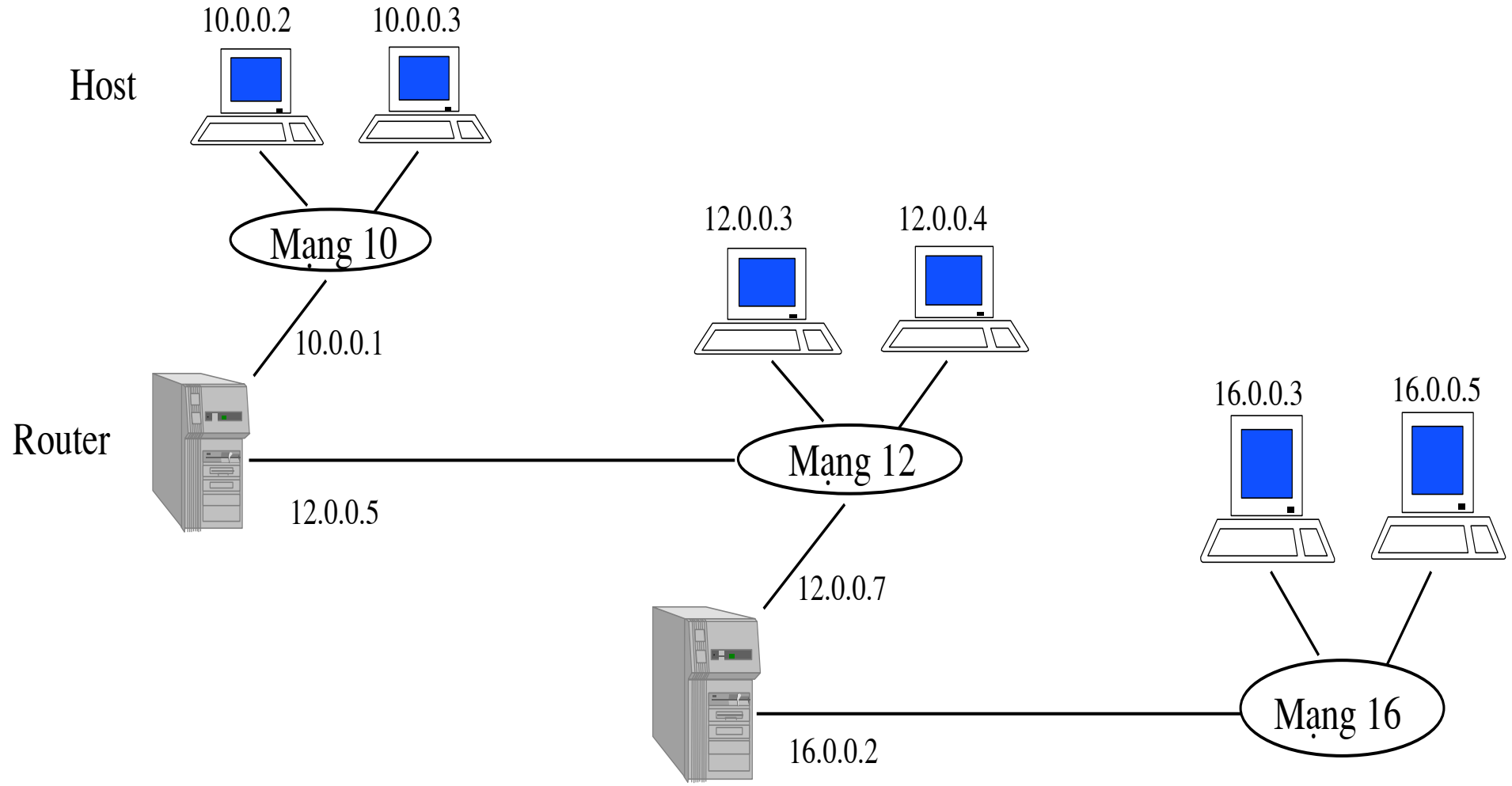
Mạng viễn thông
Chương 4: Mạng IP

Nội dung: Mạng IP

- Lý thuyết
  - Bộ giao thức TCP/IP
  - Lớp liên mạng
    - Giao thức IP
    - Cấu trúc gói tin IP
  - Lớp vận chuyển
    - Giao thức UDP
    - Giao thức TCP
  - Lớp ứng dụng
- Minh họa hoạt động của mạng IP

Mạng máy tính



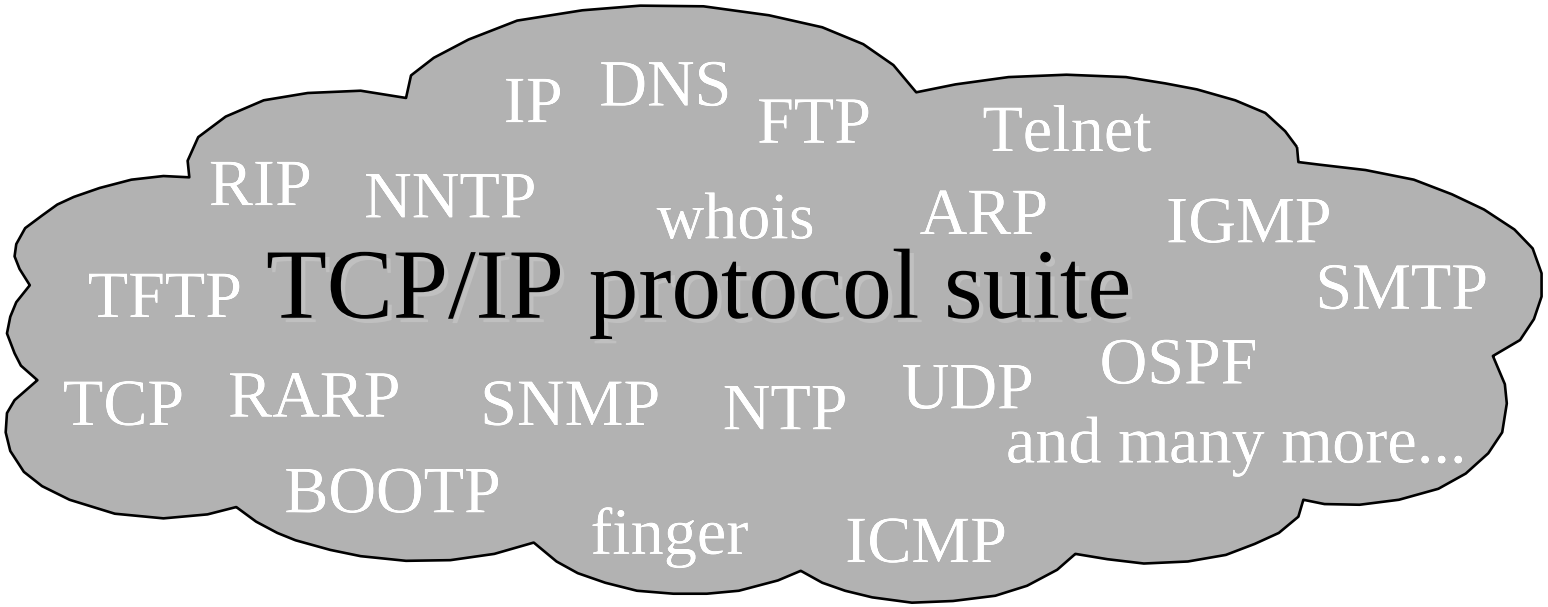
Bộ giao thức TCP/IP

- Bộ giao thức điều khiển truyền thông/giao thức Internet (TCP/IP) là một tên dùng chung cho một họ các giao thức tiêu chuẩn cho việc trao đổi thông tin giữa máy tính-máy tính.
- Hiện nay, TCP/IP được sử dụng rất phổ biến trong mạng máy tính, mà điển hình là mạng Internet.

1. TCP: Transmission Control Protocol: Giao thức điều khiển truyền dẫn
2. TCP/IP protocol suite: Chồng/ bộ giao thức TCP/IP

Bộ giao thức TCP/IP

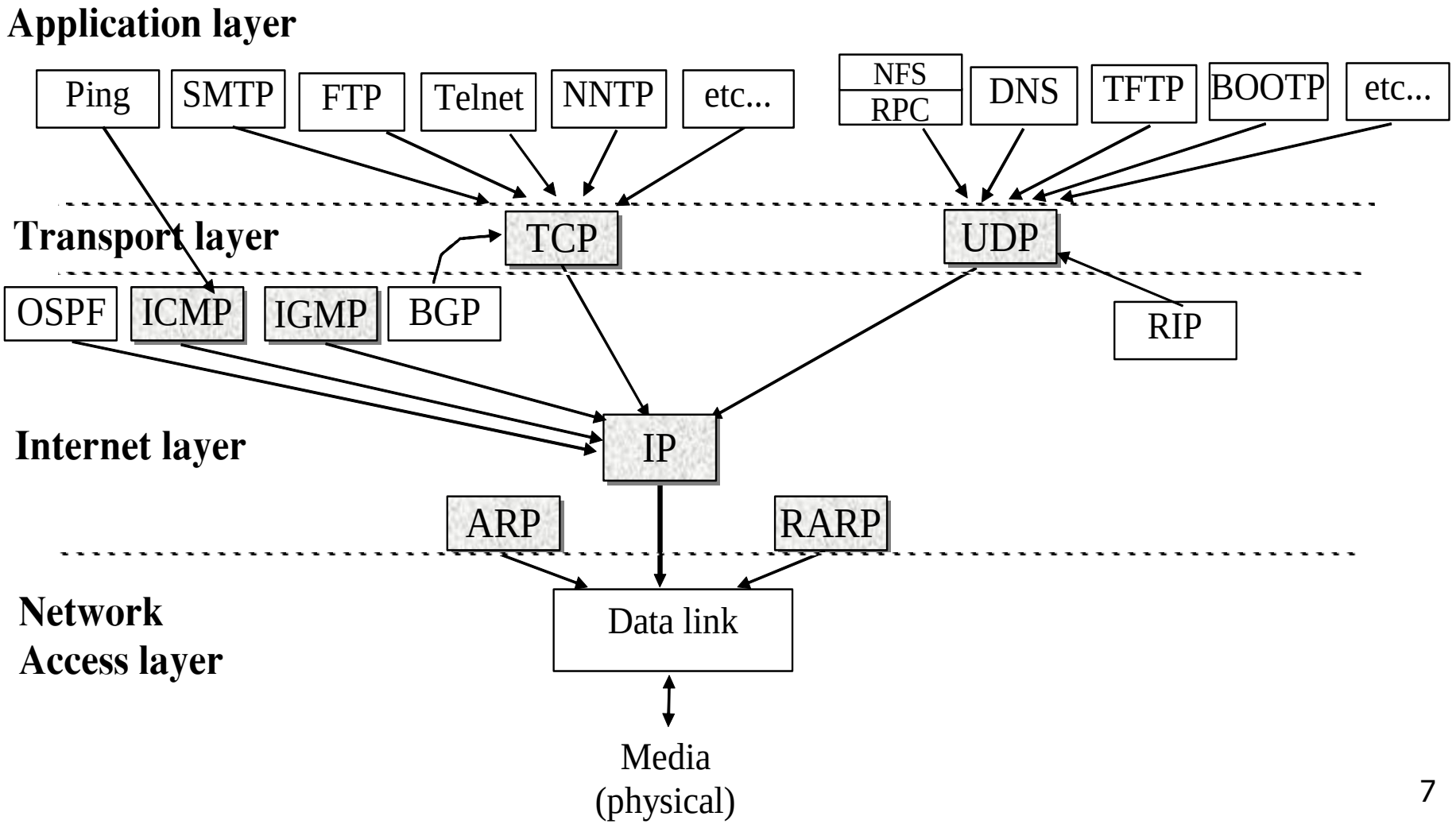
- Bộ giao thức TCP/IP: *TCP/IP Protocol Suite*



Bộ giao thức TCP/IP

- TCP/IP được phát triển trước mô hình OSI → các tầng trong TCP/IP không tương ứng hoàn toàn với các tầng trong mô hình OSI
- Bộ giao thức TCP/IP được chia thành bốn tầng:
  - Lớp 4- Application layer (lớp ứng dụng)
  - Lớp 3- Transport layer (lớp vận chuyển)
  - Lớp 2- Internet Layer (lớp Internet – đôi khi được gọi là lớp liên mạng)
  - Lớp 1- Network Access Layer/ Network Interface and Hardware (lớp truy nhập mạng, đôi khi được gọi là lớp giao diện mạng)

Bộ giao thức TCP/IP

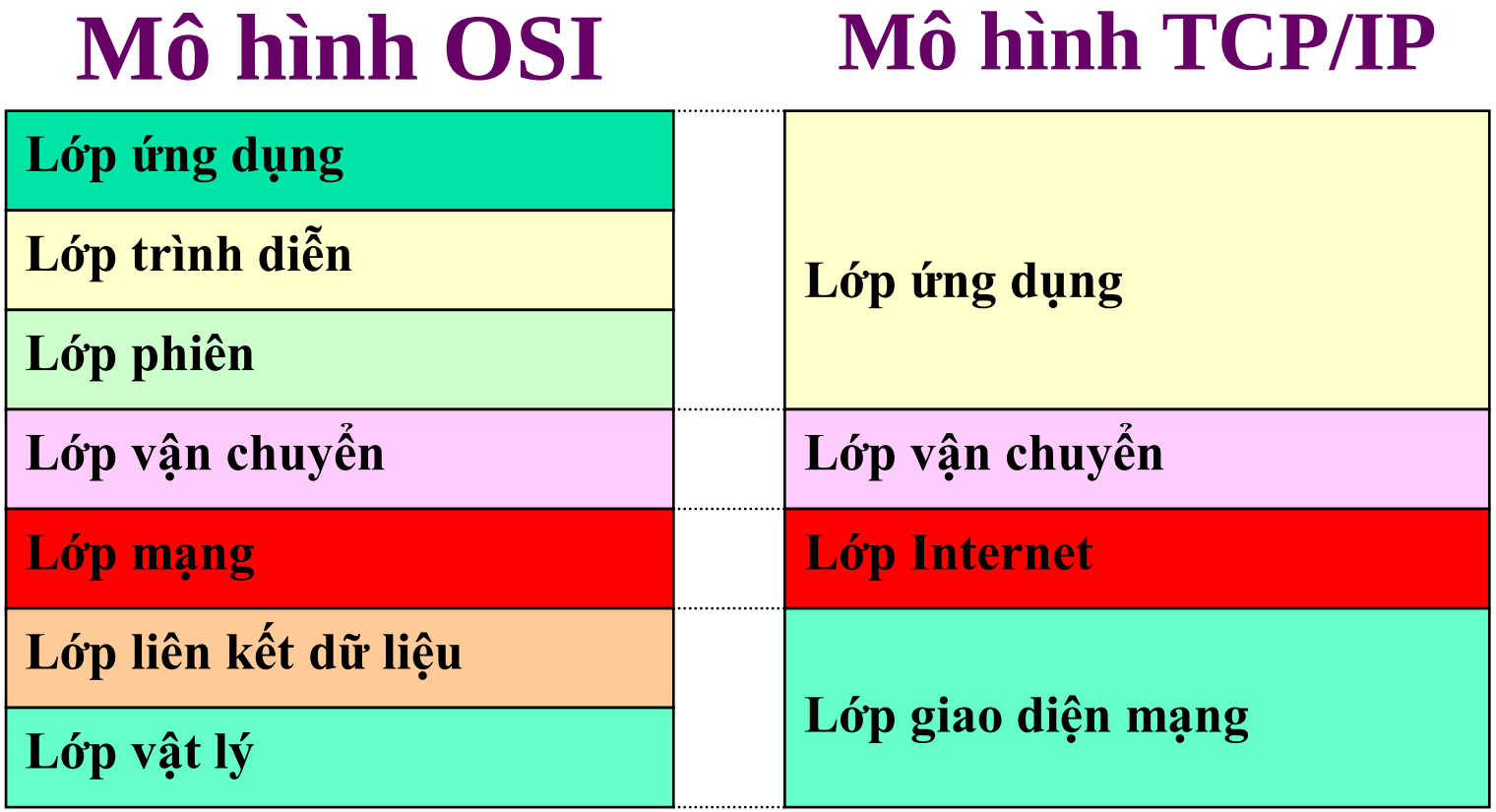


Bộ giao thức TCP/IP

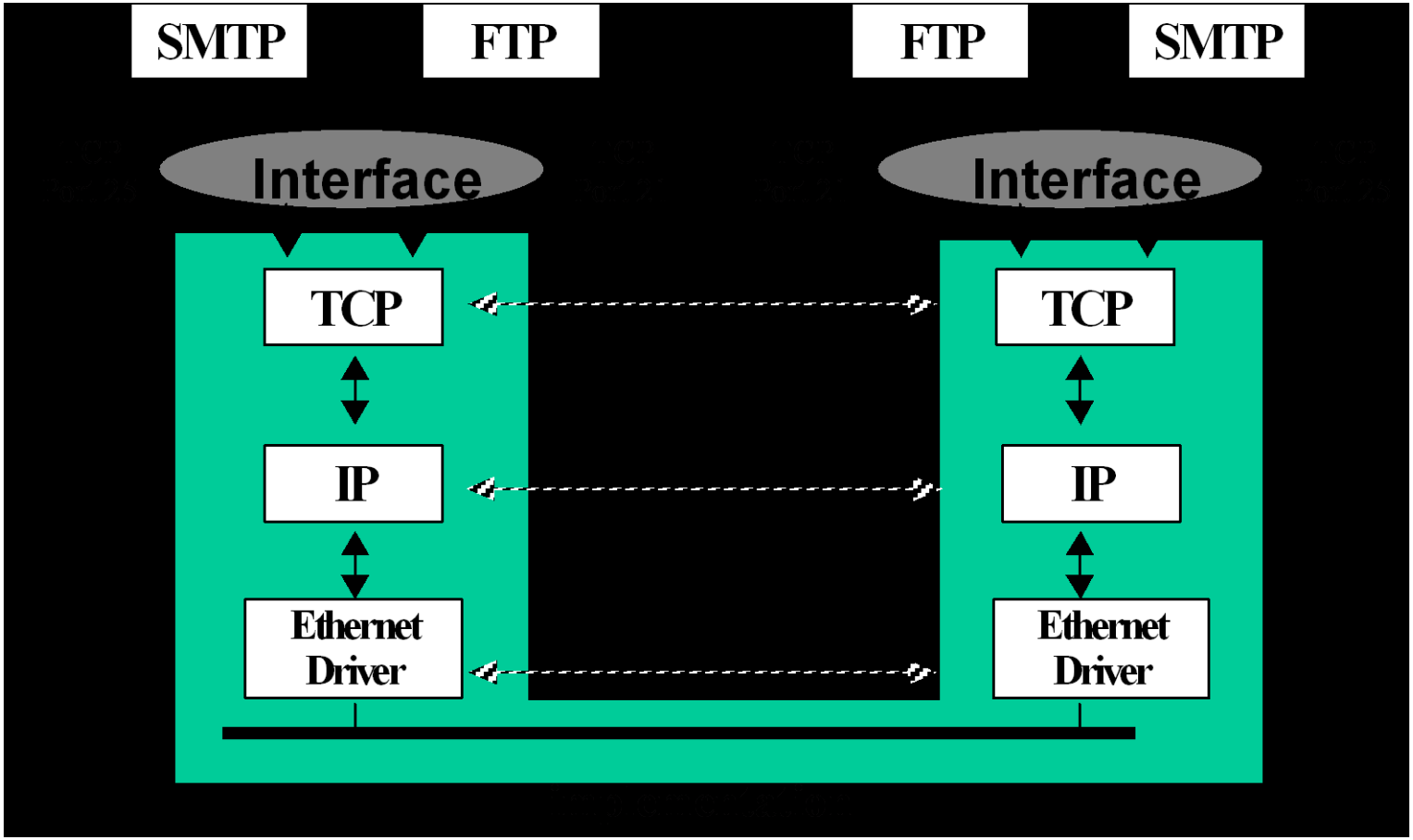
- Nhận xét:
  - *Bộ giao thức TCP/IP là sự kết hợp của các giao thức khác nhau ở các lớp khác nhau, không chỉ có các giao thức TCP và IP. Mỗi lớp lại có chức năng riêng.*
  - *Hầu hết các dữ liệu truyền trên bộ giao thức TCP/IP đều kết thúc đóng gói ở dữ liệu đồ IP (IP datagram), trừ ARP và RARP được đóng gói trực tiếp ở Khung lớp liên kết (Link Level Frames)*

