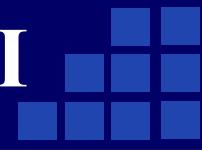




# TRƯỜNG ĐẠI HỌC THỦY LỢI



KHOA CÔNG NGHỆ THÔNG TIN

Bộ môn: Kỹ thuật máy tính và mạng

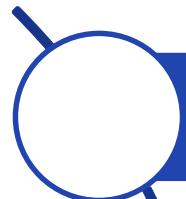
## MÔN HỌC: MẠNG MÁY TÍNH

Giảng viên: Trần Văn Hội

Email: [hoitv@tlu.edu.vn](mailto:hoitv@tlu.edu.vn)

Điện thoại: 0944.736.007

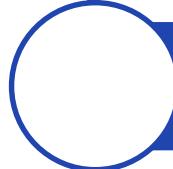
# NỘI DUNG MÔN HỌC



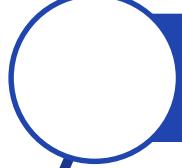
Chương 1: Tổng quan về mạng máy tính



Chương 2: Tầng ứng dụng



Chương 3: Tầng giao vận



Chương 4: Tầng mạng



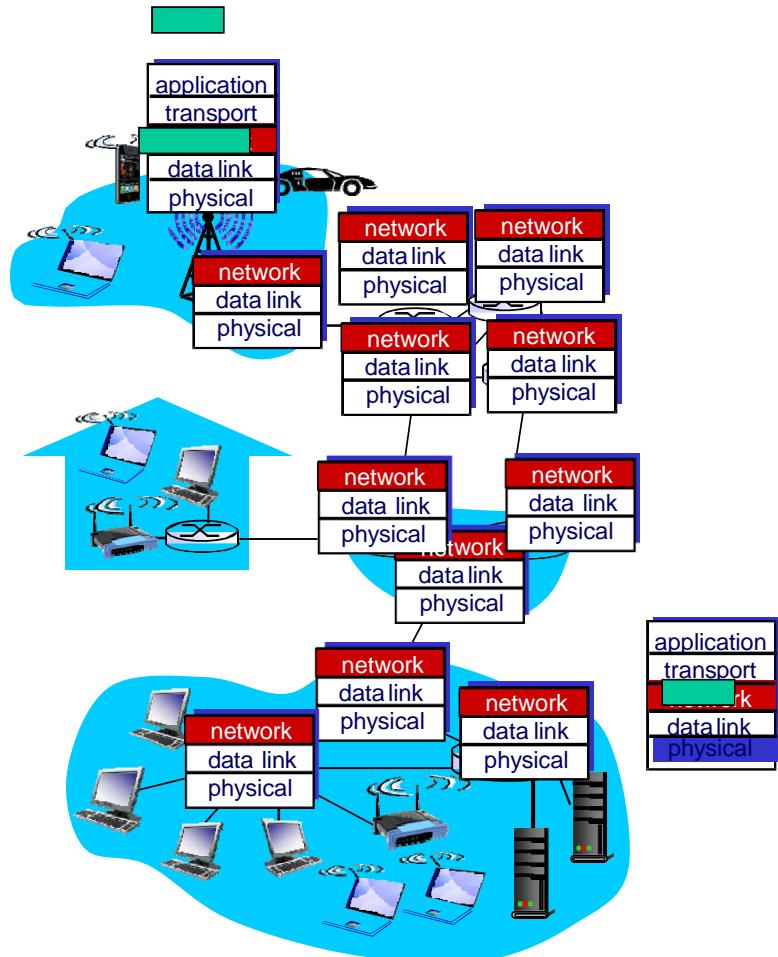
Chương 5: Tầng truy nhập mạng

# CHƯƠNG 4: TẦNG MẠNG

- 
- 1 • Giới thiệu
  - 2 • Giao thức IP
  - 3 • Định tuyến
  - 4 • Dẫn đường trong internet

# TẦNG MẠNG

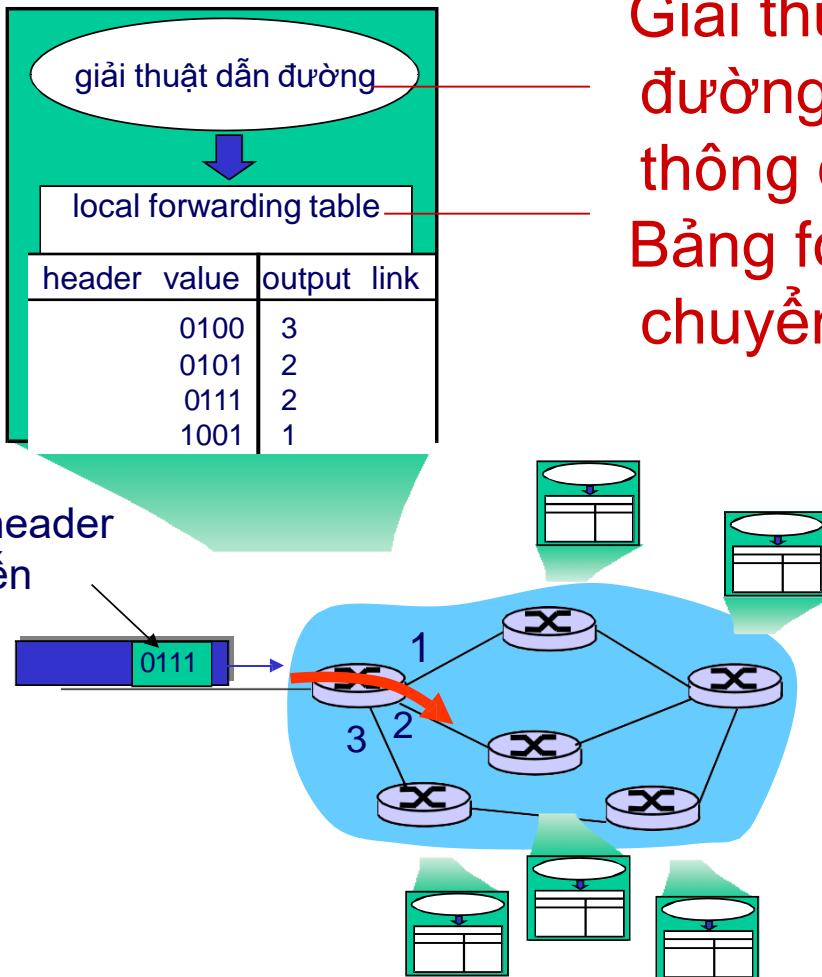
- ❖ Tầng mạng chuyển Segment của tầng giao vận từ nút gửi tới nút nhận
  - ❖ Phía gửi, đóng segment trong các datagram
  - ❖ Phía nhận, chuyển segment tới tầng giao vận
- ❖ Giao thức của tầng mạng có trong mọi host và router



# HAI CHỨC NĂNG CƠ BẢN CỦA TẦNG MẠNG

- ❖ *Định tuyến/ Dẫn đường (routing)*: xác định đường đi của gói tin từ nguồn tới đích
  - ✓ *Giải thuật dẫn đường*
- ❖ *Chuyển tiếp (forwarding)*: chuyển gói tin từ đầu vào của router ra đầu ra thích hợp của router

# DẪN ĐƯỜNG VÀ CHUYỂN TIẾP

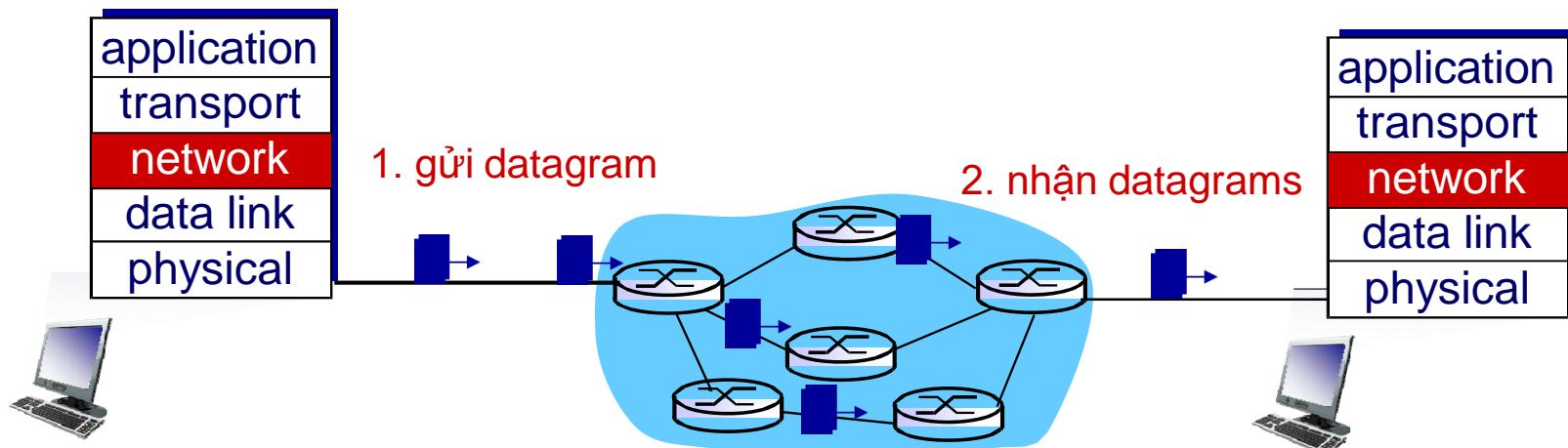


Giải thuật dẫn đường xác định  
đường đi của gói giữa hai đầu cuối  
thông qua mạng  
Bảng forwarding xác định việc  
chuyển tiếp gói tại một router

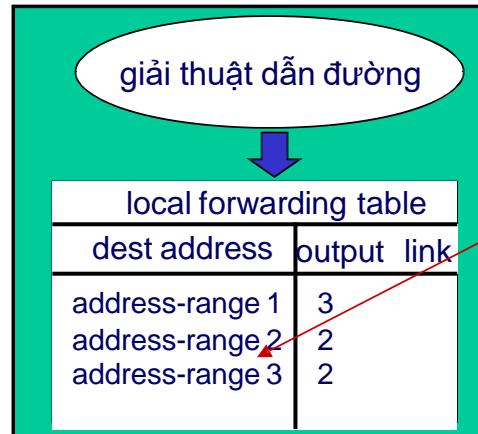
# MẠNG CHUYỂN MẠCH GÓI

## Mạng chuyển mạch gói (Datagram network)

- ❖ Gói tin được chuyển đi dựa vào địa chỉ của destination host

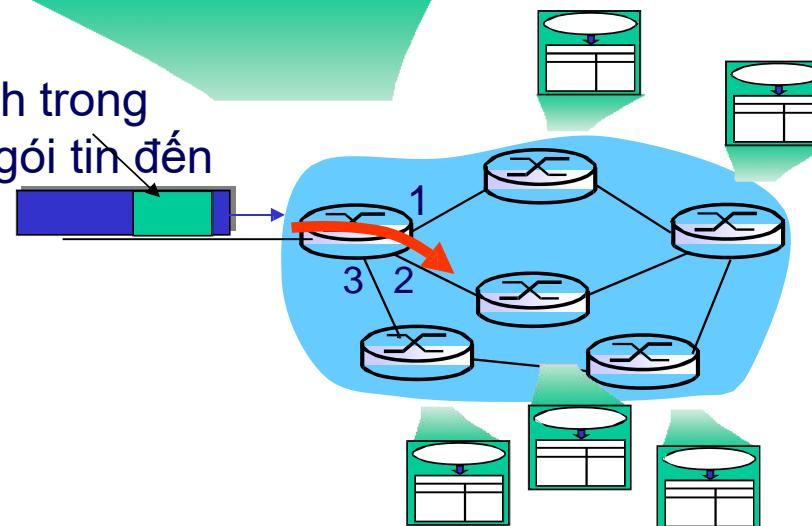


# BẢNG CHUYỂN TIẾP (FORWARDING TABLE)



Trên 4 tủ địa chỉ IP, dùng dải địa chỉ thay vì liệt kê các địa chỉ đích riêng lẻ

địa chỉ IP đích trong header của gói tin đến



# BẢNG CHUYỂN TIẾP

| Dải địa chỉ đích  | Liên kết ra |
|---|-------------|
| 11001000 00010111 00010000 00000000<br>Through<br>11001000 00010111 00010111 11111111 | 0           |
| 11001000 00010111 00011000 00000000<br>through<br>11001000 00010111 00011000 11111111 | 1           |
| 11001000 00010111 00011001 00000000<br>through<br>11001000 00010111 00011111 11111111 | 2           |

# LONGEST PREFIX MATCHING

*longest prefix matching*

khi tìm kiếm một dòng của bảng chuyển tiếp cho một địa chỉ IP đích, dùng dòng địa chỉ mà phần đầu (*prefix*) địa chỉ dài nhất giống với địa chỉ đích

| prefix                           | Liên kết ra |
|----------------------------------|-------------|
| 11001000 00010111 00010*** ***** | 0           |
| 11001000 00010111 00011000 ***** | 1           |
| 11001000 00010111 00011*** ***** | 2           |

Ví dụ:

DA: 11001000 00010111 00010110 10100001      liên kết ra?

DA: 11001000 00010111 00011000 10101010      liên kết ra?

# CÁC MÔ HÌNH DỊCH VỤ TẦNG MẠNG

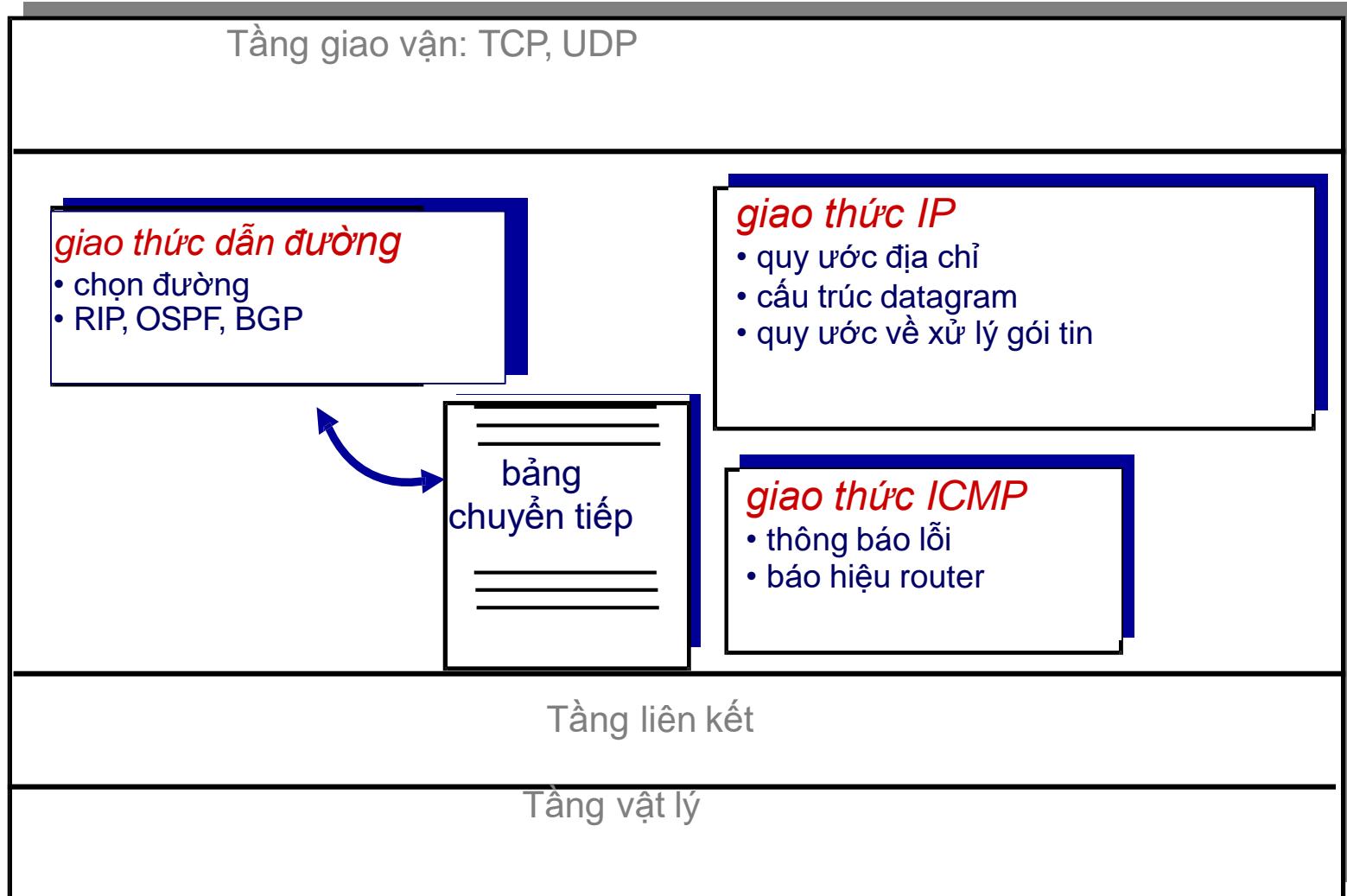
| Kiến trúc mạng                    | Mô hình dịch vụ                                      | Đảm bảo           |         |        |           | Phản hồi tắc nghẽn                  |
|-----------------------------------|--|-------------------|---------|--------|-----------|-------------------------------------|
|                                   |  | Băng thông        | Mất gói | Thứ tự | Thời gian |                                     |
| Internet                          | cố gắng tối đa (best effort)                         | không             | không   | không  | không     | không (phát hiện thông qua mất gói) |
| ATM (asynchronous transfer mode ) | Tốc độ bit cố định (CBR- Constant bit rate)          | Tốc độ không đổi  | có      | có     | có        | không tắc nghẽn                     |
| ATM                               | Tốc độ bit biến đổi (VBR -Variable bit rate)         | Tốc độ có bảo đảm | có      | có     | có        | không tắc nghẽn                     |
| ATM                               | Tốc độ bit có sẵn (ABR -Available bit rate)          | Bảo đảm tối thiểu | không   | có     | không     | có                                  |
| ATM                               | Tốc độ bit không xác định (UBR-Unspecified bit rate) | không             | không   | có     | không     | không                               |

# CHƯƠNG 4: TẦNG MẠNG

- 
- Giới thiệu
  - Giao thức IP
  - Định tuyến
  - Dẫn đường trong internet

# INTERNET PROTOCOL

Tầng  
mạng



# GIAO THÚC IP

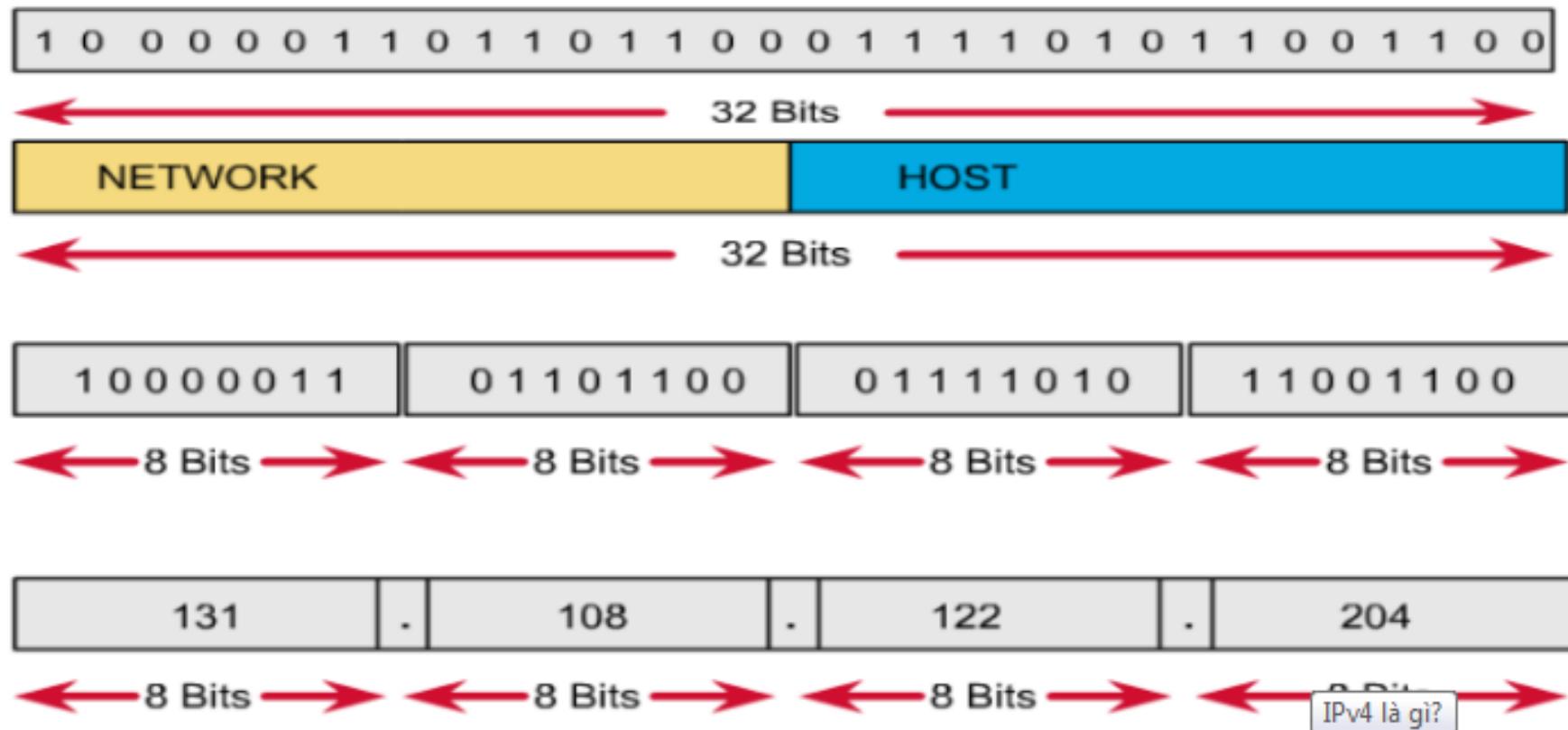
- ❖ IP (Internet Protocol) là giao thức không liên kết.
- ❖ Chức năng của IP là cung cấp khả năng kết nối các mạng con thành liên mạng để truyền dữ liệu với phương thức chuyển mạch gói IP Datagram.
- ❖ Để thực hiện được giao thức IP thực hiện tiến trình định địa chỉ và chọn đường từ nguồn tới đích (định tuyến).
- ❖ Giao thức IP có 2 version là IP version 4 (IPv4) và IP version 6 (IPv6)

# ĐỊA CHỈ IPv4

- ❖ Địa chỉ IP: Mỗi một trạm (Host) được gán một địa chỉ duy nhất gọi là địa chỉ IP.
- ❖ Mỗi địa chỉ IPv4 có độ dài 32 bit được tách thành 4 octet (mỗi octet là 8 bite), có thể được biểu diễn dưới dạng thập phân, bát phân, thập lục phân hoặc nhị phân.
- ❖ Cách viết phổ biến nhất là dưới dạng thập phân có dấu chấm để tách giữa các vùng.
- ❖ Địa chỉ IP được chia thành 5 lớp ký hiệu là A, B, C, D, E với cấu trúc mỗi lớp được xác định. Các bit đầu tiên của byte đầu tiên được dùng để định danh lớp địa chỉ (0-lớp A, 10 - lớp B, 110 - lớp C, 1110 - lớp D, 11110 - lớp E).

# Địa chỉ IPv4 và các lớp địa chỉ

Địa chỉ IPv4 có độ dài 32 bit chia thành 4 octet chia thành 2 phần là network\_id và host\_id.



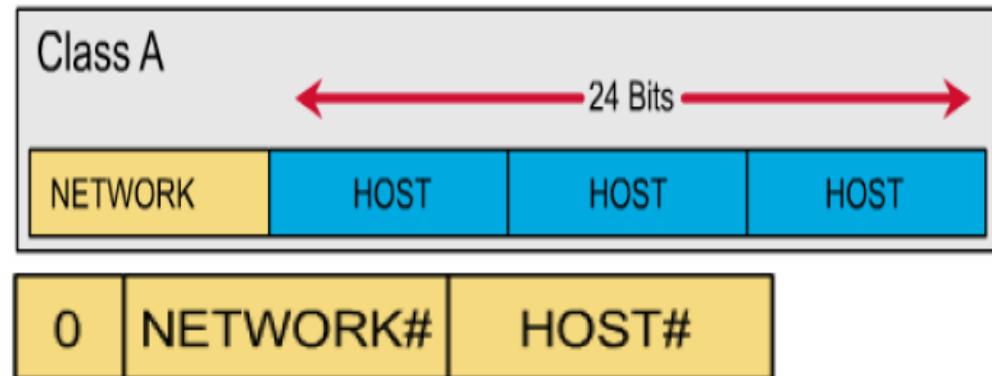
Cấu trúc địa chỉ IPv4

# Các lớp địa chỉ IPv4

- Không gian địa chỉ IPv4 được chia thành 5 lớp (class) A, B, C, D và E.
- Các lớp A, B và C được triển khai để đặt cho các host trên mạng Internet.
- Lớp D dùng cho các nhóm multicast, còn lớp E phục vụ cho mục đích nghiên cứu.

# Lớp A (Class A)

Dành 1 byte cho phần network\_id và 3 byte cho phần host\_id.



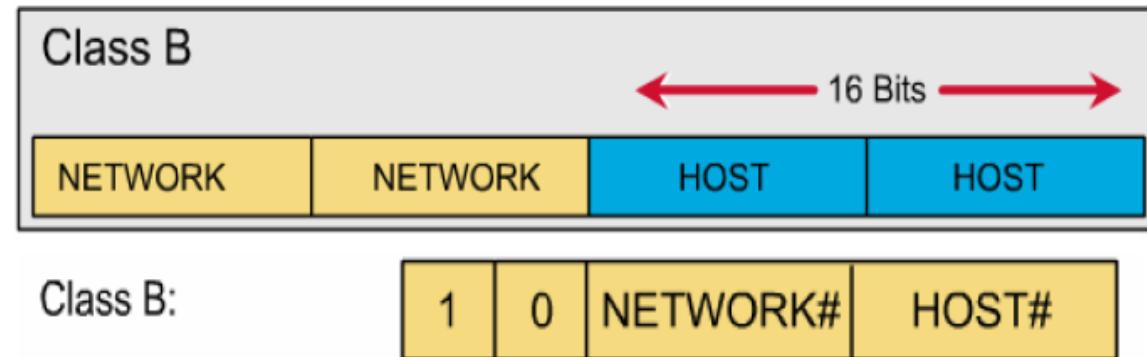
- ❖ Bit đầu tiên của byte đầu tiên phải là bit 0. Dạng nhị phân của octet này là **0xxxxxxxx**
- ❖ Những địa chỉ IP có byte đầu tiên nằm trong khoảng từ 0 ( $=00000000_{(2)}$ ) đến 127 ( $=01111111_{(2)}$ ) sẽ thuộc lớp A.
- ❖ Ví dụ: 50.14.32.8.

# Lớp A (Class A)

- ❖ Byte đầu tiên này cũng chính là network\_id, trừ đi bit đầu tiên làm ID nhận dạng lớp A, còn lại 7 bit để đánh thứ tự các mạng, ta được  $128 (=2^7)$  mạng lớp A khác nhau.
- ❖ Bỏ đi hai trường hợp đặc biệt là 0 và 127. Kết quả là lớp A chỉ còn 126 địa chỉ mạng, 1.0.0.0 đến 126.0.0.0.
- ❖ Phần host\_id chiếm 24 bit, nghĩa là có  $2^{24} = 16.777.216$  host khác nhau trong mỗi mạng. Bỏ đi hai trường hợp đặc biệt (phần host\_id chứa toàn các bit 0 và bit 1). Còn lại: 16.777.214 host.
- ❖ Ví dụ đối với mạng 10.0.0.0 thì những giá trị host hợp lệ là 10.0.0.1 đến 10.255.255.254.

# Lớp B (Class B)

Dành 2 byte cho phần network\_id và 2 byte cho phần host\_id.



