TRƯỜNG ĐẠI HỌC THỦY LỢI KHOA CÔNG NGHỆ THÔNG TIN

Mạng máy tính

ThS. Đoàn Thị Quế

Email: dtque@tlu.edu.vn

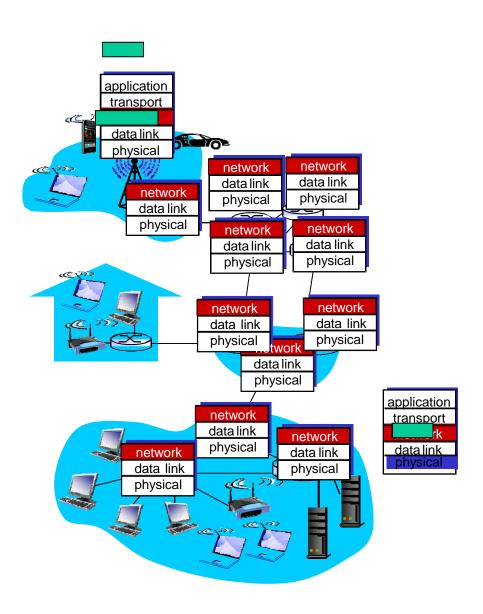
Chương 4: Tầng mạng

- Giới thiệu
- 2. IP: Internet Protocol
 - o IPv4
 - Cấu trúc datagram
 - O ICMP
 - IPv6

- 3. Định tuyến
 - Định tuyến là gì?
 - Phân loại định tuyến
 - Thuật toán định tuyến
- 4. Dẫn đường trong Internet
 - ORIP
 - OSPF
 - BGP

Tầng mạng

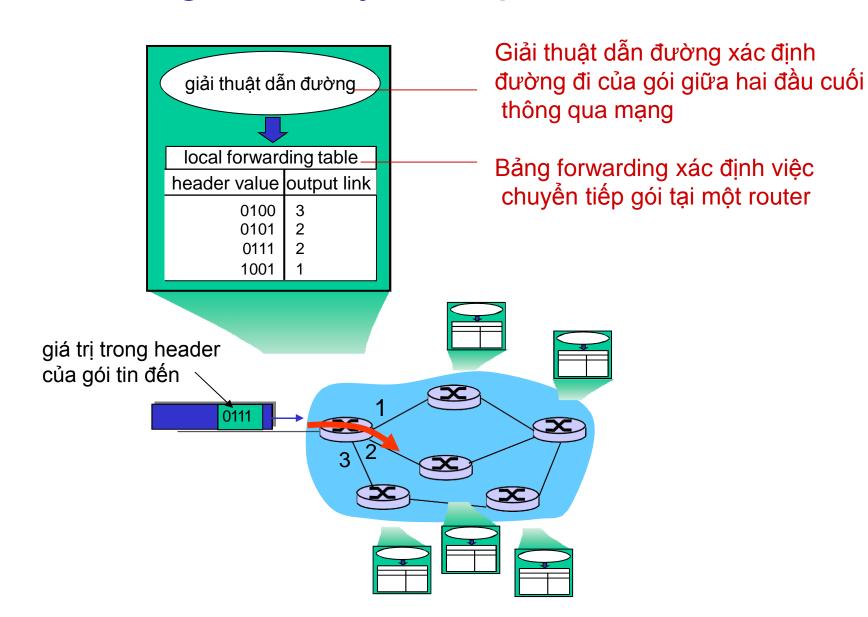
- Tầng mạng chuyển Segment của tầng giao vận từ nút gửi tới nút nhận
 - Phía gửi, đóng segment trong các datagram
 - Phía nhận, chuyển segment tới tầng giao vận
- Giao thức của tầng mạng có trong mọi host và router



Hai chức năng cơ bản của tầng mạng

- Dịnh tuyến/ Dẫn đường (routing): xác định đường đi của gói tin từ nguồn tới đích
 - o giải thuật dẫn đường
- Chuyển tiếp (forwarding): chuyển gói tin từ đầu vào của router ra đầu ra thích hợp của router

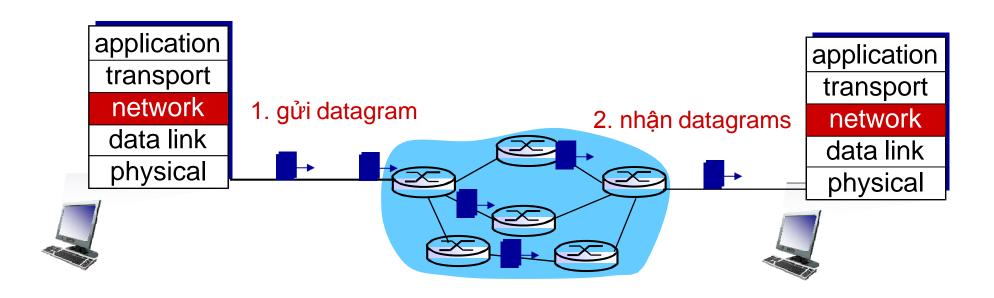
Dẫn đường và chuyển tiếp



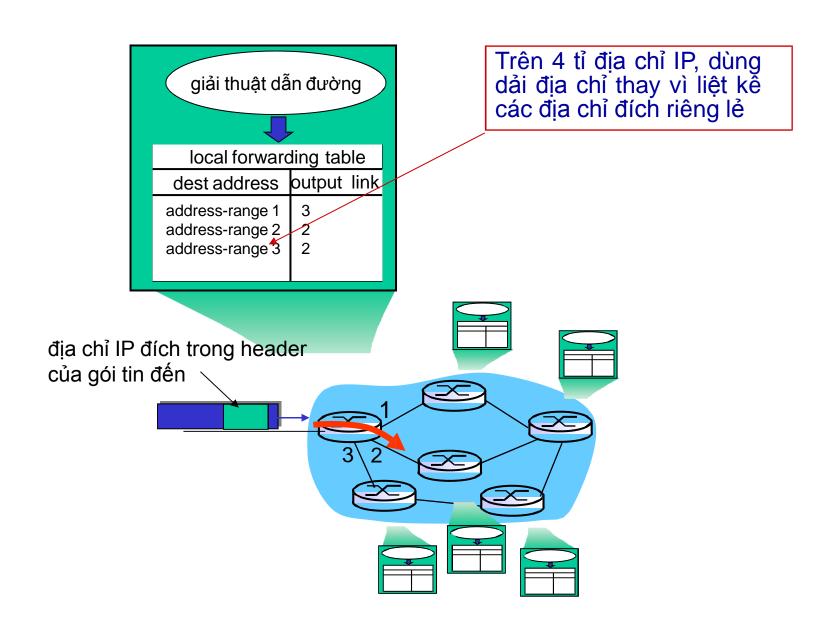
Mạng chuyển mạch gói

Mạng chuyển mạch gói (Datagram network)

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



Bảng chuyển tiếp (forwarding table)



Bảng chuyển tiếp

	Liên kết ra			
11001000 through	00010111	00010000	00000000	0
	00010111	00010111	11111111	U
	00010111	00011000	00000000	
through 11001000	00010111	00011000	11111111	1
	00010111	00011001	0000000	
through 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