Bộ giao thức TCP/IP

- Đối chiếu với mô hình OSI



Hoạt động cơ bản của chồng giao thức



Ưu điểm của kết nối mạng IP

- Giao thức IP được sử dụng rộng rãi trên phạm vi toàn cầu cho kết nối mạng dữ liệu vì:
  - mạng sử dụng giao thức IP loại bỏ ranh giới giữa dịch vụ số liệu và thoại
  - giao thức IP độc lập với lớp liên kết dữ liệu
  - các mạng IP được xây dựng dựa trên các tiêu chuẩn toàn cầu của IETF
  - phần cứng và phần mềm IP cung cấp độ tin cậy và chất lượng dịch vụ số liệu cao hơn trước đây

Lớp liên mạng: Giao thức IP

- Lớp liên mạng trong chồng giao thức TCP/IP tương ứng với lớp mạng trong mô hình OSI.
- Chức năng chính của lớp liên mạng là đánh địa chỉ lôgic và định tuyến gói tới đích.
- Giao thức đáng chú ý nhất ở lớp liên mạng chính là giao thức liên mạng (IP – Internet Protocol).
- Ngoài ra còn có một số giao thức khác như ICMP, ARP và RARP.

Các giao thức lớp liên mạng

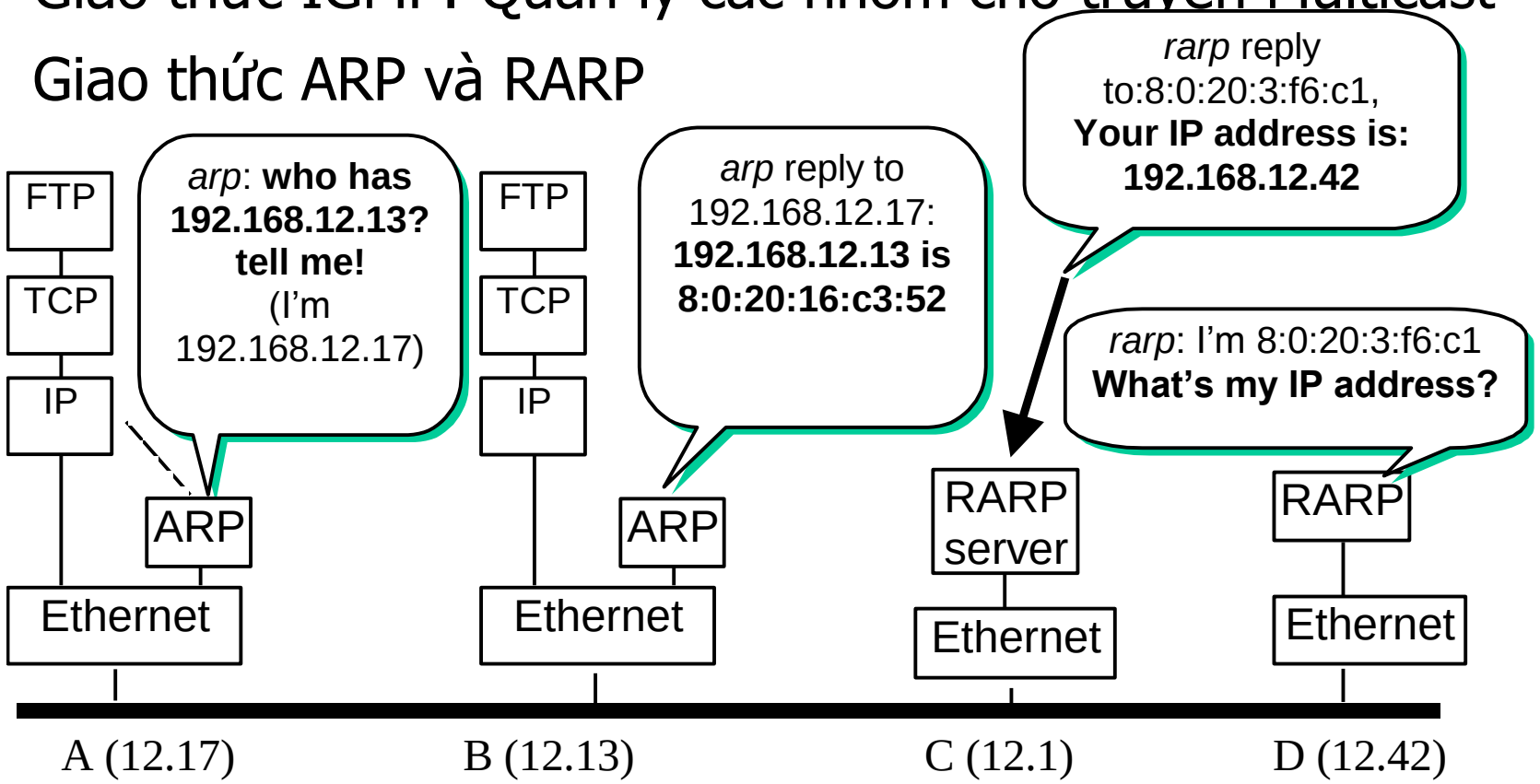
- Giao thức Internet (IP) là giao thức
  - **chuyển mạch gói phi kết nối**
  - **không tin cậy**
  - **dựa trên nguyên lý nỗ lực tốt nhất.** Nỗ lực nhất ở đây có nghĩa IP không cung cấp chức năng theo dõi và kiểm tra lỗi. Nó chỉ cố gắng chuyển gói tới đích chứ không có sự đảm bảo.
- Giao thức này làm việc tại lớp mạng, tương đương với lớp 3 trong mô hình OSI

- IP: Internet Protocol
- OSI: Open System Interconnection

- Nếu độ tin cậy là yếu tố quan trọng, IP phải hoạt động với một giao thức lớp phía trên tin cậy, chẳng hạn TCP.
- Số hiệu nhận dạng được sử dụng ở lớp liên mạng của bộ giao thức TCP/IP được gọi là địa chỉ liên mạng (địa chỉ IP).
  - địa chỉ nhị phân 32 bít
  - được thực thi trong phần mềm
  - dùng để định danh duy nhất và toàn cục một trạm hoặc một router trên liên mạng

Các giao thức lớp liên mạng

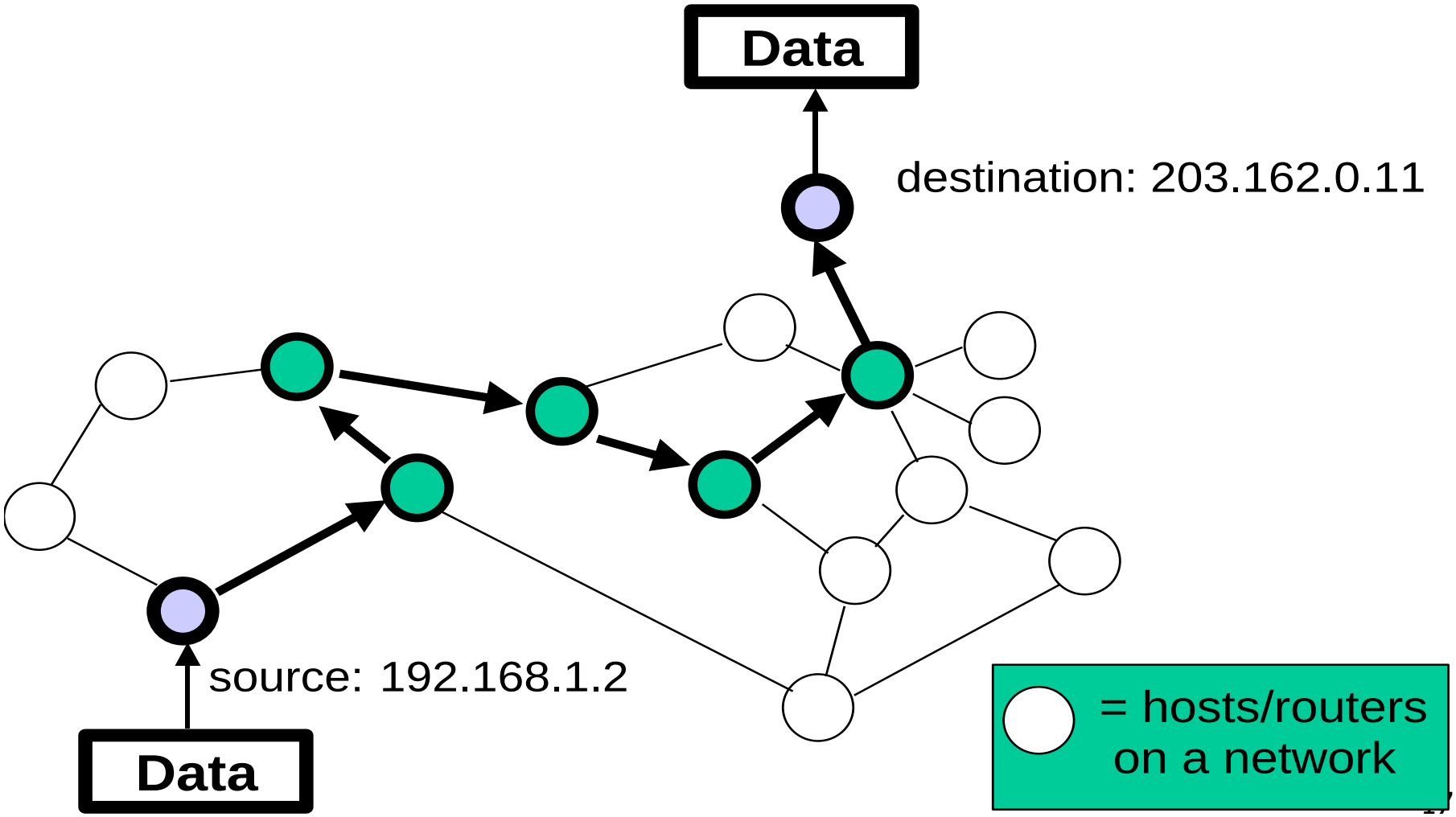
- Giao thức ICMP:Giao thức thông báo điều khiển liên mạng (Internet Control Message Protocol).
- Giao thức IGMP: Quản lý các nhóm cho truyền Multicast
- Giao thức ARP và RARP



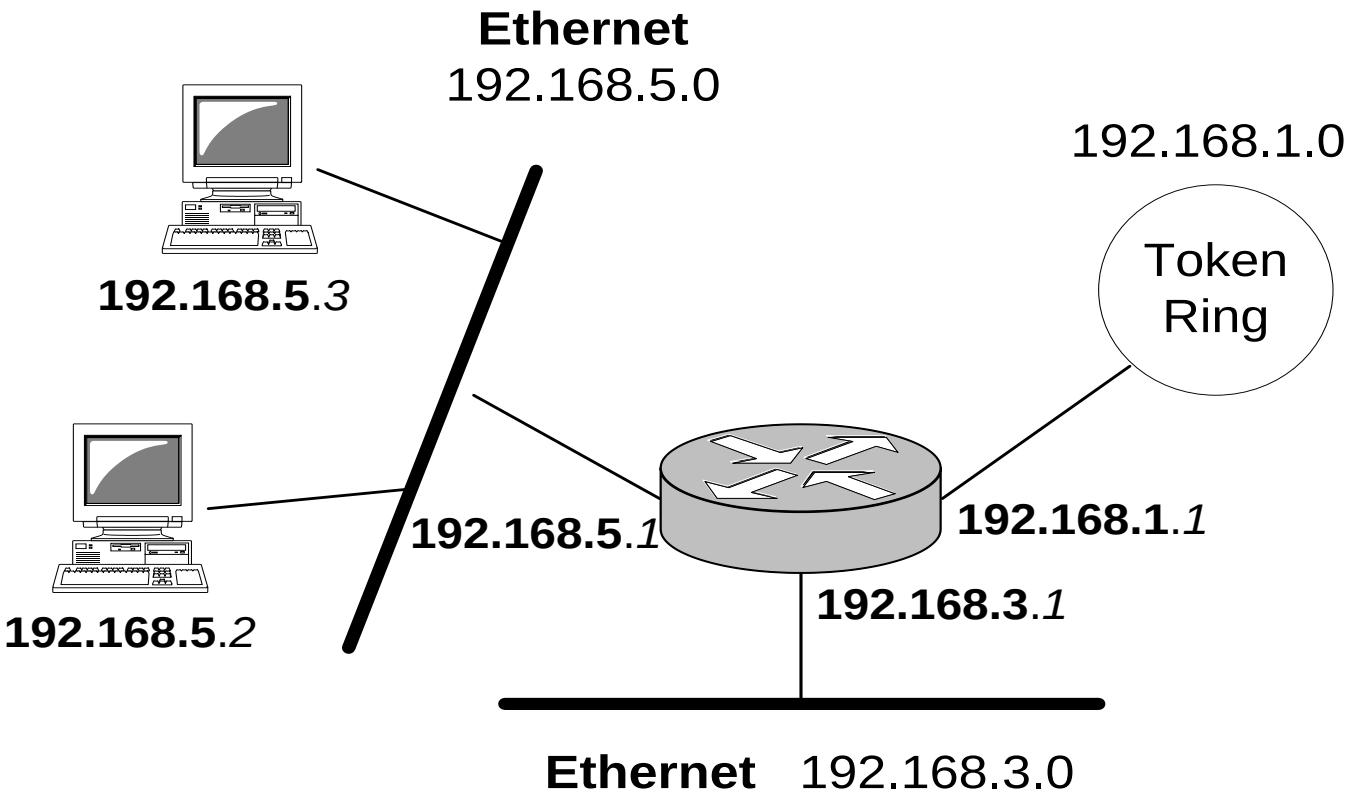
Lớp liên mạng: Giao thức IP

- Giao thức IP
  - IP được thiết kế nhằm mục đích sử dụng có hiệu quả tài nguyên mạng.
  - Giao thức này có hai thiếu hụt: thiếu điều khiển lỗi và thiếu các cơ chế hỗ trợ; IP cũng thiếu cơ chế truy vấn. Một trạm đôi khi cần xác định xem router hoặc một trạm khác có hoạt động không. Một người quản lý mạng đôi khi cần thông tin từ một trạm hoặc router khác.

