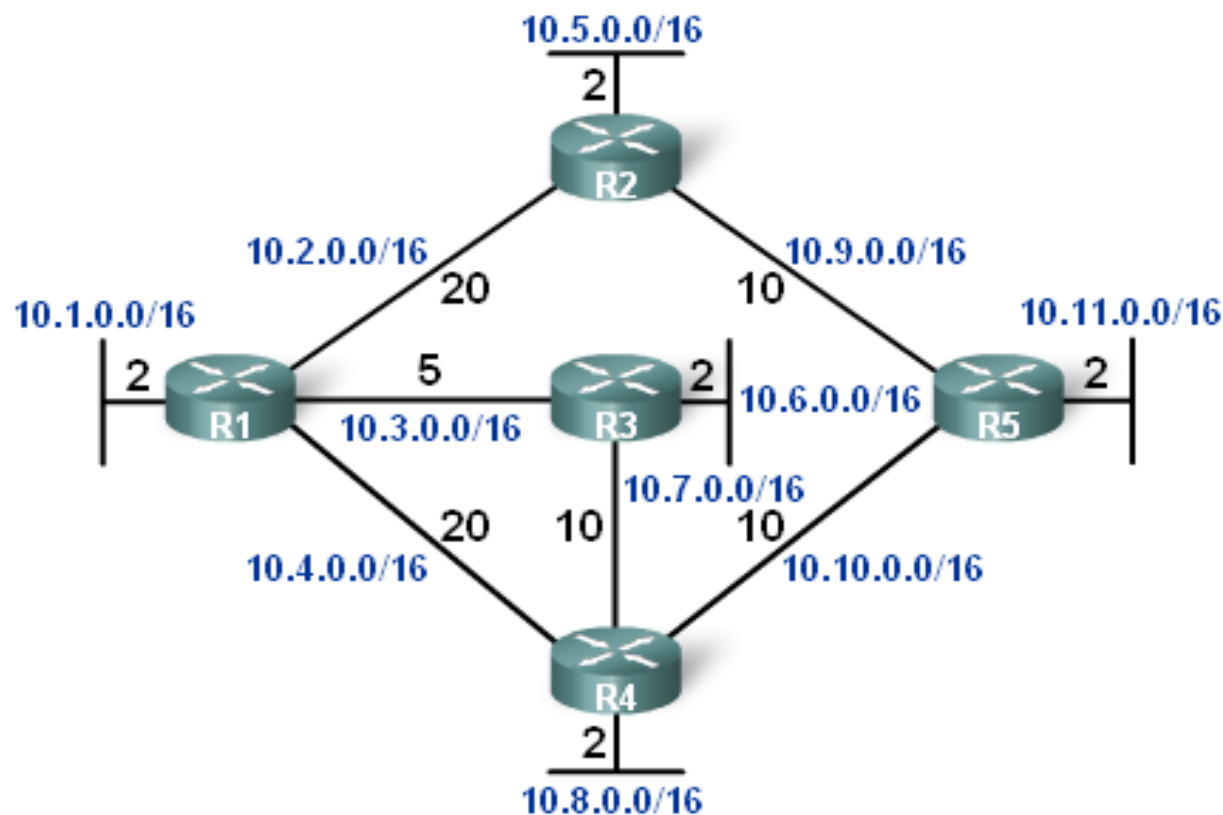
- Connected to neighbor R1 on network 10.4.0.0/16, cost of 20
- Connected to neighbor R3 on network 10.7.0.0/16, cost of 10
- Connected to neighbor R5 on network 10.10.0.0/16, cost of 10
- Has a network 10.8.0.0/16, cost of 2