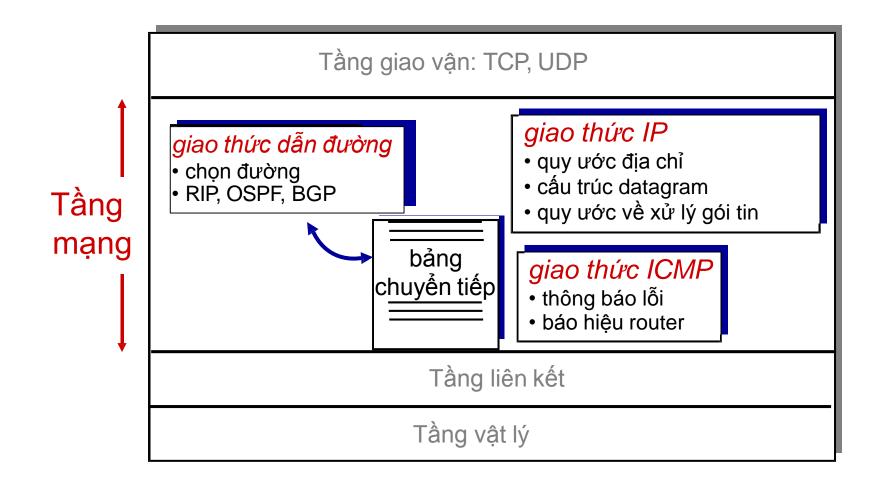
Kiến trúc mạng	Mô hình dịch vụ	Đảm bảo				Phản hồi	
		Băng thông	Mất gói	Thứ tự	Thời gian	tắc nghẽn	
Internet	cố gắng tối đa (best effort)	không	không	không	không	không (phát hiện thông qua mất gói)	
ATM (asynchronous transfer mode)	Tốc độ bit cố định (CBR- Constant bit rate)	Tốc độ không đổi	có	có	có	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	
						10	

Chương 4: Tầng mạng

- Giới thiệu
- 2. IP: Internet Protocol
 - o IPv4
 - Cấu trúc datagram
 - O ICMP
 - IPv6

- 3. Định tuyến
 - Định tuyến là gì?
 - Phân loại định tuyến
 - Thuật toán định tuyến
- 4. Dẫn đường trong Internet
 - ORIP
 - OSPF
 - BGP

Internet Protocol



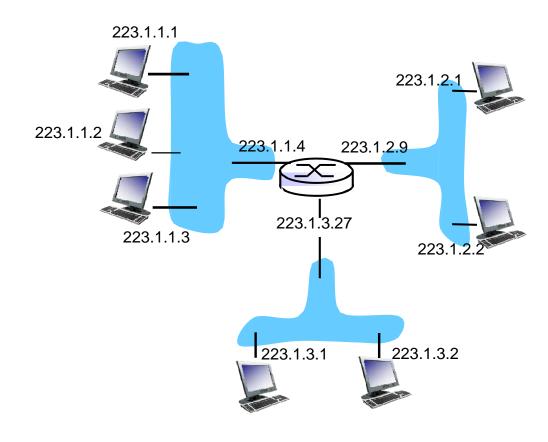
Chương 4: Tầng mạng

- Giới thiệu
- 2. IP: Internet Protocol
 - o IPv4
 - Cấu trúc datagram
 - O ICMP
 - IPv6

- 3. Định tuyến
 - Định tuyến là gì?
 - Phân loại định tuyến
 - Thuật toán định tuyến
- 4. Dẫn đường trong Internet
 - ORIP
 - OSPF
 - BGP

Địa chỉ IPv4

- □ Địa chỉ IPv4: 32 bit để định danh giao diện (interface) của host hay router
- interface: kết nối giữa host/router và liên kết vật lý
 - router thường có nhiều interface
 - host thường có 1 hoặc 2 interface (ví dụ: wired Ethernet, wireless 802.11)
- Mỗi địa chỉ IP gán cho một interface
- Địa chỉ IP có tính duy nhất



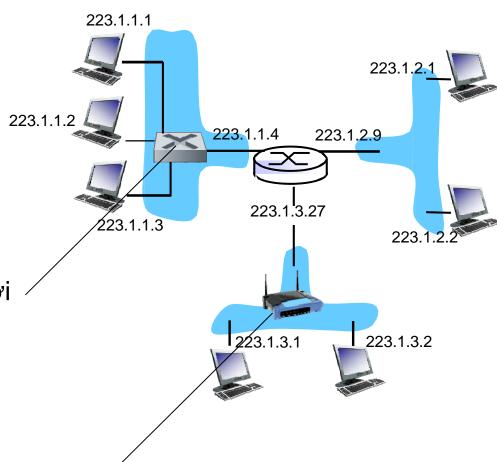
Chia làm 4 octet, mỗi octet là 8 bit có giá trị chạy từ 0 đến 255

Địa chỉ IPv4

Các interface kết nối với nhau như thế nào?

Các Ethernet interface kết nối với nhau qua Ethernet switches

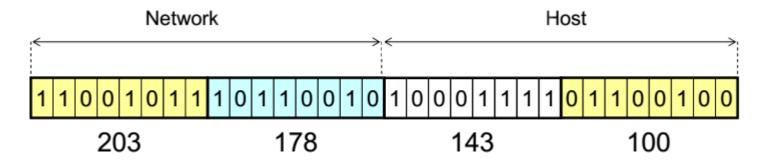
Các interface cũng có thể kết nối trực tiếp với nhau



Các WiFi interface kết nối với nhau qua WiFi base station

Địa chỉ IP v4

- ■Địa chỉ IP có hai phần
 - Host –địa chỉ host
 - Network địa chỉ mạng



- Làm thế nào biết được phần nào là cho host, phần nào dành cho mạng?
 - Phân lớp địa chỉ
 - Không phân lớp CIDR (Classless interdomain routing)

Phân lớp địa chỉ IP

Lớp	Octet1	Octet2	Octet3	Octet 4	Octet1 Thập phân	Số mạng	Số host
A	Onnn nnnn	hhhh hhhh	hhhh hhhh	hhhh hhhh	0-127	2^7 (128)	2^24-2 (16.777.214)
В	10nn nnnn	nnnn nnnn	hhhh hhhh	hhhh hhhh	128-191	2^14 (16384)	2^16-2 (65.534)
С	110n nnnn	nnnn nnnn	nnnn nnnn	hhhh hhhh	192-223	2^21 (2.097.152)	2^8-2 (254)
D	1110 xxxx	Multicast			224-239		
E	1111 xxxx	Dự phòng			240-255		

Địa chỉ mạng: phần host =0

Địa chỉ Broadcast trong một mạng: Phần host =1

Địa chỉ IPv4

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)

Lớp A: 10.0.0.0 - 10.255.255.255

Lớp B: 172.16.0.0 – 172.31.255.255

Lớp C: 192.168.0.0 – 192.168.255.255

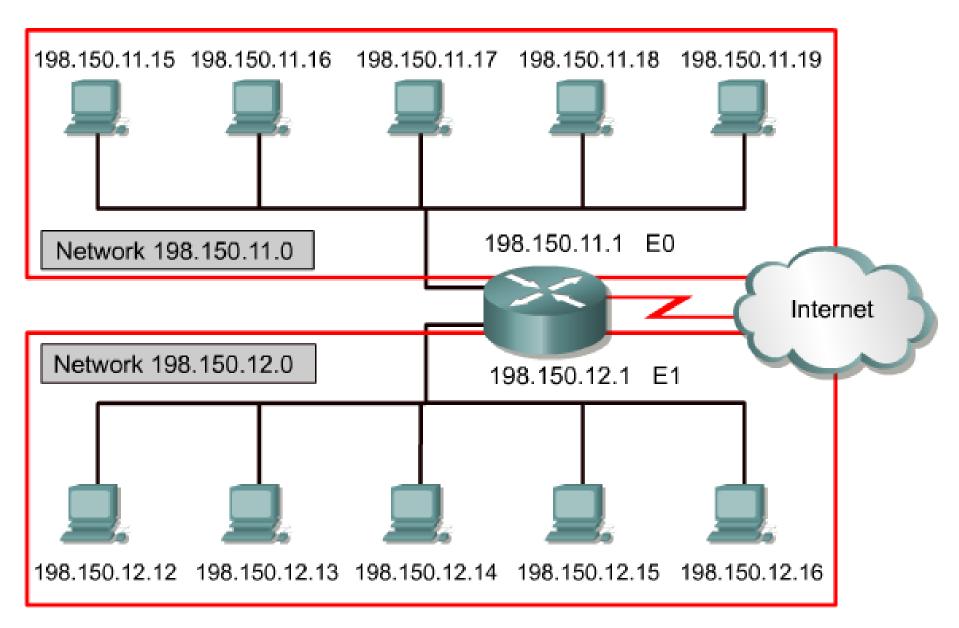
Địa chỉ nào sử dụng trong mạng nội bộ?