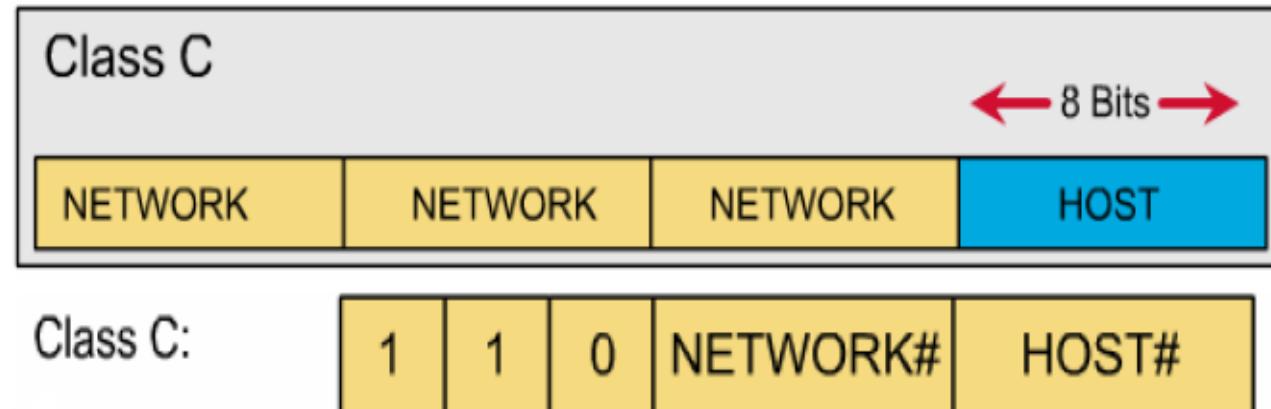
- ❖ Hai bit đầu tiên của byte đầu tiên phải là 10. Dạng nhị phân của octet này là **10xxxxxx**
- ❖ Những địa chỉ IP có byte đầu tiên nằm trong khoảng từ 128 ( $=10000000_{(2)}$ ) đến 191 ( $=10111111_{(2)}$ ) sẽ thuộc về lớp B
- ❖ Ví dụ: 172.29.10.1.

# Lớp B (Class B)

- ❖ Phần network\_id chiếm 16 bit bỏ đi 2 bit làm ID cho lớp, còn lại 14 bit cho phép ta đánh thứ tự 16.384 ( $=2^{14}$ ) mạng khác nhau (**128.0.0.0** đến **191.255.0.0**).
- ❖ Phần host\_id dài 16 bit hay có 65.536 ( $=2^{16}$ ) giá trị khác nhau. Trừ đi 2 trường hợp **đặc biệt** còn lại 65.534 host trong một mạng lớp B.
- ❖ Ví dụ đối với mạng **172.29.0.0** thì các địa chỉ host hợp lệ là từ **172.29.0.1** đến **172.29.255.254**.

# Lớp C (Class C)

Dành 3 byte cho phần network\_id và 1 byte cho phần host\_id.



- ❖ Ba bit đầu tiên của byte đầu tiên phải là 110. Dạng nhị phân của octet này là **110xxxxx**
- ❖ Những địa chỉ IP có byte đầu tiên nằm trong khoảng từ 192 ( $=11000000_{(2)}$ ) đến 223 ( $=11011111_{(2)}$ ) sẽ thuộc về lớp C.
- ❖ Ví dụ: 203.162.41.235

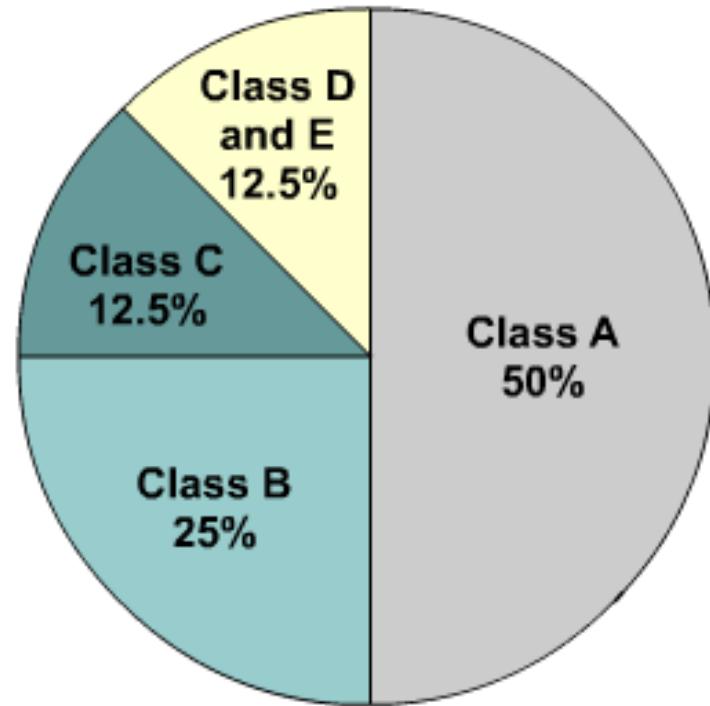
# Các lớp địa chỉ IP

| Address Class | Number of Networks | Number of Host per Network |
|---------------|--------------------|----------------------------|
| A             | 126 *              | 16,777,216                 |
| B             | 16,384             | 65,535                     |
| C             | 2,097,152          | 254                        |
| D (Multicast) | N/A                | N/A                        |

| IP Address Class | High Order Bits | First Octet Address Range | Number of Bits in the Network Address |
|------------------|-----------------|---------------------------|---------------------------------------|
| Class A          | 0               | 0 - 127 *                 | 8                                     |
| Class B          | 10              | 128 - 191                 | 16                                    |
| Class C          | 110             | 192 - 223                 | 24                                    |
| Class D          | 1110            | 224 - 239                 | 28                                    |

# Các lớp địa chỉ IP

| IP address class | IP address range<br>(First Octet Decimal Value) |
|------------------|---|
| Class A          | 1-126 (00000001-01111110) *                     |
| Class B          | 128-191 (10000000-10111111)                     |
| Class C          | 192-223 (11000000-11011111)                     |
| Class D          | 224-239 (11100000-11101111)                     |
| Class E          | 240-255 (11110000-11111111)                     |



# Địa chỉ dành riêng Private

## ❖ Địa chỉ IP Public

- Duy nhất
- Phải trả phí
- Được định tuyến

## ❖ Địa chỉ IP Private

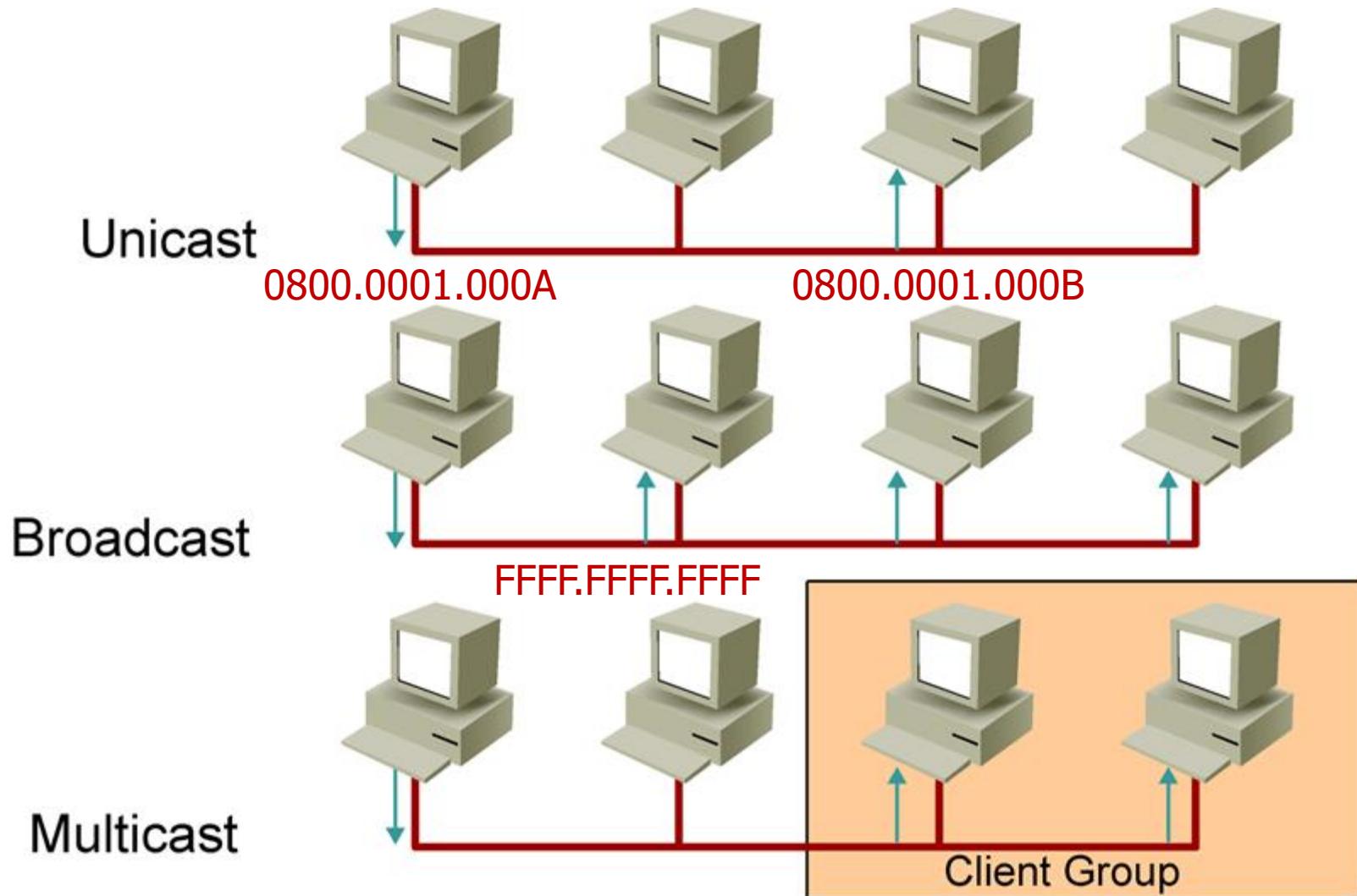
- Được sử dụng lặp
- Miễn phí
- Không được định tuyến (kết hợp NAT)

| Class | RFC 1918 internal address range |
|-------|---------------------------------|
| A     | 10.0.0.0 to 10.255.255.255      |
| B     | 172.16.0.0 to 172.31.255.255    |
| C     | 192.168.0.0 to 192.168.255.255  |

# Địa chỉ nào sử dụng trong mạng nội bộ

- ❖ **150.100.255.255**
- ❖ **172.19.255.18**
- ❖ **195.234.253.0**
- ❖ **10.10.110.23**
- ❖ **192.168.221.176**
- ❖ **127.34.25.189**
- ❖ **203.162.217.73**

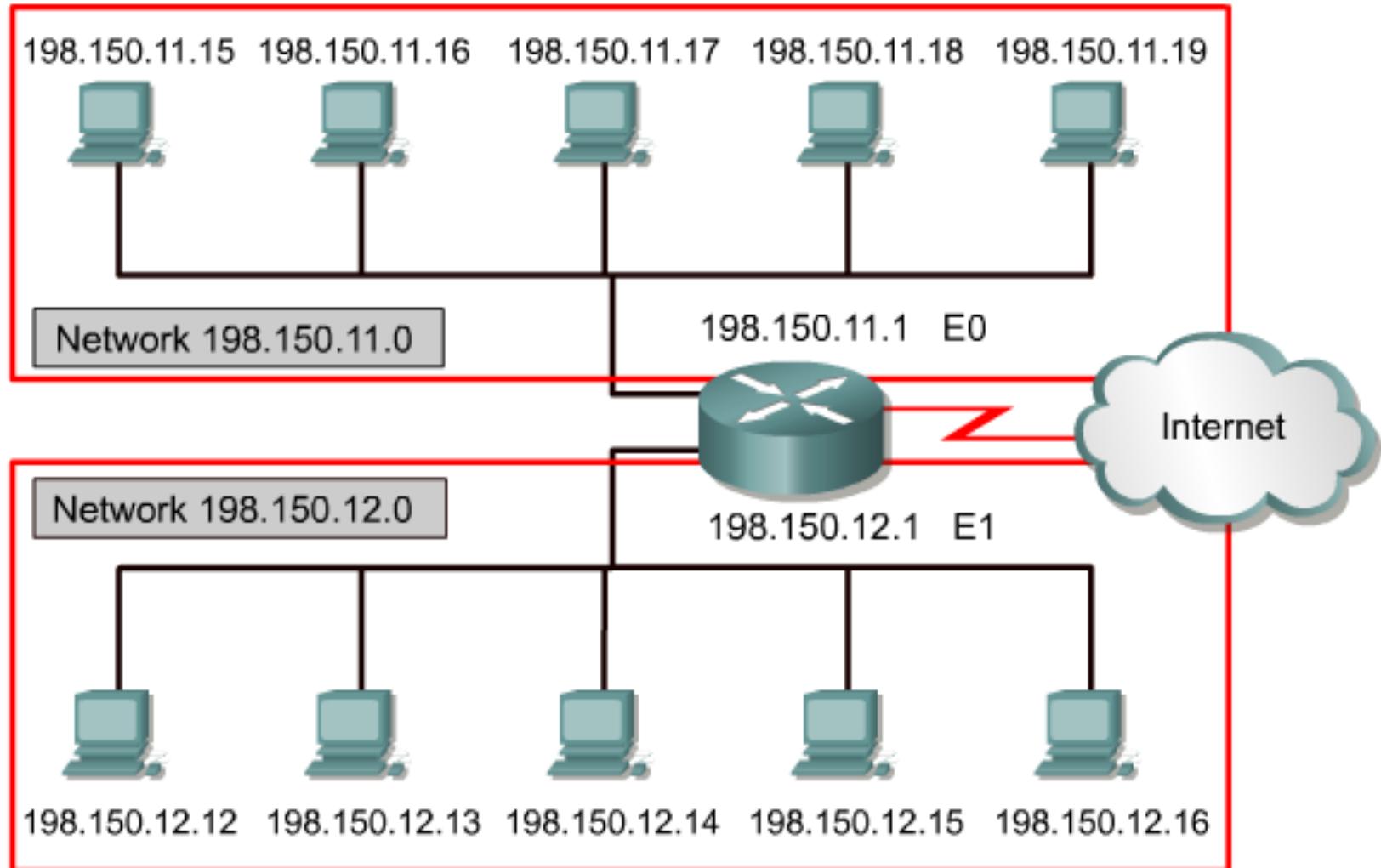
# Phân loại địa chỉ IPv4



# Địa chỉ mạng, địa chỉ Broadcast

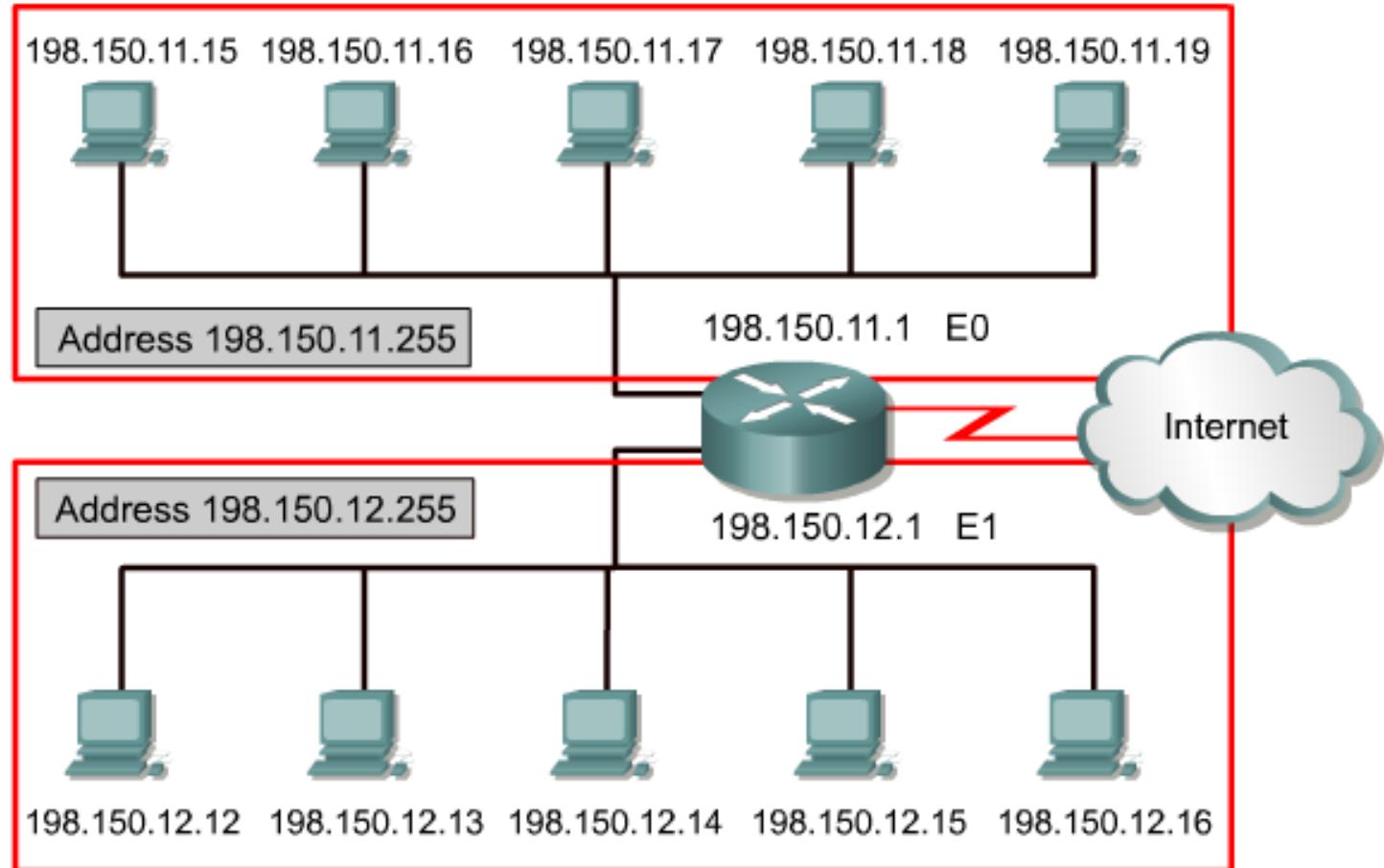
- ❖ Địa chỉ mạng (network address): là địa chỉ IP dùng để đặt cho các mạng. Phần host\_id của địa chỉ chỉ chứa các bit 0. Ví dụ: 172.29.0.0
- ❖ Địa chỉ Broadcast: là địa chỉ IP được dùng để đại diện cho tất cả các host trong mạng. Phần host\_id chỉ chứa các bit 1. Ví dụ: 172.29.255.255.

# Địa chỉ mạng (network address)



Địa chỉ mạng

# Địa chỉ Broadcast



Địa chỉ broadcast

# HẠN CHẾ CỦA VIỆC PHÂN LỚP ĐỊA CHỈ

## ❖ Lãng phí không gian địa chỉ

- Việc phân chia cứng thành các lớp (A, B, C, D, E) làm hạn chế việc sử dụng toàn bộ không gian địa chỉ

## Cách giải quyết

## ❖ Không phân lớp

- Phần địa chỉ mạng sẽ có độ dài tùy ý
- Dạng địa chỉ không phân lớp:
  - ✓ a.b.c.d/x
    - ✓ /x (subnet mask - mặt nạ mạng) là số bit trong phần ứng với địa chỉ mạng
    - ✓ Ví dụ địa chỉ IPv4: 144.28.16.17/20

# MẶT NẠ MẠNG (SUBNET MASK)

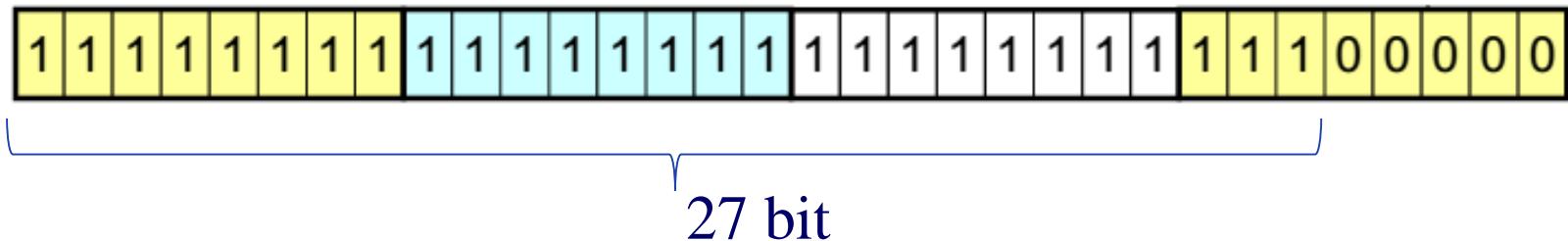
- ❖ **Mặt nạ mạng chia một địa chỉ IP thành 2 phần**
  - ❖ Phần ứng với host
  - ❖ Phần ứng với mạng
- ❖ **Tính địa chỉ mạng từ địa chỉ IP**
  - ❖ Dùng toán tử AND

# MÔ TẢ MẶT NẠ MẠNG

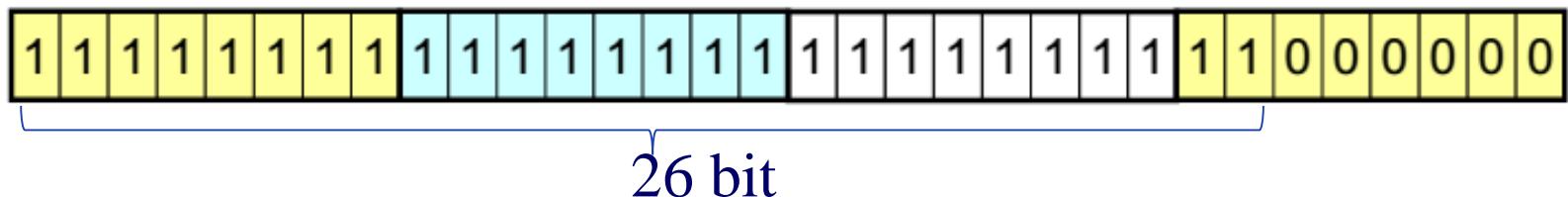
## ❖ Subnet mask

- ❖ Các bit trong phần Network bằng 1
- ❖ Các bit trong phần host bằng 0
- ❖ Ví dụ:

**/27**



**/26**



# Tính địa chỉ mạng từ địa chỉ IP

❖ Địa chỉ IP là: **144.28.16.17/20**

|                  |          |          |          |          |    |   |    |
|------------------|----------|----------|----------|----------|----|---|----|
|                  | 144      | .        | 28       | .        | 16 | . | 17 |
| IP address:      | 10010000 | 00011100 | 00010000 | 00010001 |    |   |    |
| Subnet mask:     | 11111111 | 11111111 | 11110000 | 00000000 |    |   |    |
| Network address: | 10010000 | 00011100 | 00010000 | 00000000 |    |   |    |
|                  | 144      | .        | 28       | .        | 16 | . | 0  |

❖ Địa chỉ mạng: **144.28.16.0/20**

# Bài tập áp dụng

- ❖ Cho địa chỉ IP: **203.178.142.130/27**
- ❖ Tính địa chỉ mạng?

# ĐỊA CHỈ MẠNG, ĐỊA CHỈ MÁY TRẠM?

133

27

4

160

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

133

27

4

128

# Địa chỉ máy trạm, địa chỉ mạng, địa chỉ

- 1) 203.178.142.128/25
- 2) 203.178.142.128/24
- 3) 203.178.142.127/25
- 4) 203.178.142.127/24

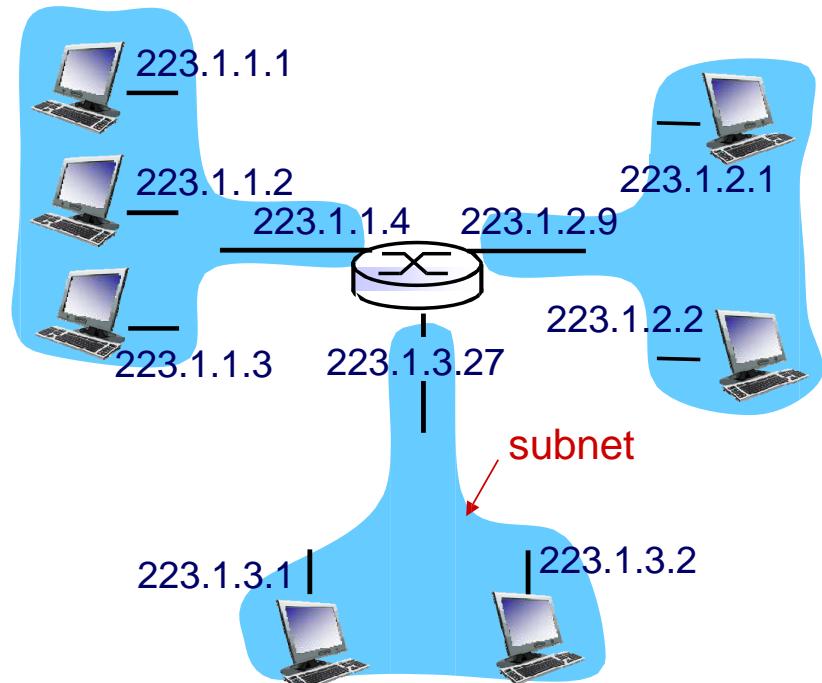
# CHIA MẠNG CON - SUBNET

## ❖ Tại sao phải chia mạng con

- Mỗi lớp mạng A có đến  $2^{24} - 2 = 16.777.214$  địa chỉ IP hay lớp B có  $2^{16} - 2 = 65534$  địa chỉ IP.
- Khó có hệ thống đạt số host quá lớn như vậy
- Khó khăn trong công tác quản lý.

# MẠNG CON - SUBNET

- ❖ Mạng con là một phần của một mạng nào đó
  - ISP thường được gán một khối địa chỉ IP
  - Chia khối địa chỉ IP thành các mạng nhỏ hơn được gọi là mạng con

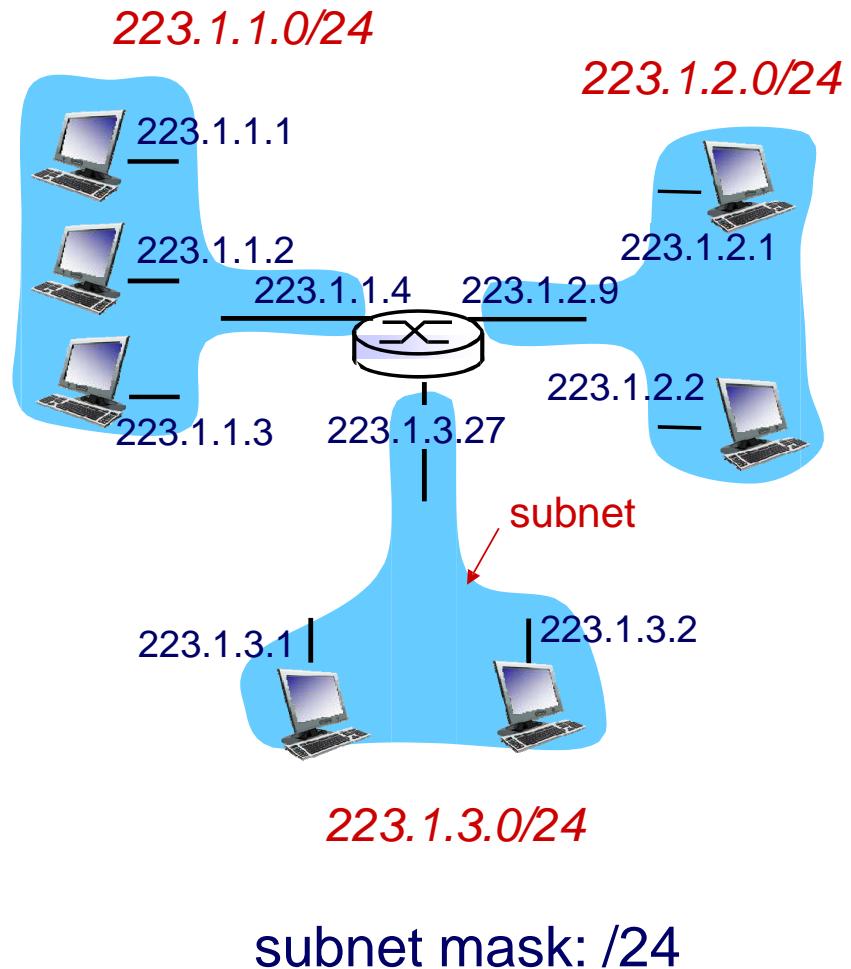


Mạng với 3 mạng con

# MẠNG CON - SUBNET

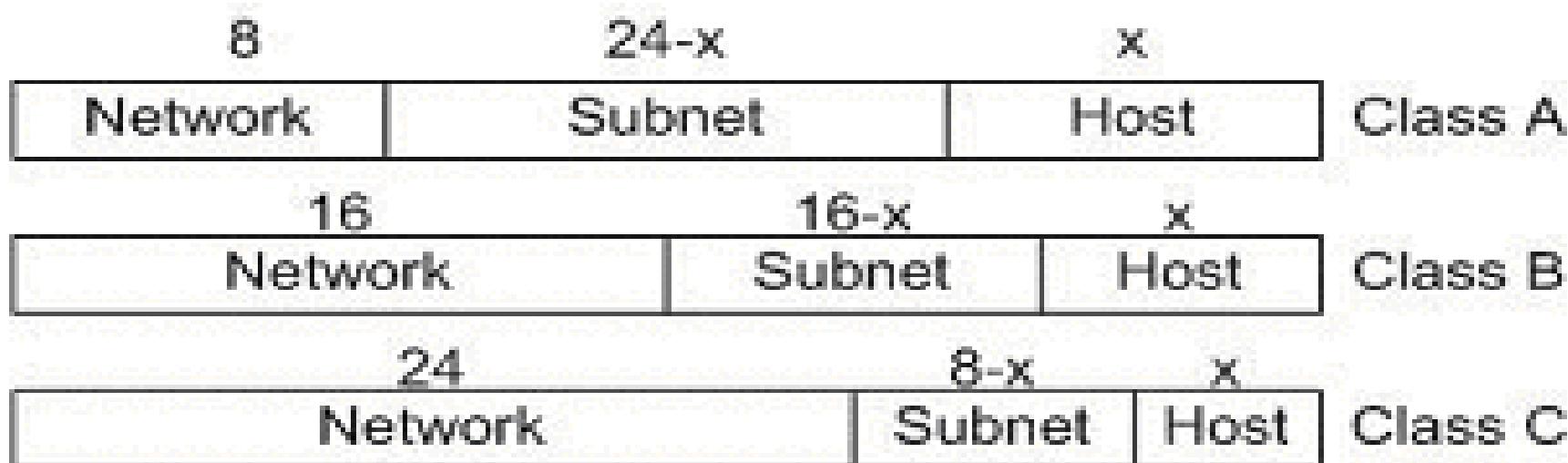
- ❖ Địa chỉ IP:

- ❖ Phần subnet: các bit cao
- ❖ Phần host: các bit thấp
- ❖ Trong một subnet
  - ❖ Các interface có cùng phần subnet
  - ❖ Có thể giao tiếp với nhau không cần qua router



# Kỹ thuật chia mạng con

- ❖ Mượn một số bit trong phần host\_id ban đầu để đặt cho các mạng con
- ❖ Cấu trúc của địa chỉ IP lúc này sẽ gồm 3 phần: network\_id, subnet\_id và host\_id.