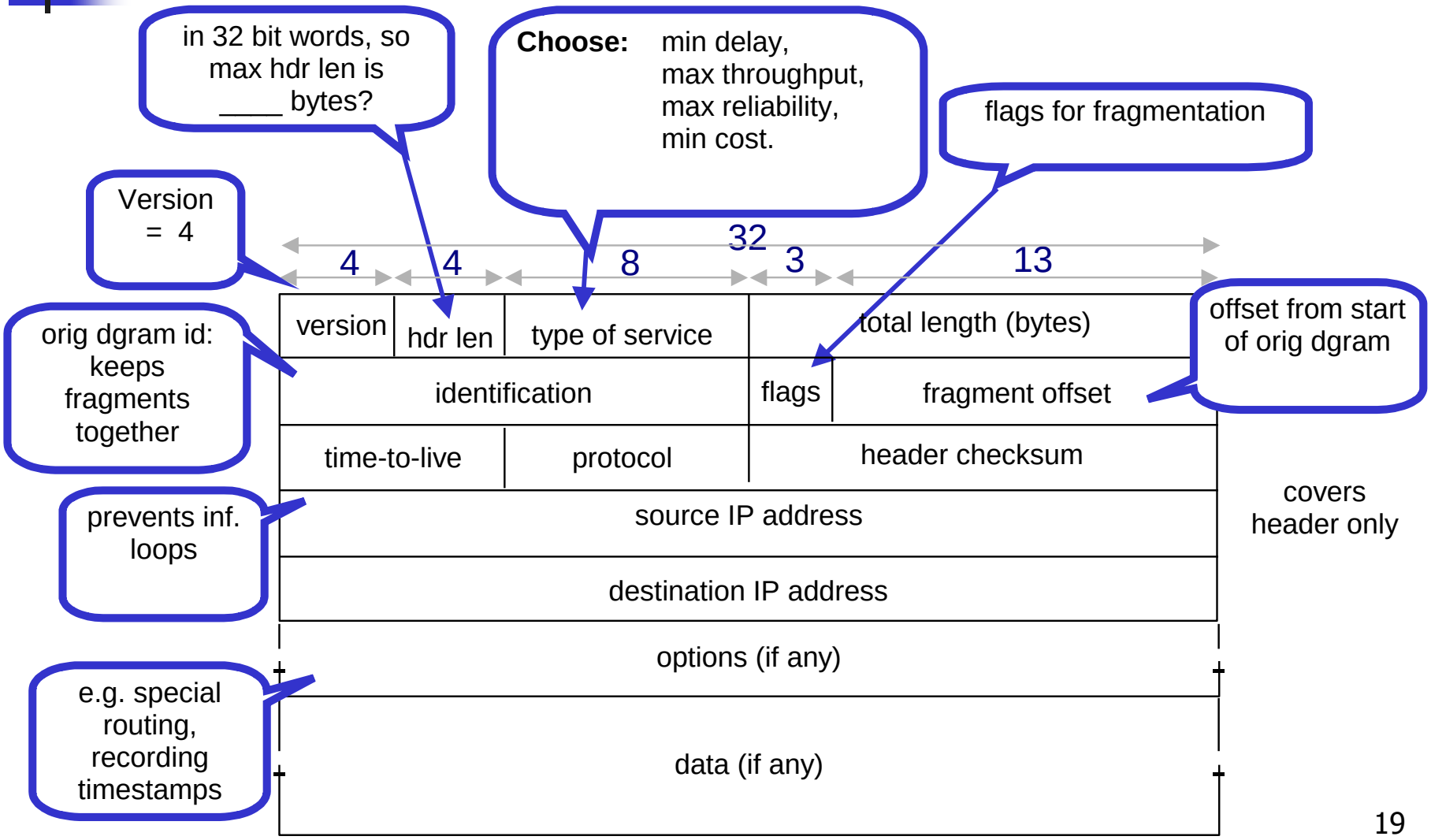
Giao thức IP


Data
destination: 203.162.0.11
source: 192.168.1.2
Data
= hosts/routers
on a network

M□ng	Tr□m
192.168.1	1
192.168.3	1
192.168.5	1 2 3



Lớp liên mạng - Cấu trúc gói tin IP



Lớp liên mạng - Cấu trúc gói tin IPv4

- Phiên bản (Version): Chỉ phiên bản của giao thức IP đã dùng để tạo datagram.
- Độ dài mào đầu IP (IP Header Length): Cung cấp thông tin về độ dài mào đầu của datagram, được tính theo các từ 32 bit.
- Loại dịch vụ (Type of Service-Service type): xác định độ ưu tiên của datagram và xác định kiểu giao thông của datagram. *delay, max throughput, max reliability, min cost.*
- Tổng độ dài (Total Length): Gồm tổng số octet của phần tiêu đề và dữ liệu.
- Nhận dạng (Identification): Giá trị ID datagram ban đầu
- Cờ (Flag): Đánh dấu phân mảnh.

Lớp liên mạng - Cấu trúc gói tin IPv4

- Fragment offset: Độ lệch từ điểm khởi đầu của datagram.
- Time to live: Thời gian sống.
- Protocol: Giao thức
- Header checksum: tổng kiểm tra (chỉ gồm phần header). Tính toán tổng kiểm tra
- **Địa chỉ IP nguồn và đích** để xác định nguồn và đích đến của gói IP, có 32 bit.
- Tùy chọn (Option-tối đa 40 byte): Khai báo các tùy chọn do bên gửi yêu cầu nếu có .
- Dữ liệu (Data): Nếu có (dữ liệu của lớp trên …)..

Lớp liên mạng - Địa chỉ IP

- Mỗi địa chỉ IP gồm 4 byte (32 bít), định nghĩa hai phần:
  - địa chỉ mạng (NetID)
  - địa chỉ trạm (HostID)
- Các phần này có chiều dài khác nhau tuỳ thuộc vào lớp địa chỉ.
- Các bít đầu tiên trong phần địa chỉ mạng xác định lớp của địa chỉ IP.

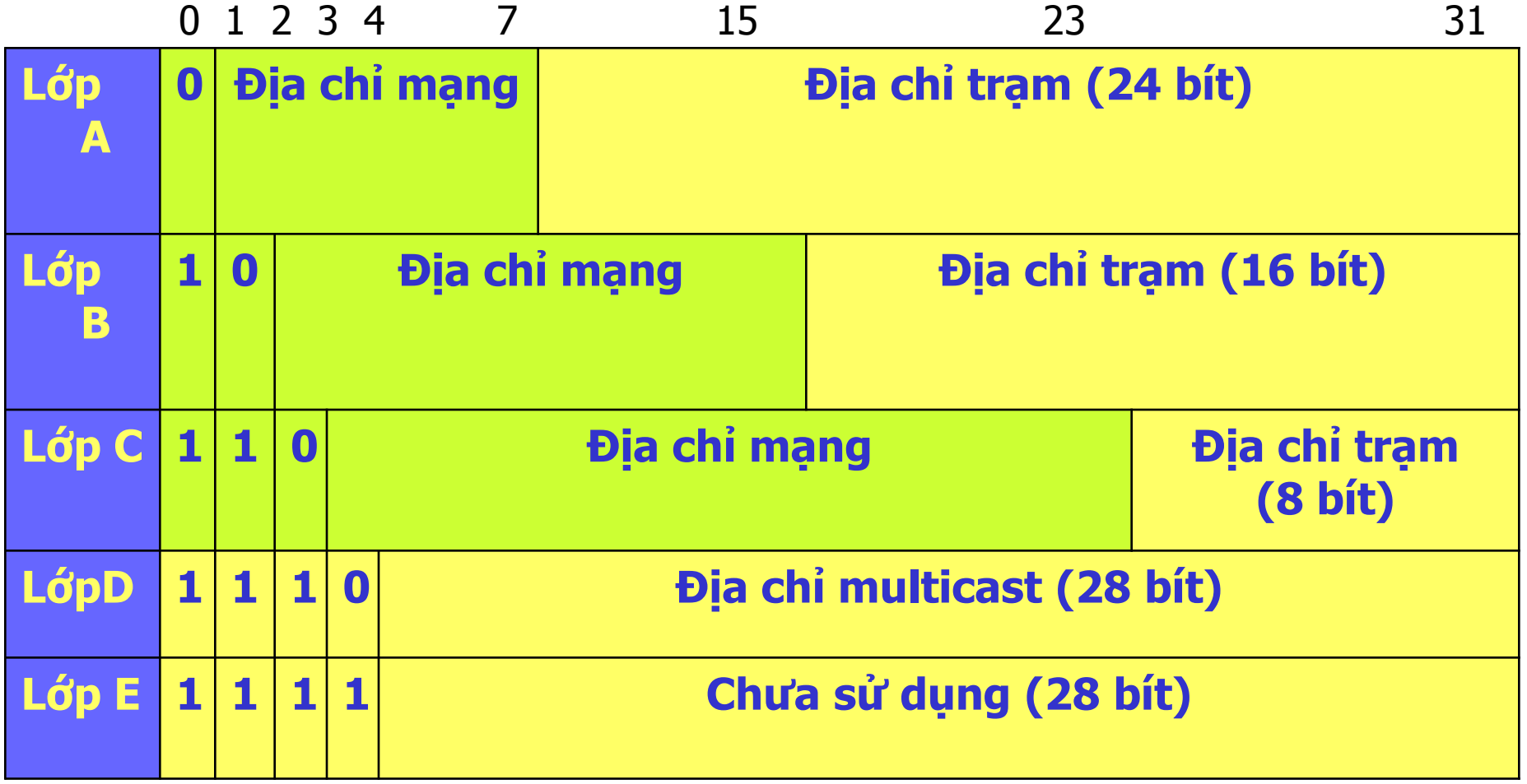
Lớp liên mạng - Địa chỉ IP

- Sự cần thiết có địa chỉ IP:
  - Ở mức ứng dụng, có thể coi liên mạng là một mạng đơn lẻ kết nối các trạm với nhau.
  - Để một trạm truyền thông với trạm khác, cần có một hệ thống định danh toàn cầu (đặt tên duy nhất cho mỗi trạm) → không thể sử dụng ở tầng mạng vì trên mạng còn có các thực thể khác như router

Lớp liên mạng - Địa chỉ IP

- Sự cần thiết có địa chỉ IP(2):
  - Một liên mạng được tạo nên từ sự kết hợp của các mạng vật lý (LAN hoặc WAN) thông qua các router
  - Khi hai trạm truyền thông với nhau, gói dữ liệu qua các mạng vật lý khác nhau bằng cách sử dụng các router này → việc truyền thông tại mức này cũng cần có một hệ thống định danh toàn cục