# Quá trình xây dựng cấu trúc mạng

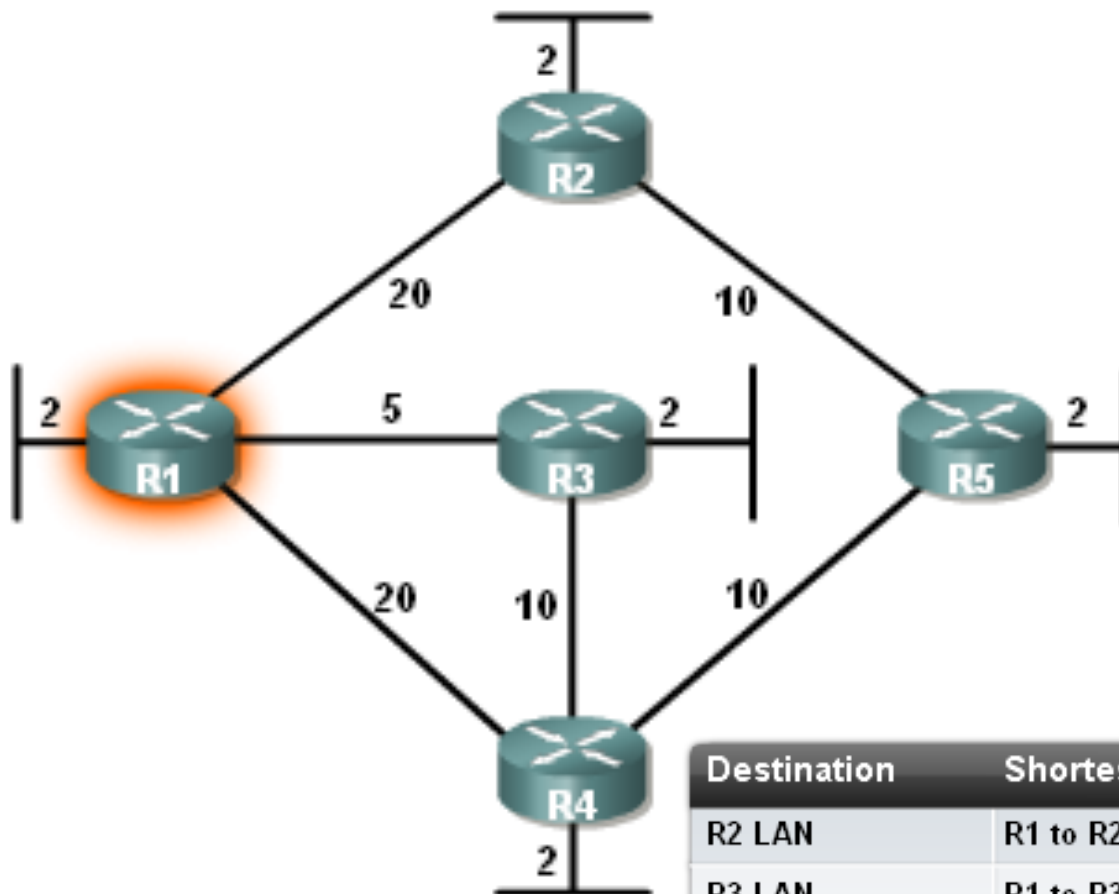
LSPs from R5:

- Connected to neighbor R2 on network 10.9.0.0/16, cost of 10
- Connected to neighbor R4 on network 10.10.0.0/16, cost of 10
- Has a network 10.11.0.0/16, cost of 2



# Shortest Path First (SPF) từ R1 đến Router khác

## Link-State Routing Protocol



Destination	Shortest Path	Cost
R2 LAN	R1 to R2	22
R3 LAN	R1 to R3	7
R4 LAN	R1 to R3 to R4	17
R5 LAN	R1 to R3 to R4 to R5	27

# Thông tin SPF và R1 Routing Table

## SPF Information

- Network 10.5.0.0/16 via R2 serial 0/0/0 at a cost of 22
- Network 10.6.0.0/16 via R3 serial 0/0/1 at a cost of 7
- Network 10.7.0.0/16 via R3 serial 0/0/1 at a cost of 15
- Network 10.8.0.0/16 via R3 serial 0/0/1 at a cost of 17
- Network 10.9.0.0/16 via R2 serial 0/0/0 at a cost of 30
- Network 10.10.0.0/16 via R3 serial 0/0/1 at a cost of 25
- Network 10.11.0.0/16 via R3 serial 0/0/1 at a cost of 27