# Kỹ thuật chia mạng con

- ❖ Số bit dùng trong subnet\_id tuỳ thuộc vào chiến lược chia mạng con. Tuy nhiên số bit tối đa có thể mượn phải tuân theo công thức:

$$\text{Subnet\_id} \leq \text{host\_id} - 2$$

- ❖ Số lượng bit tối đa có thể mượn:

- Lớp A: **22** ( $= 24 - 2$ ) bit -> chia được  $2^{22} = 4194304$  mạng con
- Lớp B: **14** ( $= 16 - 2$ ) bit -> chia được  $2^{14} = 16384$  mạng con
- Lớp C: **06** ( $= 8 - 2$ ) bit -> chia được  $2^6 = 64$  mạng con

# Kỹ thuật chia mạng con

- ❖ Số bit trong phần subnet\_id xác định số lượng mạng con. Với số bit là x thì  $2^x$  là số lượng mạng con có được.
- ❖ Ngược lại từ số lượng mạng con cần thiết theo nhu cầu, tính được phần subnet\_id cần bao nhiêu bit. Nếu muốn chia 6 mạng con thì cần 3 bit ( $2^3=8$ ), chia 12 mạng con thì cần 4 bit ( $2^4>=12$ ).

# Một số khái niệm mới

- ❖ Địa chỉ mạng con (địa chỉ đường mạng): gồm cả phần network\_id và subnet\_id, phần host\_id chỉ chứa các bit 0
- ❖ Địa chỉ broadcast trong một mạng con: tất cả các bit trong phần host\_id là 1.
- ❖ Mặt nạ mạng con (subnet mask): tất cả các bit trong phần host\_id là 0, các phần còn lại là 1.

# Quy ước ghi địa chỉ IP

- ❖ Nếu có địa chỉ IP như 172.29.8.230 thì chưa thể biết được host này nằm trong mạng nào, có chia mạng con hay không và có nếu chia thì dùng bao nhiêu bit để chia. Chính vì vậy khi ghi nhận địa chỉ IP của một host, phải cho biết subnet mask của nó
- ❖ Ví dụ: 172.29.8.230/ 255.255.255.0 hoặc 172.29.8.230/24 (có nghĩa là dùng 24 bit đầu tiên cho NetworkID).

# Kỹ thuật chia mạng con

## ❖ Thực hiện 3 bước:

- Bước 1: Xác định lớp (class) và subnet mask mặc  
nhiên của địa chỉ.
- Bước 2: Xác định số bit cần mượn và subnet mask  
mới, tính số lượng mạng con, số host thực sự có  
được.
- Bước 3: Xác định các vùng địa chỉ host và chọn  
mạng con muốn dùng.

# VÍ DỤ

Hãy xét đến một địa chỉ IP class B, 139.12.0.0, với subnet mask là 255.255.0.0. Một Network với địa chỉ thế này có thể chứa 65534 nodes hay computers. Đây là một con số quá lớn, trên mạng sẽ có đầy broadcast traffic.

Hãy chia network thành 5 mạng con.

# Bước 1: Xác định Subnet mask

- ❖ Để chia thành 5 mạng con thì cần thêm 3 bit (vì  $2^3 > 5$ ).
- ❖ Do đó Subnet mask sẽ cần: 16 (bits trước đây) + 3 (bits mới) = 19 bits
- ❖ Địa chỉ IP mới sẽ là 139.12.0.0/19 (để ý con số 19 thay vì 16 như trước đây).

## Bước 2: Liệt kê ID của các Subnet mới

Subnet mask với dạng nhị phân

11111111.11111111.11100000.00000000

Subnet mask  
với dạng thập phân

255.255.224.0

# NetworkID của bốn Subnets mới

| TT | Subnet ID với dạng nhị phân                  | Subnet ID với dạng thập phân |
|----|--|------------------------------|
| 1  | 10001011.00001100. <b>000</b> 00000.00000000 | 139.12.0.0/19                |
| 2  | 10001011.00001100. <b>001</b> 00000.00000000 | 139.12.32.0/19               |
| 3  | 10001011.00001100. <b>010</b> 00000.00000000 | 139.12.64.0/19               |
| 4  | 10001011.00001100. <b>011</b> 00000.00000000 | 139.12.96.0/19               |
| 5  | 10001011.00001100. <b>100</b> 00000.00000000 | 139.12.128.0/19              |

# Bước 3: Cho biết vùng địa chỉ IP của các HostID

| TT | Dạng nhị phân  | Dạng thập phân                         |
|----|--|--|
| 1  | 10001011.00001100.00000000.00000001<br>10001011.00001100.00011111.11111110 | 139.12.0.1/19 -<br>139.12.31.254/19    |
| 2  | 10001011.00001100.00100000.00000001<br>10001011.00001100.00111111.11111110 | 139.12.32.1/19 -<br>139.12.63.254/19   |
| 3  | 10001011.00001100.01000000.00000001<br>10001011.00001100.01011111.11111110 | 139.12.64.1/19 -<br>139.12.95.254/19   |
| 4  | 10001011.00001100.01100000.00000001<br>10001011.00001100.01111111.11111110 | 139.12.96.1/19 -<br>139.12.127.254/19  |
| 5  | 10001011.00001100.10000000.00000001<br>10001011.00001100.10011111.11111110 | 139.12.128.1/19 -<br>139.12.159.254/19 |

# Ví dụ tính nhanh vùng địa chỉ IP

- ❖ Cho địa chỉ: 192.168.0.0/24 Chia thành 16 mạng con
- ❖ Với  $n=4 \rightarrow M= 16 (= 2^{8-4}) \rightarrow$

- Network 1: 192.168.0.0. Host range: 192.168.0.1–192.168.0.14. Broadcast: 192.168.0.15
- Network 2: 192.168.0.16. Host range: 192.168.0.17–192.168.0.30. Broadcast: 192.168.0.31
- Network 3: 192.168.0.32. Host range: 192.168.0.33–192.168.0.46. Broadcast: 192.168.0.47
- Network 4: 192.168.0.48. Host range: 192.168.0.49–192.168.0.62. Broadcast: 192.168.0.63
- ....

# Tính nhanh vùng địa chỉ IP

- ❖  $n$  – số bit làm subnet
- ❖ Số mạng con:  $S = 2^n$
- ❖ Số địa chỉ host trong mạng con:  $M = 2^{8-n}$  ( $n \leq 8$ )
- Byte cuối của IP địa chỉ mạng, ví dụ lớp C:  $(k-1)*M$  (với  $k=1,2,\dots$ )
- Byte cuối của IP broadcast, ví dụ lớp C:  $k*M - 1$  (với  $k=1,2,\dots$ )
- Byte cuối của IP host đầu tiên, ví dụ lớp C:  $(k-1)*M + 1$  (với  $k=1,2,\dots$ )
- Byte cuối của IP host cuối cùng, ví dụ lớp C:  $k*M - 2$  (với  $k=1,2,\dots$ )

# Kỹ thuật chia mạng con

192.168.1.0 - 255 /24  
192.168.1.0 - 127 /25  
192.168.1.128 - 255 /25

|                     |
|---------------------|
| 192.168.1.0000 0000 |
| 192.168.1.0000 0001 |
| 192.168.1.0000 0010 |
| 192.168.1.0000 0011 |
| 192.168.1.0000 0100 |
| 192.168.1.0000 0101 |
| 192.168.1.0000 0110 |
| 192.168.1.0111 1111 |

|                     |
|---------------------|
| 192.168.1.1000 0000 |
| 192.168.1.1000 0001 |
| 192.168.1.1000 0010 |
| 192.168.1.1000 0011 |
| 192.168.1.1000 0100 |
| 192.168.1.1000 0101 |
| 192.168.1.1000 0110 |
| 192.168.1.1111 1111 |

**192.168.1.0** - 255 /24

**192.168.1.0** - 63 /26

**192.168.1.64** - 127 /26

**192.168.1.128** - 191 /26

**192.168.1.192** - 255 /26

192.168.1.0**0000** 0000  
192.168.1.0**0000** 0001  
192.168.1.0**0000** 0010  
**192.168.1.0000** 0011

192.168.1.**0011** 1111

192.168.1.**0100** 0000  
192.168.1.**0100** 0001  
192.168.1.**0100** 0010  
192.168.1.**0100** 0011

192.168.1.**0111** 1111

192.168.1.**1000** 0000  
192.168.1.**1000** 0001  
192.168.1.**1000** 0010  
192.168.1.**1000** 0011

192.168.1.**1011** 1111

192.168.1.**1100** 0000  
192.168.1.**1100** 0001  
192.168.1.**1100** 0010  
192.168.1.**1100** 0011

192.168.1.**1111** 1111

**192.168.1.0** - 255 /24

**192.168.1.0** - 31 /27

**192.168.1.32** - 63 /27

**192.168.1.64** - 95 /27

**192.168.1.96** - 127 /27

**192.168.1.128** - 159 /27

**192.168.1.160** - 191 /27

**192.168.1.192** - 223 /27

**192.168.1.224** - 255 /27

192.168.1.**0000** 0000  
192.168.1.**0001** 1111

192.168.1.**0010** 0000  
192.168.1.**0011** 1111

192.168.1.**0100** 0000  
192.168.1.**0101** 1111

192.168.1.**0110** 0000  
192.168.1.**0111** 1111

192.168.1.**1000** 0000  
192.168.1.**1001** 1111

192.168.1.**1010** 0000  
192.168.1.**1011** 1111

192.168.1.**1100** 0000  
192.168.1.**1101** 1111

192.168.1.**1110** 0000  
192.168.1.**1111** 1111

# Kỹ thuật chia mạng con

192.168.1.0 - 255 /24

192.168.1.0 - 15 /28

192.168.1.16 - 31 /28

192.168.1.32 - 47 /28

192.168.1.48 - 63 /28

192.168.1.0000 0000

192.168.1.0001 0000

192.168.1.0010 0000

192.168.1.0011 0000

192.168.1.64 - 79 /28

192.168.1.80 - 95 /28

192.168.1.96 - 111 /28

192.168.1.112 - 127 /28

192.168.1.0100 0000

192.168.1.0101 0000

192.168.1.0110 0000

192.168.1.0111 0000

192.168.1.128 - 159 /28

192.168.1.144 - 191 /28

192.168.1.160 - 223 /28

192.168.1.176 - 223 /28

192.168.1.1000 0000

192.168.1.1001 0000

192.168.1.1010 0000

192.168.1.1011 0000

192.168.1.192 - 255 /28

192.168.1.208 - 255 /28

192.168.1.224 - 255 /28

192.168.1.240 - 255 /28

192.168.1.1100 0000

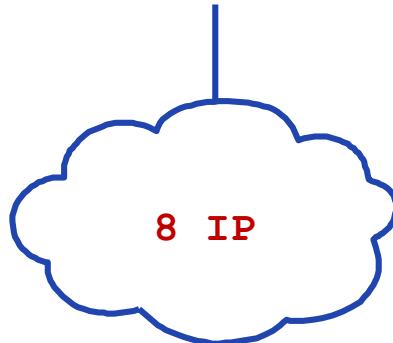
192.168.1.1101 0000

192.168.1.1110 0000

192.168.1.1111 0000

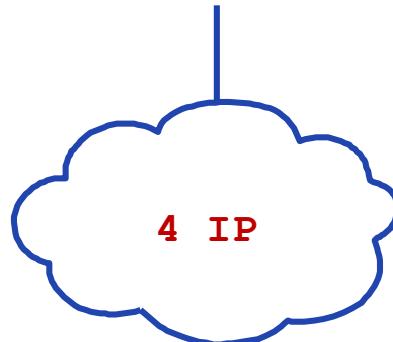
# Kỹ thuật chia mạng con

192.168.1.0/29



|                |      |      |
|----------------|------|------|
| 192.168.1.0000 | 0000 | / 29 |
| 192.168.1.0000 | 0001 | / 29 |
| 192.168.1.0000 | 0010 | / 29 |
| 192.168.1.0000 | 0011 | / 29 |
| 192.168.1.0000 | 0100 | / 29 |
| 192.168.1.0000 | 0101 | / 29 |
| 192.168.1.0000 | 0110 | / 29 |
| 192.168.1.0000 | 0111 | / 29 |

192.168.1.0/30



|                |      |      |
|----------------|------|------|
| 192.168.1.0000 | 0000 | / 30 |
| 192.168.1.0000 | 0001 | / 30 |
| 192.168.1.0000 | 0010 | / 30 |
| 192.168.1.0000 | 0011 | / 30 |

# Kỹ thuật chia mạng con

| Subnet                  | Network  | Host   |
|-------------------------|----------|--------|
| 192.168.1.0 0000000 /25 | 2 mạng   | 128 IP |
| 192.168.1.00 000000 /26 | 4 mạng   | 64 IP  |
| 192.168.1.000 00000 /27 | 8 mạng   | 32 IP  |
| 192.168.1.0000 0000 /28 | 16 mạng  | 16 IP  |
| 192.168.1.00000 000 /29 | 32 mạng  | 8 IP   |
| 192.168.1.000000 00 /30 | 64 mạng  | 4 IP   |
| 192.168.1.0000000 0 /30 | 128 mạng | 2 IP   |

|            | /24   | /25     | /26     | /27     | /28     |
|------------|-------|---------|---------|---------|---------|
| 172.16.0.0 | 0-255 | 0-127   | 0-63    | 0-31    | 0-15    |
|            |       | 128-255 | 64-127  | 32-63   | 16-31   |
|            |       |         | 128-191 | 64-95   | 32-47   |
|            |       |         | 192-255 | 96-127  | 48-63   |
|            |       |         |         | 128-159 | 64-79   |
|            |       |         |         | 160-191 | 80-95   |
|            |       |         |         | 192-223 | 96-111  |
|            |       |         |         | 224-255 | 112-127 |
|            |       |         |         |         | 128-143 |
|            |       |         |         |         | 144-159 |
|            |       |         |         |         | 160-175 |
|            |       |         |         |         | 176-191 |
|            |       |         |         |         | 192-207 |
|            |       |         |         |         | 208-223 |
|            |       |         |         |         | 224-239 |
|            |       |         |         |         | 240-255 |

## Ví dụ 2

Xác định địa chỉ mạng, địa chỉ Broadcast và dải địa chỉ của mạng sau: **172.16.0. 122 /26**

B1: Chuyển địa chỉ IP và Subnet Mask về dạng nhị phân và thực hiện phép tính IP and Subnet Mask.

|             |          |          |          |          |
|-------------|----------|----------|----------|----------|
|             | 172      | 16       | 0        | 122      |
| IP          | 10101100 | 00001000 | 00000000 | 01111010 |
| Subnet mask | 11111111 | 11111111 | 11111111 | 11000000 |
| Kết quả AND | 10101100 | 00001000 | 00000000 | 01000000 |

## B2: Xác định Network\_Id và Host\_Id, dải host.

|             |          |          |          |          |
|-------------|----------|----------|----------|----------|
| Kết quả AND | 10101100 | 00001000 | 00000000 | 01000000 |
| Network_Id  | 172      | 16       | 0        | 64       |
| Host_Id     |          |          |          | 58       |
| Host đầu    | 172      | 16       | 0        | 65       |
| Host cuối   | 172      | 16       | 0        | 126      |
| Broadcast   | 172      | 16       | 0        | 127      |

## Ví dụ 3

Xác định địa chỉ mạng, địa chỉ Broadcast và dải đại chỉ của mạng sau:

172.16.0.200 /27

172.16.0. 192 (Network Address)

172.16.0. 193

172.16.0. 194

172.16.0. 223 (Broadcast Address)

## Ví dụ 4

172.16.0. 50 /28

172.16.0. 48 (Network Address)

172.16.0. 49

172.16.0. 50

172.16.0. 63 (Broadcast Address)

# Làm thế nào để một host có địa chỉ IP?

- ❖ Khai báo cố định bởi người quản trị hệ thống
  - ❖ Windows: control-panel->network->configuration->tcp/ip->properties
  - ❖ UNIX: /etc/rc.config
  - ❖ → Cấp phát địa chỉ IP cố định (cấp phát tĩnh) cho các host sẽ dẫn đến tình trạng lãng phí địa chỉ IP
- ❖ Dịch vụ cấp phát địa chỉ động cho host (**DHCP-Dynamic Host Configuration Protocol**)

# DHCP: Dynamic Host Configuration Protocol

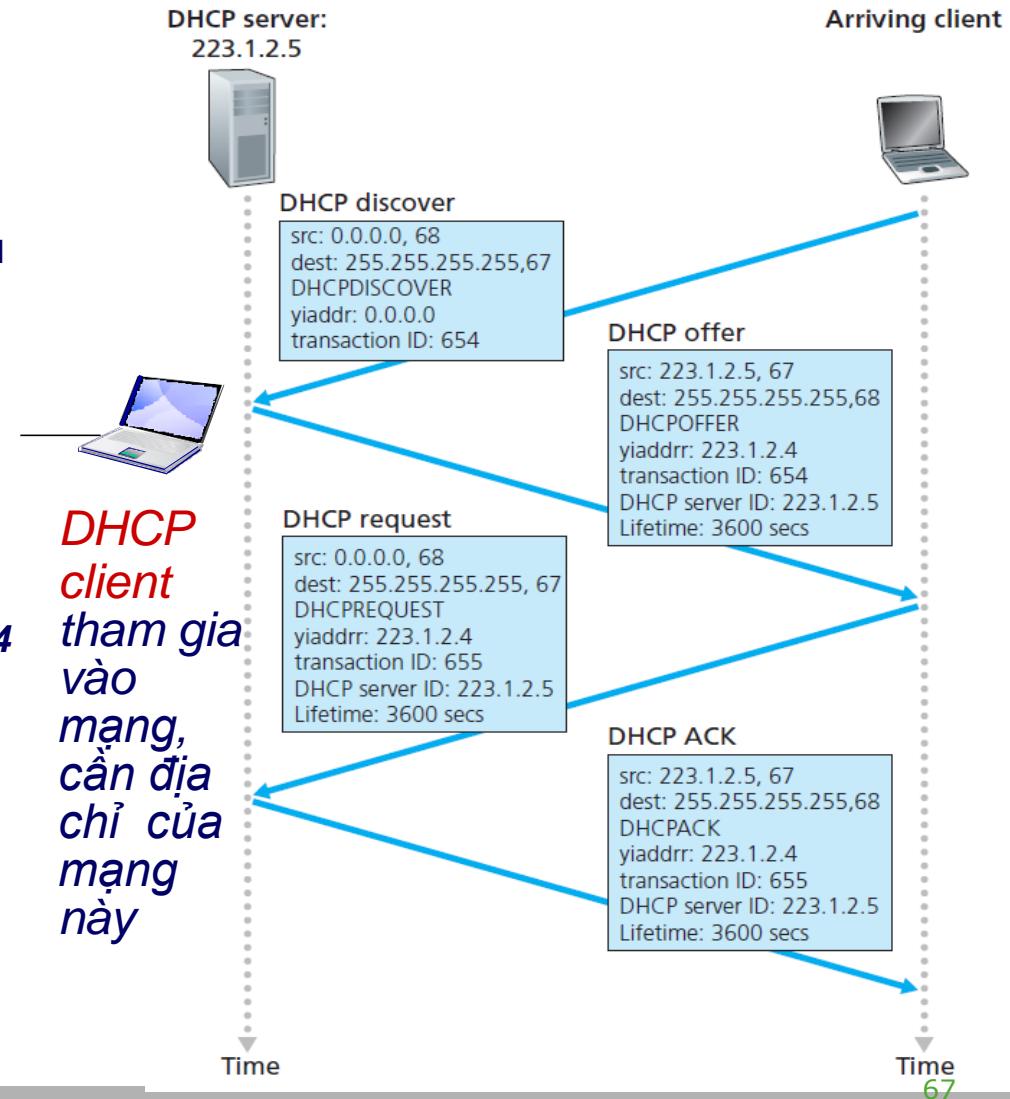
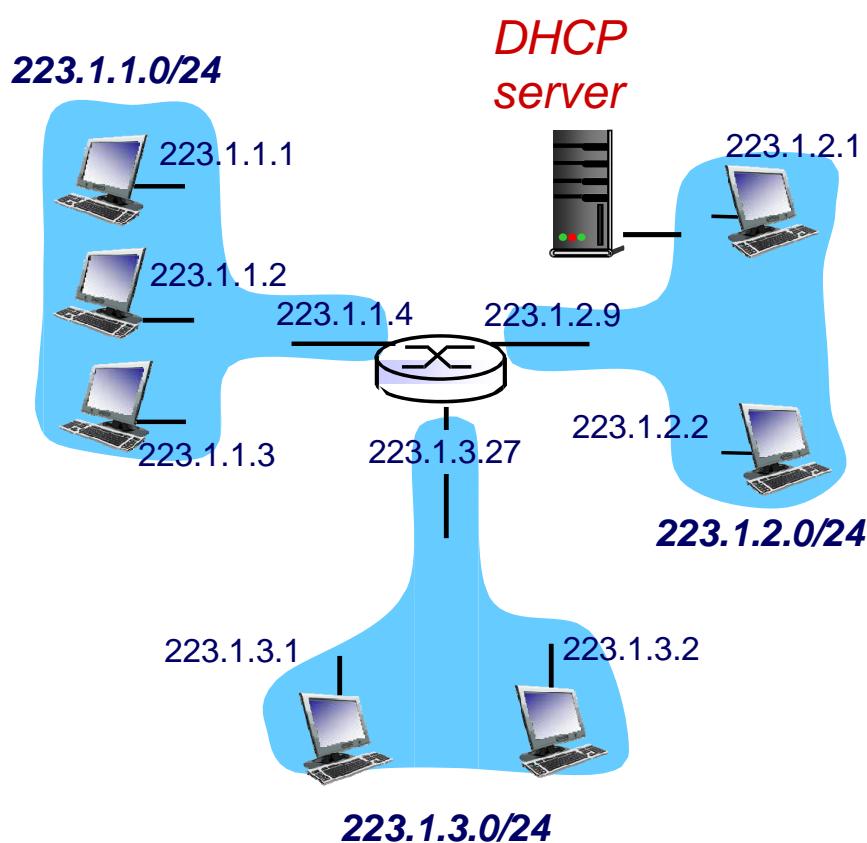
*DHCP:* cho phép host lấy địa chỉ IP động từ network server khi tham gia vào mạng

- ❖ Có thể làm mới địa chỉ đang dùng
- ❖ Có thể sử dụng lại các địa chỉ (chỉ giữ địa chỉ khi kết nối)
- ❖ Cho phép thiết bị di động tham gia vào mạng (thời gian kết nối ngắn)

*Cơ bản về hoạt động của DHCP:*

- ❖ Host quảng bá bản tin “DHCP discover”
- ❖ DHCP server trả lời bằng bản tin “DHCP offer”
- ❖ Host yêu cầu địa chỉ IP bằng bản tin “DHCP request”
- ❖ DHCP server gửi địa chỉ: bản tin “DHCP ack”

# Kịch bản DHCP client-server



# Cấp phát địa chỉ IP cho mạng?

Q: Một mạng con lấy địa chỉ IP từ đâu?

A: Lấy phần đã cấp của không gian địa chỉ của nhà cung cấp dịch vụ Internet (ISP)

|                      |  |                         |
|----------------------|--|-------------------------|
| Khối địa chỉ của ISP | <u>11001000 00010111 00010000 00000000</u> | 200.23.16.0/20          |
| Tổ chức 0            | <u>11001000 00010111 00010000</u>          | 00000000 200.23.16.0/23 |
| Tổ chức 1            | <u>11001000 00010111 00010010</u>          | 00000000 200.23.18.0/23 |
| Tổ chức 2            | <u>11001000 00010111 00010100</u>          | 00000000 200.23.20.0/23 |
| ...                  | .....                                      | .....                   |
| Tổ chức 7            | <u>11001000 00010111 00011110</u>          | 00000000 200.23.30.0/23 |

# Quản lý địa chỉ IP?

Q: ISP lấy địa chỉ IP từ đâu?