Lớp liên mạng - Địa chỉ IPv4

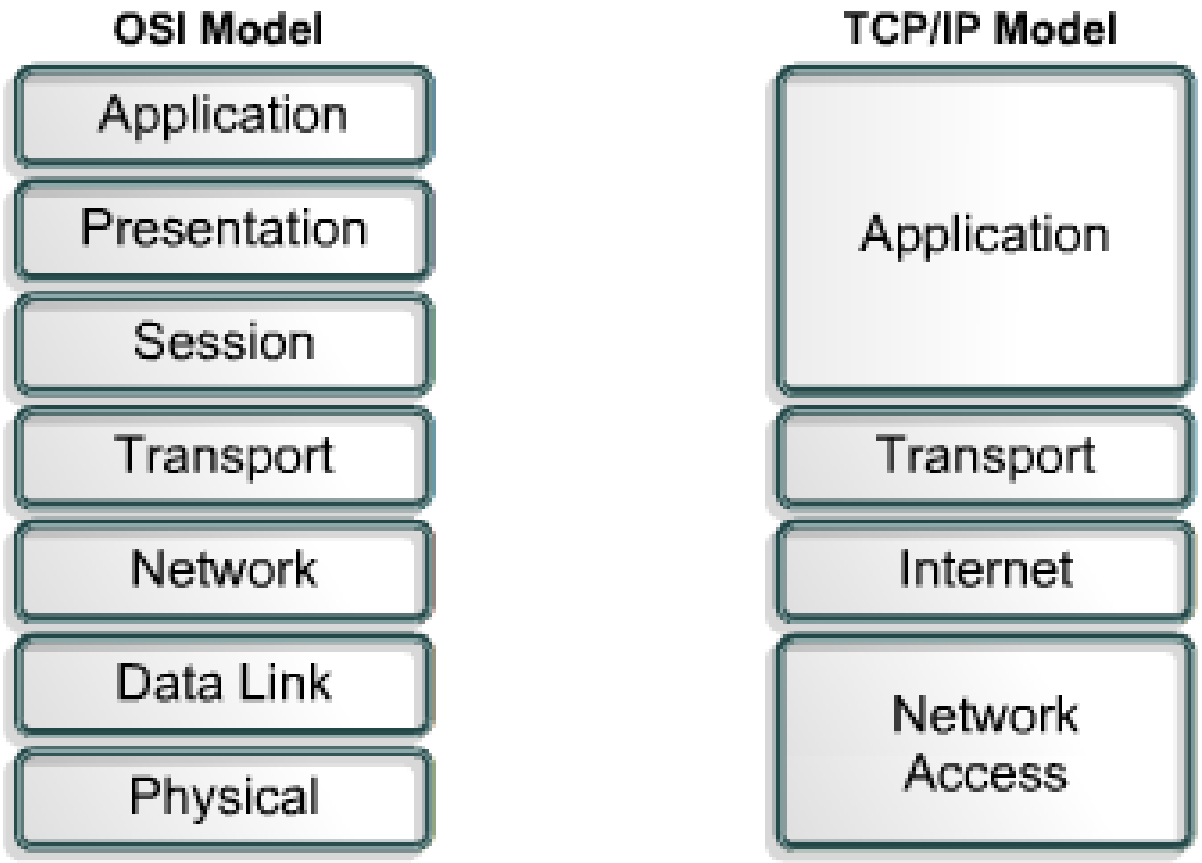


Bộ giao thức TCP/IP và Địa chỉ IP

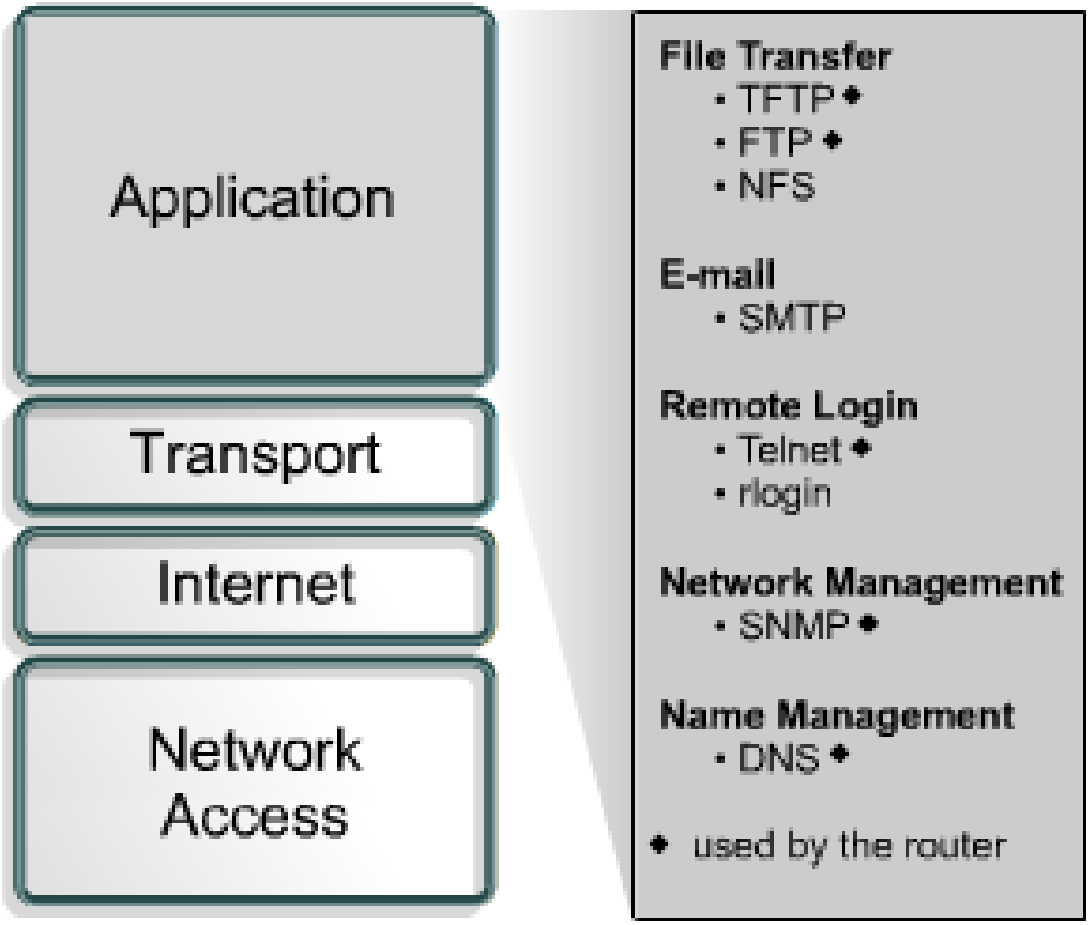
Nội dung

- Giới thiệu TCP/IP
- Địa chỉ IP
- Giới thiệu Subnetting

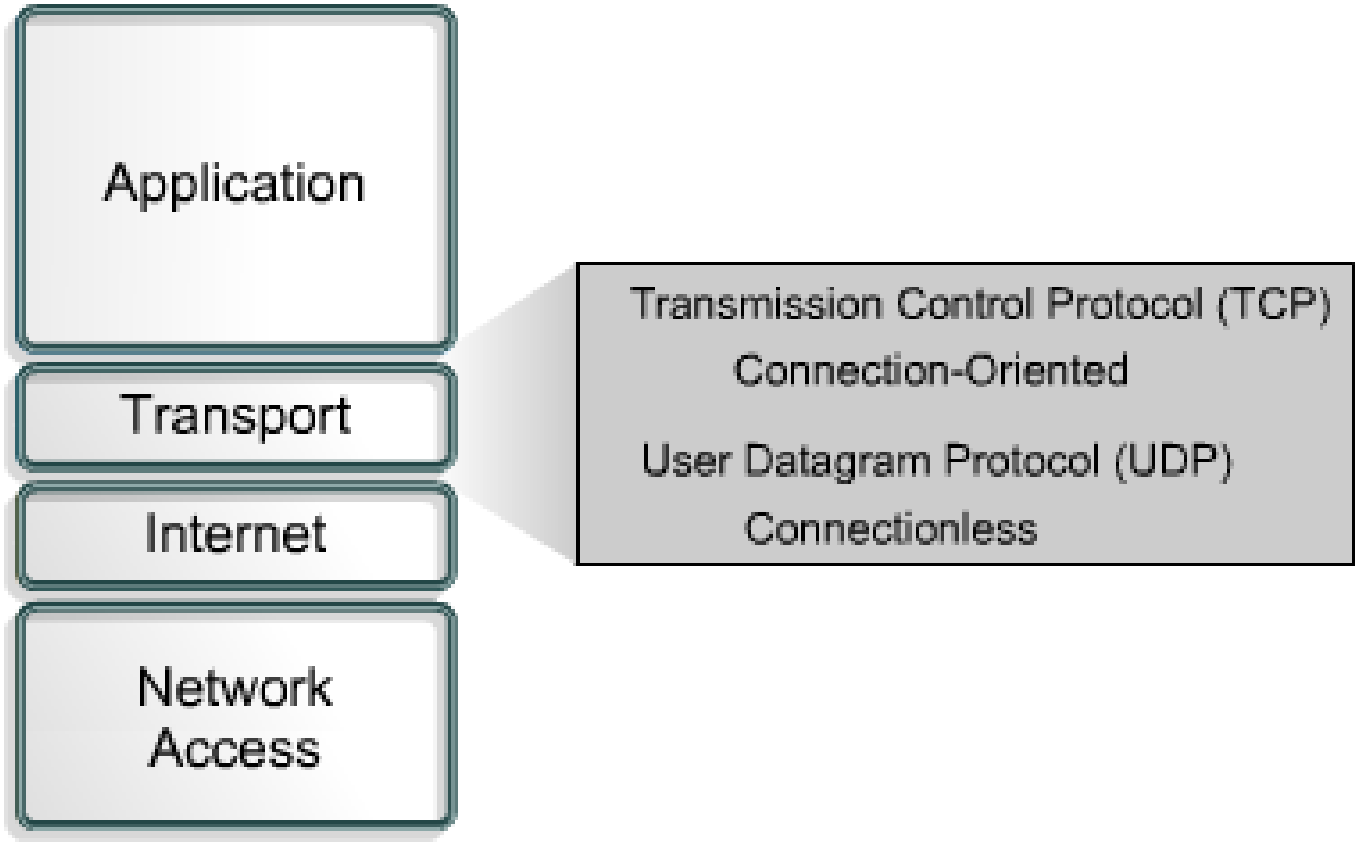
Mô hình TCP/IP



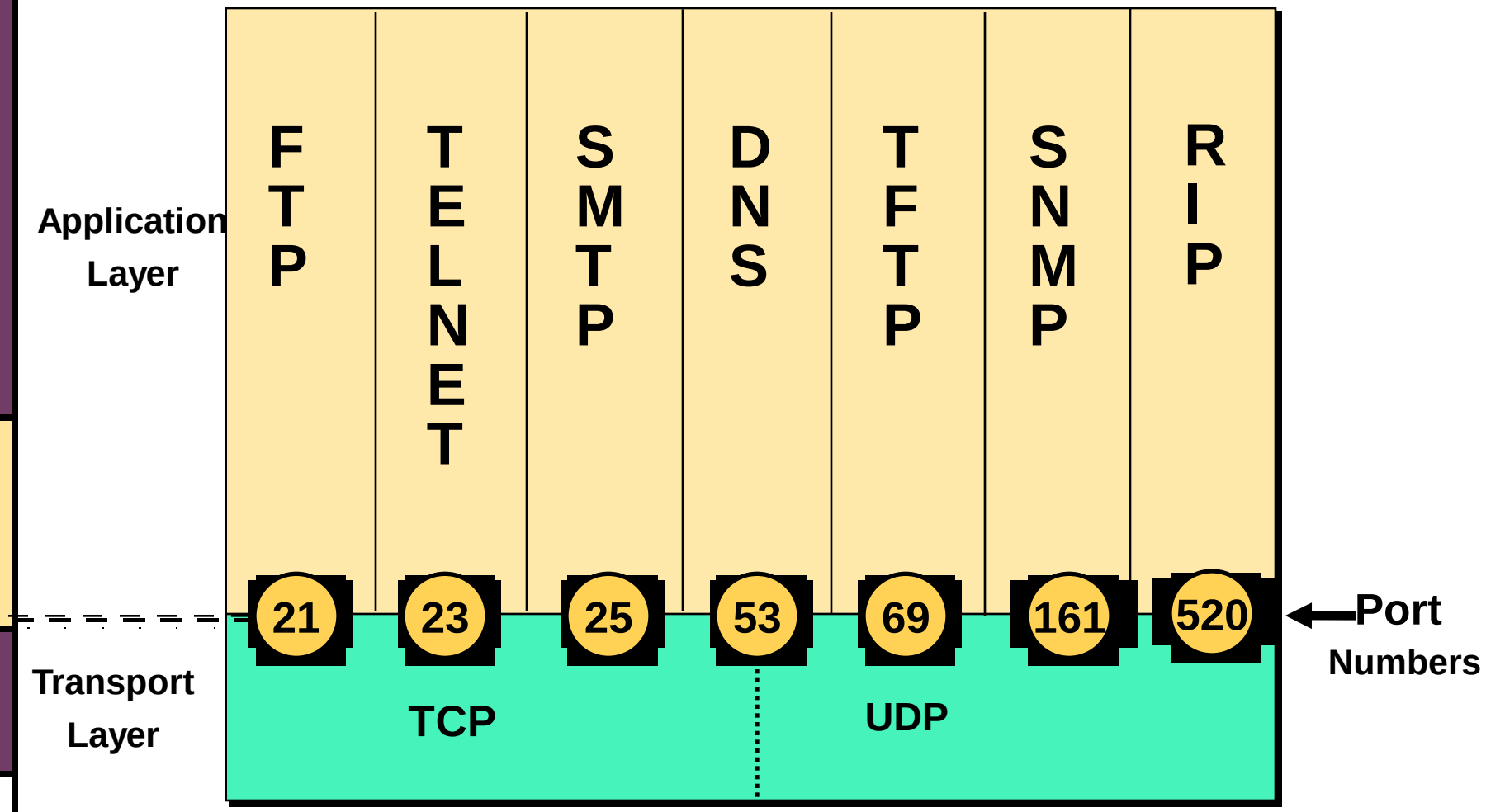
Ứng dụng TCP/IP



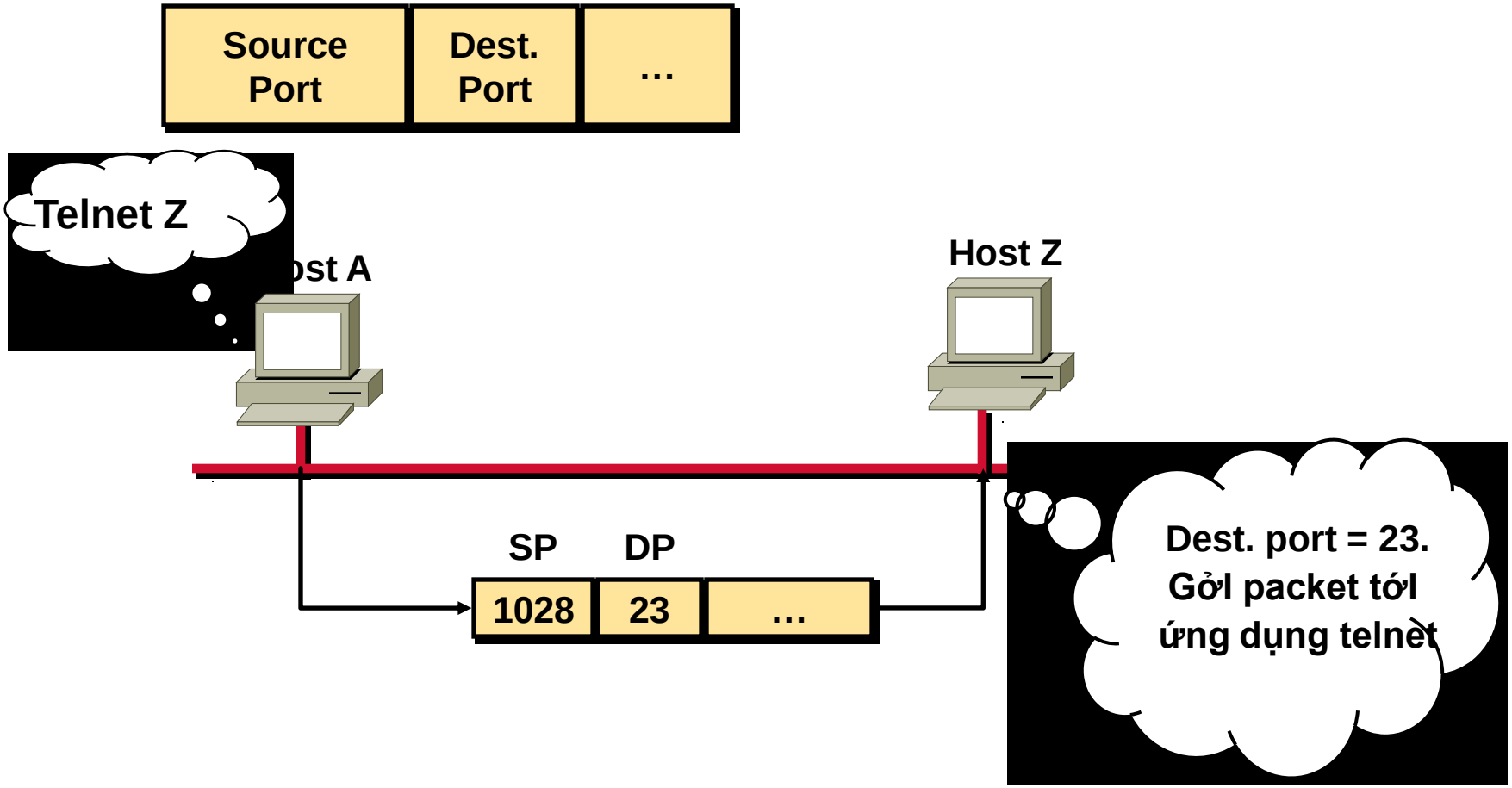
Giao thức ở tầng Transport



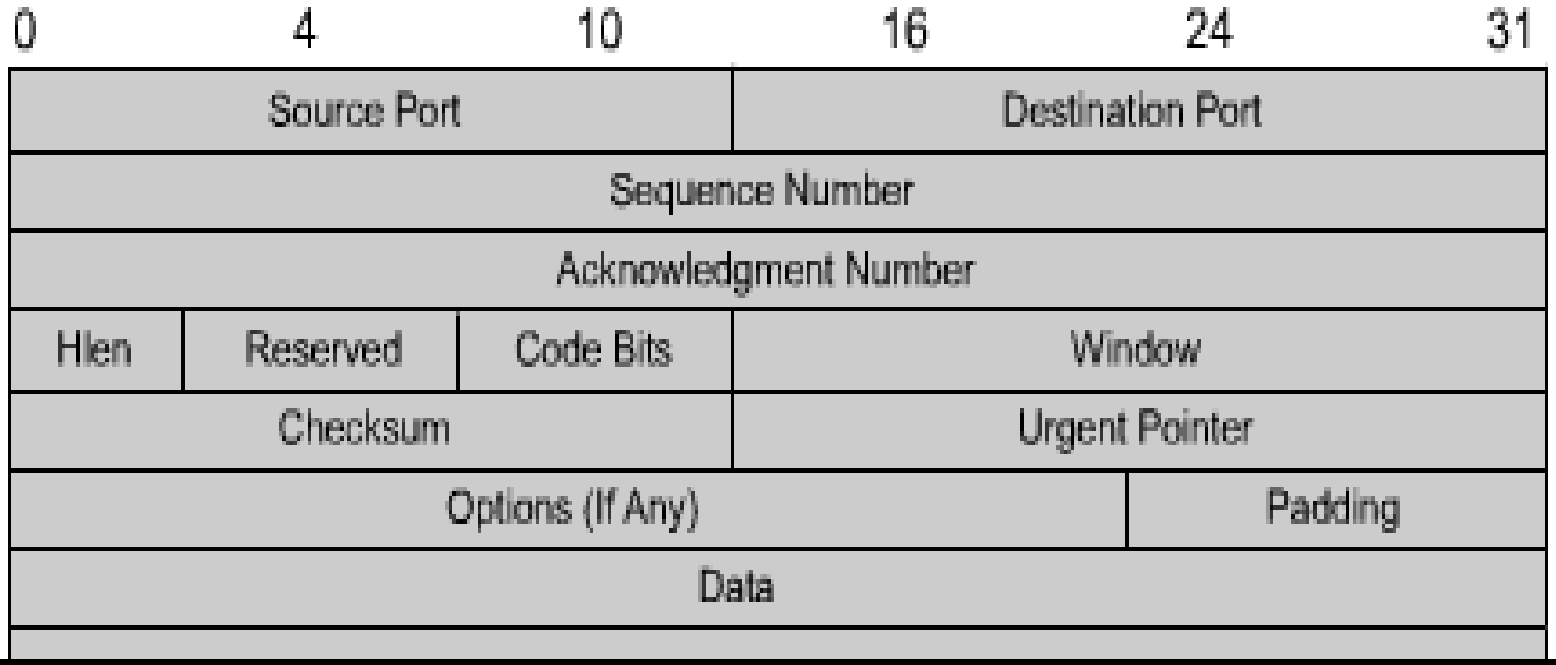
TCP,UDP và Port ứng dụng



TCP Port



Thông tin của TCP Header



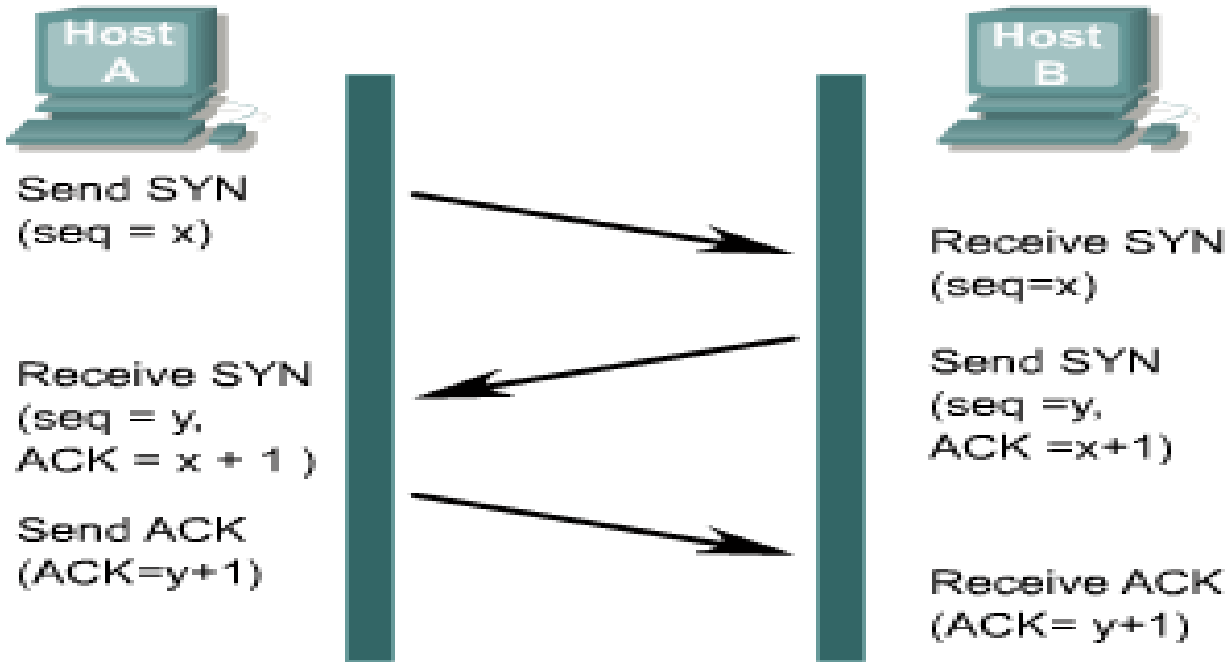
Đồng bộ hoặc ba bước bắt tay

TCP là một giao thức tin cậy (connection oriented), trước khi truyền dữ liệu hai host phải xử lý đồng bộ để tạo ra một kết nối ảo. Tiến trình này được gọi là ba bước bắt tay (Three way handshake)

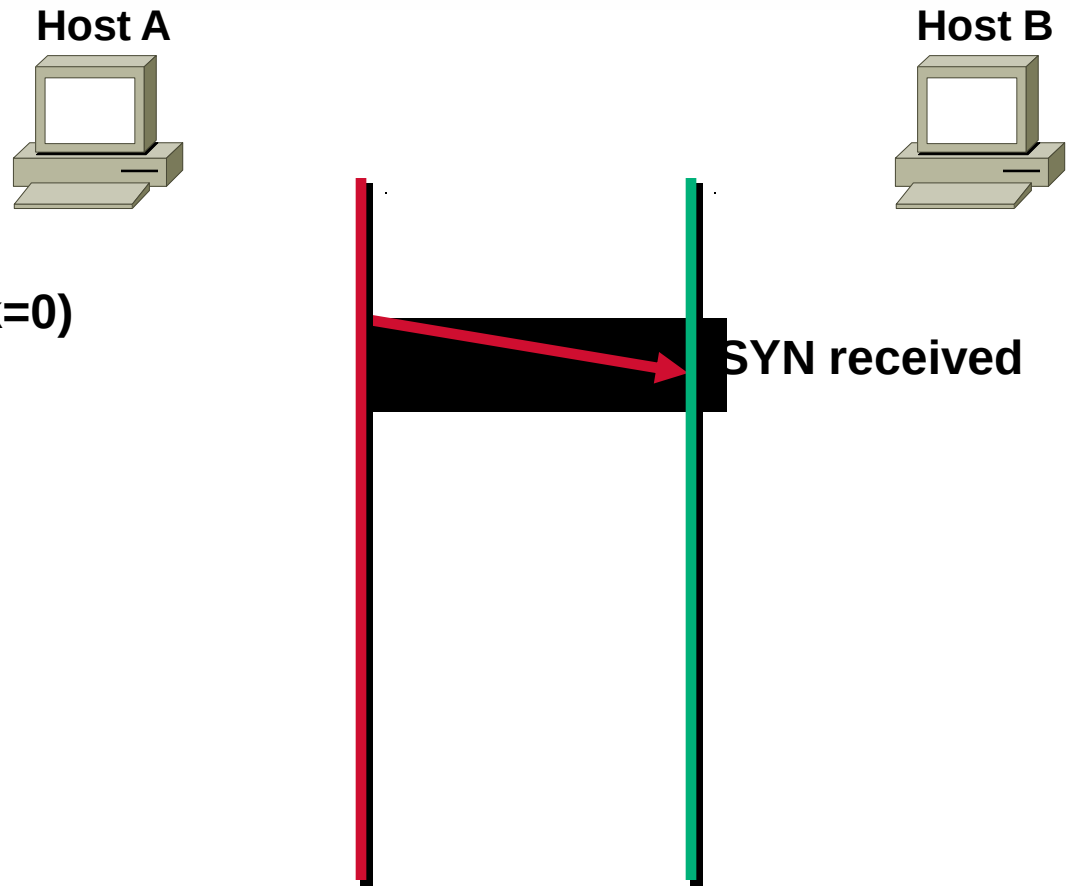
Đầu tiên host khởi tạo connection gởi packet syn (đồng bộ), thông tin này mô tả trong sequen number để báo cho bên nhận biết có nhu cầu kết nối