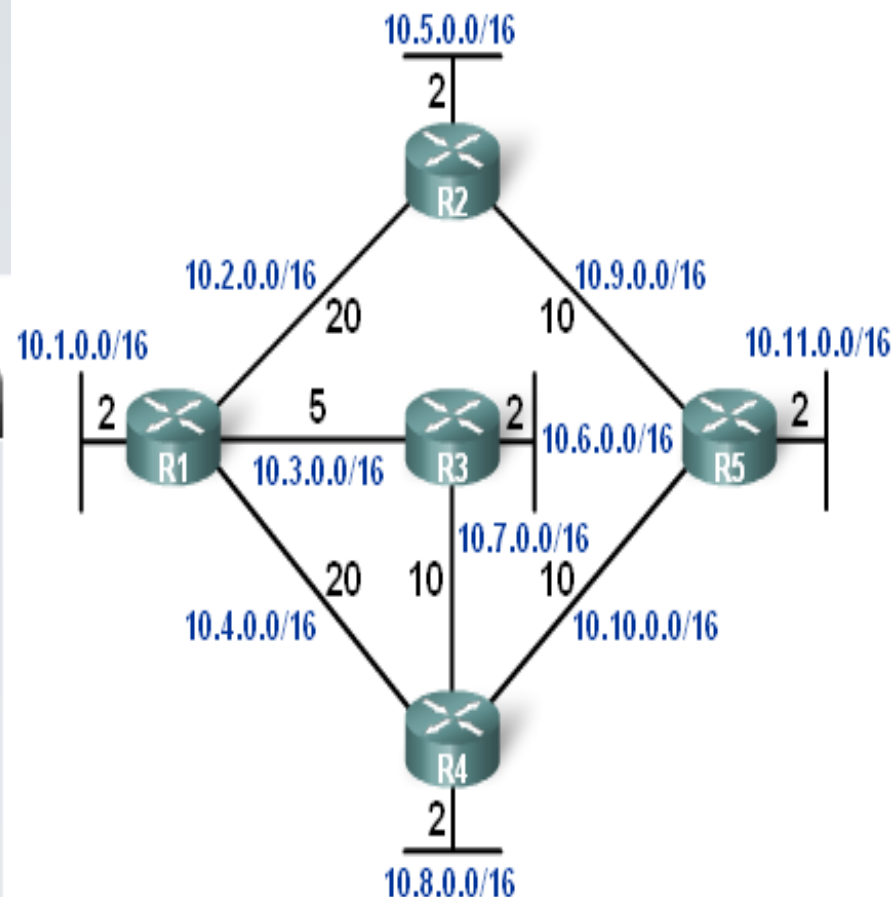
## R1 Routing Table

### Directly Connected Networks

- 10.1.0.0/16 Directly Connected Network
- 10.2.0.0/16 Directly Connected Network
- 10.3.0.0/16 Directly Connected Network
- 10.4.0.0/16 Directly Connected Network

### Remote Networks

- 10.5.0.0/16 via R2 serial 0/0/0, cost = 22
- 10.6.0.0/16 via R3 serial 0/0/1, cost = 7
- 10.7.0.0/16 via R3 serial 0/0/1, cost = 15
- 10.8.0.0/16 via R3 serial 0/0/1, cost = 17
- 10.9.0.0/16 via R2 serial 0/0/0, cost = 30
- 10.10.0.0/16 via R3 serial 0/0/1, cost = 25
- 10.11.0.0/16 via R3 serial 0/0/1, cost = 27



# Câu hỏi ôn tập

- 1) Định tuyến là gì ?
- 2) Router dựa vào thông tin gì để định tuyến gói tin ?
- 3) Router dựa vào thông tin gì để chọn đường đi tốt nhất ?
- 4) Có mấy loại Routing ? Phân biệt những loại này
- 5) Có mấy loại Routing Protocol ?
- 6) Phân tích Entry trong Routing Table
- 7) Cấu trúc câu lệnh cấu hình Static Route và Default Route
- 8) Phân biệt IGP và EGP
- 9) Trình bày qui trình cập nhật bảng định tuyến của Distance Vector Protocol ?

# Câu hỏi ôn tập

- 10) Lý do tại sao xảy ra Loop ? Trình bày những phương pháp hạn chế và khắc phục Loop
- 11) Link-state Protocol là gì ? Trình bày quá trình hoạt động của Link-state Protocol
- 12) Phân biệt Distance Vector và Link-State