A: ICANN: Internet Corporation for Assigned Names and Numbers (<http://www.icann.org/>)

- cấp phát địa chỉ
- quản lý DNS
- gán tên miền, giải quyết tranh chấp

# Cấu trúc gói tin IPv4

|                     |          |                  |              |                 |  |  |  |
|---------------------|----------|------------------|--------------|-----------------|--|--|--|
| VER                 | IHL      | Type of services | Total length |                 |  |  |  |
| Identification      |          | Flags            |              | Fragment offset |  |  |  |
| Time to live        | Protocol | Header checksum  |              |                 |  |  |  |
| Source address      |          |                  |              |                 |  |  |  |
| Destination address |          |                  |              |                 |  |  |  |
| Options + Padding   |          |                  |              |                 |  |  |  |
| Data                |          |                  |              |                 |  |  |  |

# CHỨC NĂNG CÁC TRƯỜNG

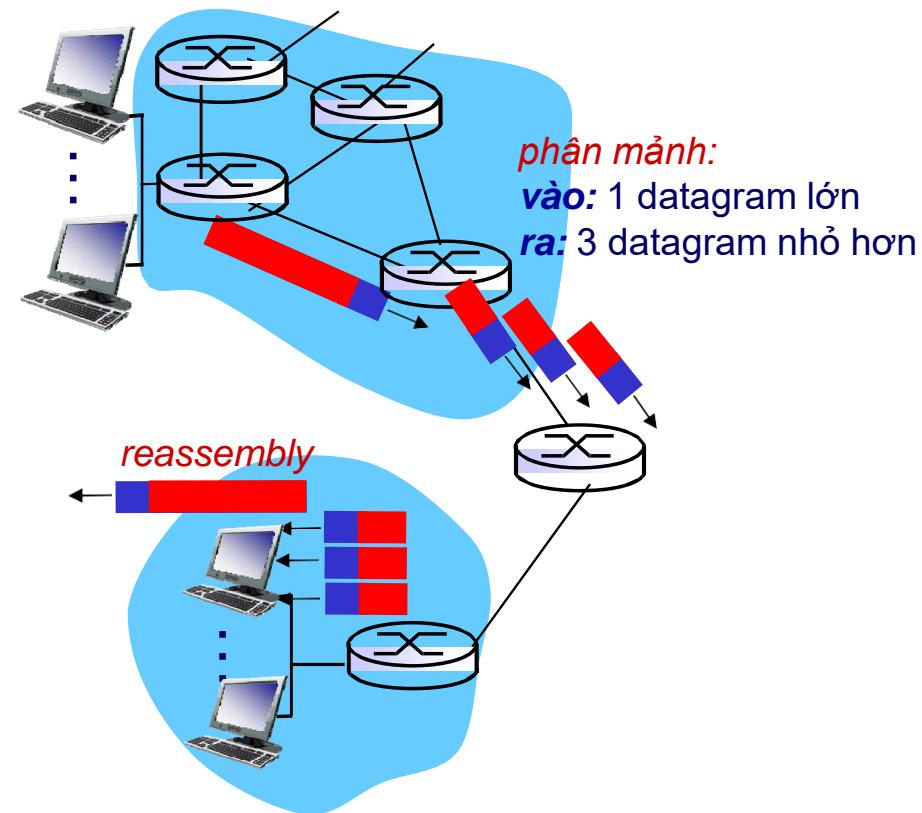
- ❖ VER (4 bits): Version hiện hành của IP được cài đặt.
- ❖ IHL(4 bits): Internet Header Length của Datagram, tính theo đơn vị word (32 bits).
- ❖ Type of service (8 bits): Thông tin về loại dịch vụ và mức ưu tiên của gói IP:
- ❖ Total Length (16 bits): Chỉ độ dài Datagram,
- ❖ Identification (16bits): Định danh cho một Datagram trong thời gian sống của nó.
- ❖ Flags (3 bits): Liên quan đến sự phân đoạn (Fragment) các Datagram.
- ❖ Fragment Offset (13 bits): Chỉ vị trí của Fragment trong Datagram.

# CHỨC NĂNG CÁC TRƯỜNG

- ❖ Time To Live (TTL-8 bits): Thời gian sống của một gói dữ liệu.
- ❖ Protocol (8 bits): Chỉ giao thức sử dụng TCP hay UDP.
- ❖ Header Checksum (16 bits): Mã kiểm soát lỗi CRC (Cycle Redundancy Check).
- ❖ Source Address (32 bits): Địa chỉ của trạm nguồn.
- ❖ Destination Address (32 bits): Địa chỉ của trạm đích.
- ❖ Option (độ dài thay đổi): Sử dụng trong trường hợp bảo mật, định tuyến đặc biệt.
- ❖ Padding (độ dài thay đổi): Vùng đệm cho phần Header luôn kết thúc ở 32 bits
- ❖ Data (độ dài thay đổi): Độ dài dữ liệu tối đa là 65.535 bytes, tối thiểu là 8 bytes.

# PHÂN MẢNH IP DATAGRAM

- ❖ Liên kết mạng có MTU (max.transfer size)
  - ❖ Kiểu liên kết khác nhau có MTU khác nhau
- ❖ IP datagram lớn được chia nhỏ trong mạng
  - ❖ 1 datagram thành nhiều datagram
  - ❖ Được ghép lại tại đích
  - ❖ Các bit trong IP header dùng để ghép các phân mảnh



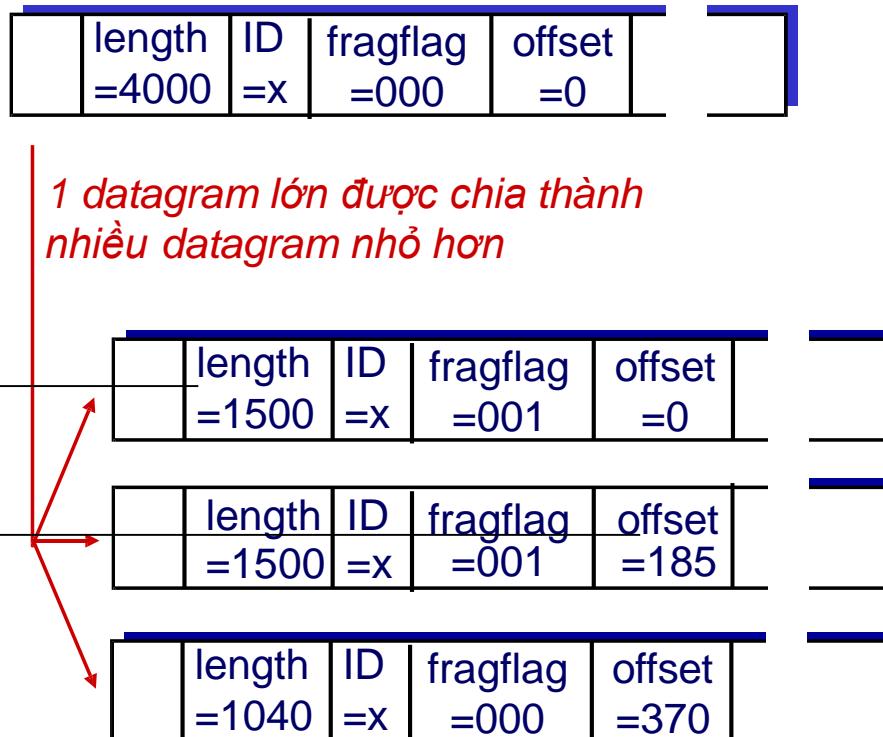
# GHÉP PHÂN MÃNH

ví dụ:

- ❖ datagram kích thước 4000 byte
- ❖ MTU = 1500 byte

1480 byte phần dữ liệu

$$\text{offset} = \frac{1480}{8}$$



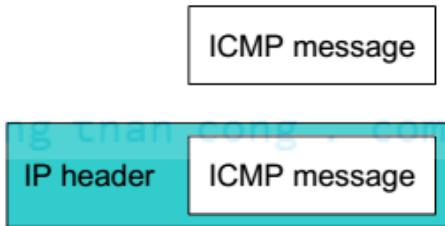
# GIAO THÚC ICMP

- ❖ Giao thức thông báo điều khiển mạng ICMP (Internet Control Message Protocol) là giao thức điều khiển của tầng IP, sử dụng để trao đổi các thông tin điều khiển dòng dữ liệu, thông báo lỗi và các thông tin trạng thái khác của bộ giao thức TCP/IP.
- ❖ Các loại thông điệp ICMP: Các thông điệp ICMP được chia thành hai nhóm: các thông điệp truy vấn và các thông điệp thông báo lỗi.
- ❖ ICMP sử dụng địa chỉ IP nguồn để gửi thông điệp thông báo lỗi cho node nguồn của gói IP.

# ICMP: internet control message protocol

- ❑ Dùng bởi các host và router để giao tiếp thông tin ở mức mạng
  - Thông báo lỗi: unreachable host, network, port, protocol
  - echo request/reply (dùng bởi ping)

- ❑ Gói tin ICMP chứa trong các IP datagram



- ❑ ICMP message: type, code và 8 byte đầu tiên của IP datagram gây ra lỗi

| Type | Code | Mô tả  |
|------|------|--|
| 0    | 0    | echo reply (ping)                                |
| 3    | 0    | không tới được mạng đích                         |
| 3    | 1    | không tới được host đích                         |
| 3    | 2    | không tới được giao thức đích                    |
| 3    | 3    | không tới được cổng đích                         |
| 3    | 6    | không biết mạng đích                             |
| 3    | 7    | không biết host đích                             |
| 4    | 0    | source quench (điều khiển tắc nghẽn, không dùng) |
| 8    | 0    | echo request (ping)                              |
| 9    | 0    | route advertisement                              |
| 10   | 0    | router discovery                                 |
| 11   | 0    | TTL quá hạn                                      |
| 12   | 0    | IP header không hợp lệ                           |

# GIAO THỨC IPv6

## ❖ Những hạn chế của IPv4

- Sự thiếu hụt địa chỉ
- Cấu trúc định tuyến không hiệu quả
- Hạn chế tính bảo mật và kết nối đầu cuối – đầu cuối

IPv6 là phiên bản kế thừa phát triển từ IPv4 còn gọi là giao thức IPng (Next General Internet Protocol).

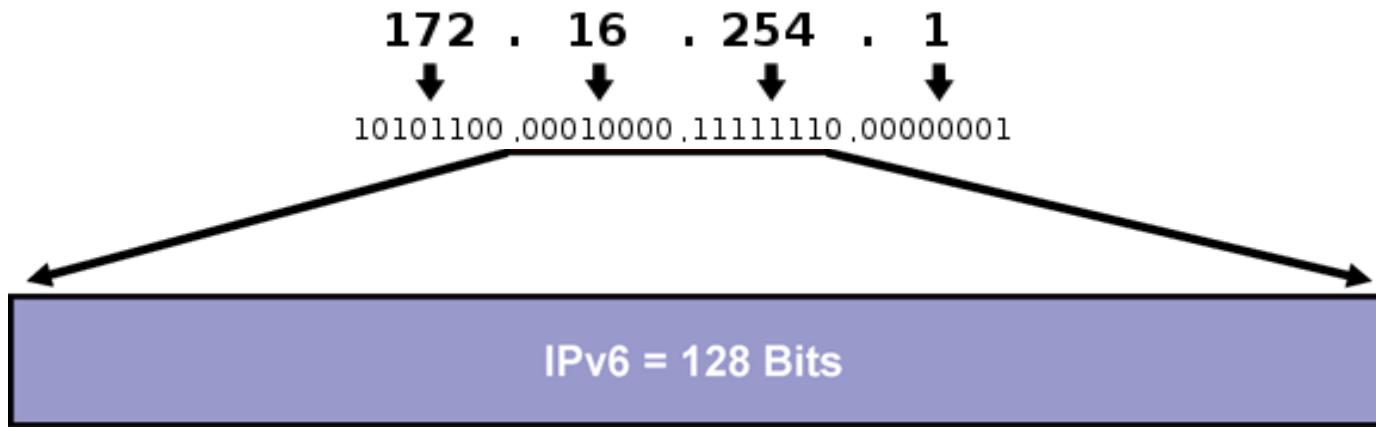
# CÁC ĐẶC TRƯNG CỦA IPV6

- ❖ Đơn giản hoá Header: Một số trường trong Header của IPv4 bị bỏ hoặc chuyển thành các trường tùy chọn. Giảm thời gian xử lý và tăng thời gian truyền.
- ❖ Không gian địa chỉ lớn: Độ dài địa chỉ IPv6 là 128 bit, gấp 4 lần độ dài địa chỉ IPv4. Địa chỉ IPv6 không bị thiếu hụt trong tương lai.
- ❖ Khả năng địa chỉ hoá và chọn đường linh hoạt: IPv6 cho phép nhiều lớp địa chỉ với số lượng các node. Cho phép các mạng đa mức và phân chia địa chỉ thành các mạng con riêng lẻ.
- ❖ Có khả năng tự động trong việc đánh địa chỉ. Mở rộng khả năng chọn đường bằng cách thêm trường “Scop” vào địa chỉ quảng bá (Multicast).

# CÁC ĐẶC TRƯNG CỦA IPV6

- ❖ Tự động cấu hình địa chỉ: Khả năng tự cấu hình của IPv6 được gọi là khả năng cắm và chạy (Plug and Play). Tính năng này cho phép tự cấu hình địa chỉ cho giao diện mà không cần sử dụng các giao thức DHCP.
- ❖ Khả năng bảo mật: IPsec bảo vệ và xác nhận các gói tin IP:
- ❖ Chất lượng dịch vụ QoS (Quanlity Of Service): Do cơ chế xác nhận gói tin ngay trong Header nên việc hỗ trợ QoS có thể thực hiện được ngay cả khi gói tin được mã hóa qua IPsec.
- ❖ Khả năng mở rộng: Thêm vào trường Header mở rộng tiếp ngay sau Header, IPv6 có thể được mở rộng thêm các tính năng mới một cách .
- ❖ Tính di động: IPv6 cho phép nhiều thiết bị di động kết nối vào Internet qua mạng công cộng nhờ sóng vô tuyến.

# ĐẠI CHỈ IPV6



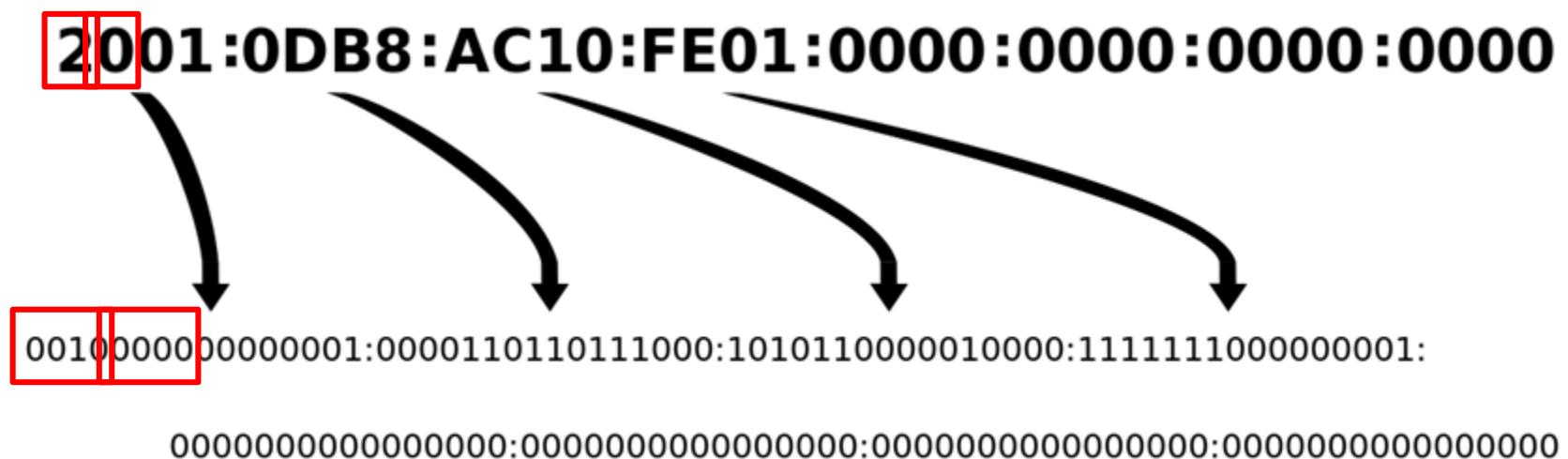
IPv4: 32 bits chia làm 4 octat, mỗi octat 1 bytes

- $\approx 4,200,000,000$  địa chỉ

IPv6

- 128 bits or 16 bytes: số bít gấp 4 lần IPv4
  - $\approx 3.4 * 10^{38}$  địa chỉ
  - $\approx 340,282,366,920,938,463,374,607,432,768,211,456$
  - $\approx 5 * 10^{28}$  địa chỉ cho một người

# BIỂU DIỄN ĐỊA CHỈ IPV6



| Decimal | Hexadecimal | Binary |
|---------|-------------|--------|
| 0       | 0           | 0000   |
| 1       | 1           | 0001   |
| 2       | 2           | 0010   |
| 3       | 3           | 0011   |
| 4       | 4           | 0100   |
| 5       | 5           | 0101   |
| 6       | 6           | 0110   |
| 7       | 7           | 0111   |
| 8       | 8           | 1000   |
| 9       | 9           | 1001   |
| 10      | A           | 1010   |
| 11      | B           | 1011   |
| 12      | C           | 1100   |
| 13      | D           | 1101   |
| 14      | E           | 1110   |
| 15      | F           | 1111   |

# BIỂU DIỄN ĐỊA CHỈ IPV6

**2001:0ABC:00AB:000A:0000:0000:0000:1001**

**2001:ABC:AB:A:0:0:0:1001** :: 0: 0: 0:1001

**2001:ABC:AB:A::1001**

# CẤU TRÚC IPV6

## 32-bit IPv4 address

|     |     |     |     |
|-----|-----|-----|-----|
| YYY | YYY | YYY | YYY |
|-----|-----|-----|-----|

YYY = 8 bits

## 128-bit IPv6 address



|      |      |      |      |      |      |      |      |      |
|------|------|------|------|------|------|------|------|------|
| XXXX |
|------|------|------|------|------|------|------|------|------|

XXXX = 16 bits

# TIÊU ĐỀ ĐỊA CHỈ IPv6

## IPv4 Header

| Version             | IHL      | Type of Service | Total Length    |         |
|---------------------|----------|-----------------|-----------------|---------|
| Identification      |          | Flags           | Fragment Offset |         |
| Time to Live        | Protocol | Header Checksum |                 |         |
| Source Address      |          |                 |                 | Options |
| Destination Address |          |                 |                 | Padding |

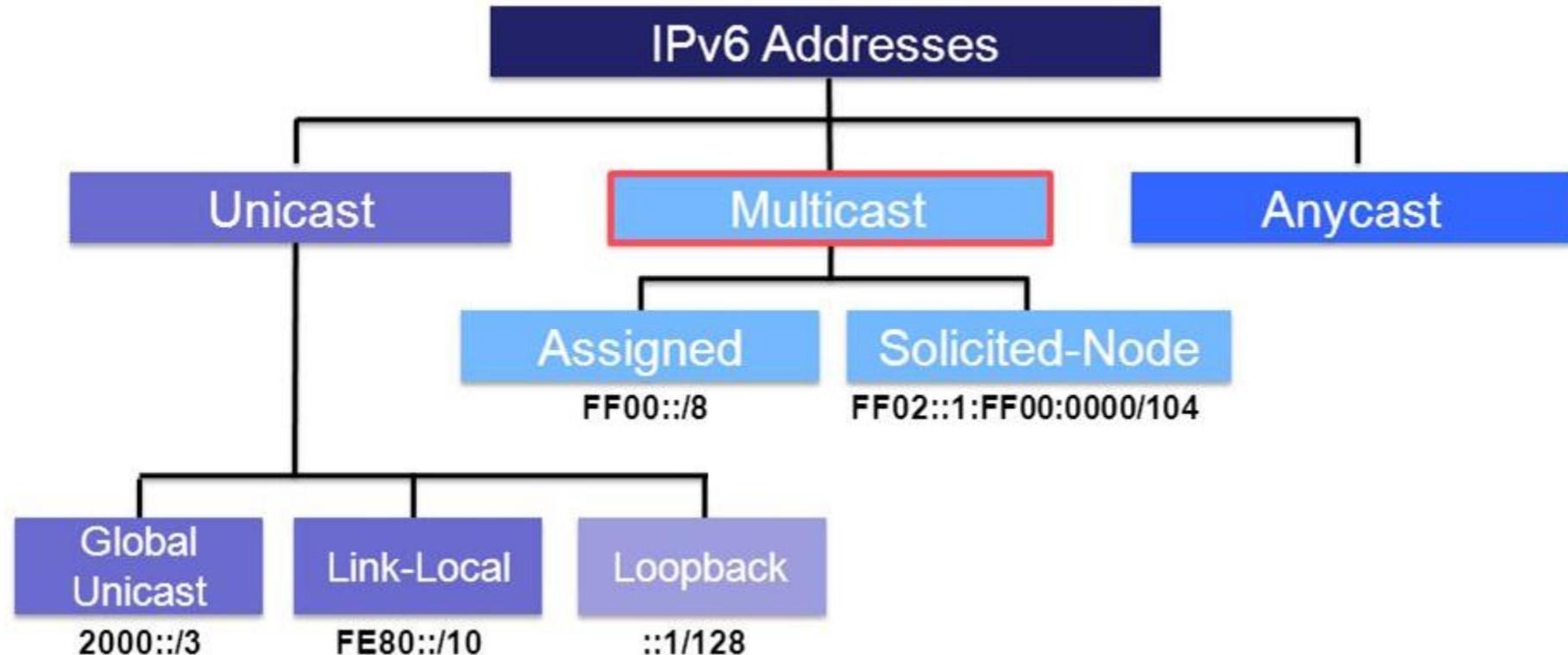
## IPv6 Header

| Version        | Traffic Class | Flow Label  |                     |
|----------------|---------------|-------------|---------------------|
| Payload Length |               | Next Header | Hop Limit           |
| Source Address |               |             | Destination Address |

### Legend

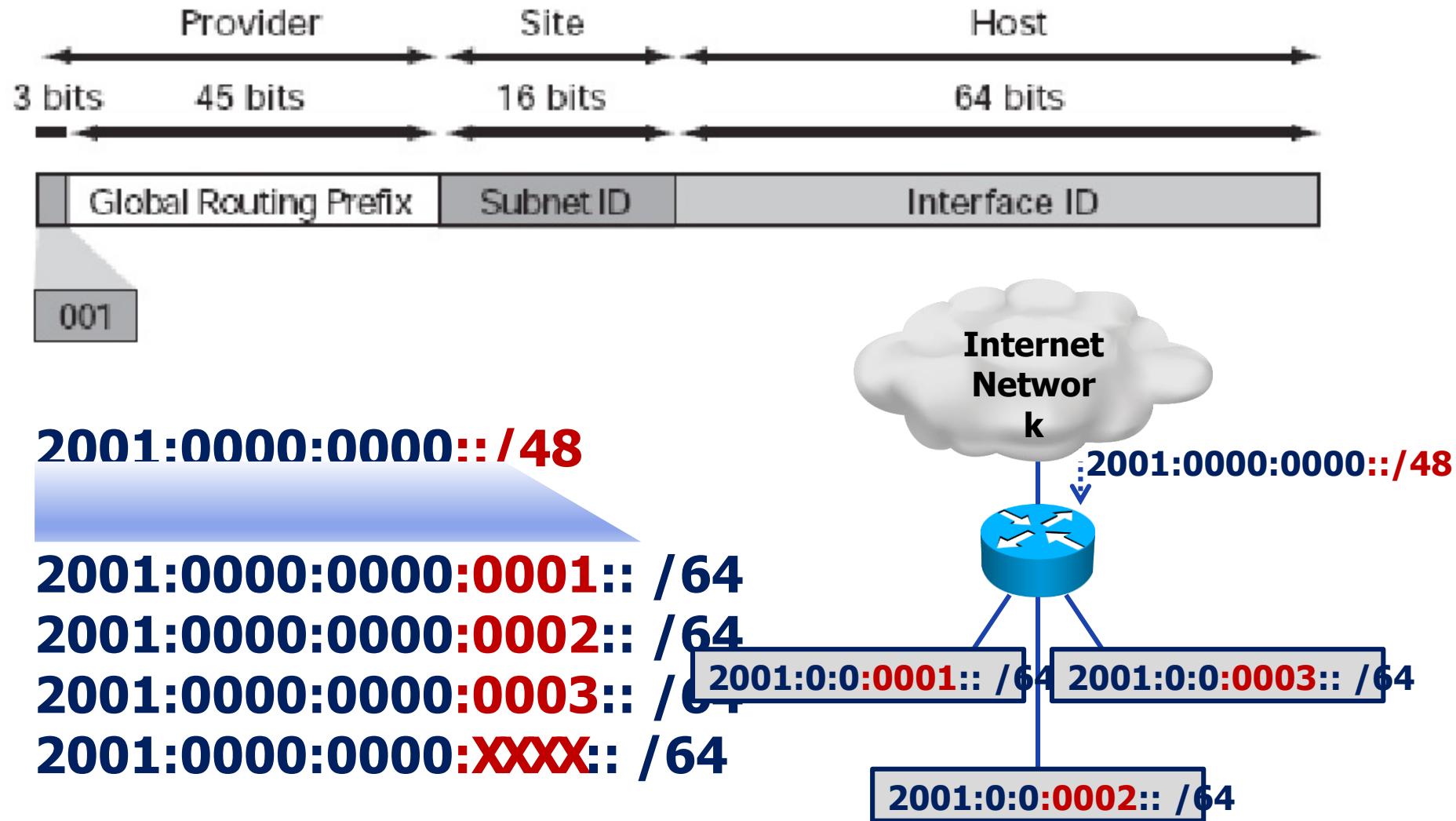
- Field's Name Kept from IPv4 to IPv6
- Fields Not Kept in IPv6
- Name and Position Changed in IPv6
- New Field in IPv6

# PHÂN LOẠI IPV6



IPv6 does not have a Broadcast Address

# IPV6: GLOBAL UNICAST ADDRESS (2000::/3)



# IPV6: MULTICAST ADDRESS

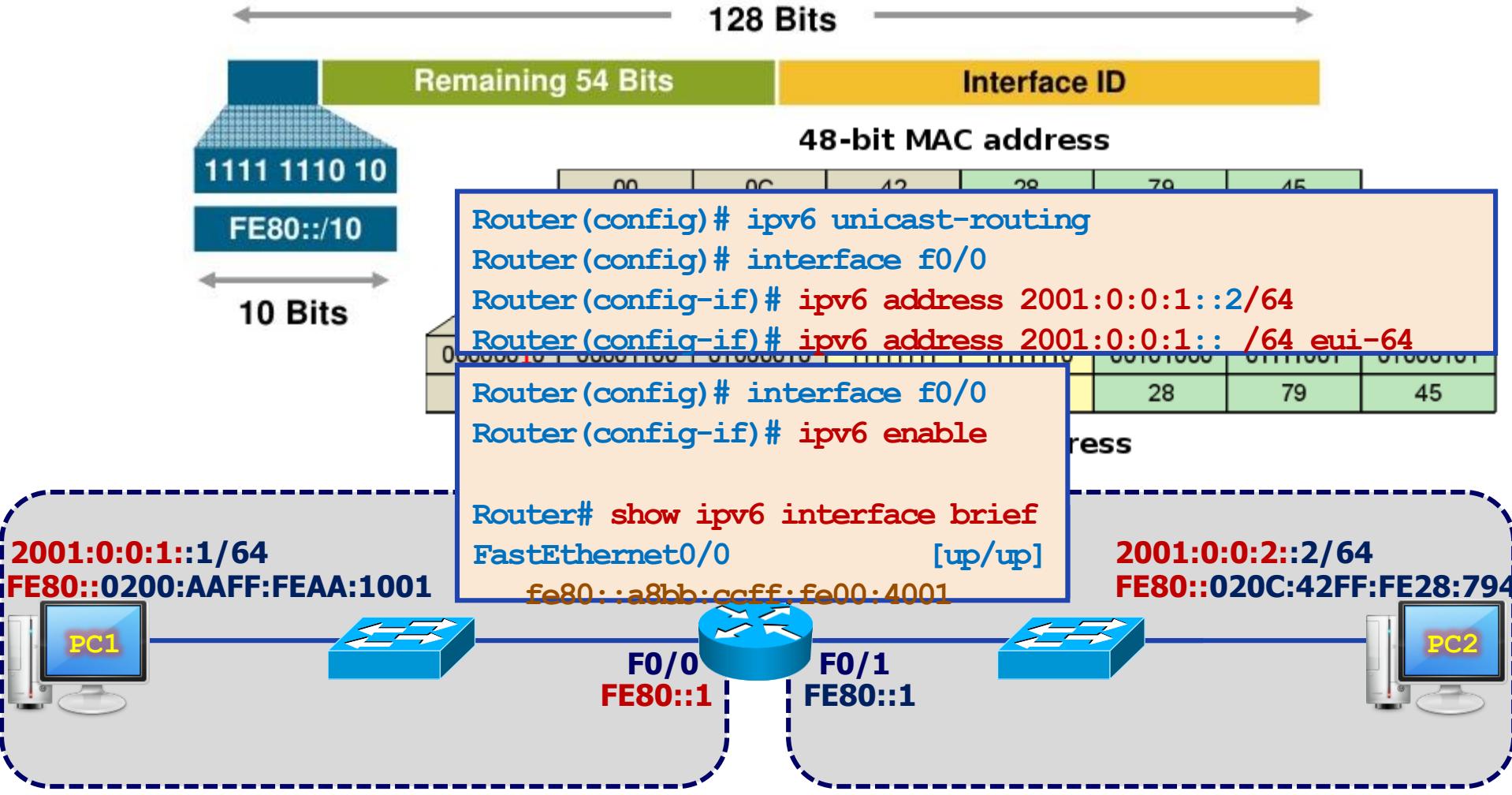
## IPv6

| Address(s)        | Description                |
|-------------------|----------------------------|
| FF02:0:0:0:0:0:1  | All Nodes Address          |
| FF02:0:0:0:0:0:2  | All Routers Address        |
| FF02:0:0:0:0:0:3  | Unassigned                 |
| FF02:0:0:0:0:0:4  | DVMRP Routers              |
| FF02:0:0:0:0:0:5  | OSPFIGP                    |
| FF02:0:0:0:0:0:6  | OSPFIGP Designated Routers |
| FF02:0:0:0:0:0:7  | ST Routers                 |
| FF02:0:0:0:0:0:8  | ST Hosts                   |
| FF02:0:0:0:0:0:9  | RIP Routers                |
| FF02:0:0:0:0:0:10 | EIGRP Routers              |
| FF02:0:0:0:0:0:11 | Mobile-Agents              |
| FF02:0:0:0:0:0:12 | SSDP                       |
| FF02:0:0:0:0:0:13 | All PIM Routers            |
| FF02:0:0:0:0:0:14 | RSVP-ENCAPSULATION         |