- 1) 150.100.255.255
- 2) 172.19.255.18
- 3) 195.234.253.0
- 4) 10.10.110.23
- 5) 192.168.221.176
- 6) 127.34.25.189
- 7) 203.162.217.73

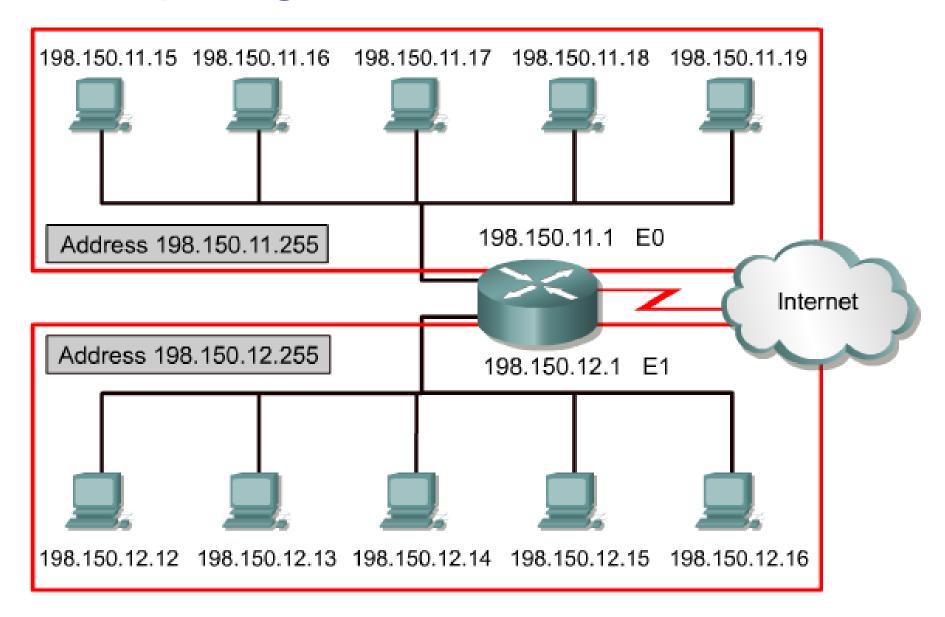
Các dạng địa chỉ

- ■Địa chỉ mạng
 - Địa chỉ IP gán cho một mạng (các bit phần host bằng 0). Ví dụ: 172.29.0.0
- ■Địa chỉ host (địa chỉ máy trạm)
 - Địa chỉ IP gán cho interface của host
- ■Địa chỉ quảng bá
 - Địa chỉ dùng để đại diện cho tất cả các host trong mạng (các bit phần host bằng 1). Ví dụ: 172.29.255.255.

Địa chỉ mạng (Network Address)



Địa chỉ quảng bá 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

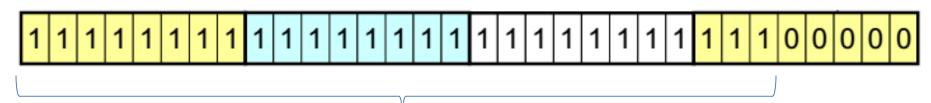
Mặt nạ mạng

- 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

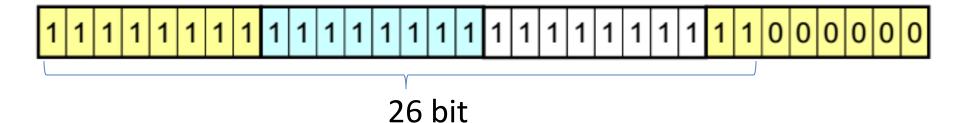
- Subnet mask
 - Các bit trong phần Network bằng 1
 - Các bit trong phần host bằng 0
 - Ví dụ:

/27



27 bit

/26



Tính địa chỉ mạng từ địa chỉ IP

□Địa chỉ IP là: 144.28.16.17/20

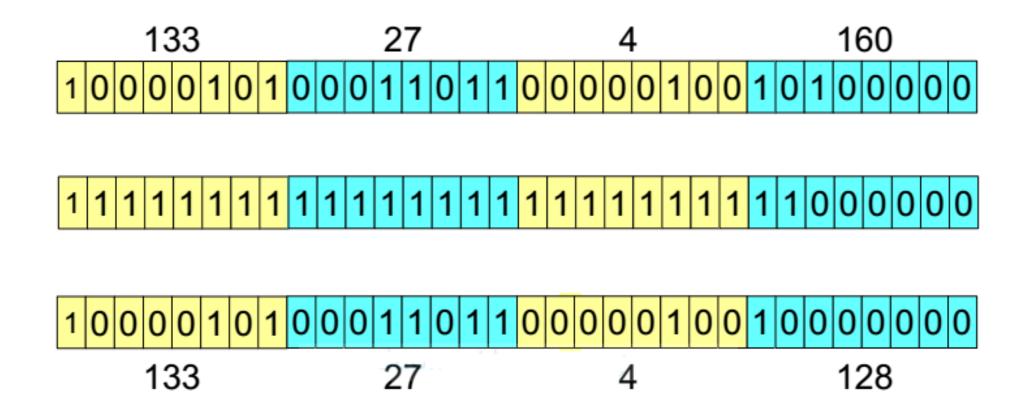
```
144 . 28 . 16 . 17
IP address: 10010000 00011100 00010000 00010001
Subnet mask: 11111111 1111111 11110000 00000000
Network address: 10010000 00011100 00010000 00000000
144 . 28 . 16 . 0
```

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?



Địa chỉ máy trạm, địa chỉ mạng, địa chỉ quảng bá?

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

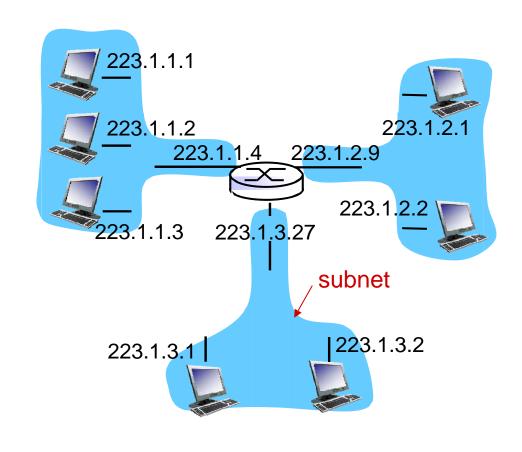
Mang con - Subnet

Sự cần thiết phải phân chia mạng con

- ➤ Mỗi mạng ở lớp A có đến $2^{24} 2 = 16.777.214$ địa chỉ IP Mỗi mạng ở lớp B có $2^{16} 2 = 65534$ địa chỉ IP.
- Trong một hệ thống mạng, số host nhỏ hơn số lượng địa chỉ trong một mạng lớp B rất nhiều.
- Việc quản trị trên một mạng có quá nhiều host gặp khó khăn lớn.

Mang con - Subnet

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



Mạng với 3 mạng con

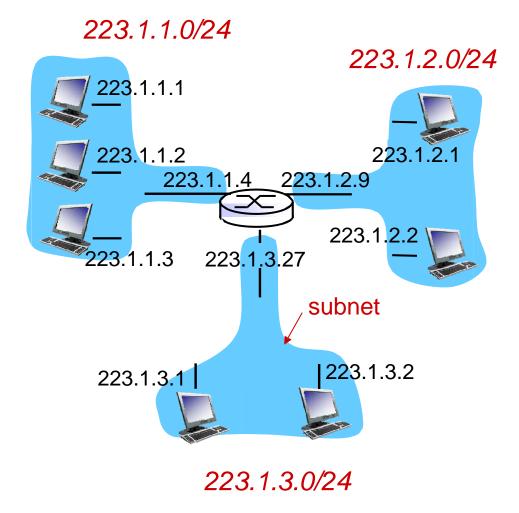
Mang con - Subnet

■ Địa chỉ IP:

- Phần subnet: các bit cao
- OPhần host: các bit thấp

Trong một subnet

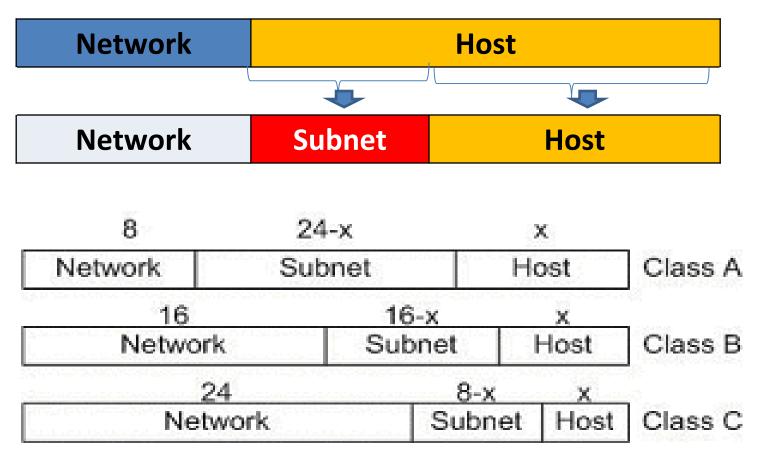
- ocác interface có cùng phần subnet
- ocó thể giao tiếp với nhau không cần qua router



subnet mask: /24

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 tối đa có thể mượn phải tuân theo công thức:

Subnet_id ≤ host_id - 2

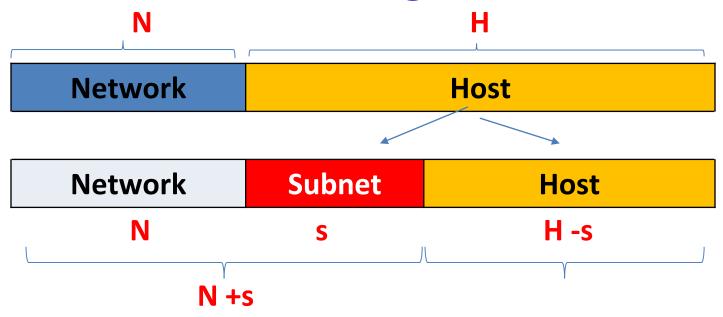
- ☐ Số lượng bit tối đa có thể mượn: lớp A: N=8,H=24
 - \triangleright Lớp A: 22 (= 24 2) bit -> chia được 2^{22} = 4194304 mạng con
 - \triangleright Lớp B: 14 (= 16 2) bit -> chia được 2^{14} = 16384 mạng con
 - ightharpoonup Lớp C: 6 (= 8 2) bit -> chia được 2^6 = 64 mạng con

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 thế biết được host này nằm trong mạng nào. 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).



Bước 1	Xác định Subnet mask ban đầu: N (bit) Xác định Host_ID ban đầu: H (bit)		
Bước 2	 Subnet_ID (số bit cần mượn): s (bit) Subnet mask mới: N + s (bit) Số lượng mạng con chia được: 2s Host_ID mới: H - s (bit) Số lượng host thực sự có được 2^{H-s} - 2 (2, trong đó 1 là địa chỉ mạng, 1 là địa chỉ quảng bá) 		
Bước 3	- Xác định các vùng địa chỉ host và chọn mạng con muốn dùng		

Ví dụ 1

- Cho một địa chỉ IP lớp B, 139.12.0.0, với subnet mask là 255.255.0.0.
- Hãy chia network thành 5 mạng con.

Bước 1

```
    □ Bước 1: Địa chỉ mạng 139.12.0.0/16
    ⇒ Subnet mask ban đầu: N = 16 (bit)
    Host_ID ban đầu: H = 16 (bit)
    139 . 12 . 0 . 0
    10001011 00001100 00000000 00000000
    Network ID
```

Bước 2

```
      139
      .
      12
      .
      0
      .
      0

      10001011
      000001100
      00000000
      00000000
```

Network_ID

- Subnet_ID (số bit cần mượn):
 cần chia 5 mạng con ⇒ s = 3 (bit)
- Số lượng mạng con chia được: 2^s = 2³ = 8
 Subnet_ID: 000 → 111
- Subnet mask mới: N + s = 16 + 3 = 19 (bit)
- Host_ID mới: H s = 16-3 = 13 (bit)

```
10001011 00001100 ssshhhhh hhhhhhhh
Network_ID subnet_ID host_ID
```

Bước 2: Địa chỉ mạng của tám Subnet mới

TT	Dạng nhị phân	Dạng thập phân
1	10001011.00001100.00000000.00000000	139.12.0.0/19
2	10001011.00001100.00100000.00000000	139.12.32.0/19
3	10001011.00001100.01000000.00000000	139.12.64.0/19
4	10001011.00001100.01100000.00000000	139.12.96.0/19
5	10001011.00001100.10000000.00000000	139.12.128.0/19
6	10001011.00001100.10100000.00000000	139.12.160.0/19
7	10001011.00001100.11000000.00000000	139.12.192.0/19
8	10001011.00001100.11100000.00000000	139.12.224.0/19

Bước 3: Cho biết vùng địa chỉ host của các mạng con

ТТ	Dạng nhị phân	Dạng thập phân
1	10001011.00001100.00000000.00000001 10001011.00001100.0001111.11111110	139.12.0.1/19 - 139.12.31.254/19
2	10001011.00001100.00100000.00000001 10001011.00001100.0011111.11111110	139.12.32.1/19 139.12.63.254/19
3	10001011.00001100.01000000.00000001 10001011.00001100.0101111.11111110	139.12.64.1/19 139.12.95.254/19
4	10001011.00001100.01100000.00000001 10001011.00001100.01111111.11111110	139.12.96.1/19 139.12.127.254/19
5	10001011.00001100.10000000.00000001 10001011.00001100.1001111.11111110	139.12.128.1/19 139.12.159.254/19
6	10001011.00001100.10100000.00000001 10001011.00001100.1011111.11111110	139.12.160.1/19 139.12.191.254/19
7	10001011.00001100.11000000.00000001 10001011.00001100.1101111.11111110	139.12.192.1/19 139.12.223.254/19
8	10001011.00001100.11100000.00000001 10001011.00001100.11111111	139.12.224.1/19 139.12.255.254/19

Bài tập áp dụng

Cho địa chỉ mạng 139.12.0.0/16:

- a) Chia mạng trên thành 4 mạng con
- b) Xác định địa chỉ mạng và địa chỉ quảng bá của mỗi mạng con
- c) Xác định dải địa chỉ host cho mỗi mạng con
- d) Tính số lượng địa chỉ host trong mỗi mạng con

Ví dụ tính nhanh vùng địa chỉ IP của các host

Cho địa chỉ: 192.168.0.0/24 Chia thành 16 mạng con Giải:

/24 => N=24, H=8

16 mạng con \Rightarrow s = 4, H'=8-4=4

Số lượng địa chỉ host trong một mạng con $M = 2^{H'} = 2^4 = 16$

Số lượng host thực sự có được: 16 - 2 = 14

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 host

- Địa chỉ mạng thuộc lớp C
- s số bit làm subnet
- Số mạng con: 2^s
- Số địa chỉ host trong mạng con: $M = 2^{8-s}$
- Số lượng host thực sự có được: M 2
 - ➤ Byte cuối của địa chỉ IP mạng: (i-1)*M (với i=1,2,...)
 - ▶Byte cuối của địa chỉ IP host đầu tiên: (i-1)*M + 1 (với i=1,2,...)
 - ▶Byte cuối của địa chỉ IP host cuối cùng: i*M 2 (với i=1,2,...)
 - ▶Byte cuối của IP broadcast, ví dụ lớp C: i*M 1 (với i=1,2,...)

```
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
```

```
192.168.1.0000 0000
192.168.1.0000 0001
192.168.1.0000 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
```

```
192.168.1.0
              - 255 /24
                                192.168.1.0000 0000
192.168.1.0 - 15 /28
                                192.168.1.0001 0000
192.168.1.16 - 31 /28
                                192.168.1.0010 0000
192.168.1.32 - 47 /28
                                192.168.1.0011 0000
192.168.1.48 - 63 /28
                                192.168.1.0100 0000
192.168.1.64 - 79 /28
                                192.168.1.0101 0000
192.168.1.80 - 95 /28
                                192.168.1.0110 0000
192.168.1.96 - 111 /28
                                192.168.1.0111 0000
192.168.1.112 - 127 /28
                                192.168.1.1000 0000
192.168.1.128 - 159 /28
                                192.168.1.1001 0000
192.168.1.144 - 191 /28
                                192.168.1.1010 0000
192.168.1.160 - 223 /28
                                192.168.1.1011 0000
192.168.1.176 - 223 /28
                                192.168.1.1100 0000
192.168.1.192 - 255 /28
                                192.168.1.1101 0000
192.168.1.208 - 255 /28
                                192.168.1.1110 0000
192.168.1.224 - 255 /28
                                192.168.1.1111 0000
192.168.1.240 - 255 /28
```

```
192.168.1.0/29
               192.168.1.0000 0000 /
               192.168.1.0000 0001 / 29
8 IP address
               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
 4 IP address
                192.168.1.0000 0010 / 30
                192.168.1.0000 0011 / 30
```