Khi bên nhận được packet nó gởi lại packet xác nhận là đã nhận được packet thông tin này được lưu trong Acknowledgment number (ACK) và số sequence number yêu cầu bên gởi gởi

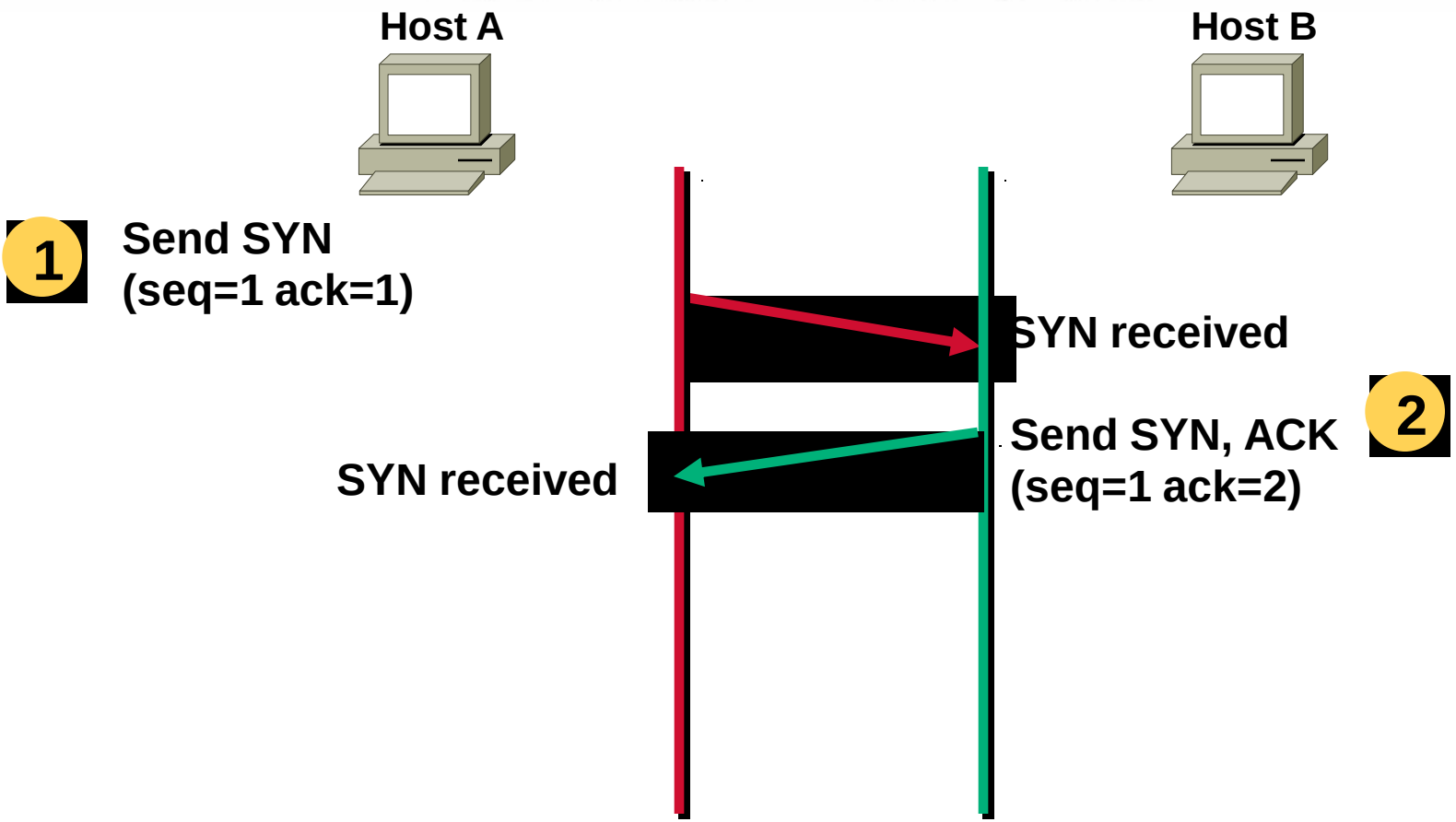
Sau khi host có nhu cầu kết nối nhận được thông tin trả lời nó sẽ gởi lại một packet ACK. Sau đó 2 bên thực hiện truyền nhận



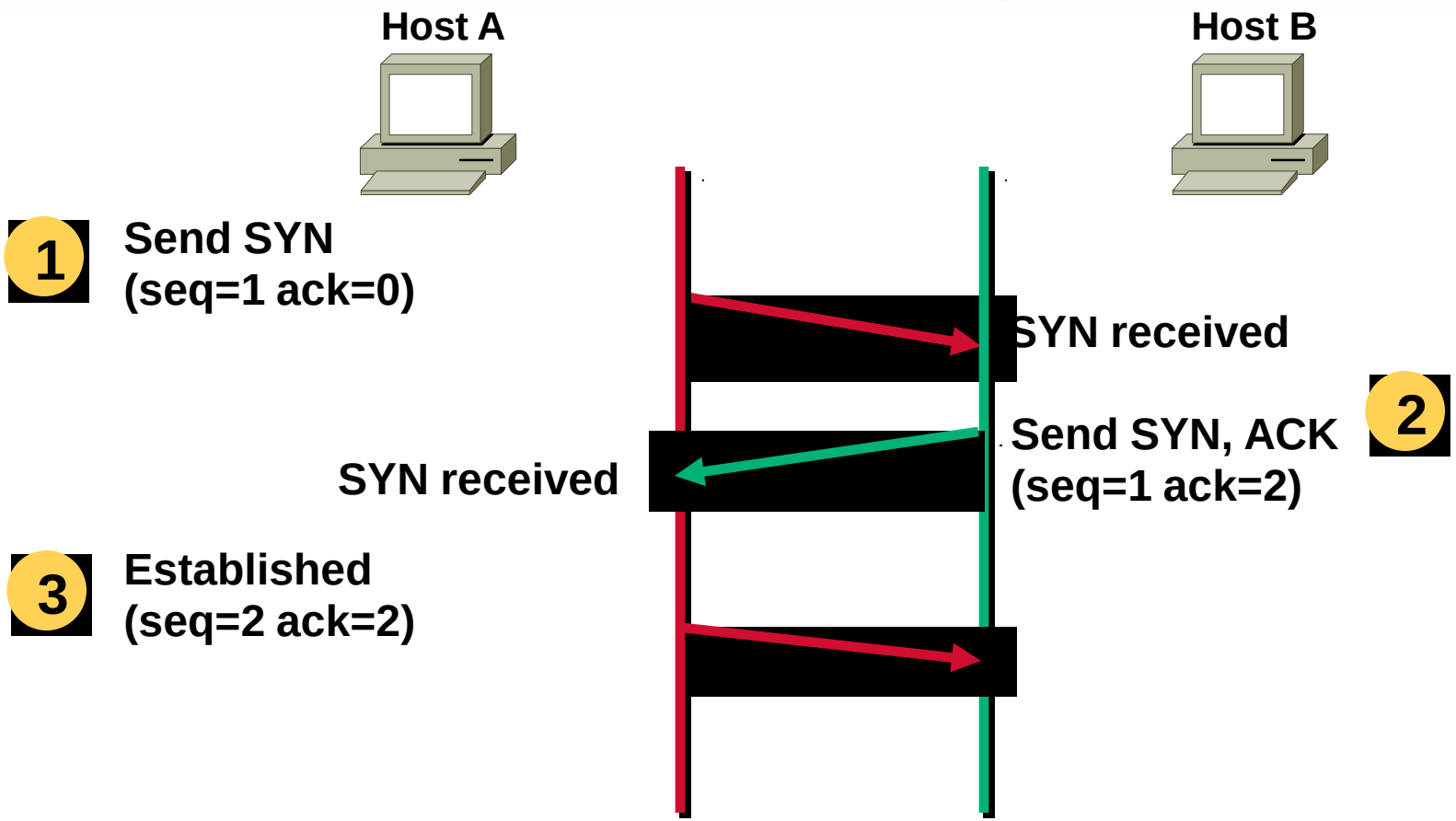
Ba bước bắt tay của TCP để mở một kết nối



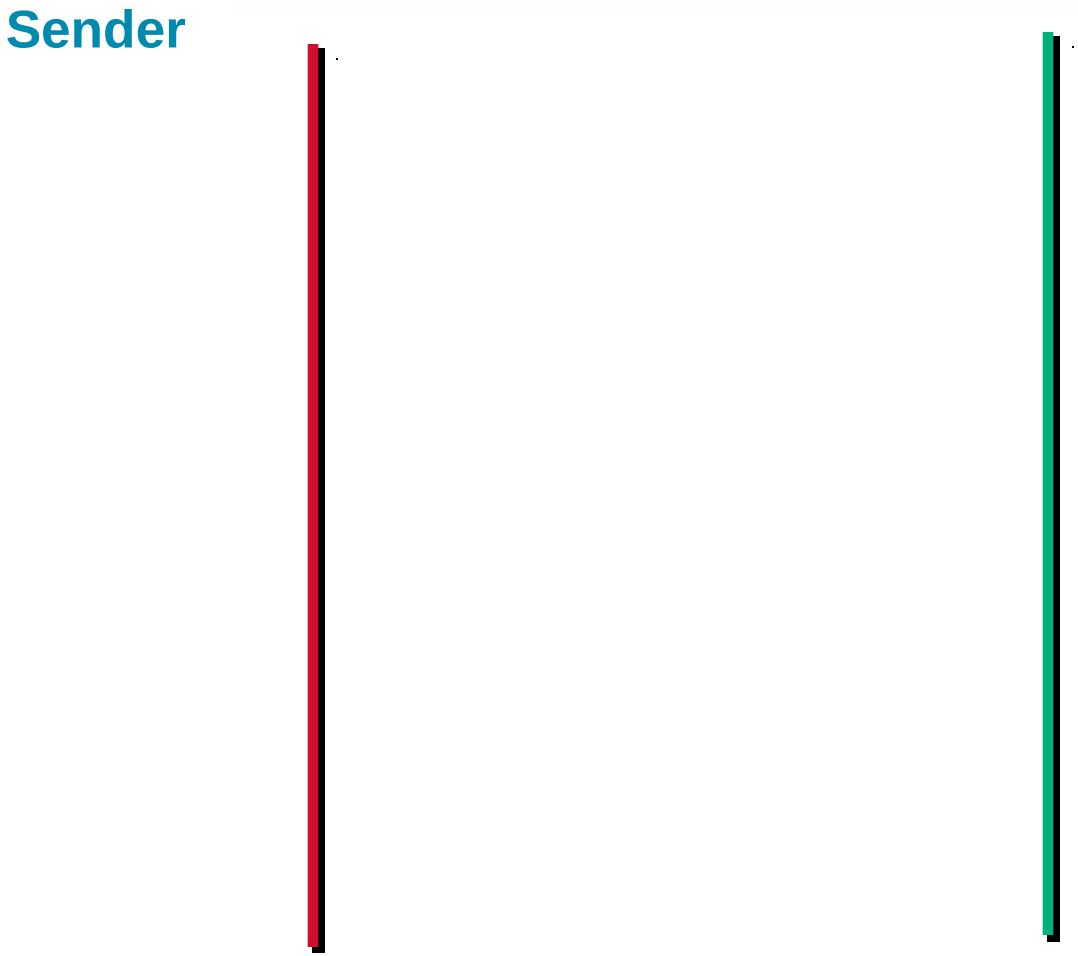
Ba bước bắt tay của TCP để mở một kết nối