## IPv4

| Address(es) | Description                        |
|-------------|------------------------------------|
| 224.0.0.0   | Base Address (Reserved)            |
| 224.0.0.1   | All Systems on this Subnet         |
| 224.0.0.2   | All Routers on this Subnet         |
| 224.0.0.3   | Unassigned                         |
| 224.0.0.4   | DVMRP Routers                      |
| 224.0.0.5   | OSPFIGP OSPFIGP All Routers        |
| 224.0.0.6   | OSPFIGP OSPFIGP Designated Routers |
| 224.0.0.7   | ST Routers                         |
| 224.0.0.8   | ST Hosts                           |
| 224.0.0.9   | RIP2 Routers                       |
| 224.0.0.10  | EIGRP Routers                      |
| 224.0.0.11  | Mobile-Agents                      |
| 224.0.0.12  | DHCP Server / Relay Agent          |
| 224.0.0.13  | All PIM Routers                    |
| 224.0.0.14  | RSVP-ENCAPSULATION                 |

# IPV6: LINK-LOCAL ADDRESS



# IPV6: LINK-LOCAL ADDRESS

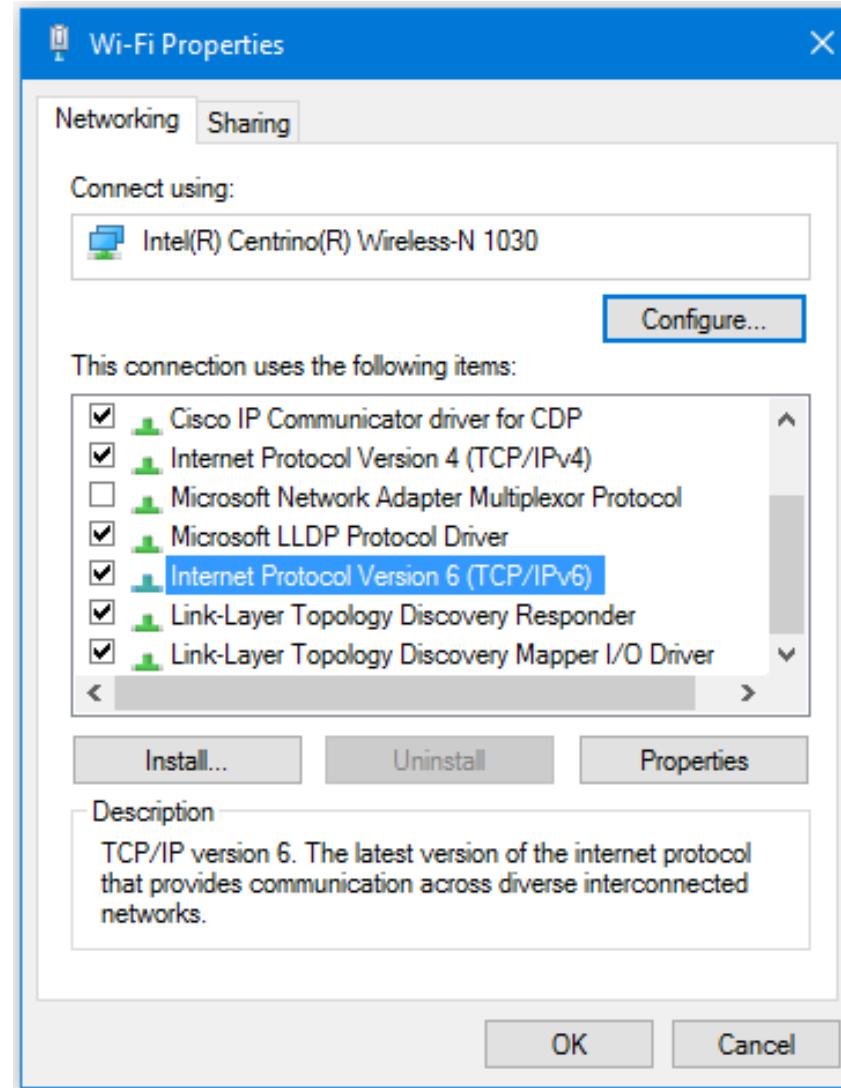


```
C:\Users\Admin>ipconfig  
Windows IP Configuration
```

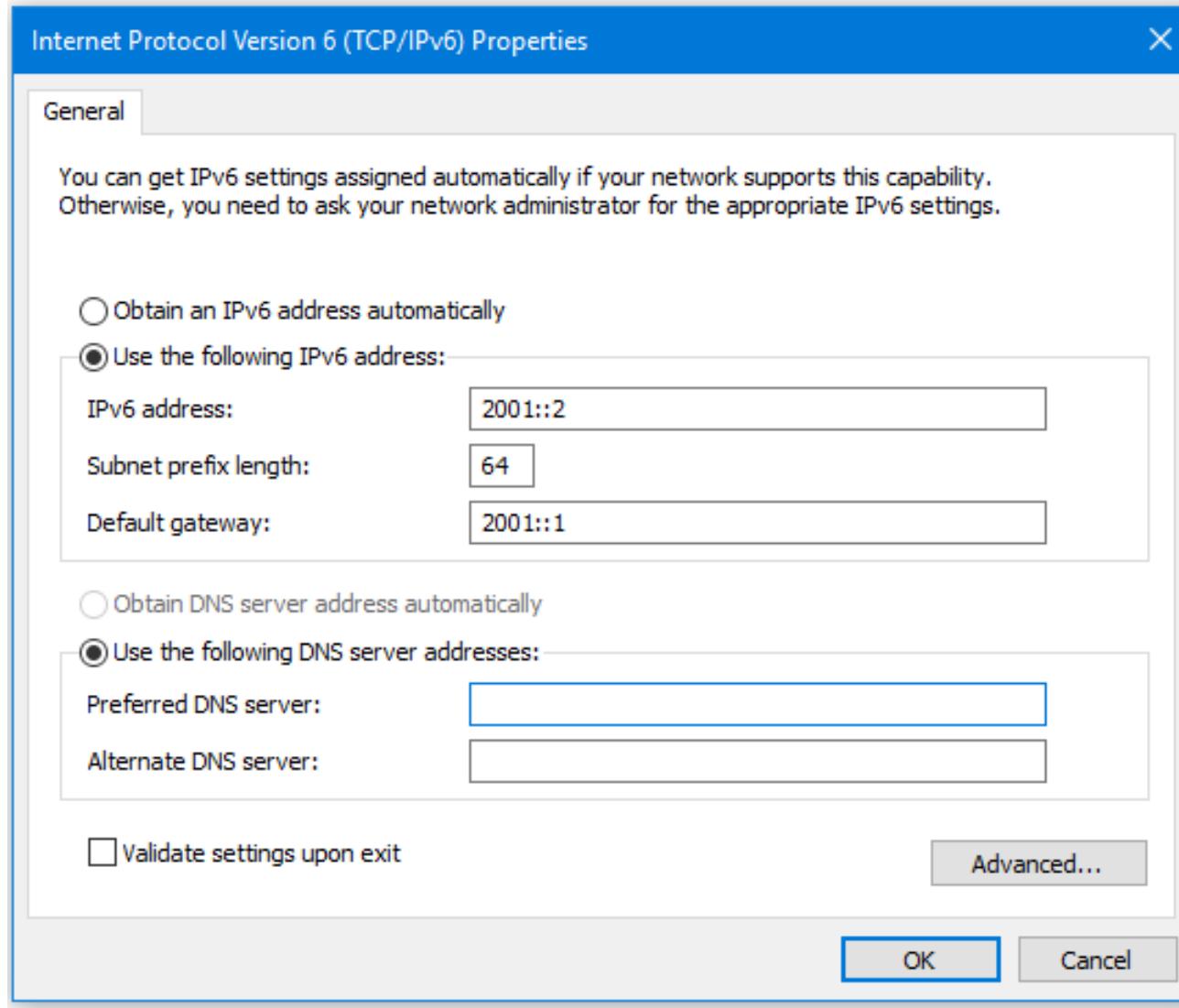
```
Ethernet adapter Local Area Connection:
```

|  |   |                              |
|--|---|------------------------------|
| Connection-specific DNS Suffix . . . . . | : |                              |
| IPv6 Address . . . . .                   | : | fcab:bebc:abac:100::1000     |
| Link-local IPv6 Address . . . . .        | : | fe80::88c2:66c3:3049:1172%10 |
| IPv4 Address . . . . .                   | : | 192.168.1.145                |
| Subnet Mask . . . . .                    | : | 255.255.255.0                |
| Default Gateway . . . . .                | : | 192.168.1.1                  |

# IPV6: CONFIGURING ADDRESS



# IPV6: CONFIGURING ADDRESS



# SO SÁNH IPv4 VÀ IPv6

| IPv4 Address  | IPv6 Address   |
|---|--|
| Phân lớp địa chỉ (Lớp A, B, C và D)                                 | Không phân lớp địa chỉ. Cấp phát theo tiền tố  |
| Lớp D là Multicast (224.0.0.0/4)                                    | Địa chỉ multicast có tiền tố FF00::/8  |
| Sử dụng địa chỉ Broadcast   | Không có Broadcast, thay bằng Anycast  |
| Địa chỉ unspecified là 0.0.0.0                                      | Địa chỉ Unspecified là ::  |
| Địa chỉ Loopback 127.0.0.1  | Địa chỉ Loopback là ::1  |
| Sử dụng địa chỉ Public  | Tương ứng là địa chỉ Unicast toàn cầu  |
| Địa chỉ IP riêng (10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16)    | Địa chỉ Site-Local (FEC0::/48)   |
| Địa chỉ tự cấu hình (169.254.0.0/16)                                | Địa chỉ Link-Local (FE80::/64)   |
| Dạng biểu diễn: chuỗi số thập phân cách nhau bởi dấu chấm           | Dạng biểu diễn: chuỗi số Hexa cách nhau bởi dấu hai chấm; có thể nhóm chuỗi số 0 liền nhau vào một kí tự |
| Sử dụng mặt nạ mạng con   | Chỉ sử dụng kí hiệu tiền tố để chỉ mạng  |
| Phân giải tên miền DNS: bản ghi tài nguyên địa chỉ máy chủ IPv4 (A) | Phân giải tên miền DNS: bản ghi tài nguyên địa chỉ máy chủ IPv6 (AAAA)                                   |
| Tên miền ngược: IN-ADDR.ARPA  | Tên miền ngược: IP6.INT domain   |

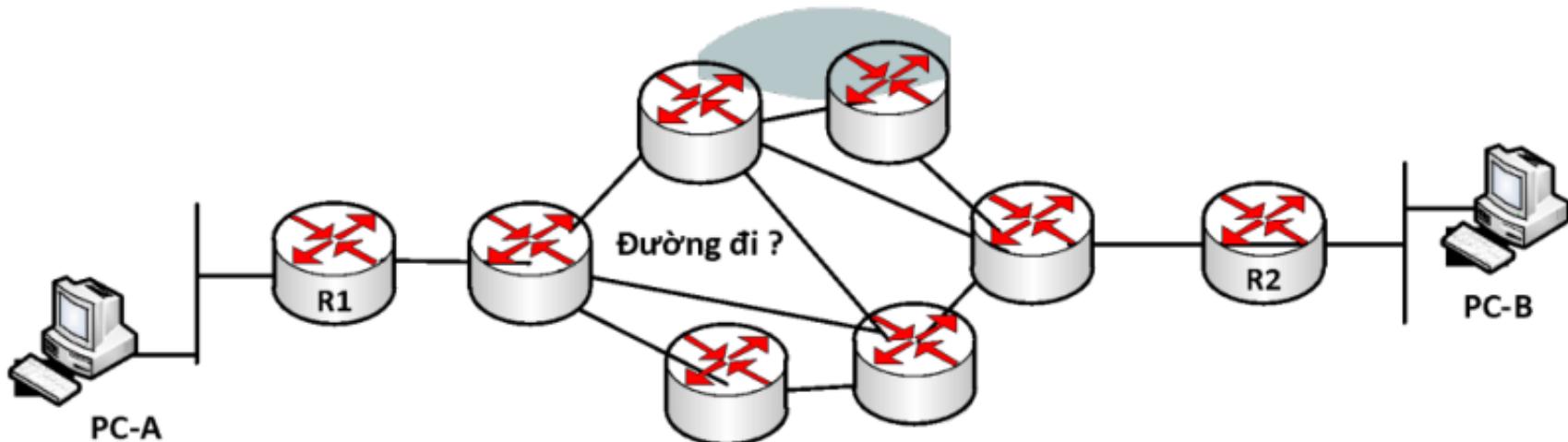
# CHƯƠNG 4: TẦNG MẠNG

- 
- 1 • Giới thiệu
  - 2 • Giao thức IP
  - 3 • Định tuyến
  - 4 • Dẫn đường trong internet

# VII KỸ THUẬT ĐỊNH TUYẾN

## 1. ROUTING - ĐỊNH TUYẾN

- Định tuyến là chức năng của Router giúp xác định quá trình tìm đường đi cho các gói tin để truyền dữ liệu từ nguồn tới đúng đích cần gửi.



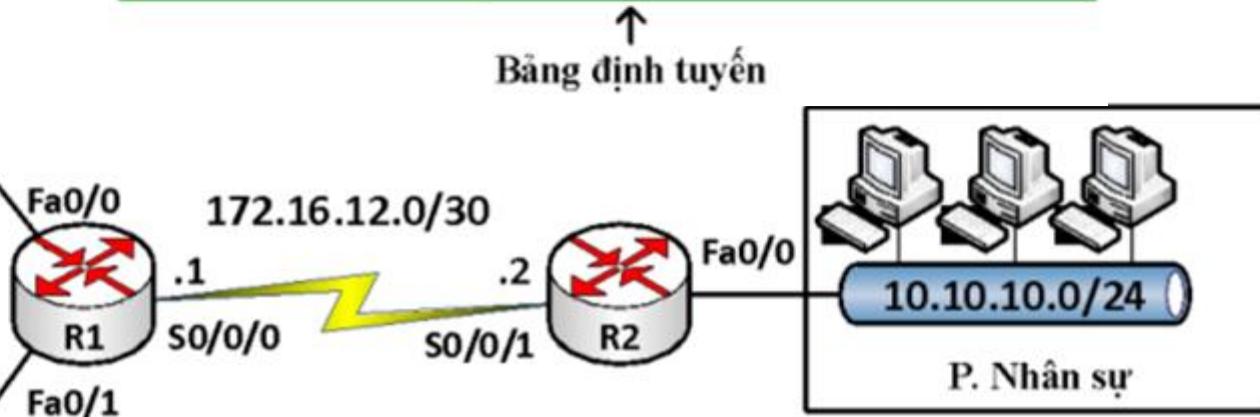
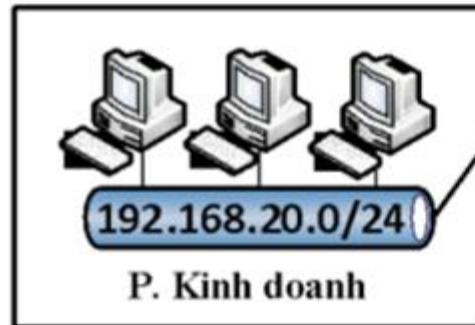
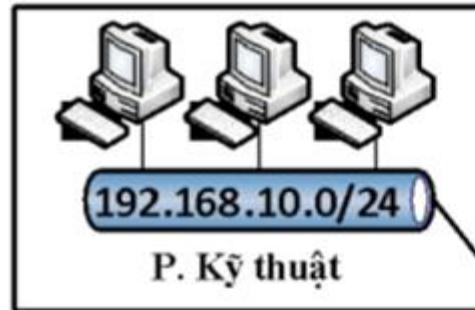
# ROUTER

- ❖ Router là một thiết bị ở lớp 3. Router quyết định chuyển gói dựa trên địa chỉ mạng của gói dữ liệu. Router sử dụng bảng định tuyến để ghi lại địa chỉ lớp 3 của các mạng kết nối trực tiếp vào router và các mạng mà router học được từ các router láng giềng.
- ❖ Mục tiêu của router là thực hiện các việc sau:
  - Kiểm tra dữ liệu lớp 3 của gói nhận được
  - Chọn đường tốt nhất cho gói dữ liệu
  - Chuyển mạch gói ra cổng tương ứng
- ❖ Router ko bị bắt buộc phải chuyển các gói quảng bá

# ROUTING - ĐỊNH TUYẾN

- Router dựa vào địa chỉ IP đích (destination IP) trong các gói tin và sử dụng bảng định tuyến (routing table) để xác định đường đi cho chúng.
- Trong bảng định tuyến, mỗi mạng mà router có thể chuyển đi (mạng đích) thể hiện bằng một dòng.
- Mỗi mạng này có được có thể do chúng đang kết nối trực tiếp với router đang xét hay router học được thông qua việc cấu hình định tuyến.

# BẢNG ĐỊNH TUYẾN



| Network         | Interface or Next hop |
|-----------------|-----------------------|
| 172.16.12.0/30  | Connected – S0/0/1    |
| 10.10.10.0/24   | Connected – Fa0/0     |
| 192.168.10.0/24 | 172.16.12.1           |
| 192.168.20.0/24 | 172.16.12.1           |

Bảng định tuyến

↓  
Bảng định tuyến

| Network         | Interface or Next hop |
|-----------------|-----------------------|
| 172.16.12.0/30  | Connected – S0/0/0    |
| 192.168.10.0/24 | Connected – Fa0/0     |
| 192.168.20.0/24 | Connected – Fa0/1     |
| 10.10.10.0/24   | 172.16.12.2           |

```
Router>en
Router#show ip route
Codes: L - local, C - connected, S - static, 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
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is 0.0.0.0 to network 0.0.0.0

      20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        20.0.0.0/24 is directly connected, Serial0/1/0
L        20.0.0.2/32 is directly connected, Serial0/1/0
S*      0.0.0.0/0 is directly connected, Serial0/1/0
                  [1/0] via 20.0.0.1

Router#
```

## 2. PHÂN LOẠI ĐỊNH TUYẾN

- ❖ Có 2 phương pháp định tuyến:
  - Định tuyến tĩnh (Static Route)
  - Định tuyến động (Dynamic Route)
- ❖ **Định tuyến tĩnh (Static Route)**
  - Định tuyến tĩnh là loại định tuyến mà trong đó router sử dụng các tuyến đường đi tĩnh để vận chuyển dữ liệu đi.
  - Các tuyến đường đi tĩnh này có được do người quản trị cấu hình thủ công vào các router.

- ❖ Định tuyến động là loại định tuyến mà trong đó router sử dụng các tuyến đường đi động để vận chuyển dữ liệu đi.
- ❖ Các tuyến đường đi động này có được do các router sử dụng các giao thức định tuyến động trao đổi thông tin định tuyến với nhau tạo ra.
- ❖ Một số giao thức định tuyến động phổ biến: RIP, OSPF, EIGRP,...

# ĐỊNH TUYẾN TĨNH

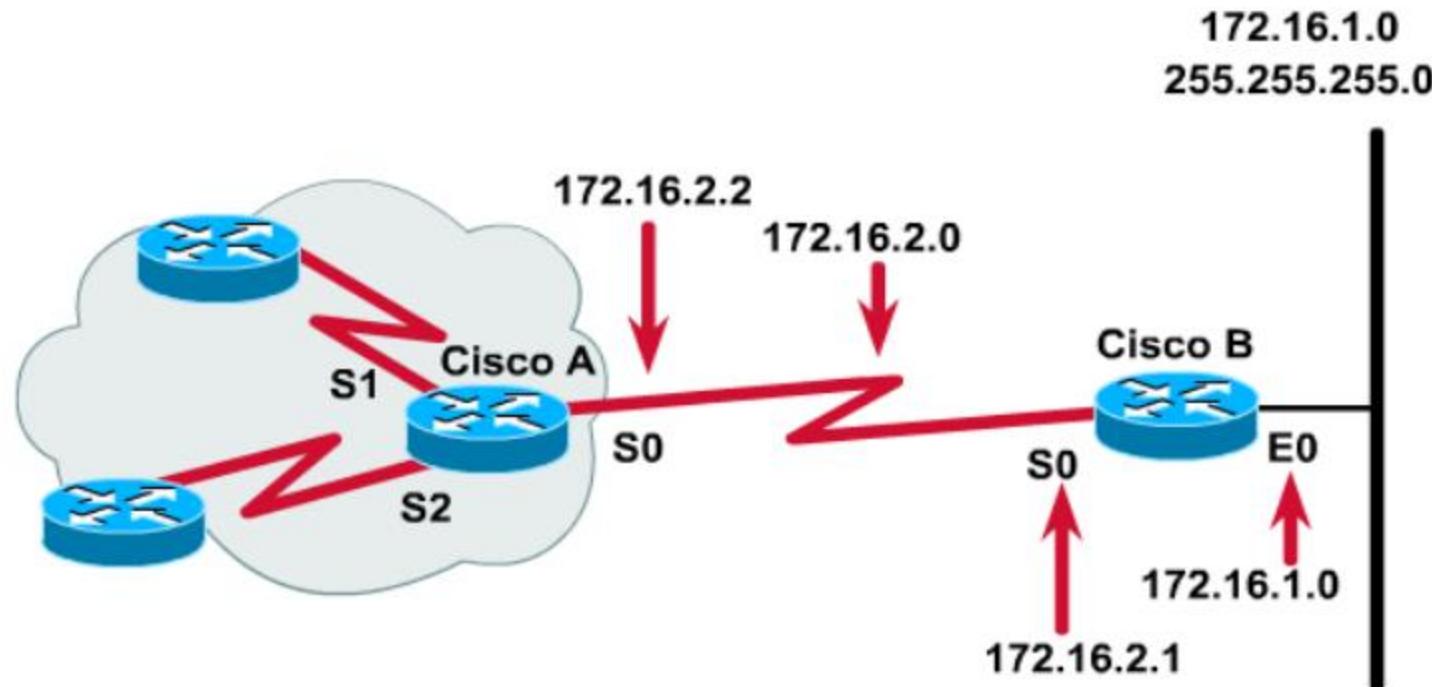
## 1. Định tuyến tĩnh IPv4:

```
R(config)#ip route <destination-net> <subnet-mask> <NextHop  
|OutPort>
```

- ❖ Trong đó:
  - destination-network: Là địa chỉ mạng cần đi tới
  - subnet-mask: subnet mask của destination-network
  - next-hop: địa chỉ IP của router kế tiếp kết nối trực tiếp với router đang xét
  - OutPort: cổng của router mà packet sẽ đi ra

# ĐỊNH TUYẾN TĨNH (STATIC ROUTING)

- Ví dụ: Cấu hình trên router Cisco A để học mạng 172.16.1.0/24



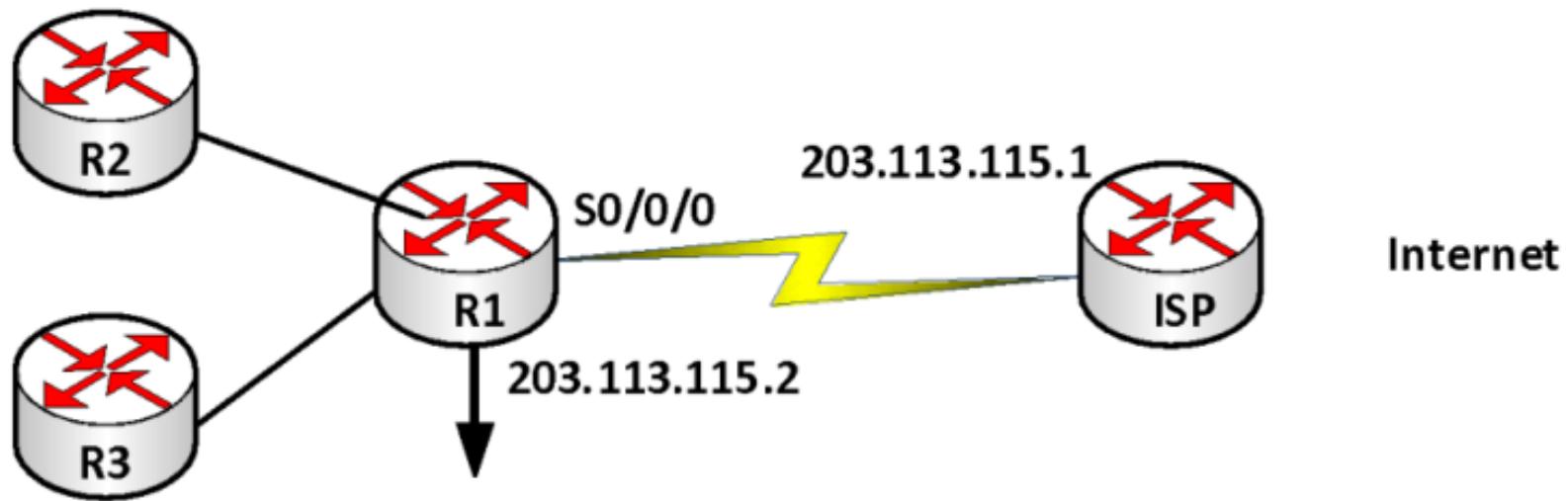
```
CiscoA(config)#ip route 172.16.1.0 255.255.255.0 172.16.2.1
```

```
CiscoA(config)#ip route 172.16.1.0 255.255.255.0 S0
```

# DEFAULT ROUTE

- ❖ Default route nằm ở cuối bảng định tuyến và được sử dụng để gửi các gói tin đi trong trường hợp mạng đích không tìm thấy trong bảng định tuyến.
- ❖ Nó rất hữu dụng trong các mạng dạng “stub network” như kết nối từ mạng nội bộ ra ngoài Internet.