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

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

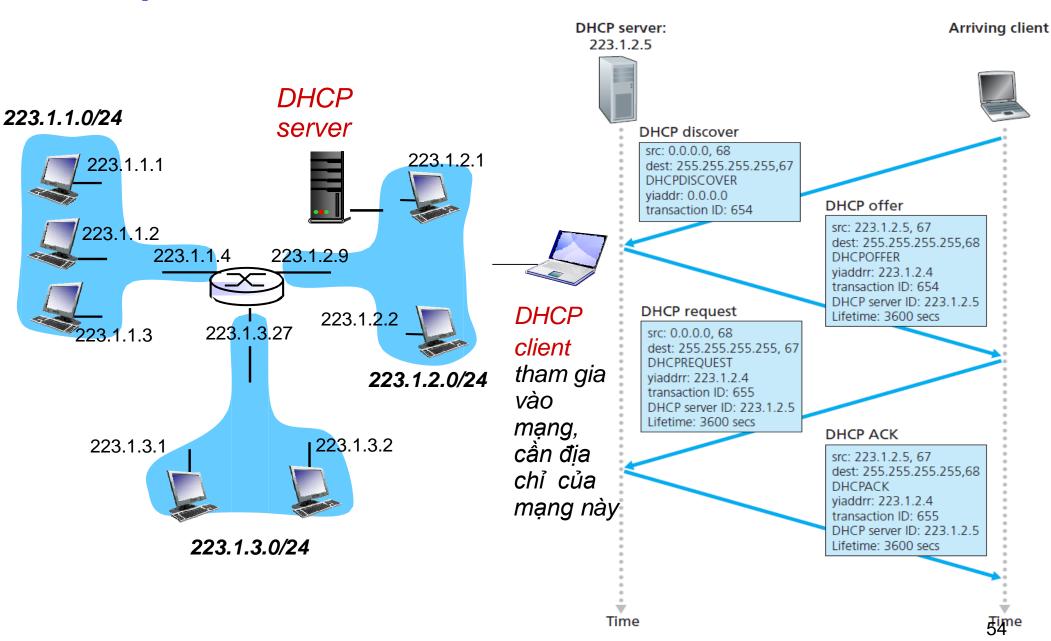
DHCP: cho phép host lấy địa chỉ IP động từ network server khi tham gia vào mạng

- o có thể làm mới địa chỉ đang dùng
- o 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 0001</u>0000 00000000 200.23.16.0/20

 Tổ chức 0
 11001000 00010111 0001000
 00000000
 200.23.16.0/23

 Tổ chức 1
 11001000 00010111 0001001
 00000000
 200.23.18.0/23

 Tổ chức 2
 11001000 00010111 0001010
 00000000
 200.23.20.0/23

 Tổ chức 7
 11001000 00010111 0001111
 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/)

- o cấp phát địa chỉ
- oquản lý DNS
- ogán tên miền, giải quyết tranh chấp

Chương 4: Tầng mạng

- Giới thiệu
- 2. IP: Internet Protocol
 - o IPv4
 - Cấu trúc datagram
 - O ICMP
 - IPv6

- 3. Định tuyến
 - Định tuyến là gì?
 - Phân loại định tuyến
 - Thuật toán định tuyến
- 4. Dẫn đường trong Internet
 - o RIP
 - OSPF
 - BGP

Cấu trúc của IPv4 datagram

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

Chức năng các trường

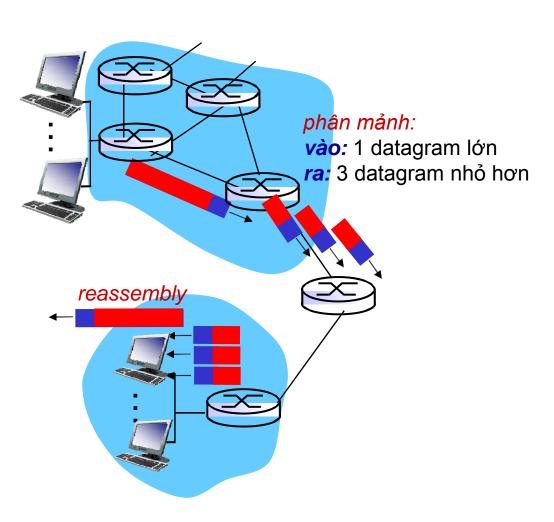
- VER (4 bits): Version hiện hành của IP được cài đặt.
- ➤ IHL(4 bits): Internet Header Length của Datagram, tính theo đơn vị word (32 bits).
- Type of service (8 bits): Thông tin về loại dịch vụ và mức ưu tiên của gói IP:
- Datagram Length (16 bits): Chỉ độ dài toàn bộ Datagram (bytes),
- Identification (16bits), Flags (3 bits), Fragment Offset (13 bits): được sử dụng khi phân mảnh gói IP
- Time To Live (TTL-8 bits): Thời gian sống của một gói dữ liệu (số hop tối đa còn).

Chức năng các trường

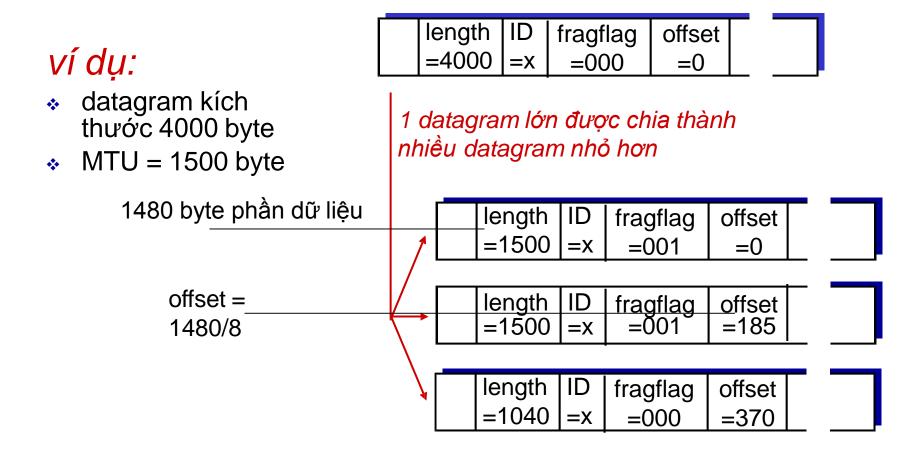
- > Protocol (8 bits): Chỉ giao thức sử dụng TCP hay UDP.
- Header Checksum (16 bits): giá trị checksum của trường tiêu đề.
- Source Address (32 bits): địa chỉ của trạm nguồn.
- Destination Address (32 bits): Địa chỉ của trạm đích.
- Option (có độ dài thay đổi): Sử dụng trong trường hợp bảo mật, định 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 bít trong IP header dùng để ghép các phân mảnh



Ghép phân mảnh



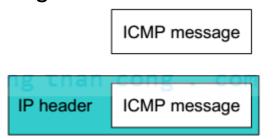
Chương 4: Tầng mạng

- Giới thiệu
- 2. IP: Internet Protocol
 - o IPv4
 - Cấu trúc datagram
 - O ICMP
 - IPv6