Ba bước bắt tay của TCP để mở một kết nối

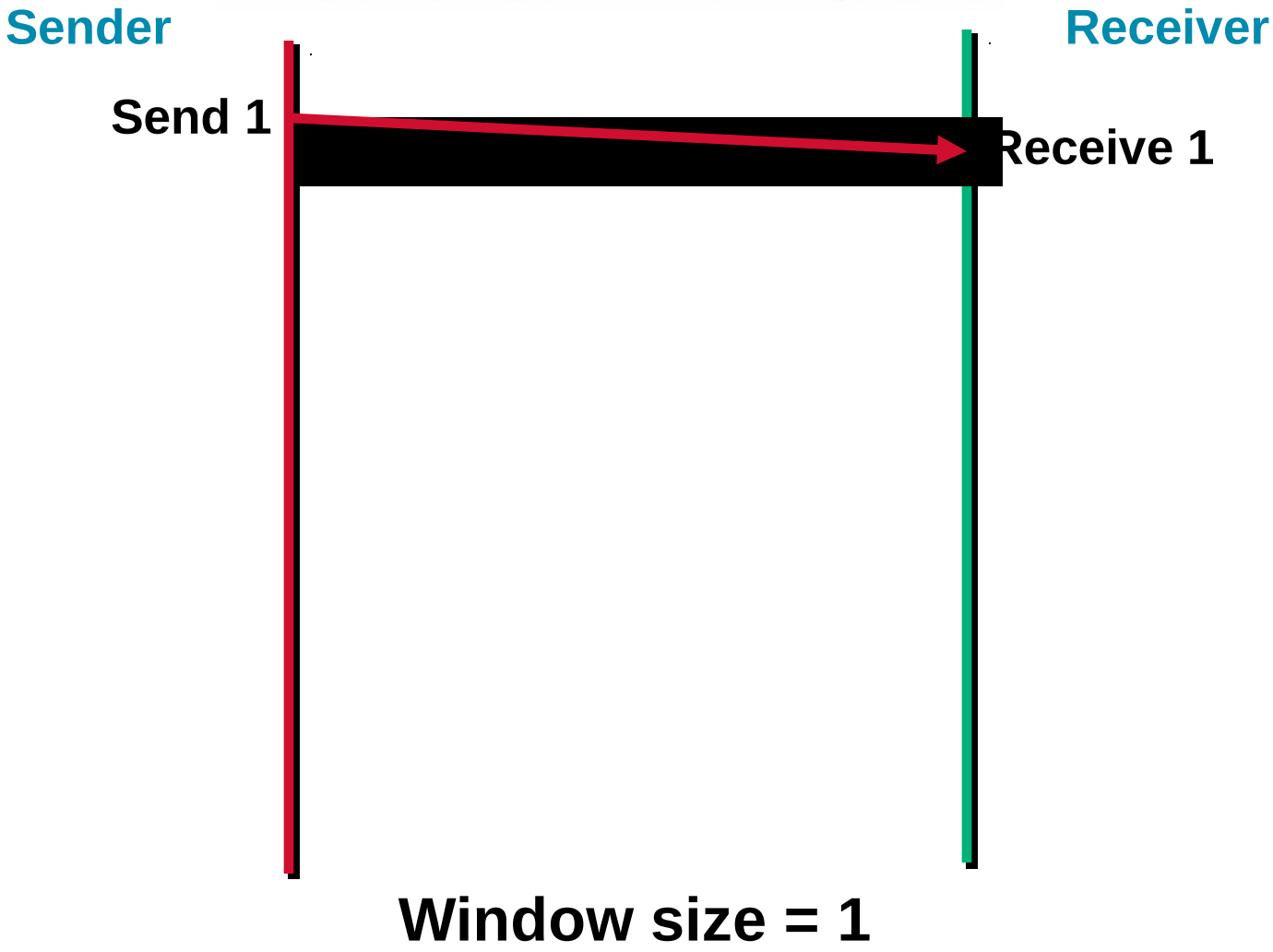


TCP Simple Acknowledgment

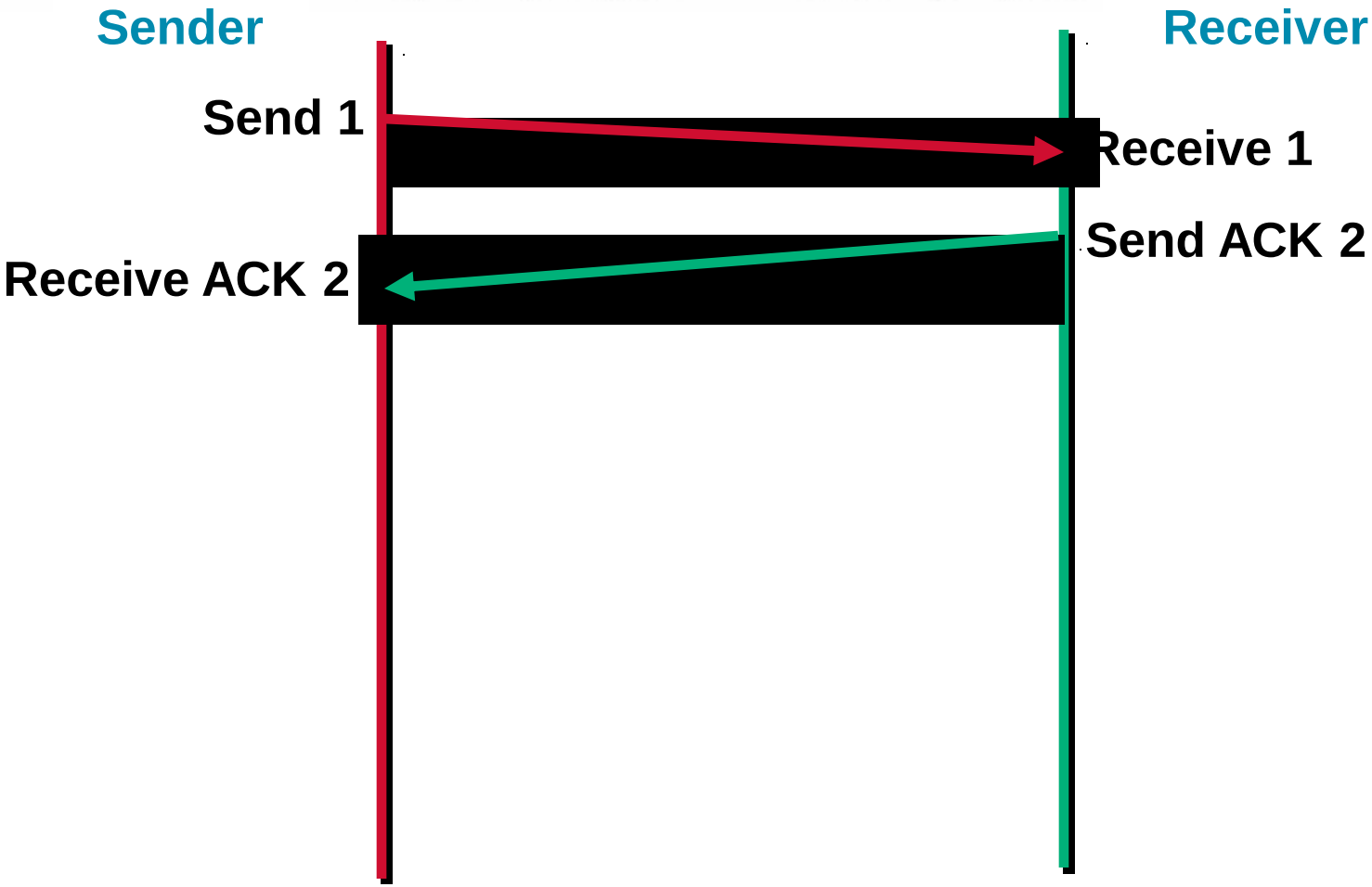


Window size = 1

TCP Simple Acknowledgment

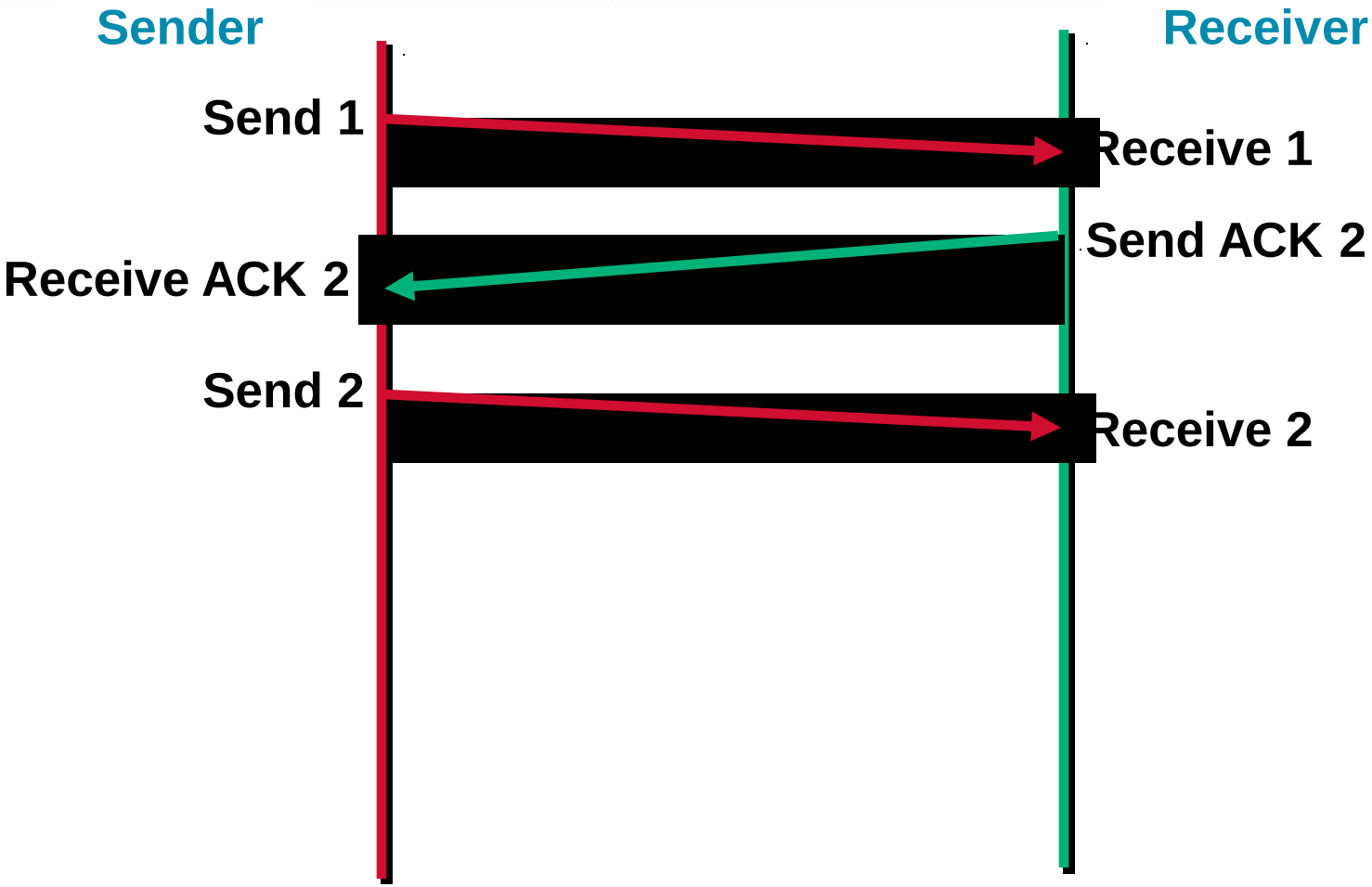


TCP Simple Acknowledgment



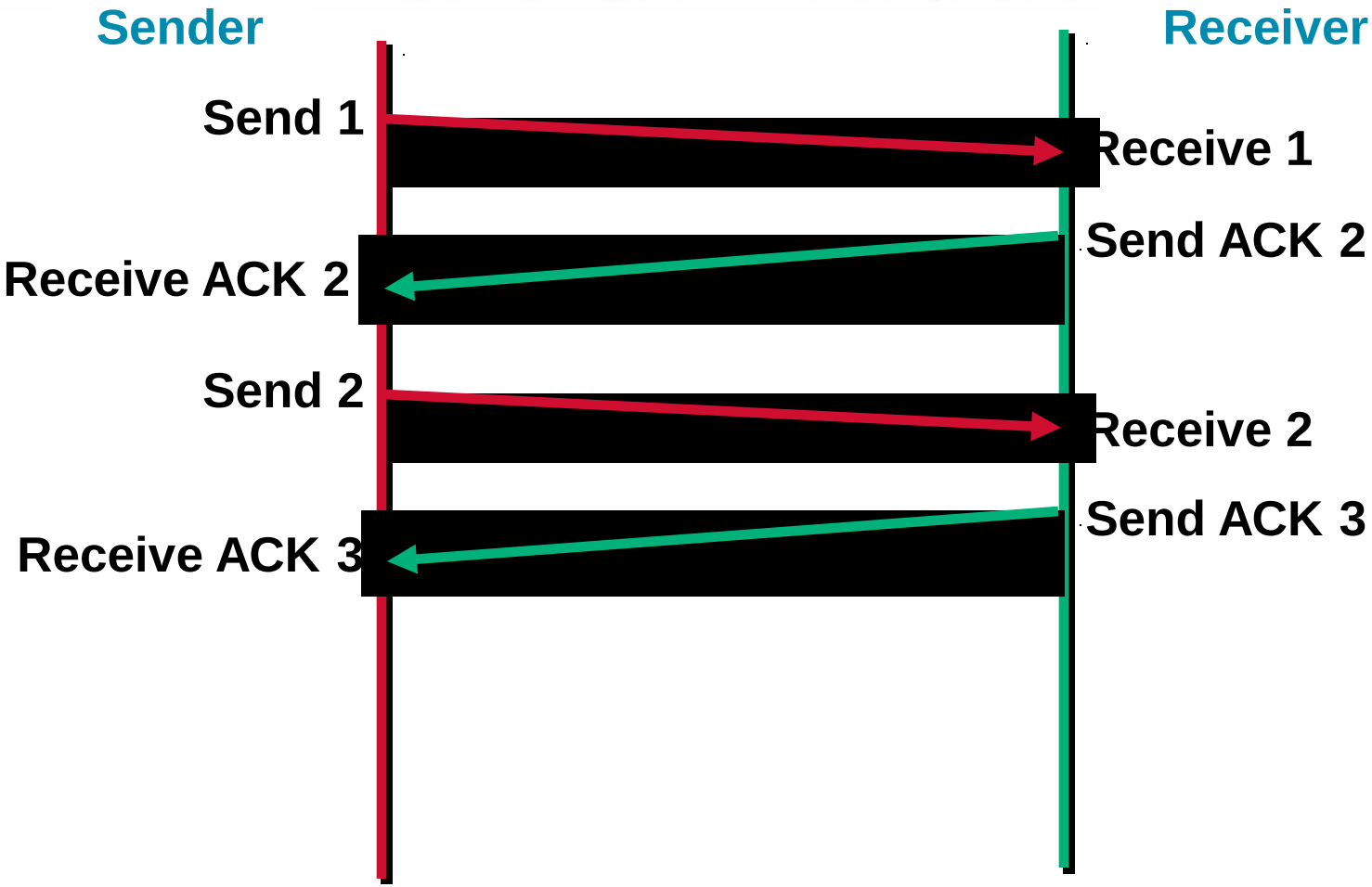
Window size = 1

TCP Simple Acknowledgment



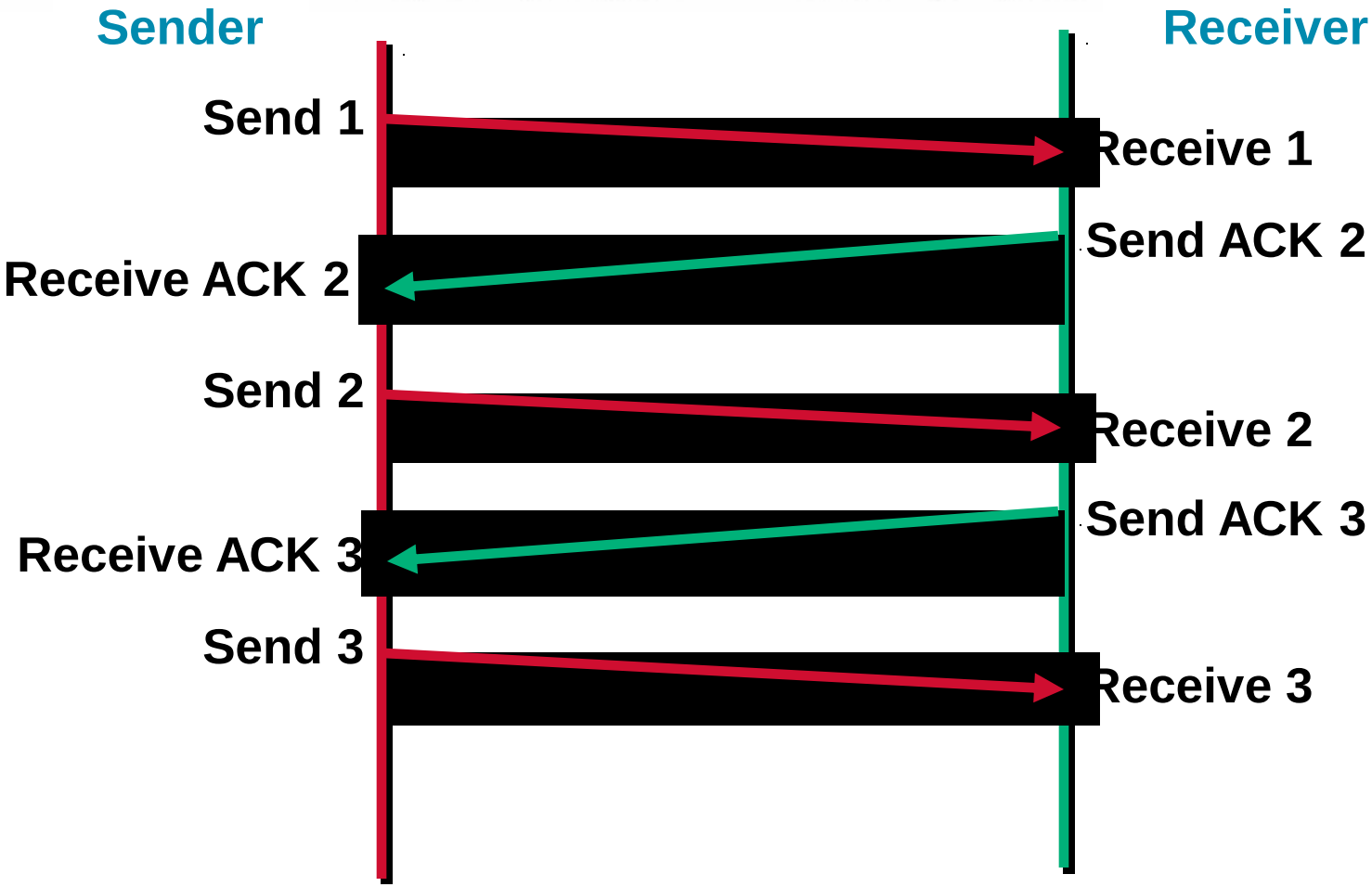
Window size = 1

TCP Simple Acknowledgment



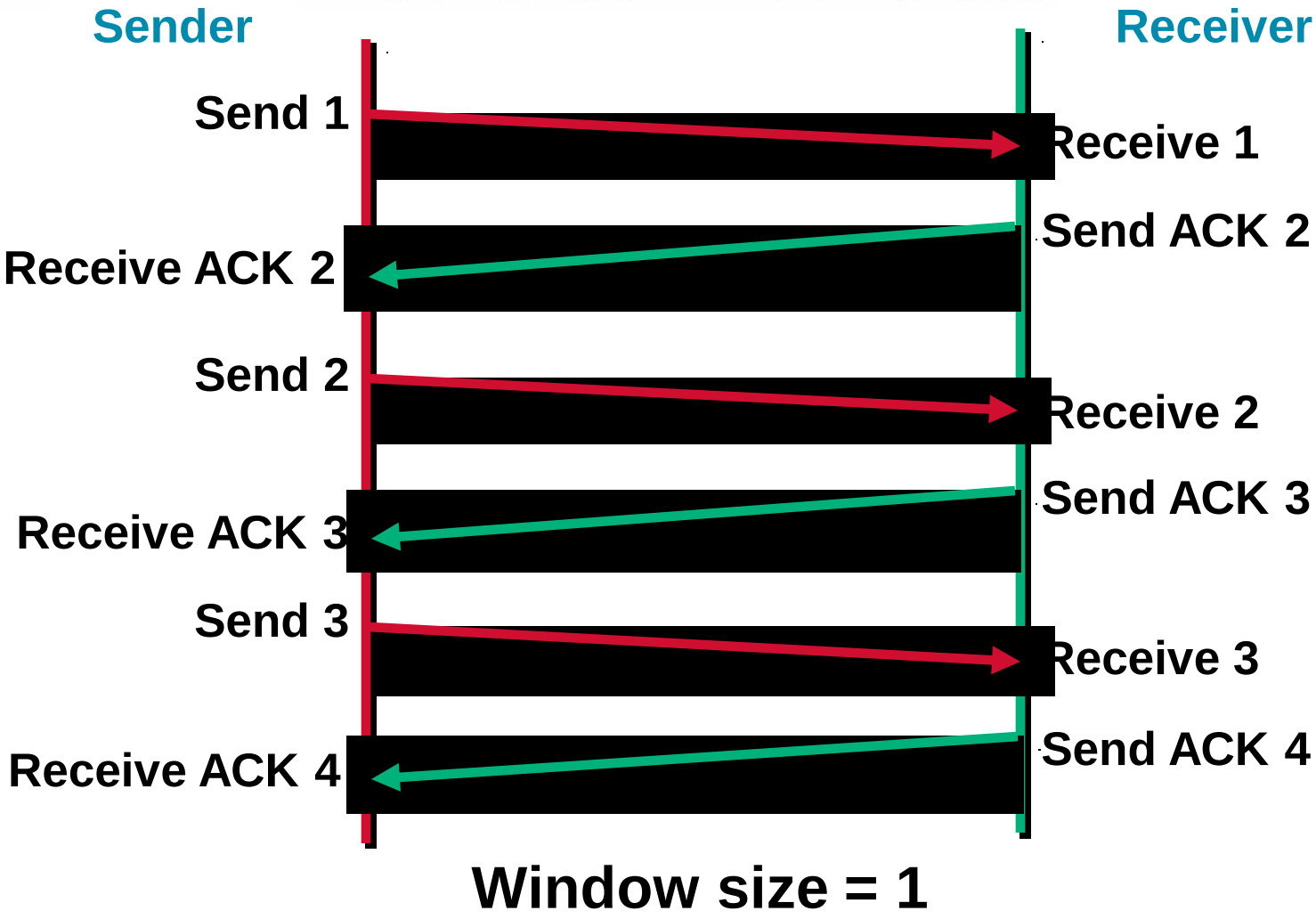
Window size = 1

TCP Simple Acknowledgment



Window size = 1

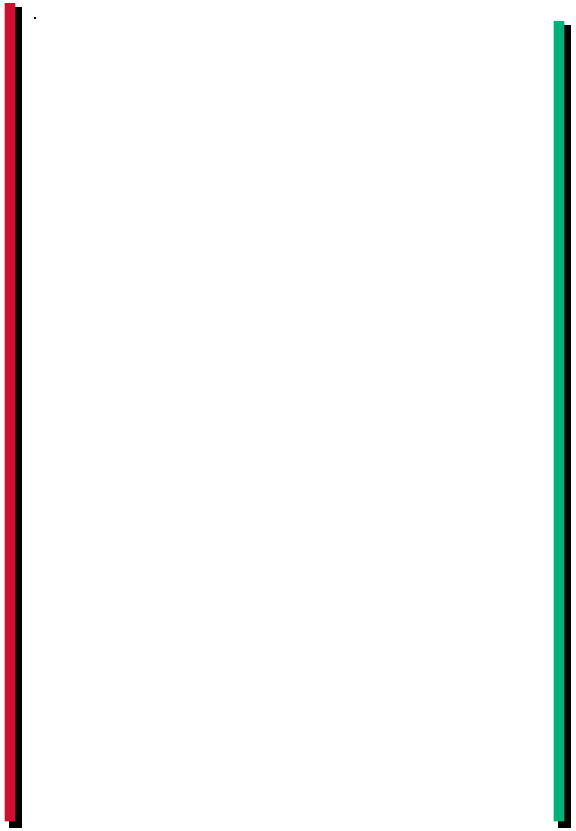
TCP Simple Acknowledgment



TCP Windowing

Sender