# DEFAULT ROUTE



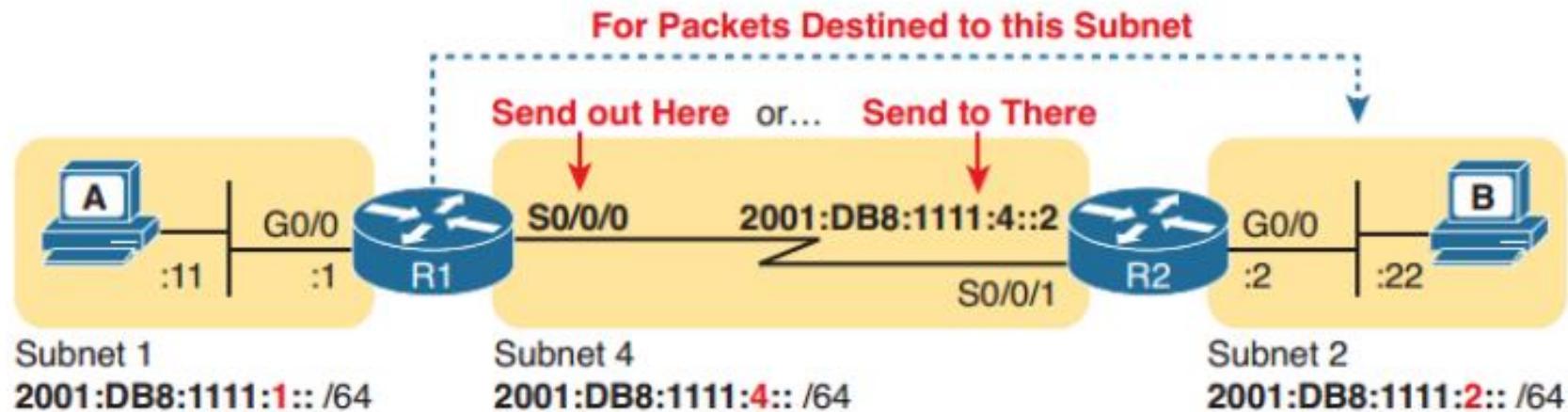
```
R1(config)#ip route 0.0.0.0 0.0.0.0 203.113.115.1
```

```
R1(config)#ip route 0.0.0.0 0.0.0.0 S0/0/0
```

# ĐỊNH TUYẾN TÍNH CHO IPv6

R(config)#Ipv6 unicast-routing

R(config)#ipv6 route prefix-network/prefix-length [OutGoing Interface | Next-Hop]



```
! Static route on router R1
```

```
R1(config)# ipv6 route 2001:db8:1111:2::/64 s0/0/0
```

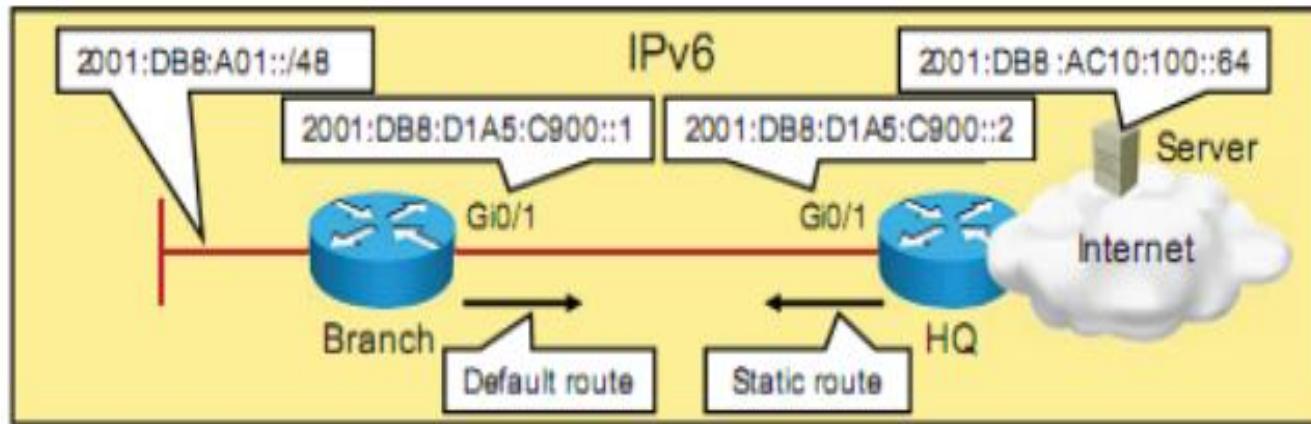
```
! The first command is on router R1, listing R2's global unicast address
```

```
R1(config)# ipv6 route 2001:db8:1111:2::/64 2001:DB8:1111:4::2
```

```
! The first command is on router R1, listing R2's link-local address
```

```
R1(config)# ipv6 route 2001:db8:1111:2::/64 S0/0/0 FE80::FF:FE00:2
```

# DEFAULT ROUTE VIPv6



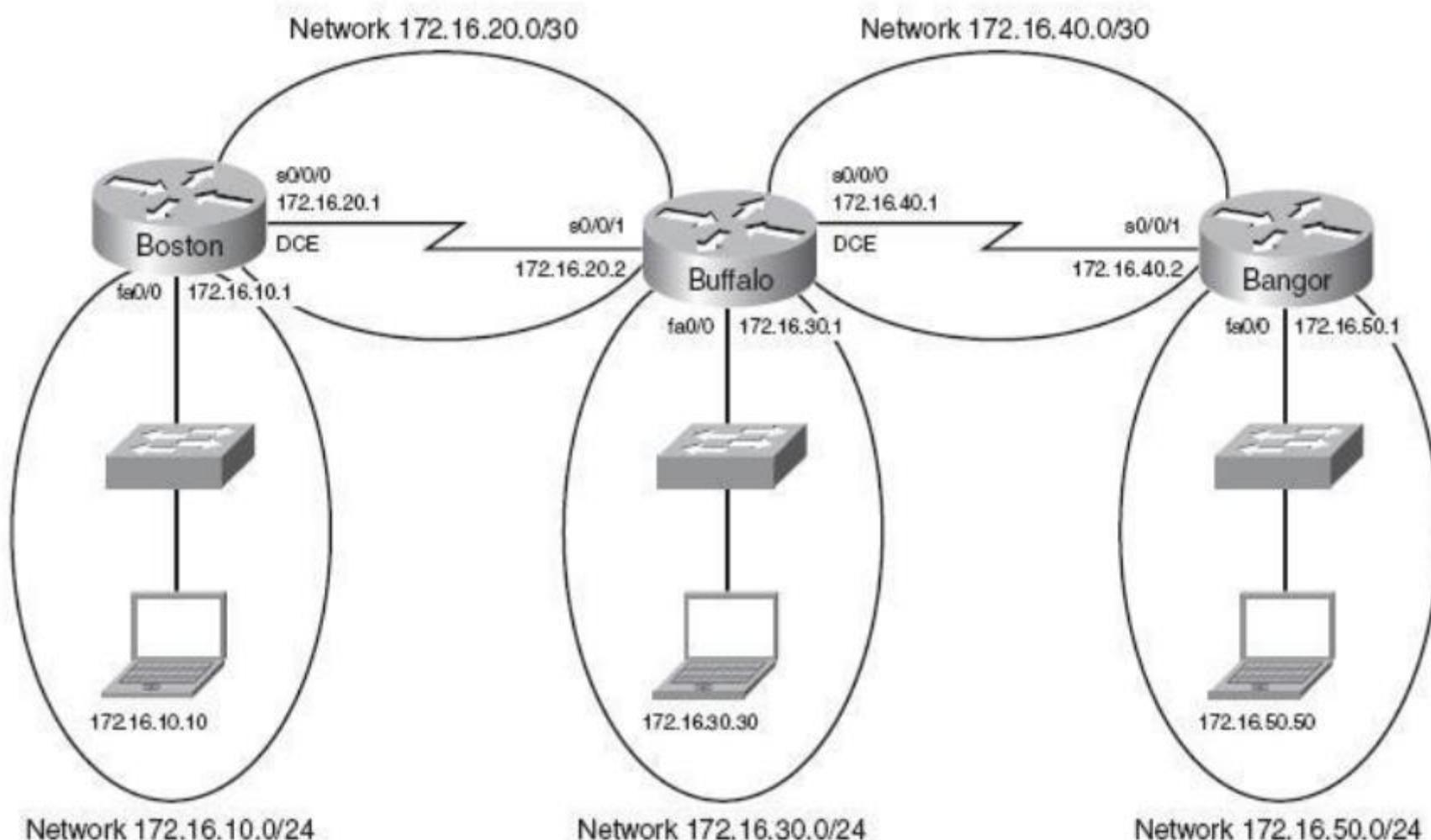
The static IPv6 route is configured on the HQ router:

```
HQ(config)#ipv6 route 2001:DB8:A01::/48 Gi0/1 2001:DB8:D1A5:C900::1
```

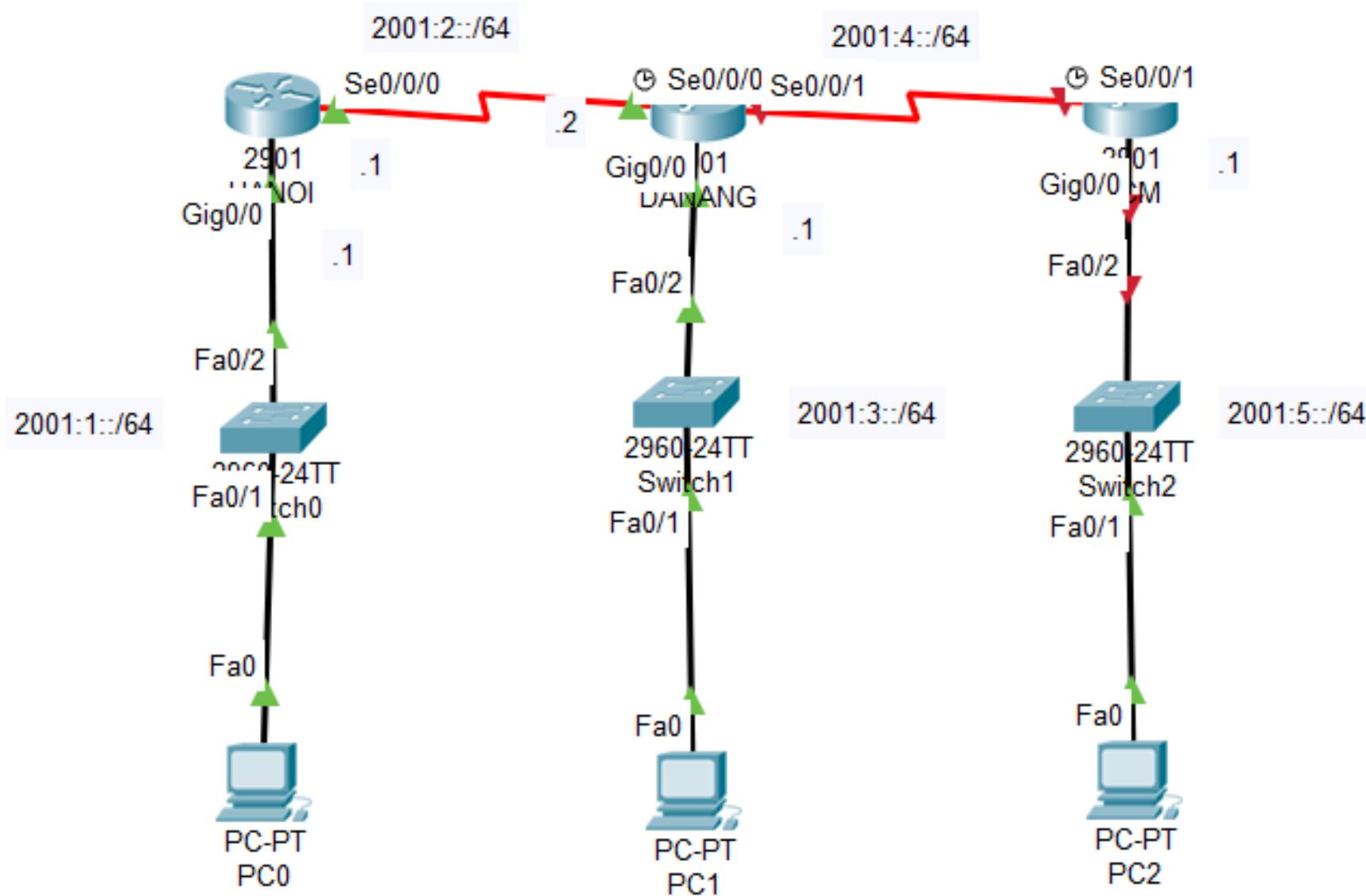
The default IPv6 route is configured on the Branch router:

```
Branch(config)#ipv6 route ::/0 Gi0/1 2001:DB8:D1A5:C900::2
```

# BÀI TẬP 1: ĐỊNH TUYẾN TĨNH IPv4



# BÀI TẬP 2: ĐỊNH TUYỀN TĨNH IPv6



# Boston Router

|   |  |
|---|--|
| Boston> enable  | Chuyển vào chế độ Privileged                       |
| Boston# configure terminal  | Chuyển vào chế độ cấu hình Global Configuration    |
| Boston(config)# ip route 172.16.30.0<br>255.255.255.0 172.16.20.2   | Cấu hình một static route sử dụng địa chỉ next-hop |
| Boston(config)# ip route 172.16.40.0<br>255.255.255.252 172.16.20.2 | Cấu hình một static route sử dụng địa chỉ next-hop |
| Boston(config)# ip route 172.16.50.0<br>255.255.255.0 172.16.20.2   | Cấu hình một static route sử dụng địa chỉ next-hop |
| Boston(config)# exit  | Chuyển về chế độ cấu hình Privileged               |
| Boston# copy run start  | Lưu file cấu hình đang chạy trên RAM vào NVRAM.    |

# Buffalo Router

|   |   |
|---|---|
| Buffalo> enable   | Chuyển về chế độ cấu hình Privileged.                         |
| Buffalo# configure terminal                                   | Chuyển vào chế độ cấu hình Global Configuration               |
| Buffalo(config)# ip route<br>172.16.10.0 255.255.255.0 s0/0/1 | Cấu hình một static route sử dụng một interface đang tồn tại. |
| Buffalo(config)# ip route<br>172.16.50.0 255.255.255.0 s0/0/0 | Cấu hình một static route sử dụng một interface đang tồn tại. |
| Buffalo(config)# exit   | Thoát ra chế độ Privileged.                                   |
| Buffalo# copy run start                                       | Lưu file cấu hình đang chạy trên RAM vào NVRAM                |

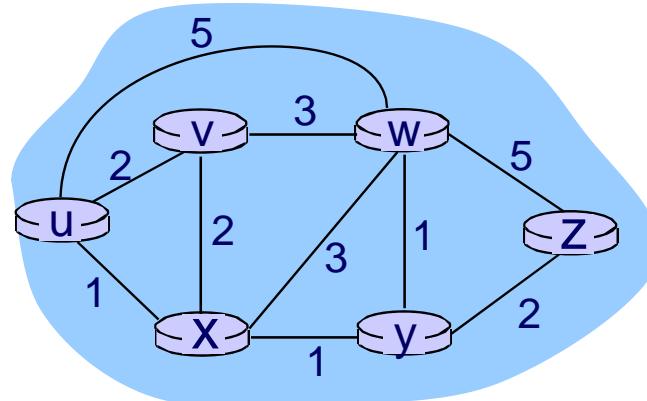
# Bangor Router

|  |  |
|--|--|
| Bangor> enable                                     | Chuyển vào chế độ cấu hình Privileged.           |
| Bangor# configure terminal                         | Chuyển vào chế độ cấu hình Global Configuration. |
| Bangor(config)# ip route 0.0.0.0<br>0.0.0.0 s0/0/1 | Cấu hình static route sử dụng default route      |
| Bangor# copy run start                             | Lưu file cấu hình đang chạy trên RAM vào NVRAM.  |

|                      |                     |
|----------------------|---------------------|
| Bangor#show ip route | Xem bảng định tuyến |
|----------------------|---------------------|

# THUẬT TOÁN ĐỊNH TUYẾN

- ❖ Thuật toán định tuyến: xác định đường đi có chi phí nhỏ nhất giữa hai nút bất kỳ
- ❖ Sử dụng đồ thị để xây dựng thuật toán định tuyến



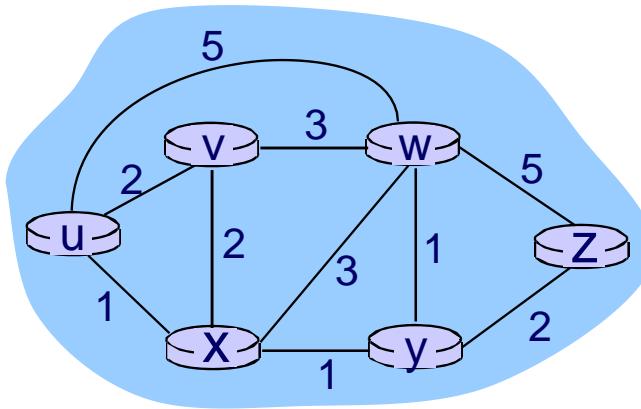
Đồ thị:  $G = (N, E)$

$N = \text{tập các router} = \{ u, v, w, x, y, z \}$

$E = \text{tập các liên kết} = \{ (u,v), (u,x), (v,x), (v,w), (x,w), (x,y), (w,y), (w,z), (y,z) \}$

Chi phí cho việc gửi gói tin qua mỗi liên kết:  $c(x,y), c(w,z), \dots$

# CHI PHÍ CỦA LIÊN KẾT



$c(x,x')$  = chi phí của liên kết  $(x,x')$

ví dụ  $c(w,z) = 5$

Chi phí có thể phản ánh thời gian trễ trung bình hoặc có thể phản ánh khoảng cách thực sự giữa hai router

Chi phí của đường đi  $(x_1, x_2, x_3, \dots, x_p) = c(x_1, x_2) + c(x_2, x_3) + \dots + c(x_{p-1}, x_p)$

**Giải thuật định tuyến:** Giải thuật tìm đường có chi phí nhỏ nhất

# PHÂN LOẠI THUẬT TOÁN ĐỊNH TUYẾN

## ❖ *Thuật toán tập trung:*

- ✓ Mọi router có mô hình đầy đủ, thông tin chi phí của các liên kết trong mạng

- ✓ Thuật toán “link state”

## ❖ *Thuật toán phân tán*

- ✓ Router chỉ biết các láng giềng có liên kết vật lý tới nó và chi phí liên kết tương ứng

- ✓ Liên tục lặp lại việc tính toán và trao đổi thông tin với láng giềng

- ✓ Thuật toán “distance vector”

# THUẬT TOÁN ĐỊNH TUYẾN LINK-STATE

## Thuật toán Dijkstra

- ❖ Mọi nút đều biết topology của mạng, chi phí của liên kết
  - Thực hiện bằng “link state broadcast”
  - Mọi nút có thông tin giống nhau
- ❖ Tính toán đường đi có chi phí thấp nhất từ một nút (nguồn) tới tất cả các nút khác.
  - Cho kết quả là bảng chuyển tiếp của nút đó
- ❖ Lặp: sau k vòng lặp, biết đường đi có chi phí thấp nhất tới k đích

## Chú thích:

- ❖  $c(x,y)$ : chi phí liên kết từ nút x tới nút y;  $= \infty$  nếu không là hàng xóm trực tiếp
- ❖  $D(v)$ : giá trị hiện tại của chi phí đường đi từ nguồn tới đích v
- ❖  $p(v)$ : nút trước của v trong đường đi từ nguồn tới v
- ❖  $N'$ : tập các nút mà đường đi có chi phí thấp nhất đã xác định

# THUẬT TOÁN DIJSKTRA

1 *Khởi tạo:*

2  $N' = \{u\}$

3 for <mọi nút v>

4 if <v kề u>

5      then  $D(v) = c(u,v)$

6 else  $D(v) = \infty$

7

8 *Lặp*

9 Tìm w không trong  $N'$  thỏa mãn  $D(w)$  nhỏ nhất

10 Thêm w vào  $N'$

11 Cập nhật  $D(v)$  với mọi v kề với w và không trong  $N'$  :

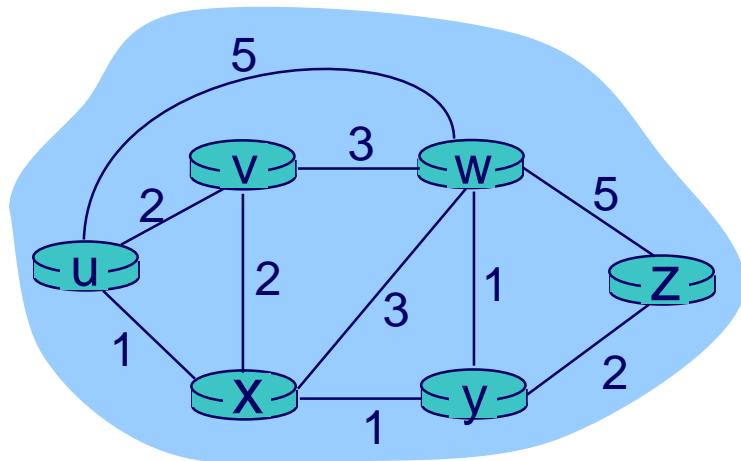
12  $D(v) = \min(D(v), D(w) + c(w,v))$

13 /\* Chi phí mới tới v là chi phí cũ tới v hoặc là chi phí đường đi ngắn nhất tới w đã biết cộng với chi phí từ w tới v \*/

15 *Tới khi mọi nút trong  $N'$*

# THUẬT TOÁN DIJKSTRA: VÍ DỤ

| Bước | $N'$   | $D(v), p(v)$ | $D(w), p(w)$ | $D(x), p(x)$ | $D(y), p(y)$ | $D(z), p(z)$ |
|------|--------|--------------|--------------|--------------|--------------|--------------|
| 0    | u      | 2,u          | 5,u          | 1,u          | $\infty$     | $\infty$     |
| 1    | ux     | 2,u          | 4,x          |              | 2,x          | $\infty$     |
| 2    | uxy    | 2,u          | 3,y          |              |              | 4,y          |
| 3    | uxyyv  |              | 3,y          |              |              | 4,y          |
| 4    | uxyvw  |              |              |              |              | 4,y          |
| 5    | uxyvwz |              |              |              |              |              |



Bước 0: Khởi tạo  $N' = \{u\}$

Tính  $D(v)$  và  $P(v)$  với  $v$  là nút liền kề với nút  $u$

$D(v) = 2, P(v) = u; D(w) = 5, P(w) = u \dots\dots$

Bước 1: Thấy  $D(x)$  nhỏ nhất, thêm  $x$  vào  $N'$

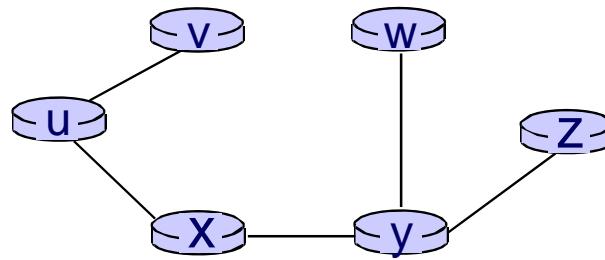
Tính lại  $D(v) = \min(D(v), D(x) + c(x,v)) = \min(2, 3) = 2; P(v) = u$

$D(w) = \min(D(w), D(x) + c(x,w)) = \min(5, 4) = 4; P(w) = x$

$D(y) = \min(D(y), D(x) + c(x,y)) = \min(\infty, 1+1) = 2; P(y) = x$

# THUẬT TOÁN DIJKSTRA: VÍ DỤ

Kết quả cây đường đi ngắn nhất từ u:

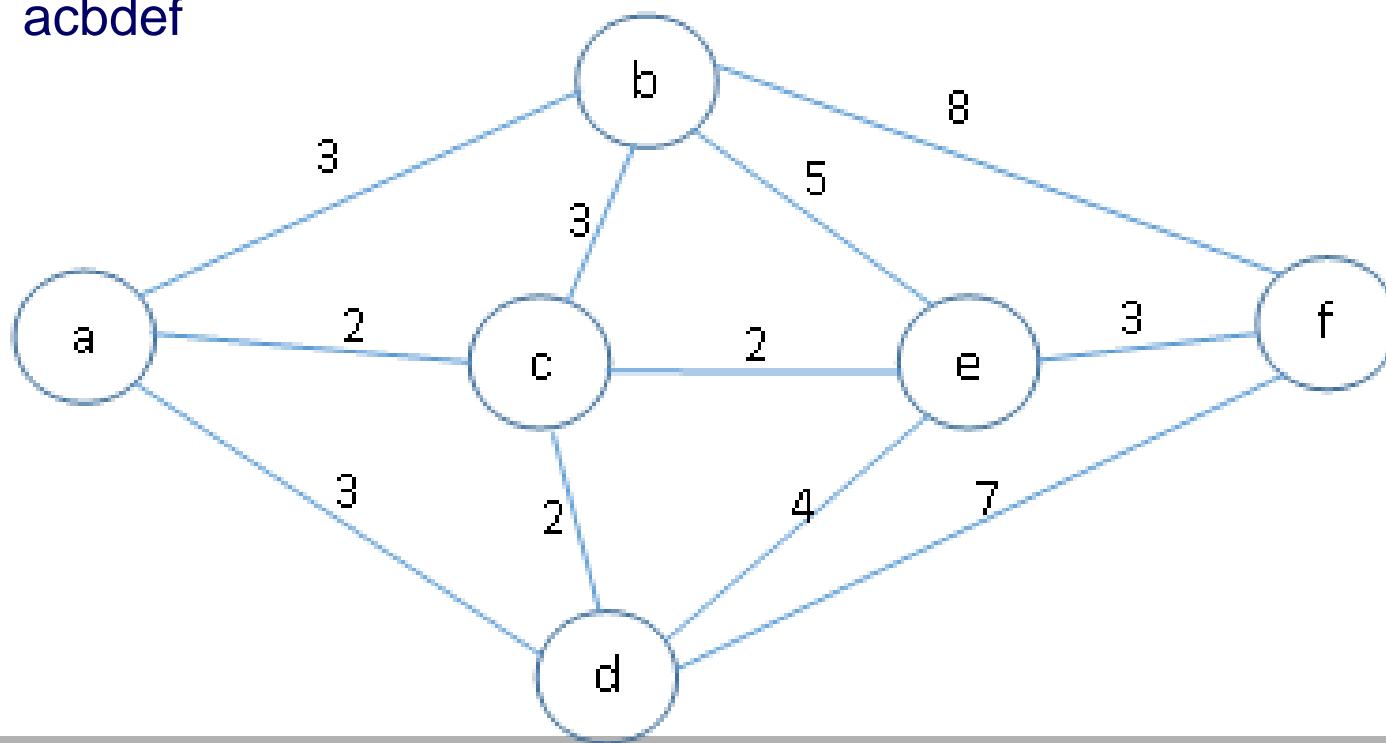


Kết quả forwarding table tại u:

| đích | liên kết |
|------|----------|
| v    | (u,v)    |
| x.   | (u,x)    |
| y.   | (u,x)    |
| w    | (u,x)    |
| z    | (u,x)    |

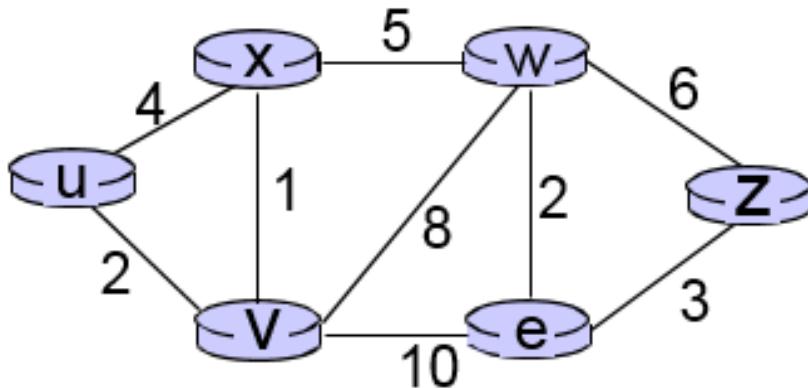
# BÀI TẬP Thuật toán Dijkstra: ví dụ

| Bước | N'     | D(b),p(b) | D(c),p(c) | D(d),p(d) | D(e),p(e) | D(f),p(f) |
|------|--------|-----------|-----------|-----------|-----------|-----------|
| 0    | a      | 3,a       | 2,a       | 3,a       | 00        | 00        |
| 1    | ac     | 3,a       |           | 3,a       | 4,c       | 00        |
| 2    | Acb    |           |           | 3,a       | 4,c       | 11,b      |
| 3    | acbd   |           |           |           | 4,c       | 10,d      |
| 4    | Acbde  |           |           |           |           | 7e        |
| 5    | acbdef |           |           |           |           |           |

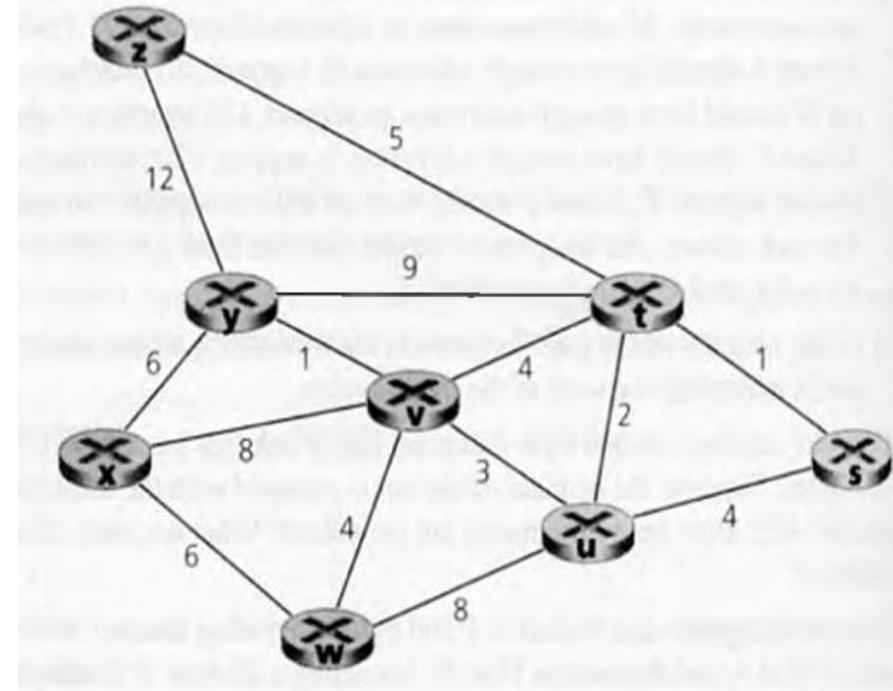


# Bài tập áp dụng

- ❖ Dùng giải thuật Dijkstra trong định tuyến link state để xác định đường đi ngắn nhất từ một nút tới các nút còn lại trong các sơ đồ mạng dưới đây.