- 3. Định tuyến
 - Định tuyến là gì?
 - Phân loại định tuyến
 - Thuật toán định tuyến
- 4. Dẫn đường trong Internet
 - o RIP
 - OSPF
 - BGP

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 cácIP datagram



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

<u>Type</u>	<u>Code</u>	<u>Mô tả</u>
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

Chương 4: Tầng mạng

- 1. Giới thiệu
- 2. IP: Internet Protocol
 - o IPv4
 - Cấu trúc datagram
 - O ICMP
 - o IPv6

- 3. Định tuyến
 - Định tuyến là gì?
 - Phân loại định tuyến
 - Thuật toán định tuyến
- 4. Dẫn đường trong Internet
 - o RIP
 - OSPF
 - BGP

IPv6

- Lý do ban đầu: Không gian địa chỉ 32-bit của IPv4 sẽ nhanh chóng dùng hết
- Lý do khác:
 - Đơn giản hóa header để tăng tốc độ xử lý/ chuyển tiếp
 - thay đổi header để hỗ trợ chất lượng dịch vụ (QoS -Quality of Service)

Cấu trúc của IPv6 datagram:

- o header có chiều dài cố định 40 byte
- o không cho phép phân mảnh

Cấu trúc IPv6 datagram

priority: xác định ưu tiên giữa các datagram trong flow flow Label: xác định các datagram trong cùng flow (khái niệm "flow" không định nghĩa rõ)

Payload length: độ dài của gói datagram

next header: xác định giao thức của tầng cao hơn cho

data

ver	pri	flow label			
Ķ	bayload	len	next hdr	hop limit	
	source address (128 bits)				
destination address (128 bits)					
data					
32 bits					

Cấu trúc IPv6 datagram

Địa chỉ IPv6: 128 bit, biểu diễn ở hệ hex

Ví dụ:

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

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

2001:ABC:AB:A::1001

Các thay đổi khác so với IPv4

- checksum: loại bỏ để giảm thời gian xử lý tại mỗi hop
- options: cho phép, nhưng ở ngoài phần header, xác định bởi trường "Next Header"
- □ ICMPv6: phiên bản mới của ICMP
 - o thêm kiểu bản tin, ví dụ "Packet Too Big"
 - o chức năng quản lý nhóm multicast

Chương 4: Tầng mạng

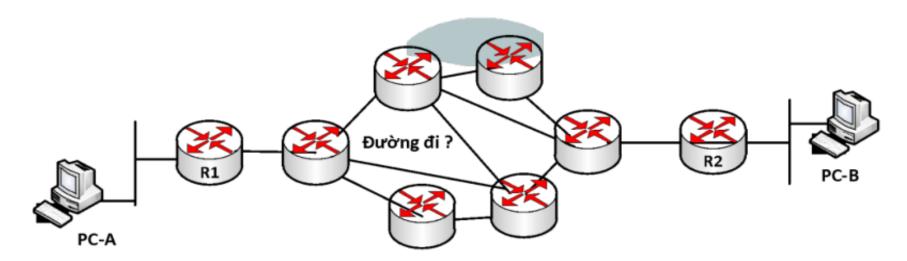
- Giới thiệu
- 2. IP: Internet Protocol
 - Cấu trúc datagram
 - IPv4
 - Cấu trúc datagram
 - ICMP
 - o IPv6

3. Định tuyến

- Định tuyến là gì?
- Phân loại định tuyến
- Thuật toán định tuyến
- 4. Dẫn đường trong Internet
 - ORIP
 - OSPF
 - BGP

Định tuyến là gì?

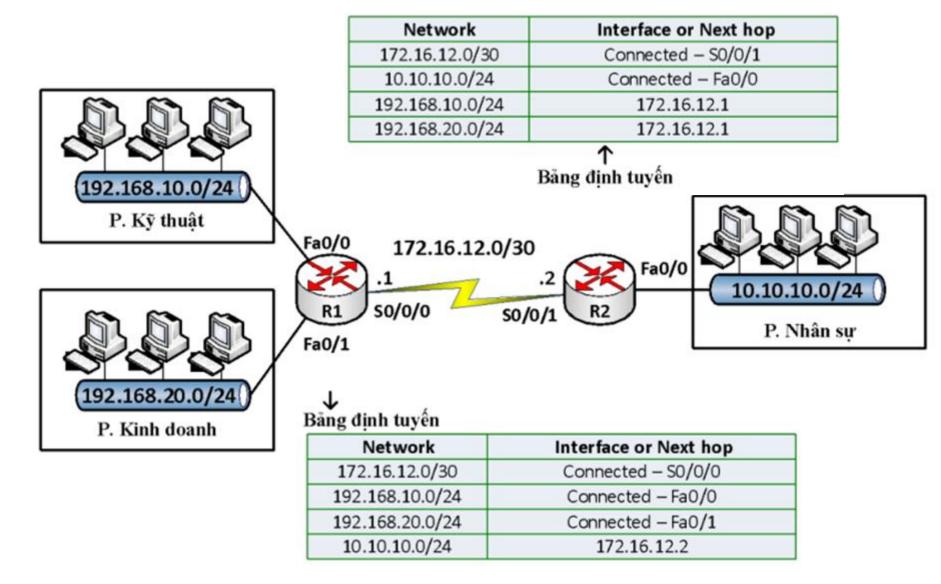
- ☐ Khi một host gửi một gói tin IP tới một host khác
 - Nếu địa chỉ đích nằm trên cùng một đường truyền vật lý: chuyển trực tiếp
 - Nếu địa chỉ đích nằm trên một mạng khác: chuyển gián tiếp qua bộ định tuyến Router
- Định tuyến là quá trình xác định đường đi tốt nhất của gói tin từ nút gửi đến nút nhận



Thông tin định tuyến được lưu trữ ở đâu?

- ☐ Thông tin về tuyến đường đi được lưu trữ trong bảng định tuyến (routing table)
- □ Bảng định tuyến thường gồm những thông tin sau:
 - ✓ Địa chỉ đích của mạng, mạng con/mặt nạ mạng
 - ✓ Địa chỉ IP của router chặng kế tiếp phải đến
 - ✓ Giao tiếp vật lí phải sử dụng để đi đến Router kế tiếp
 - ✓ Số lượng chặng để đến đích

Ví dụ bảng định tuyến



Lưu ý: Khi có sự thay đổi như thêm mạng mới, một nút mạng nào đó bị mất điện thì phải cập nhật bảng định tuyến

Nhiệm vụ của Router

- □Router thực hiện các việc sau:
 - ≻Kiểm tra tiêu đề lớp 3 của gói nhận được (địa chỉ IP đích)
 - ➤ Chọn đường tốt nhất cho gói dữ liệu
 - ➤ Chuyển gói dữ liệu ra cổng tương ứng

Chương 4: Tầng mạng

- Giới thiệu
- 2. IP: Internet Protocol
 - Cấu trúc datagram
 - IPv4
 - Cấu trúc datagram
 - ICMP
 - o IPv6

- 3. Định tuyến
 - Định tuyến là gì?
 - Phân loại định tuyến
 - Thuật toán định tuyến
- 4. Dẫn đường trong Internet
 - ORIP
 - OSPF
 - BGP

Phân loại định tuyến

- □Định tuyến tĩnh (Static Routing):
 - Các thông tin trong bảng định tuyến được sửa đổi thủ công bởi người quản trị
- ☐ Định tuyến động (Dynamic Routing):
 - Các thông tin trong bảng định tuyến được tự động cập nhật bằng các giao thức định tuyến

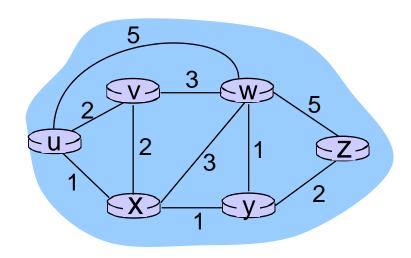
Chương 4: Tầng mạng

- Giới thiệu
- 2. IP: Internet Protocol
 - Cấu trúc datagram
 - IPv4
 - Cấu trúc datagram
 - ICMP
 - o IPv6

- 3. Định tuyến
 - Định tuyến là gì?
 - Phân loại định tuyến
 - Thuật toán định tuyến
- 4. Dẫn đường trong Internet
 - ORIP
 - OSPF
 - BGP

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

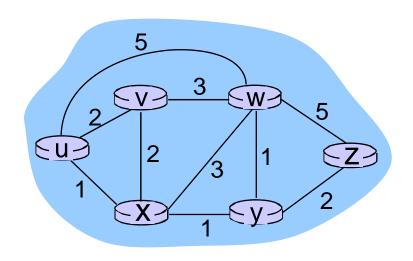


Đồ thị: G = (N,E)

 $N = t\hat{a}p các router = \{ u, v, w, x, y, z \}$

E = 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), ...