Receiver



Window size = 3

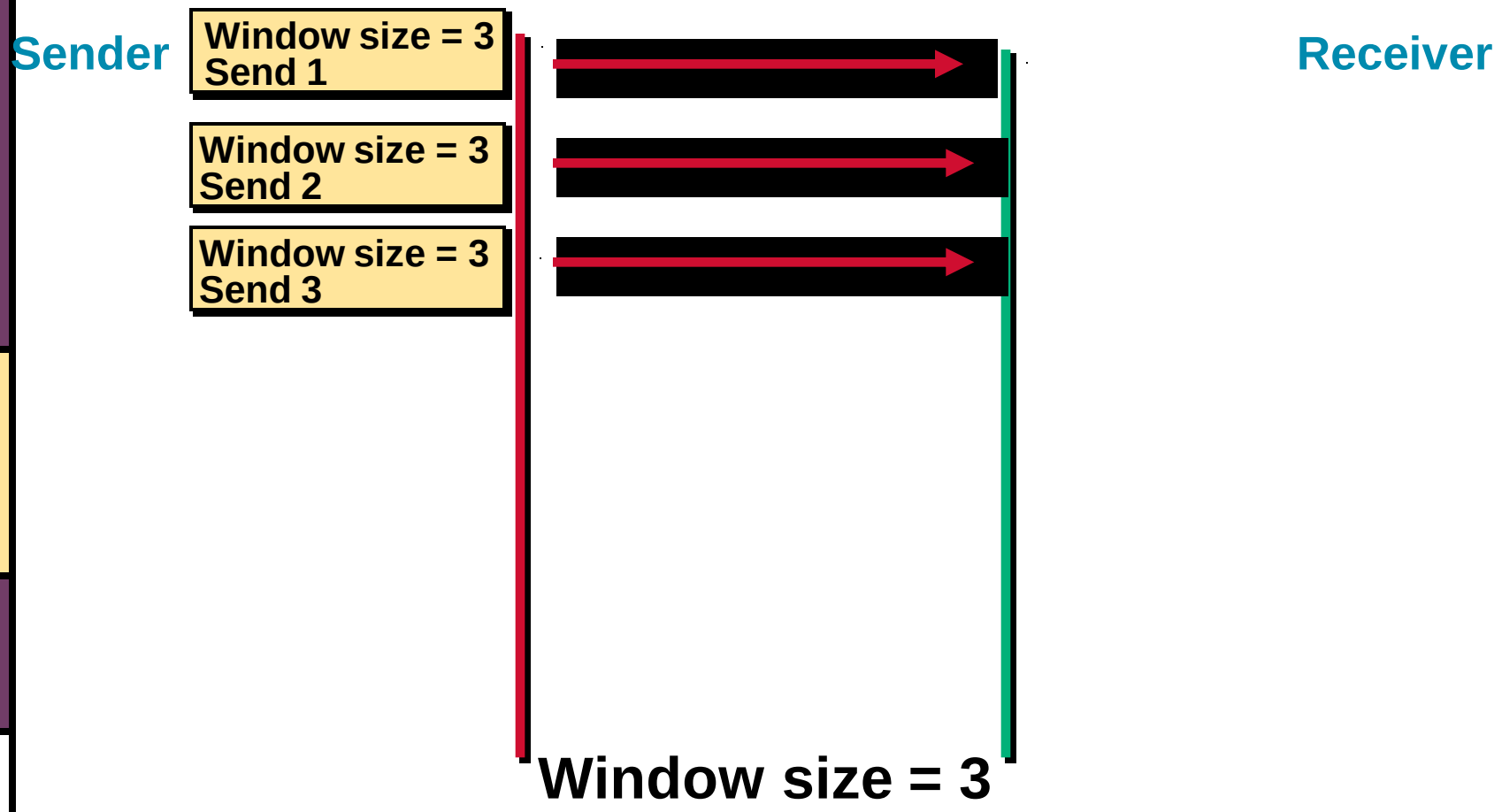
Windows và windows size

Tổng dữ liệu cần truyền quá lớn được gởi trong segment. Trong trường hợp này dữ liệu phải được chia nhỏ thành những mảnh nhỏ để truyền dữ liệu hiệu quả hơn. TCP chịu trách nhiệm thực hiện nhiệm vụ này

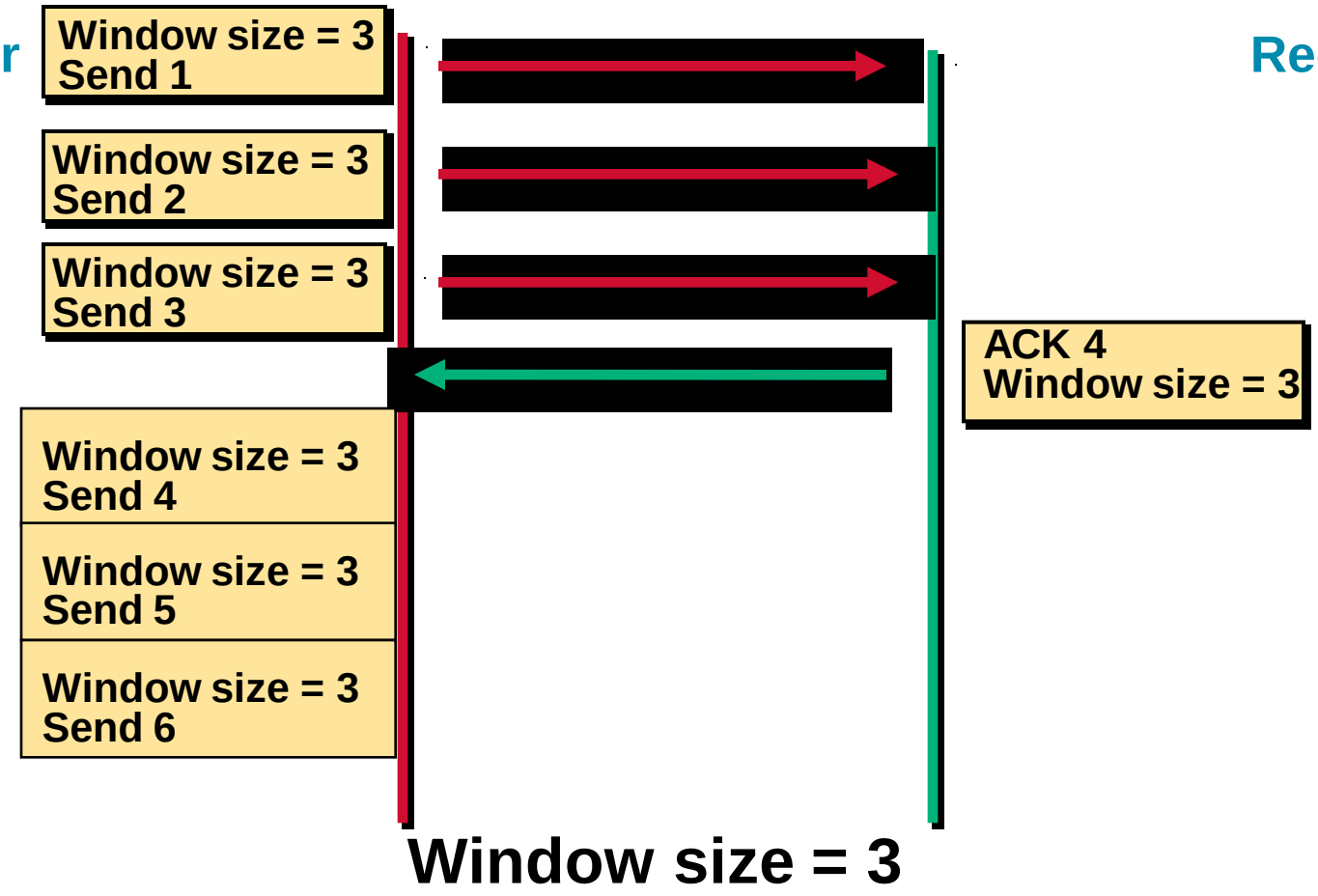
Khi truyền và nhận dữ liệu, một dịch vụ được cung cấp bởi TCP là điều khiển dòng (flow control). Có nghĩa bao nhiêu dữ liệu được truyền trong một khỏang thời gian, tiến trình điều khiển được biết là windowing

Window size xác định tổng dữ liệu có thể truyền tại một thời điểm trước khi nhận ACK từ destination