a) Từ nút nút u tới các nút còn lại



b) Từ nút nút s tới các nút còn lại

# THUẬT TOÁN DISTANCE VECTOR (1)

## Công thức Bellman-Ford (quy hoạch động)

**Định nghĩa**

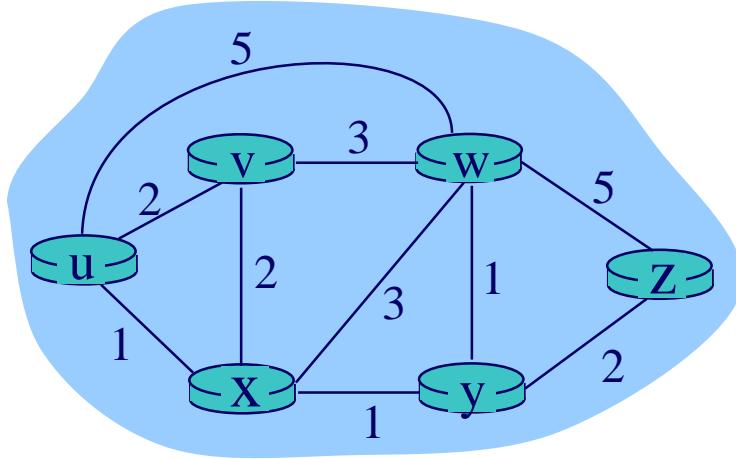
$d_x(y) :=$  chi phí của đường đi chi phí thấp nhất từ x tới y

**Thì**

$$d_x(y) = \min \{c(x,v) + d_v(y) \}$$

Trong đó, min được tính với tất cả các hàng xóm của x

# Ví dụ Bellman-Ford (2)



Dễ thấy,  $d_v(z) = 5$ ,  $d_x(z) = 3$ ,  $d_w(z) = 3$

Công thức B-F:

$$\begin{aligned}d_u(z) &= \min \{ c(u,v) + d_v(z), \\&\quad c(u,x) + d_x(z), \\&\quad c(u,w) + d_w(z) \} \\&= \min \{ 2 + 5, \\&\quad 1 + 3, \\&\quad 5 + 3 \} = 4\end{aligned}$$

Nút đạt giá trị nhỏ nhất là next hop trong đường đi ngắn nhất  
→ bảng chuyển tiếp

# Thuật toán Distance Vector (3)

- ❖  $D_x(y) = \text{chi phí thấp nhất từ } x \text{ tới } y$
- ❖ Distance vector:  $D_x = [D_x(y): y \in N]$
- ❖ Nút  $x$  biết chi phí tới mỗi hàng xóm  $v$ :  $c(x,v)$
- ❖ Node  $x$  duy trì  $D_x = [D_x(y): y \in N]$
- ❖ Node  $x$  cũng duy trì các distance vector của các hàng xóm của nó
  - Đối với hàng xóm  $v$ ,  $x$  duy trì $D_v = [D_v(y): y \in N]$

# Thuật toán Distance vector (4)

## Ý tưởng:

- ❖ Mỗi nút định kỳ gửi tính toán distance vector của nó tới các hàng xóm
- ❖ Khi nút x nhận tính toán DV mới từ hàng xóm, nó cập nhật DV của nó sử dụng công thức B-F:

$$D_x(y) \leftarrow \min_v \{c(x,v) + D_v(y)\} \quad \text{với mỗi } y \in N$$

- Ước lượng  $D_x(y)$  hội tụ tới chi phí thấp nhất thực tế của  $d_x(y)$

# Thuật toán Distance Vector (5)

Lặp, không đồng bộ: Mỗi vòng

lặp cục bộ thực hiện khi:

- ❖ Chi phí liên kết cục bộ thay đổi
- ❖ Bản tin cập nhật DV từ hàng xóm

Phân tán:

- ❖ Mỗi nút thông báo cho hàng xóm chỉ khi DV của nó thay đổi
  - Sau đó, các hàng xóm thông báo cho các hàng xóm của nó nếu cần

Mỗi nút:

*đợi* for (thay đổi chi phí liên kết cục bộ, bản tin từ hàng xóm)

*tính lại*

Nếu DV tới đích nào đó thay đổi, *thông báo* cho hàng xóm

$$D_x(y) = \min\{c(x,y) + D_y(y), c(x,z) + D_z(z)\}$$

$$= \min\{2+0, 7+1\} = 2$$

$$D_x(z) = \min\{c(x,y) + D_y(z), c(x,z) + D_z(z)\}$$

$$= \min\{2+1, 7+0\} = 3$$

## Bảng nút x

| chi phí tới |          |          |          |
|-------------|----------|----------|----------|
|             | x        | y        | z        |
| x           | 0        | 2        | 7        |
| y           | $\infty$ | $\infty$ | $\infty$ |
| z           | $\infty$ | $\infty$ | $\infty$ |

## Bảng nút y

| chi phí tới |          |          |          |
|-------------|----------|----------|----------|
|             | x        | y        | z        |
| x           | $\infty$ | $\infty$ | $\infty$ |
| y           | 2        | 0        | 1        |
| z           | $\infty$ | $\infty$ | $\infty$ |

## Bảng nút z

| chi phí tới |          |          |          |
|-------------|----------|----------|----------|
|             | x        | y        | z        |
| x           | $\infty$ | $\infty$ | $\infty$ |
| y           | $\infty$ | $\infty$ | $\infty$ |
| z           | 7        | 1        | 0        |

| chi phí tới |   |   |   |
|-------------|---|---|---|
|             | x | y | z |
| x           | 0 | 2 | 3 |
| y           | 2 | 0 | 1 |
| z           | 7 | 1 | 0 |

| chi phí tới |   |   |   |
|-------------|---|---|---|
|             | x | y | z |
| x           | 0 | 2 | 7 |
| y           | 2 | 0 | 1 |
| z           | 7 | 1 | 0 |

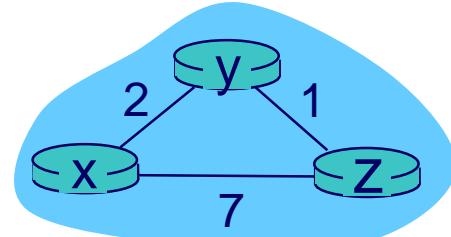
| chi phí tới |   |   |   |
|-------------|---|---|---|
|             | x | y | z |
| x           | 0 | 2 | 7 |
| y           | 2 | 0 | 1 |
| z           | 3 | 1 | 0 |

| chi phí tới |   |   |   |
|-------------|---|---|---|
|             | x | y | z |
| x           | 0 | 2 | 3 |
| y           | 2 | 0 | 1 |
| z           | 3 | 1 | 0 |

| chi phí tới |   |   |   |
|-------------|---|---|---|
|             | x | y | z |
| x           | 0 | 2 | 3 |
| y           | 2 | 0 | 1 |
| z           | 3 | 1 | 0 |

| chi phí tới |   |   |   |
|-------------|---|---|---|
|             | x | y | z |
| x           | 0 | 2 | 3 |
| y           | 2 | 0 | 1 |
| z           | 3 | 1 | 0 |

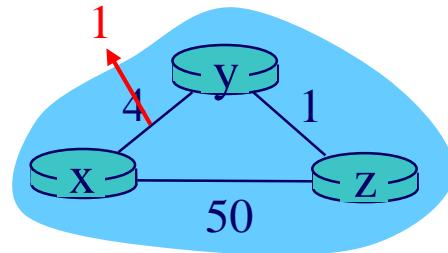
thời gian



# Distance Vector: chi phí liên kết thay đổi

## Chi phí liên kết thay đổi:

- ❑ Nút phát hiện chi phí liên kết cục bộ thay đổi
- ❑ Cập nhật thông tin dẫn đường, tính lại DV
- ❑ Nếu DV thay đổi, thông báo cho các hàng xóm



Tại thời điểm  $t_0$ , y phát hiện chi phí liên kết thay đổi, cập nhật DV của nó và thông báo cho hàng xóm của nó.

“Tin mới  
đi nhanh”

Tại thời điểm  $t_1$ , z nhận cập nhật từ y và cập nhật bảng của nó. Nó tính chi phí thấp nhất mới tới x và gửi DV của nó tới các hàng xóm của nó.

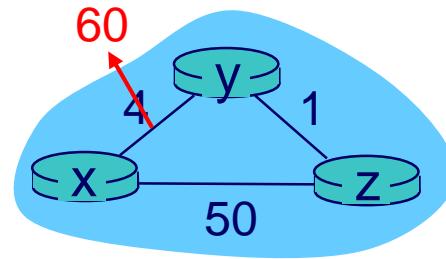
Tại thời điểm  $t_2$ , y nhận cập nhật của z và cập nhật bảng khoảng cách của nó.

Chi phí thấp nhất của y không thay đổi và vì vậy y *không* gửi bất kỳ bản tin nào tới z

# Distance Vector: chi phí liên kết thay đổi

## Chi phí liên kết thay đổi:

- ❑ Tin mới đi nhanh
- ❑ Tin xấu đi chậm – vấn đề “đếm vô hạn”!



## Kỹ thuật “Poissoned reverse”:

- ❑ Nếu Z qua Y tới X :
  - Z nói với Y khoảng cách của Z tới X là vô hạn (vì vậy, Y không dẫn đường tới X qua Z)
- ❑ Giải quyết bài toán “đếm vô hạn”?

# So sánh thuật toán LS và DV

## Sự phức tạp bản tin

- ❖ LS:  $n$  nút,  $E$  liên kết,  $O(nE)$  bản tin gửi đi
- ❖ DV: chỉ trao đổi giữa các hàng xóm
  - Thời gian hội tụ thay đổi

## Tốc độ hội tụ

- ❖ LS: Thuật toán  $O(n^2)$  đòi hỏi  $O(nE)$  bản tin
  - Có thể không ổn định
- ❖ DV: Thời gian hội tụ thay đổi
  - Có thể dẫn đường lặp
  - Vấn đề “đếm vô hạn”

**Tính chịu lỗi:** Điều gì xảy ra khi router bị lỗi?

### LS:

- Nút có thể thông báo chi phí *liên kết* sai
- Mỗi nút chỉ tính toán bảng của nó

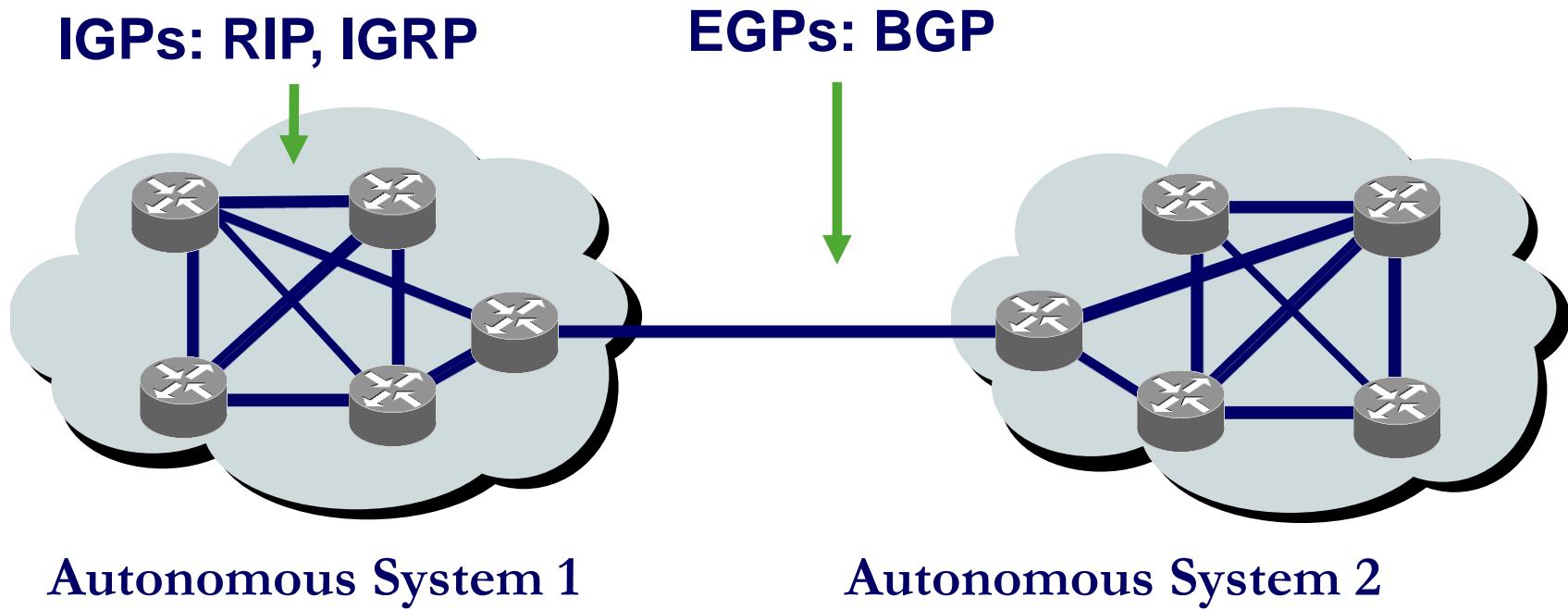
### DV:

- Nút có thể thông báo chi phí *đường đi* sai
- Mỗi bảng của nút sử dụng bởi nút khác
  - Lan truyền lỗi qua mạng

# CHƯƠNG 4: TẦNG MẠNG

- 
- 1 • Giới thiệu
  - 2 • Giao thức IP
  - 3 • Định tuyến
  - 4 • Dẫn đường trong internet

# ĐỊNH TUYẾN TRONG INTERNET



# ĐỊNH TUYẾN TRONG INTERNET

- ❖ **Định tuyến ngoài EGP (Exterior Gateway Protocol):** là giao thức trao đổi thông tin giữa các AS (Autonomous System) khác nhau. Tiêu biểu là giao thức BGP (Border Gateway Protocol)
- **Định tuyến trong IGP (Interior Gateway Protocol):** là giao thức định tuyến bên trong 1 AS như (RIP, IGRP, EIGRP, OSPF...).
- **Autonomous System:** Hệ thống tự trị là hệ thống do cơ quan tổ chức tự quản lý và chạy một giao thức riêng (VD ISP)

# IGP (Interior Gateway Protocol)

## ❖ IGP lại được chia thành 3 nhóm:

- ❖ Distance – vector: Mỗi router gửi cho láng giềng của nó toàn bộ bảng định tuyến của nó theo định kì. Giao thức tiêu biểu là RIP.
- ❖ Link – state: Mỗi router sẽ gửi bản tin trạng thái đường link (LSA) cho các router khác. Các Router sau khi xây dựng xong bảng định tuyến sẽ vẽ ra được một bản đồ mạng của toàn bộ hệ thống. Tiêu biểu là giao thức OSPF.
- ❖ Hybrid: tiêu biểu là giao thức EIGRP. Loại hình này kết hợp các đặc điểm của hai loại trên.

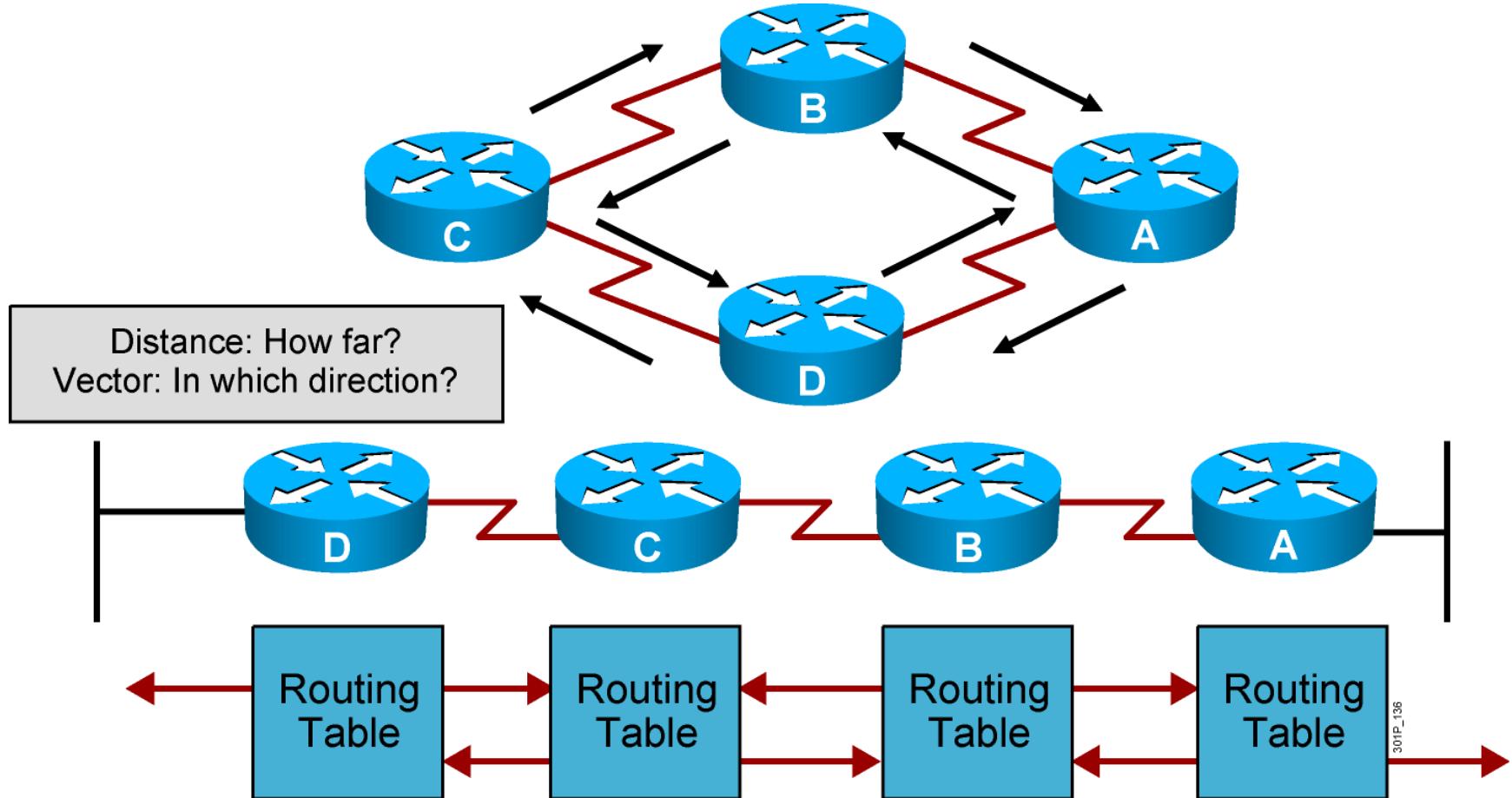
# GIAO THỨC ĐỊNH TUYẾN RIP

- Giao thức định tuyến nội vùng (Interior routing protocols)
- Sử dụng thuật toán tìm đường Bellman Ford.
- 30 giây các router update thông tin định tuyến.
- Metric = hop count (số nút mạng đi qua).
  - (Maximum is 16 equal-cost paths, if metric =16 -> infinity.)
- Đường đi tốt nhất là đường đi có metric nhỏ nhất.
- Dùng UDP-port 520 để chuyển có gói tin update với IPv4 và 521 với IPv6.
- AD (Administrative Distance) = 120 (độ tin cậy).

# RIP (Routing Information Protocol)

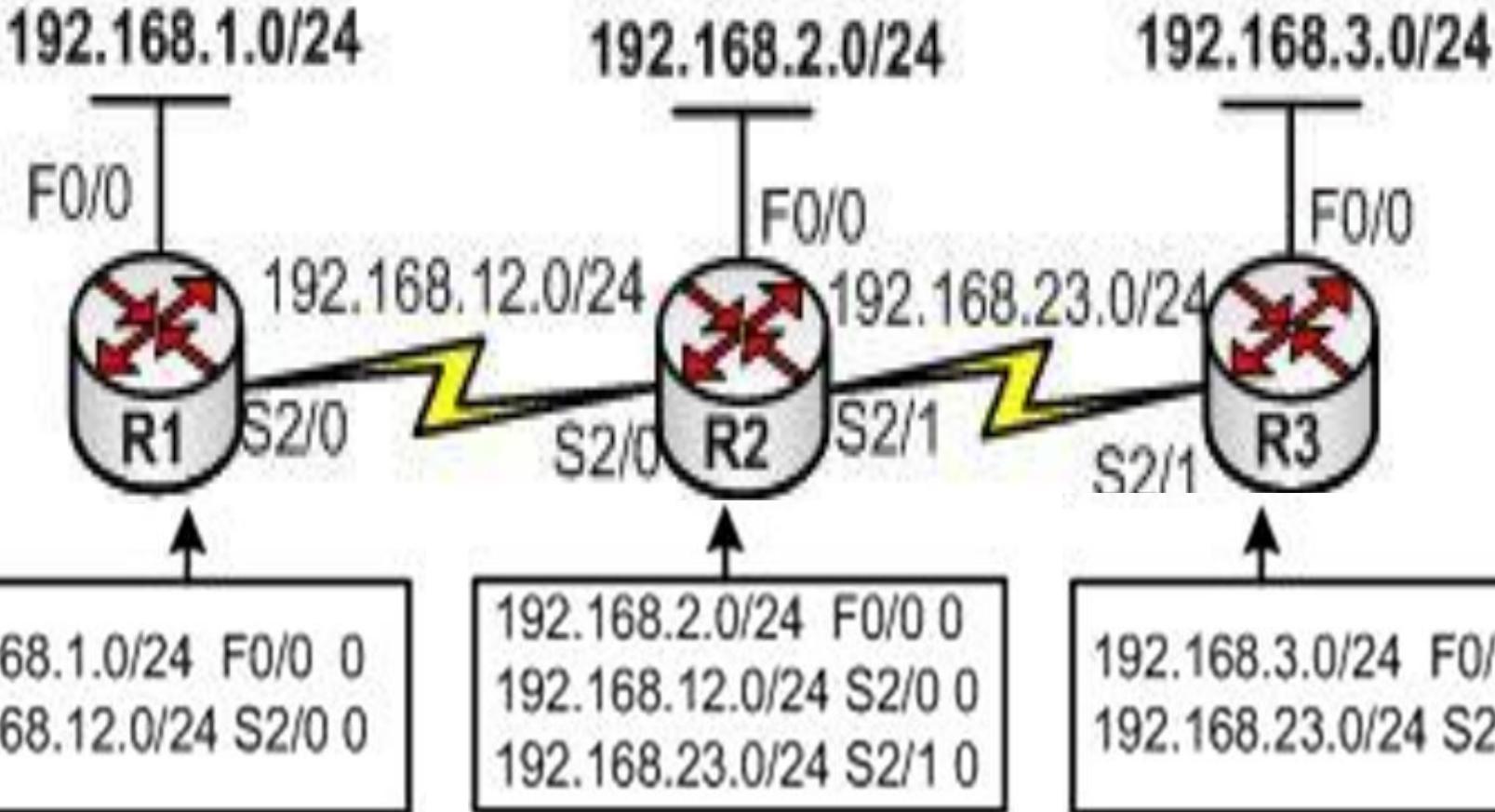
- ❖ RIP v1 và RIP v2, RIPng
- ❖ Thông tin cập nhật routing table theo địa chỉ 224.0.0.9  
đối với IPv4 và FF02::09 với IPv6
- ❖ Người quản trị có thể tính lại đường đi một cách thủ công.
- ❖ Các gói tin RIP với router được chứng thực

# Distance Vector

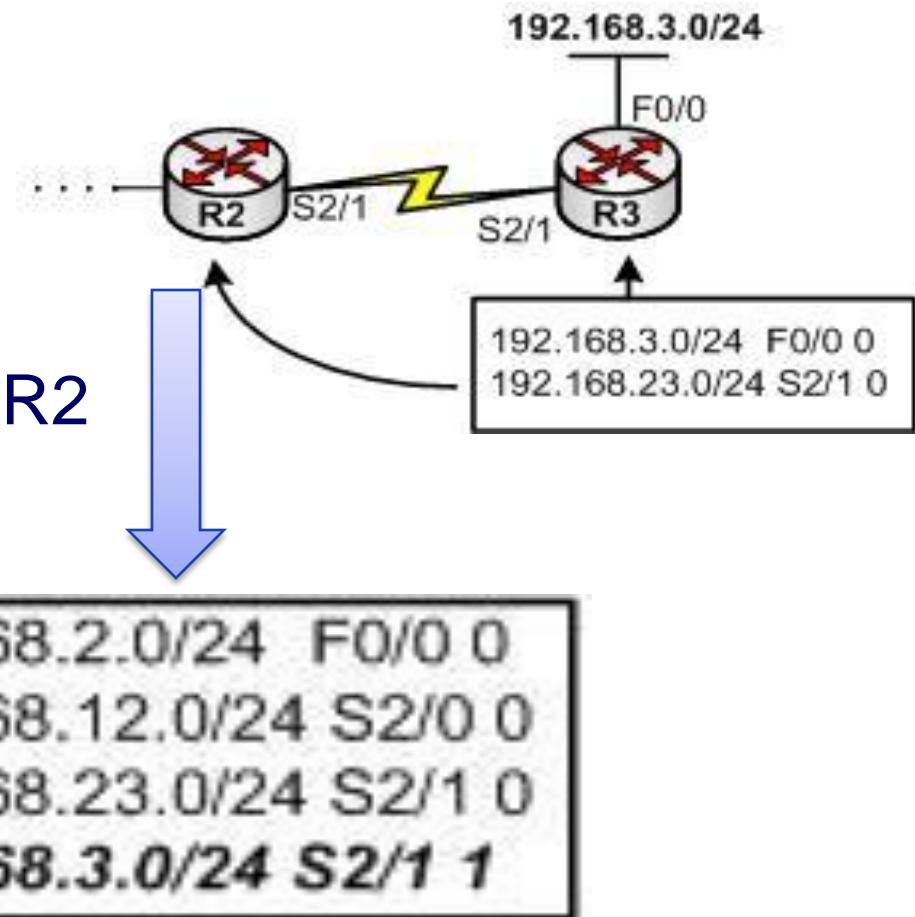


Các router định kỳ gửi các routing table đến các router láng giềng

# Hoạt Động RIP



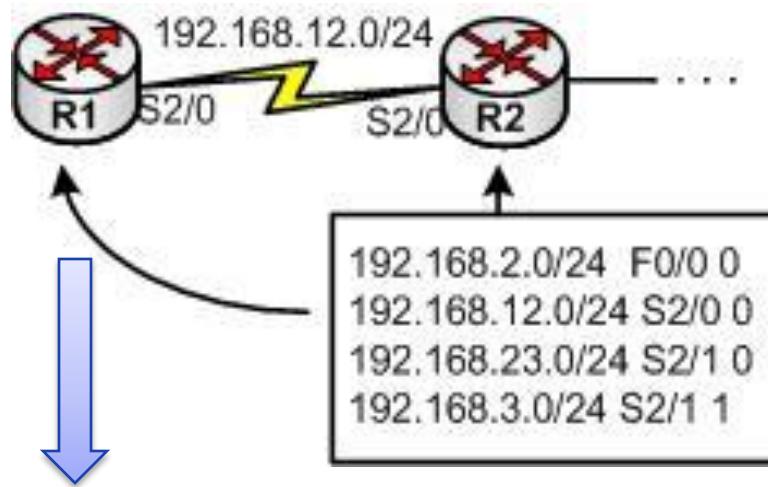
# Hoạt Động RIP tt



Routing table send from R3 to R2

Routing table of R2 ?