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 $(x_1, x_2, x_3, ..., x_p) = c(x_1, x_2) + c(x_2, x_3) + ... + c(x_{p-1}, x_p)$

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 kiểu Link-State

Thuật toán Dijkstra

- Mọi nút mạng có đầy đủ thông tin về sơ đồ mạng và chi phí các liên kết
 - Quảng bá "link state"
 - Mọi nút có cùng thông tin
- Tính toán đường đi có chi phí thấp nhất từ một nút tới các nút khác
 - Gửi forwarding table cho nút đó
- Lặp: sau k bước lặp, tính được đường đi có chi phí thấp nhất tới k đích

Ký hiệu:

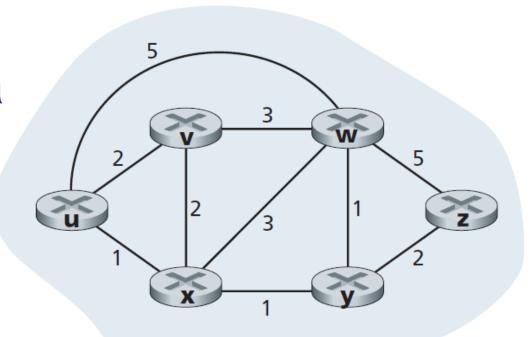
- C(X,y): chi phí liên kết từ nút x tới nút y; = ∞ nếu không là nút kể
- D(v): giá trị hiện tại thấp nhất của chi phí đường đi từ nguồn tới nút v
- p(v): nút trước v trên đường đi có giá thấp nhất từ nguồn tới v
- N': tập các nút mà đường đi với chi phí nhỏ nhất tới nút đã được xác định

Thuật toán Dijsktra

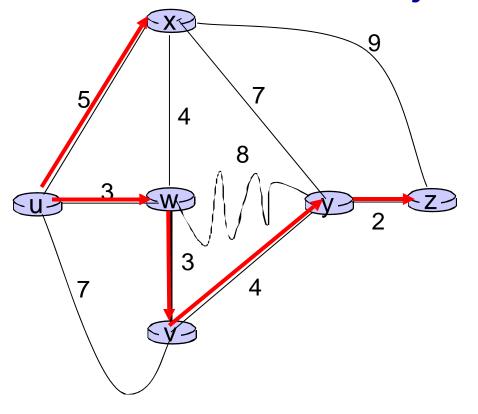
- 1. Khởi tạo:
- 2. $N' = \{u\}$
- 3. for all nodes v
- 4. if v kề u then D(v) = c(u,v), P(v) = u
- 5. else
- 6. $D(v) = \infty$
- 7. Loop
- 8. tìm w không trong N' mà D(w) nhỏ nhất
- 9. thêm w vào N'
- 10. cập nhật D(v) cho mọi v kề với w và không trong N':

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

- 11. P(v) = w n'eu D(w) + c(w,v) < D(v)
- 12. /* chi phí mới tới v sẽ hoặc là chi phí cũ tới v hoặc là chi phí đường đi ngắn nhất tới w cộng với chi phí từ w tới v */
- 13. until mọi nút nằm trong N'



Thuật toán Dijkstra: Ví dụ 1



		D(v)	D(w)	$D(\mathbf{x})$	D(y)	D(z)
B <u>ướ</u>	c N'	p(v)	p(w)	p(x)	p(y)	p(z)
0	u	7,u	(3,u)	5,u	∞	∞
1	uw	6,w		5,u) 11,W	∞
2	uwx	6,w			11,W	14,x
3	uwxv				10,V	14,x
4	uwxvy					(12,y)
5	uwxvyz					

U->v:6, v,w,u->u,w,v

U->w: 3, w,u->u, w

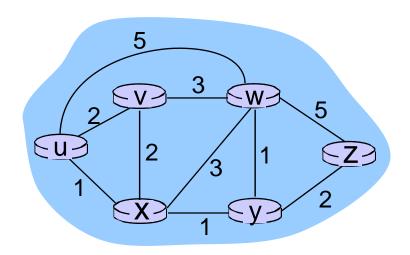
U->x:5: x,u -> u,x

U->y:10: y,v,w,u->u,w,v,y

U->z: 12: z,y,v,w,u -> u,w,v,y,z

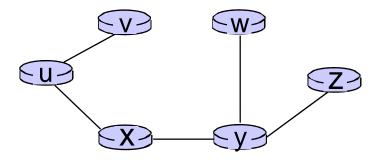
Thuật toán: Ví dụ 2

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	∞	∞
1	ux ←	2,u	4,x		2,x	∞
2	uxy <mark>←</mark>	2,u	3,y			4,y
3	uxyv 🗸		3,y			4,y
4	uxyvw 🗸					4,y
5	uxyvwz ←					



Thuật toán Dijkstra: Ví dụ 2

Kết quả cây đường đi ngắn nhất từ u:

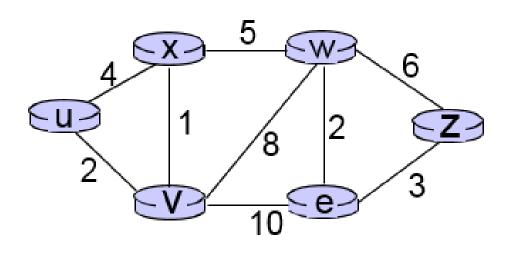


Kết quả forwarding table tại u:

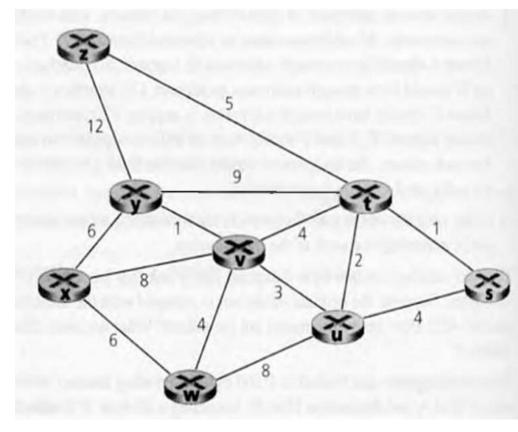
đích		liên kết
	V	(u,v)
	Χ.	(u,x)
	у.	(u,x)
	W	(u,x)
	Z	(u,x)

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

Đáp án

b)

Step	N'	D(x), $p(x)$	$D(t)_{i}p(t)$	$D(u)_{,p}(u)$	D(v),p(v)	D(w),p(w)	D(y),p(y)	D(z),p(z)
0	S	00	1,s	4,s	00	00	00	00
1	st	00	1,s	3,t	5,t	00	8,t	6,t
2	stu	00	1,s	3,t	5,t	6,u	8,t	6,t
3	stuv	8,v	1.s	3,t	5,t	6,u	6.v	6,t
4	stuvy	8,v	1,s	3,t	5,t	6, u	6.v	6,t
5	stuvyz	8,v	1,s	3,t	5,t	6, u	6,v	6,t
6	stuvyzw	8,v	1.s	3,t	5,t	6, u	6.v	6,t
7	stuvyzwx	8,v	1,s	3,t	5,t	6, u	6,v	6,t

Đáp án a)