# Hoạt Động RIP tt

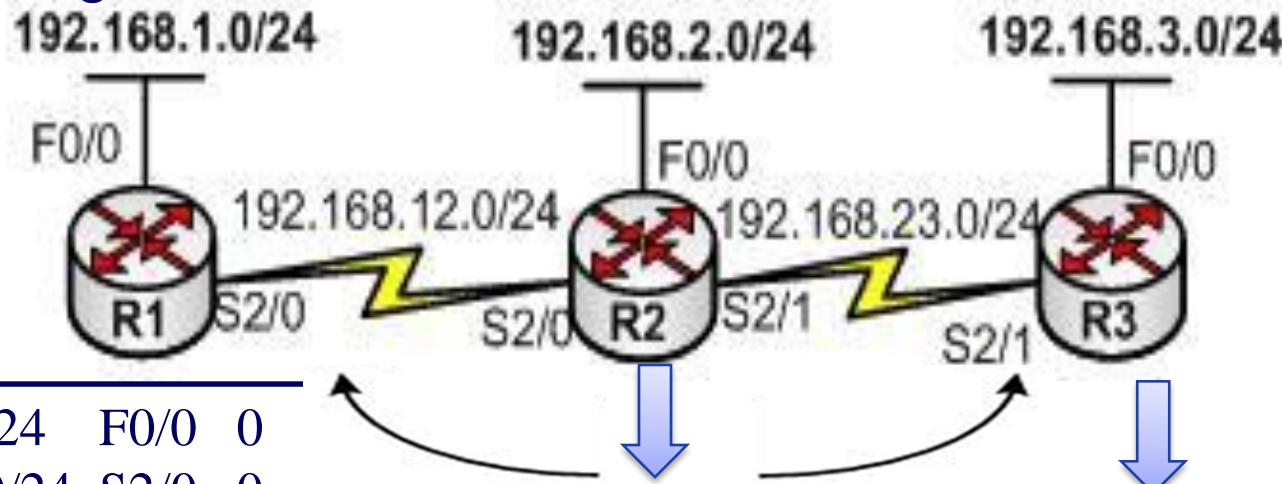


|                        |             |          |
|------------------------|-------------|----------|
| 192.168.1.0/24         | F0/0        | 0        |
| 192.168.12.0/24        | S2/0        | 0        |
| <b>192.168.23.0/24</b> | <b>S2/0</b> | <b>1</b> |
| <b>192.168.2.0/24</b>  | <b>S2/0</b> | <b>1</b> |
| <b>192.168.3.0/24</b>  | <b>S2/0</b> | <b>2</b> |

Routing table send from R2 to **R1** and **R3**  
Routing table of **R1**

# Hoạt Động RIP tt

Routing table send from R1 to R2



|                        |             |          |
|------------------------|-------------|----------|
| 192.168.1.0/24         | F0/0        | 0        |
| 192.168.12.0/24        | S2/0        | 0        |
| <b>192.168.23.0/24</b> | <b>S2/0</b> | <b>1</b> |
| <b>192.168.2.0/24</b>  | <b>S2/0</b> | <b>1</b> |
| <b>192.168.3.0/24</b>  | <b>S2/0</b> | <b>2</b> |

|                       |             |          |
|-----------------------|-------------|----------|
| 192.168.12.0/24       | S2/0        | 0        |
| 192.168.2.0/24        | F0/0        | 0        |
| 192.168.23.0/24       | S2/1        | 0        |
| <b>192.168.3.0/24</b> | <b>S2/1</b> | <b>1</b> |
| <b>192.168.1.0/24</b> | <b>S2/0</b> | <b>1</b> |

|                       |  |  |
|-----------------------|--|--|
| 192.168.12.0/24       |  |  |
| 192.168.2.0/24        |  |  |
| 192.168.23.0/24       |  |  |
| <b>192.168.3.0/24</b> |  |  |
| <b>192.168.1.0/24</b> |  |  |

Routing table of R2

# Cấu hình RIPv2

- Bước 1: Khởi tạo tiến trình định tuyến RIP

```
RouterX(config)# router rip
```

- Bước 2: Thiết lập version cho RIP

```
RouterX(config-router)# version 2
```

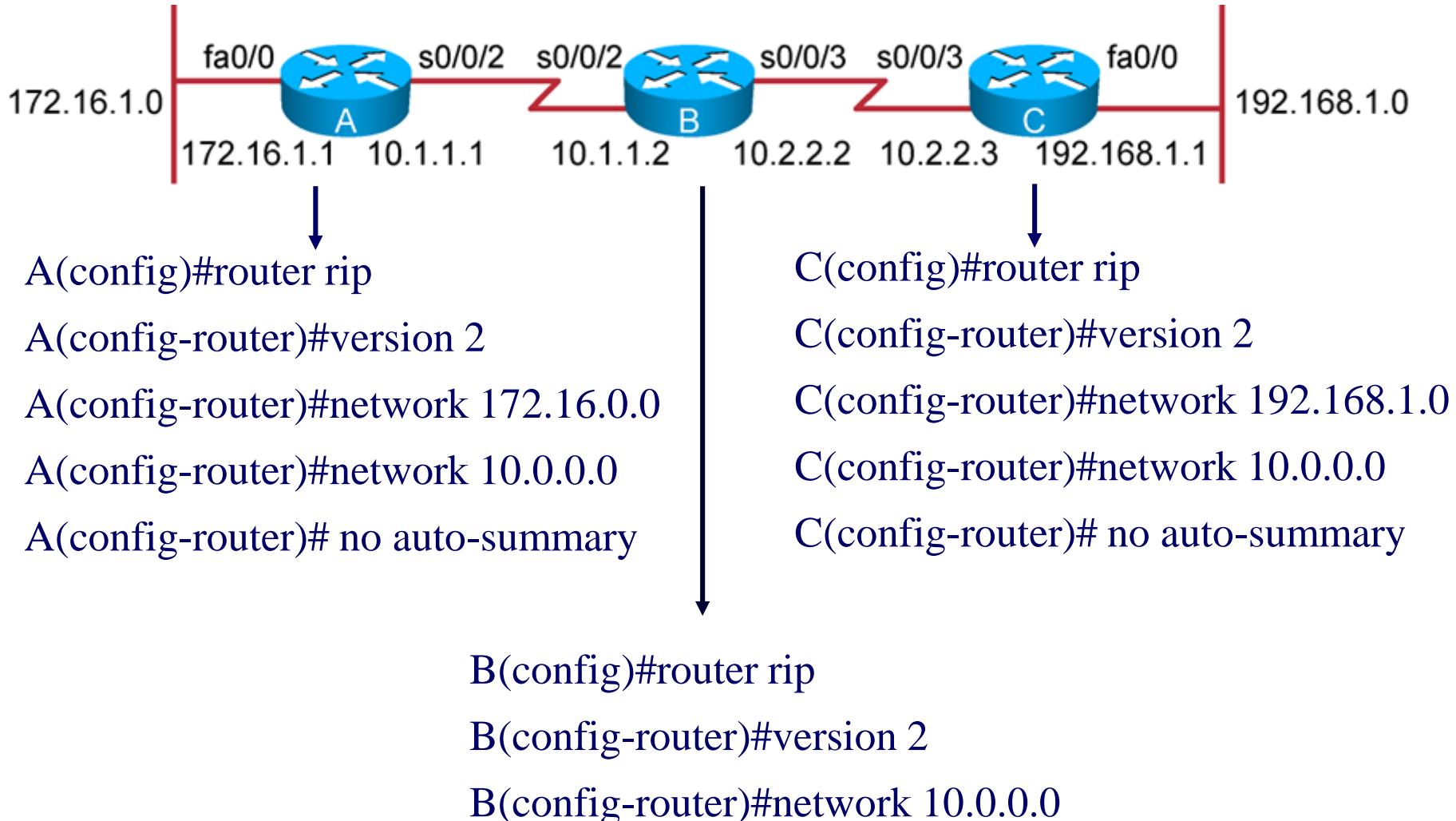
- Bước 3: Thiết lập cổng tham gia định tuyến (yêu cầu cấu mạng chính theo đúng lớp mạng)

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

Để quảng bá các mạng con và hỗ trợ mạng không liên tục, chúng ta phải sử dụng lệnh sau:

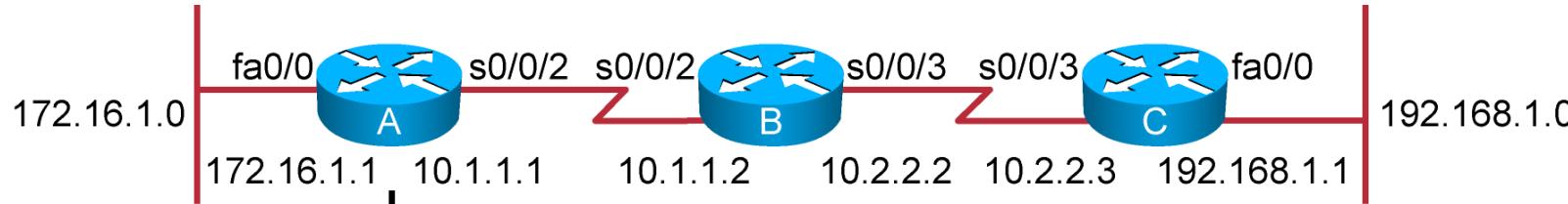
```
Router(config-router)#no auto-summary
```

# Ví dụ cấu hình RIP



# Thông tin cấu hình RIP

Router#Show ip protocol



30IP\_97%

Routing Protocol is "rip"

Sending updates every 30 seconds, next due in 6 seconds  
Invalid after 180 seconds, hold down 180, flushed after 240  
Outgoing update filter list for all interfaces is not set  
Incoming update filter list for all interfaces is not set  
Redistributing: rip

Default version control: send version 2, receive version 2  
Interface Send Recv Triggered RIP Key-chain  
FastEthernet0/0 2 2  
Serial0/0/2 2 2

Automatic network summarization is in effect

Maximum path: 4

Routing for Networks:

10.0.0.0  
172.16.0.0

Routing Information Sources:

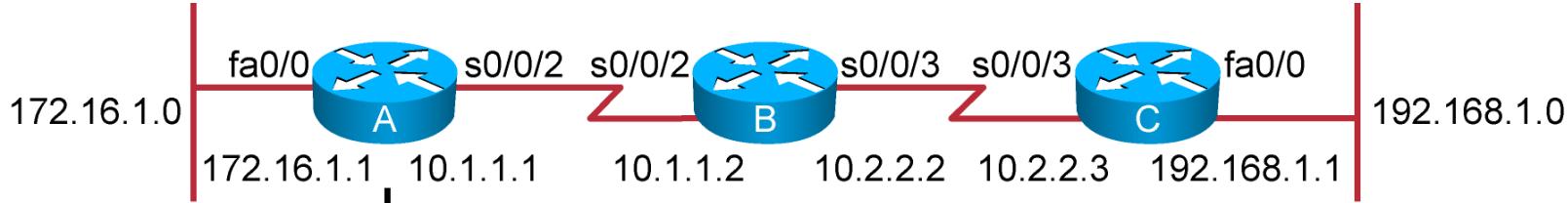
| Gateway  | Distance | Last Update |
|----------|----------|-------------|
| 10.1.1.2 | 120      | 00:00:25    |

Distance: (default is 120)

RouterA#

# Hiển thị bảng định tuyến

Router#Show ip route



301P\_976

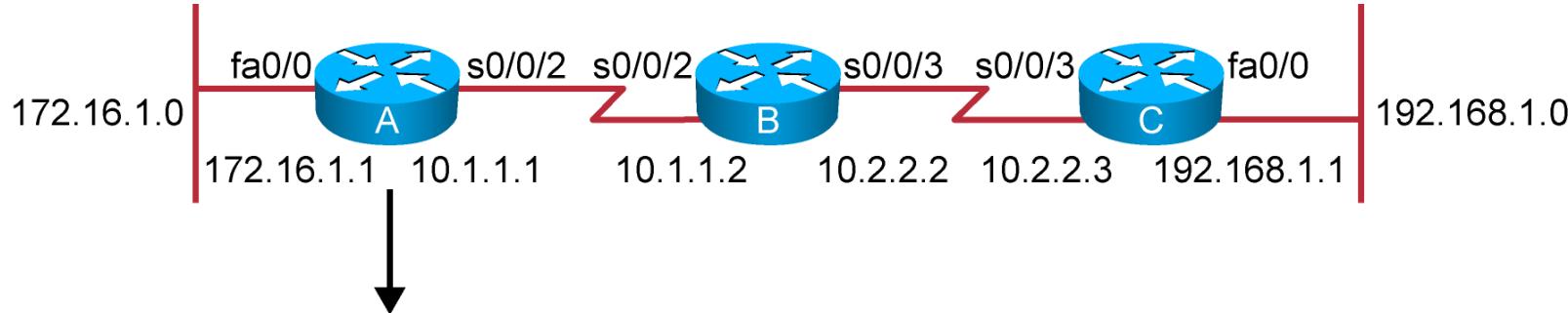
```
RouterA# show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
      U - per-user static route, o - ODR
      T - traffic engineered route
```

Gateway of last resort is not set

```
      172.16.0.0/24 is subnetted, 1 subnets
C        172.16.1.0 is directly connected, fastethernet0/0
      10.0.0.0/24 is subnetted, 2 subnets
R        10.2.2.0 [120/1] via 10.1.1.2, 00:00:07, Serial0/0/2
C        10.1.1.0 is directly connected, Serial0/0/2
R        192.168.1.0/24 [120/2] via 10.1.1.2, 00:00:07, Serial0/0/2
```

# Hiển thị thông tin gõ rối IP RIP

RouterA# debug ip rip



30IP\_976

```
RouterA# debug ip rip
RIP protocol debugging is on
RouterA#
00:06:24: RIP: received v1 update from 10.1.1.2 on Serial0/0/2
00:06:24:      10.2.2.0 in 1 hops
00:06:24:      192.168.1.0 in 2 hops
00:06:33: RIP: sending v1 update to 255.255.255.255 via FastEthernet0/0 (172.16.1.1)
00:06:34:      network 10.0.0.0, metric 1
00:06:34:      network 192.168.1.0, metric 3
00:06:34: RIP: sending v1 update to 255.255.255.255 via Serial0/0/2 (10.1.1.1)
00:06:34:      network 172.16.0.0, metric 1
```

# Cấu hình RIP IPv6

B1: Cho phép định tuyến IPv6: R(config)#**ipv6 unicast-routing**

B2: Chọn giao thức định tuyến: R(config)#**ipv6 router rip tag**

Trong đó: tag là một chuỗi định danh, do người cấu hình tự đặt

B3. Chọn cổng tham gia vào quá trình trao đổi thông tin định tuyến: R(config-if)#**ipv6 rip tag enable**

Các lệnh kiểm tra cấu hình:

R#**show ipv6 rip**

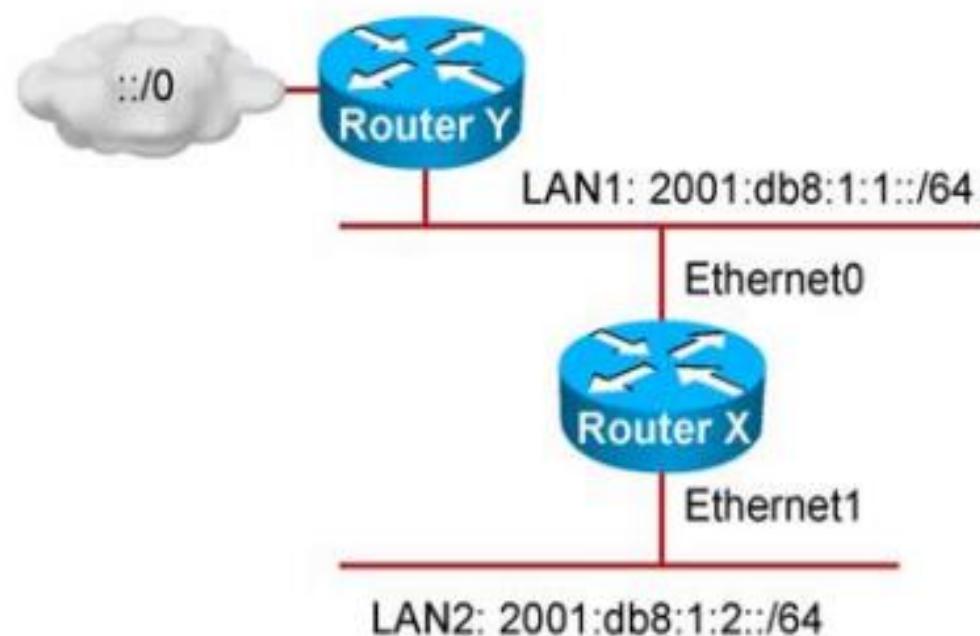
R#**show ipv6 route [rip]**

# VÍ DỤ

## ❖ Trên Router Y

```
Y(config)#ipv6 unicast-routing
```

```
Y(config)#ipv6 router rip R1
```



```
Y#config terminal
```

```
Y(config)#int Ethernet0
```

```
Y(config-if)#ipv6 address 2001:db8:1:1::/64 eui-64
```

```
Y(config-if)#ipv6 rip R1 enable
```

## ❖ Trên Router X

```
X(config)#ipv6 unicast-routing
```

```
X(config)#ipv6 router rip R1
```

```
X#conf t
```

```
X(config)#int Ethernet0
```

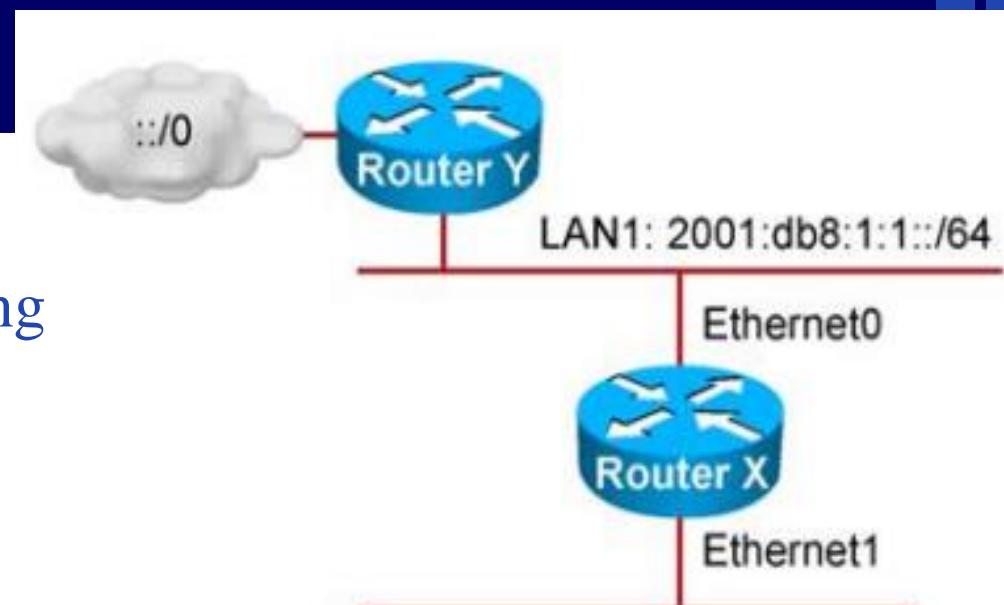
```
X(config-if)#ipv6 address 2001:db8:1:1::/64 eui-64
```

```
X(config-if)#ipv6 rip R1 enable
```

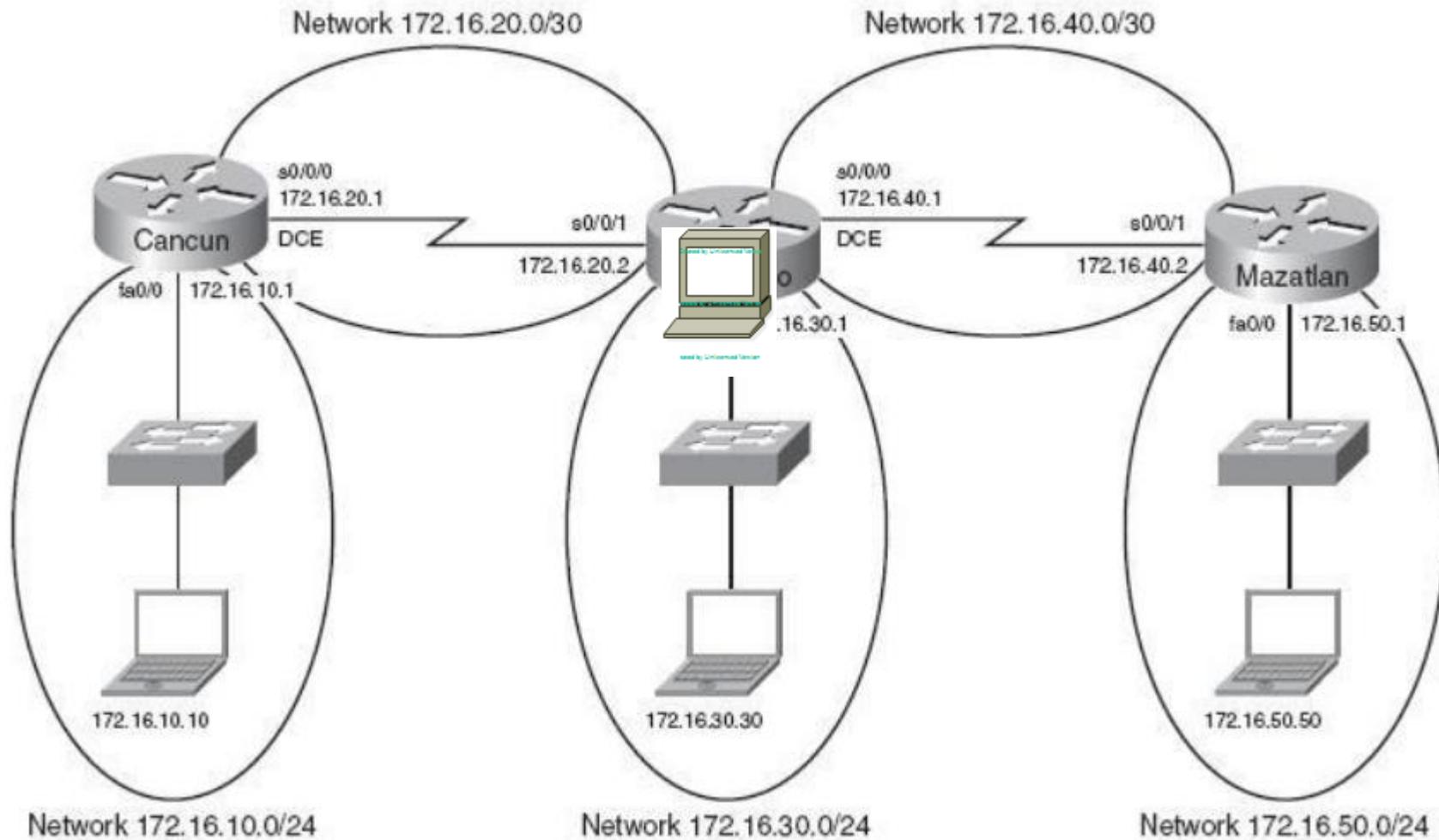
```
X(config)#int Ethernet1
```

```
X(config-if)#ipv6 address 2001:db8:1:2::/64 eui-64
```

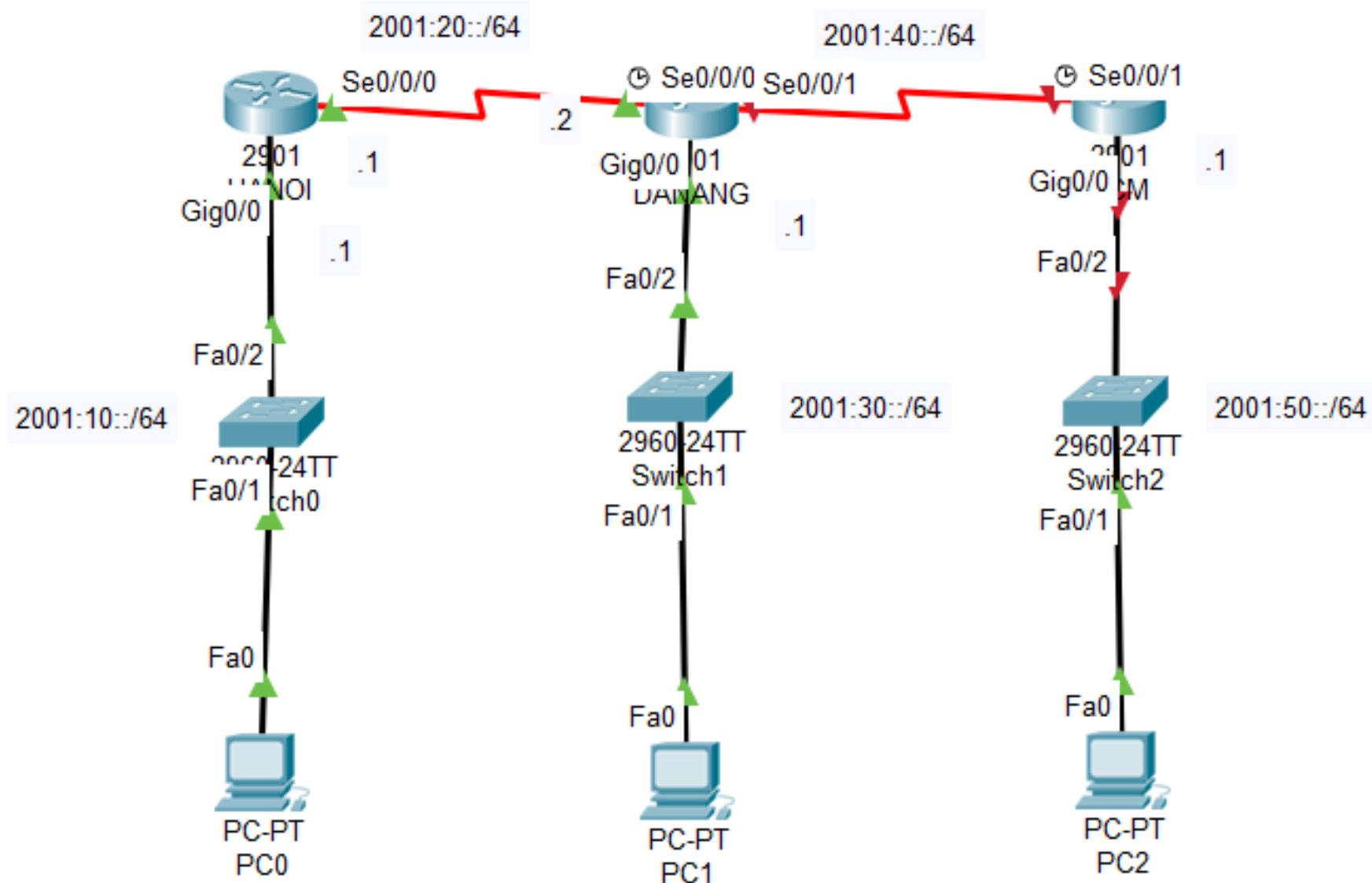
```
X(config-if)#ipv6 rip R1 enable
```



# BÀI TẬP 1: CẤU HÌNH RIPv2



# BÀI TẬP 2: CẤU HÌNH RIPng



# BÀI TẬP 3

## ❖ Cho địa chỉ IP **192.168.1.200/26**

1. Hãy tính Subnet Mask dạng thập phân
2. Hãy cho biết địa chỉ IP trên thuộc mạng con nào.
3. Liệt kê dải IP của mạng con đó
4. Tìm địa chỉ broadcast của mạng con đó.

# BÀI TẬP 4

- ❖ Cho địa chỉ mạng **172.16.10.0/18**
  1. Hãy chia mạng trên thành 4 mạng con.
  2. Xác định Subnets Mask mới dạng thập phân
  3. Liệt kê các dải IP trong từng mạng con.
  4. Xác định địa chỉ Broadcast của từng mạng con

- ❖ Một mạng lớp B cần chia thành 9 mạng con sử dụng Subnet Mask nào sau đây:
  - a. 255.255.240.0
  - b. 255.0.0.255
  - c. 255.224.255.0
  - d. 255.255.255.224

**Cho địa chỉ IP của một máy tính là:  
192.168.1.150; Subnet mask: 255.255.255.192.  
Địa chỉ mạng mà máy tính này thuộc về ?**

- ❖ 192.168.1.128
- ❖ 192.168.1.0
- ❖ 192.168.1.160
- ❖ 192.168.1.192

❖ Địa chỉ IP nào nằm cùng mạng subnet với IP 200.200.200.150, subnet mask: 255.255.255.240:

- A. 200.200.200.145
- B. 200.200.200.144
- C. 200.200.200.160
- D. 200.200.200.170

- ❖ Trong các địa chỉ sau sẽ có một địa chỉ không cùng nằm chung mạng con với bốn địa chỉ còn lại khi sử dụng subnet mask là **255.255.224.0**:
  - a. 192.16.158.130**
  - b. 192.16.165.141**
  - c. 192.16.171.152**
  - d. 192.16.185.200**