Bước	N'	D(x),p(x)	D(v),p(v)	D(w),p(w)	D(e),p(e)	D(z),p(z)
0	u	4,u (2,u	00	8	8
1	uv (3,v		10,v	12,v	8
2	uvx		(8,x	12,v	8
3	uvxw				(10,w)	14,w
4	uvxwe)	13,e)
5	uvxwez					

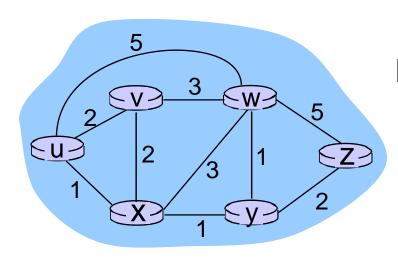
Thuật toán Distance vector

Công thức Bellman-Ford (quy hoạch động)

```
let d_x(y) := \text{chi phí của đường đi có chi phí nhỏ} \\ \text{nhất từ x tới y} \\ \text{then} \qquad \qquad \text{chi phí từ nút kề v tới đích y} \\ d_x(y) = \min_v \left\{ c(x,v) + d_v(y) \right\} \\ \text{chi phí từ x tới nút kề v} \\
```

Cho tất cả các v là nút liến kề với x

Ví dụ Bellman-Ford



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

Công thức B-F:

$$d_{u}(z) = \min \{ c(u,v) + d_{v}(z), \\ c(u,x) + d_{x}(z), \\ c(u,w) + d_{w}(z) \}$$

$$= \min \{ 2 + 5, \\ 1 + 3, \\ 5 + 3 \} = 4$$

Nút nào làm giá trị trên nhỏ nhất ⇒ lựa chọn là nút kế tiếp trong bảng chọn đường (forwarding table)

Thuật toán Distance vector

- $\square D_x(y) = danh giá chi phí thấp nhất từ x tới y$
 - \circ x duy trì véc tơ khoảng cách $D_x = [D_x(y): y \in N]$
- □ nút x:
 - obiết chi phí tới nút kề v: c(x,v)
 - oduy trì véc tơ khoảng cách của các nút kề. Với mỗi nút kề v, x duy trì
 - $\mathbf{D}_{\mathsf{v}} = [\mathsf{D}_{\mathsf{v}}(\mathsf{y}): \mathsf{y} \in \mathsf{N}]$

Thuật toán Distance vector

Ý tưởng:

- DV: vector khoảng cảnh, tạm gọi là đường đi ngắn nhất từ một nút tới những nút khác
- Định kì, mỗi nút gửi DV của nó tới các nút kề
- Khi nút x nhận được một DV, nó cập nhật DV của nó theo công thức B-F:

$$D_x(y) \leftarrow \min_{v} \{c(x,v) + D_v(y)\} \ v \circ i \ y \in \mathbb{N}$$

Với một số điều kiện, D_x(y) sẽ dần hội tụ tới chi phí thấp nhất d_x(y)

mỗi nút:

đợi thay đổi trong DV của nút kề

tính lại DV của nó

Nếu DV tới bất kì đích nào thay đổi, *thông báo* cho các nút kề

Distance-Vector (DV) Algorithm

At each node, *x*:

```
Initialization:
       for all destinations y in N:
          D_{x}(y) = c(x,y) /* if y is not a neighbor then c(x,y) = \infty */
       for each neighbor w
          D_{w}(y) = ? for all destinations y in N
       for each neighbor w
6
          send distance vector \mathbf{D}_{x} = [D_{x}(y): y \ in \ N] to w
8
   loop
9
10
      wait (until I see a link cost change to some neighbor w or
11
              until I receive a distance vector from some neighbor w)
12
13
       for each y in N:
14
          D_{x}(y) = \min_{v} \{c(x,v) + D_{v}(y)\}
15
      if D_x(y) changed for any destination y
16
          send distance vector \mathbf{D}_{x} = [D_{x}(y): y \text{ in N}] to all neighbors
17
18
19 forever
```

$$D_{x}(y) = \min\{c(x,y) + D_{y}(y), c(x,z) + D_{z}(y)\}$$

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

$$D_{y}(z) = \min\{c(x,y) + D_{y}(z), c(x,z) + D_{z}(z)\}$$

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

$$D_{x}(z) = \min\{c(x,y) + D_{y}(z), c(x,z) + D_{z}(z)\}$$

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

$$D_{x}(z) = \min\{c(x,y) + D_{y}(z), c(x,z) + D_{z}(z)\}$$

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

$$D_{x}(z) = \min\{c(x,y) + D_{y}(z), c(x,z) + D_{z}(z)\}$$

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

$$D_{x}(z) = \min\{c(x,y) + D_{y}(z), c(x,z) + D_{z}(z)\}$$

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

$$D_{x}(z) = \min\{c(x,y) + D_{y}(z), c(x,z) + D_{z}(z)\}$$

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

$$D_{x}(z) = \min\{c(x,y) + D_{y}(z), c(x,z) + D_{z}(z)\}$$

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

$$D_{x}(z) = \min\{c(x,y) + D_{y}(z), c(x,z) + D_{z}(z)\}$$

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

$$D_{x}(z) = \min\{c(x,y) + D_{y}(z), c(x,z) + D_{z}(z)\}$$

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

$$D_{x}(z) = \min\{c(x,y) + D_{y}(z), c(x,z) + D_{z}(z)\}$$

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

$$D_{x}(z) = \min\{c(x,y) + D_{z}(z)\}$$

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

$$D_{x}(z) = \min\{c(x,y) + D_{z}(z)\}$$

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

$$D_{x}(z) = \min\{c(x,y) + D_{z}(z)\}$$

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

$$D_{x}(z) = \min\{c(x,y) + D_{z}(z)\}$$

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

$$D_{x}(z) = \min\{c(x,y) + D_{z}(z)\}$$

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

$$D_{x}(z) = \min\{c(x,y) + D_{z}(z)\}$$

$$= \min\{c(x,y) + D_{z}(x) + D_{z}(x)\}$$

$$= \min\{c(x,y) + D_{z}(x) + D_{z}(x)$$

$$= \min\{c(x,y) + D_{z}(x) + D_$$

thời gian

$$D_{x}(y) = \min\{c(x,y) + D_{y}(y), c(x,z) + D_{z}(y)\}$$

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

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

$$\begin{vmatrix}
b_{ang} & chi phitoi \\
nút & x & y & z \\
x & 0 & 2 & 7 \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x & y & z \\
x & 0 & 2 & 3
\end{vmatrix}$$

$$\begin{vmatrix}
chi phitoi \\
x$$

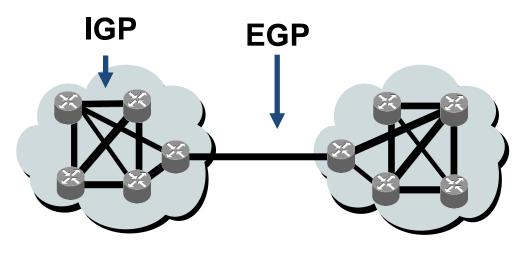
Chương 4: Tầng mạng

- Giới thiệu
- 2. IP: Internet Protocol
 - Cấu trúc datagram
 - IPv4
 - Cấu trúc datagram
 - ICMP
 - o IPv6

- 3. Định tuyến
 - Định tuyến là gì?
 - Phân loại định tuyến
 - Thuật toán định tuyến
- 4. Dẫn đường trong Internet
 - ORIP
 - OSPF
 - BGP

Định tuyến trong Internet

- ➡ Định tuyến trong IGP (Interior Gateway Protocol): là giao thức định tuyến bên trong một hệ thống tự quản (AS Autonomous System)
 - Phổ biến là RIP (Routing Information Protocol), OSPF (Open Shortest Path First), IGRP (Interior Gateway Routing Protocol)
- □ Định tuyến ngoài EGP (Exterior Gateway Protocol): là giao thức trao đổi thông tin giữa các AS khác nhau.
 - ➤ Phổ biến là BGP (Border Gateway Protocol)



Autonomous System 1

Autonomous System 2

Hết chương 4