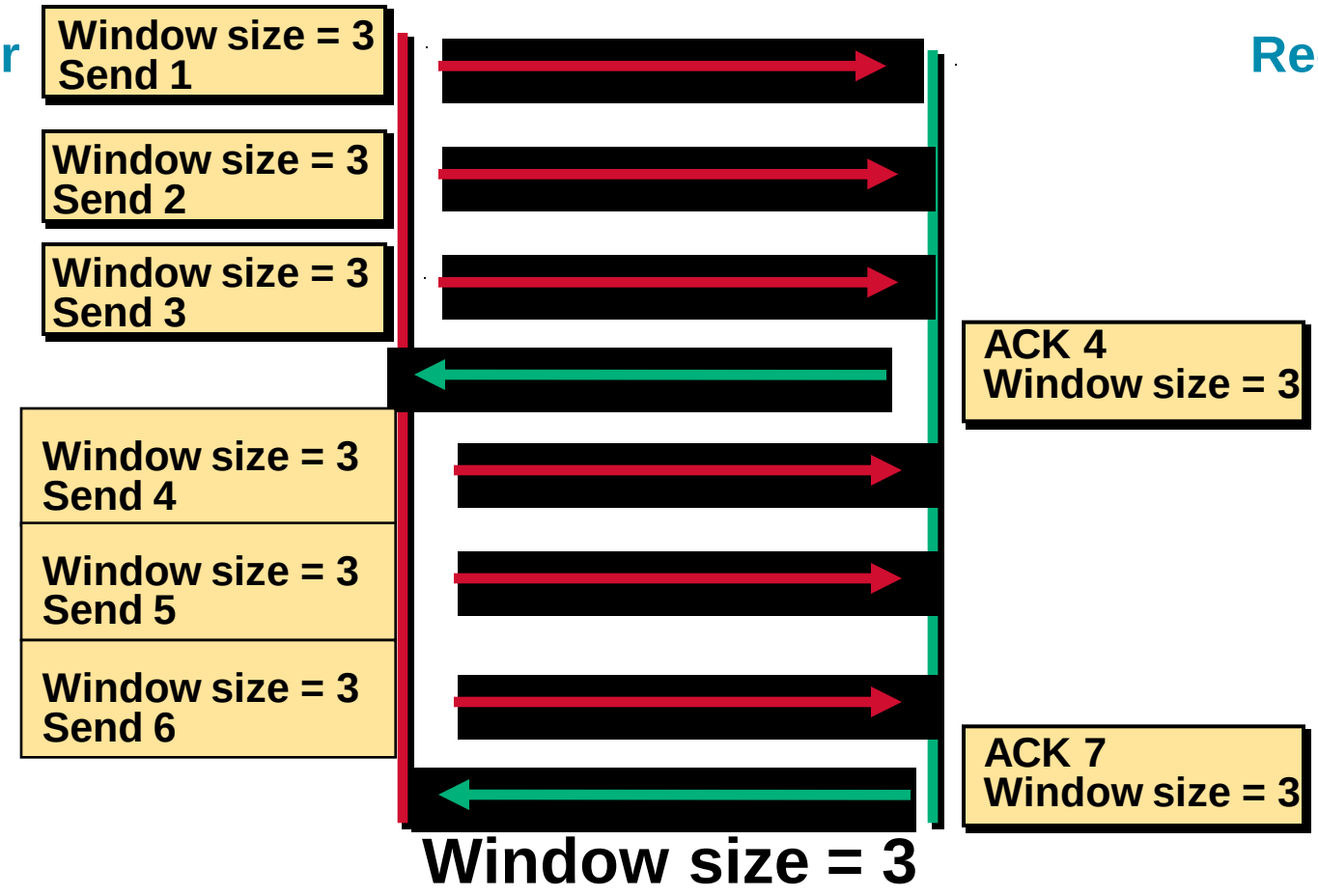
TCP Windowing



TCP Windowing



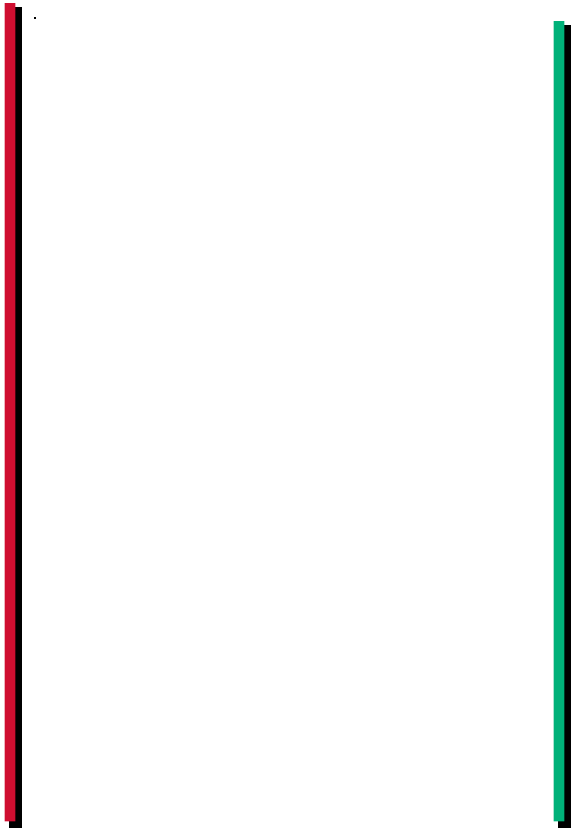
TCP Windowing



TCP Windowing

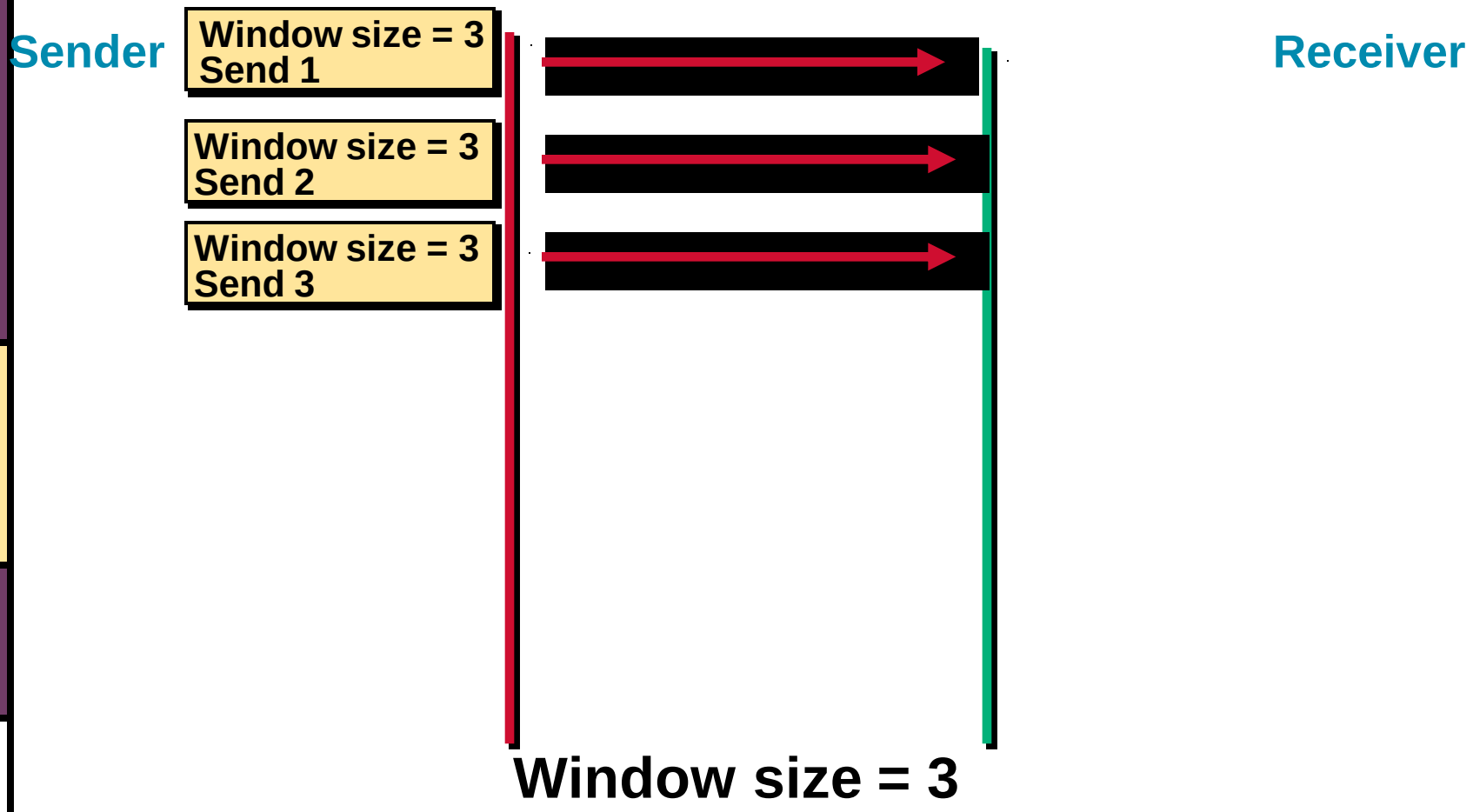
Sender

Receiver

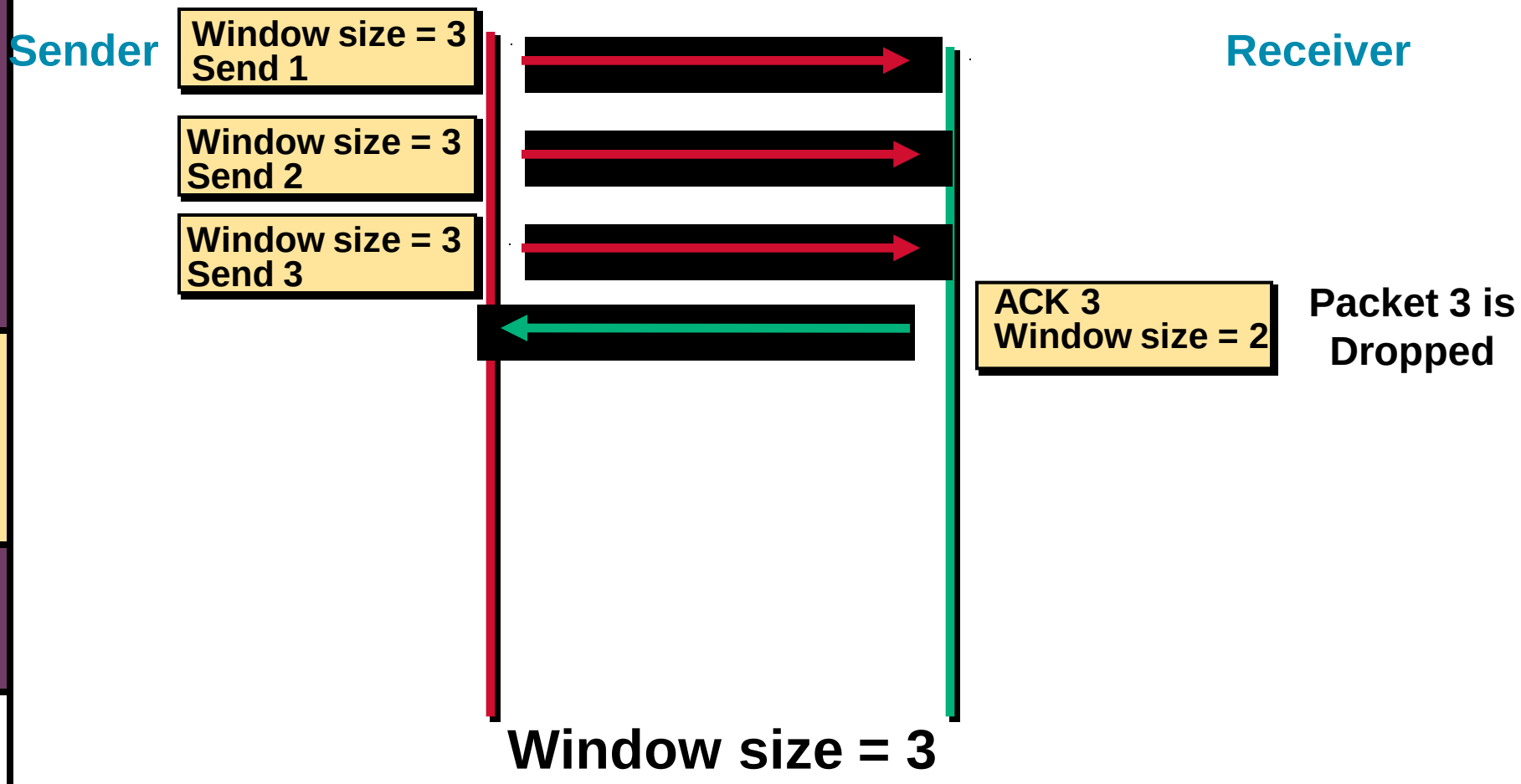


Window size = 3

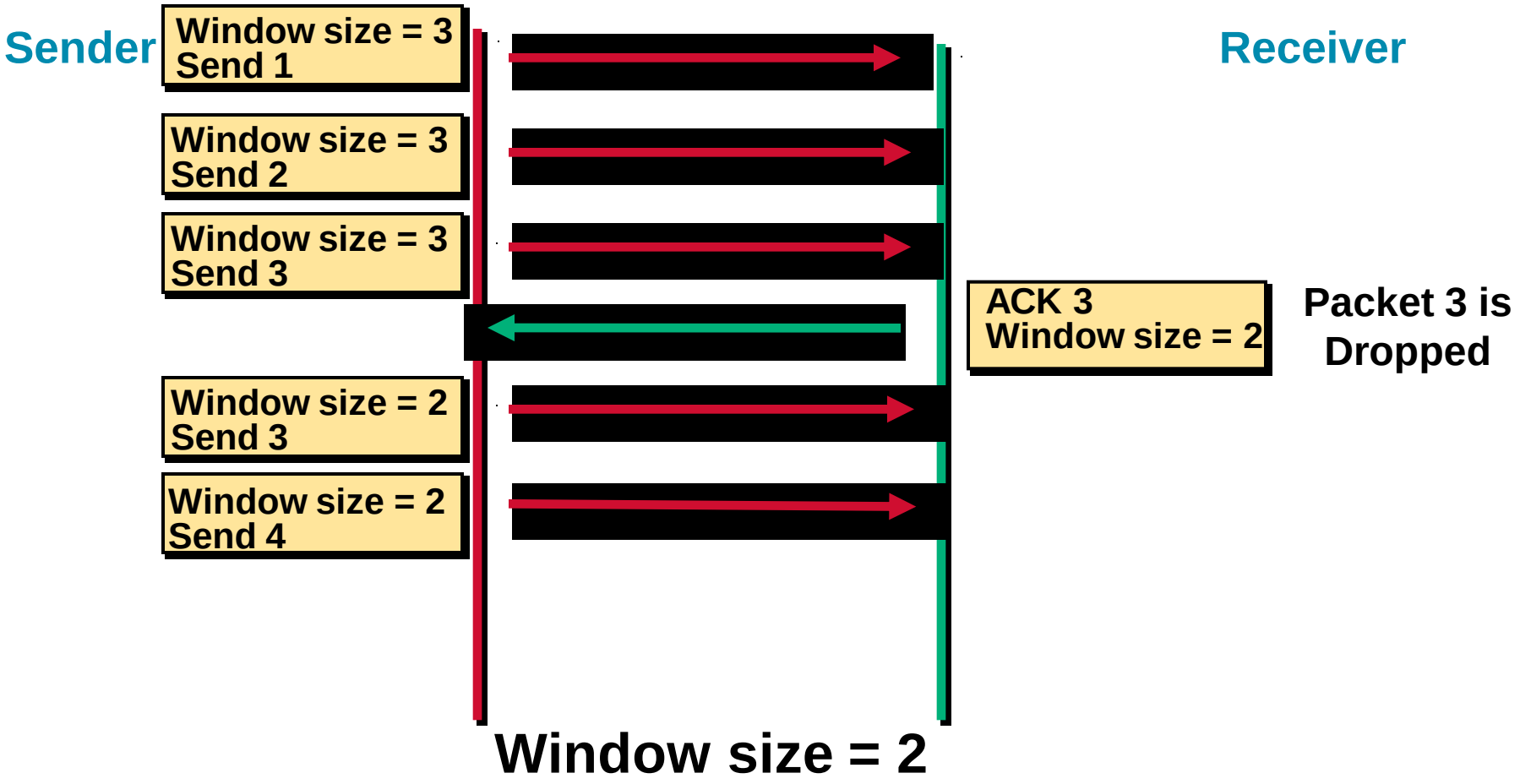
TCP Windowing



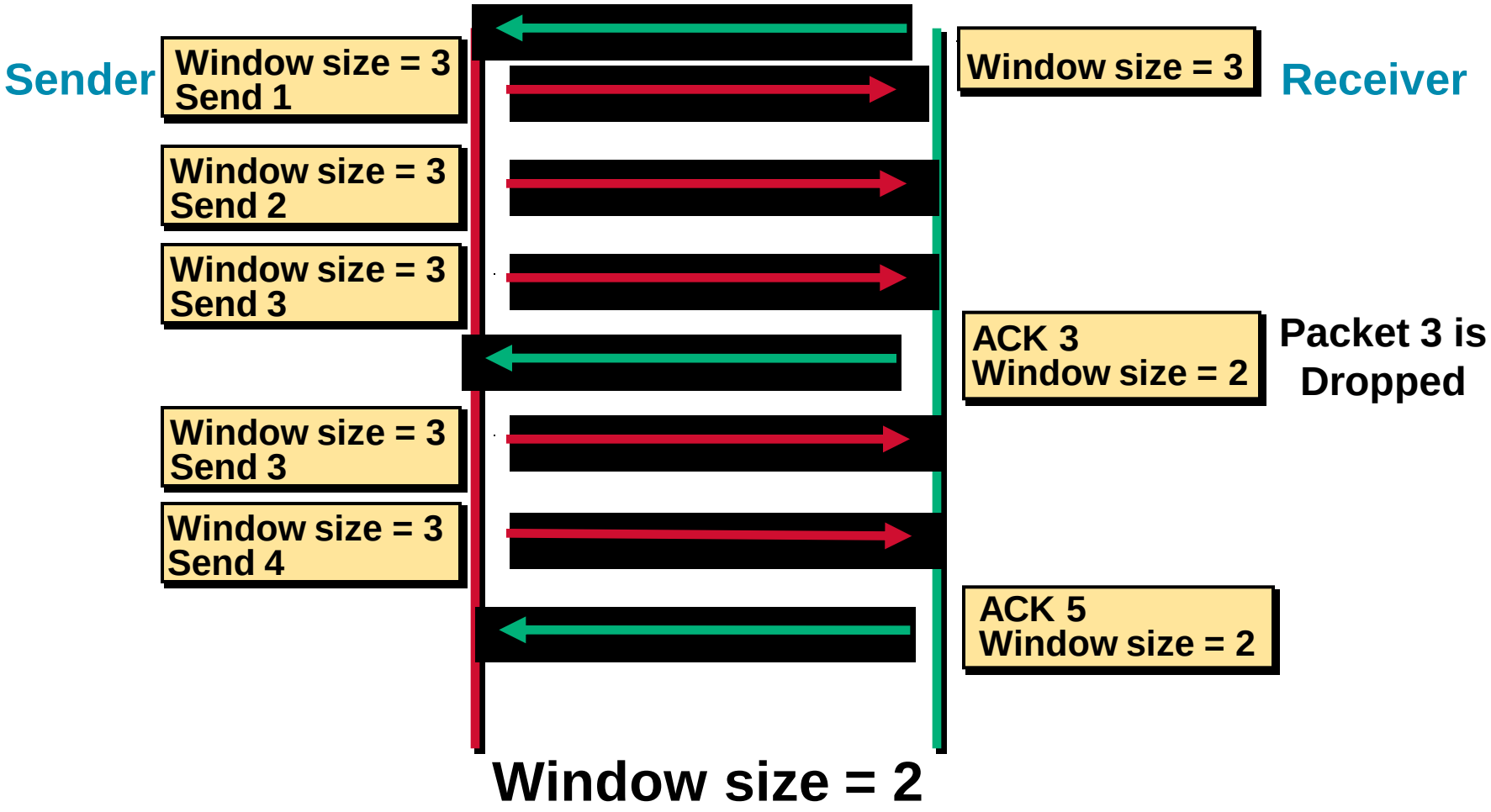
TCP Windowing



TCP Windowing



TCP Windowing



TCP sequence và Acknowledgement

TCP chia nhỏ dữ liệu thành những segment. Những segment được truyền từ nơi gởi đến nơi nhận, nhưng trong quá trình truyền bên nhận có thể nhận không theo thứ tự

Để bên nhận tái lắp ghép lạl data thì khi bởl mỗl segent đều được ghi một số thứ tự gọl là *sequence number*

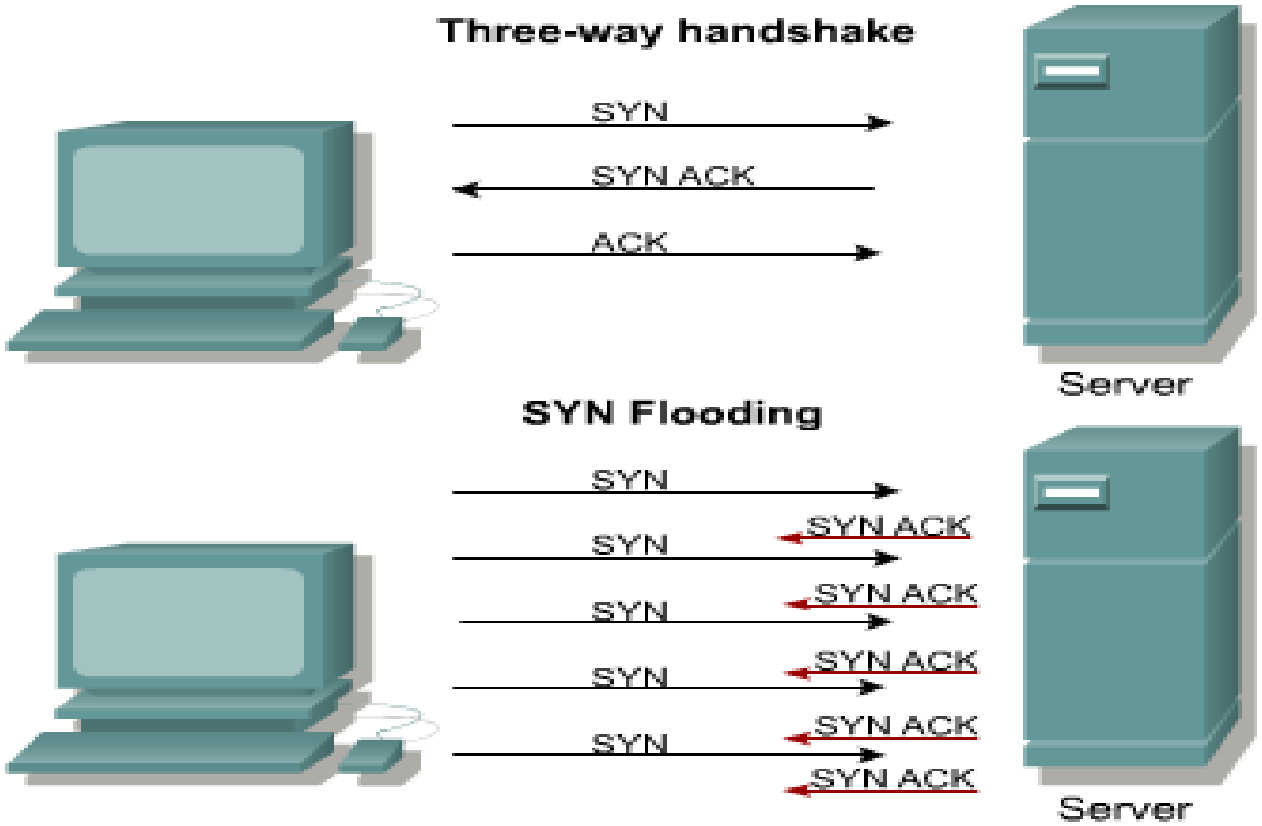
Tấn công từ chối dịch vụ (Denial of Service)

Denial of service(DoS) là kiểu tấn công theo kiểu từ chối dịch vụ bằng cách cố gắng tạo connection.

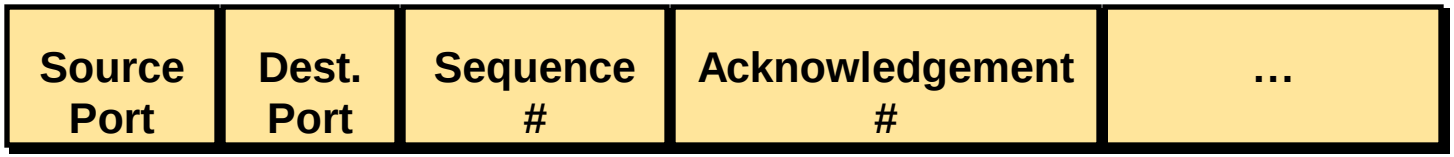
Là một kiểu tấn công của hacker nhằm mục đích làm cho server quá tải bằng cách khởi tạo kết nối đến host muốn tấn công nhưng với IP không tồn tại, khi host bị tấn công nhận segment nó sẽ gởi ACK cho host không tồn tại

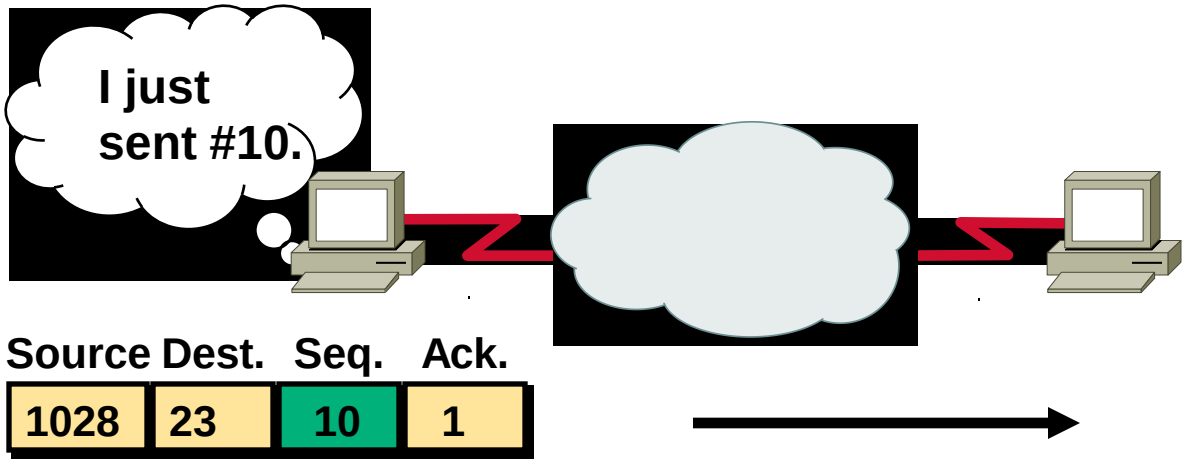
Kết quả không nhận được trả lời trong khi đó hacker liên tục gởi làm cho bên nhận rơi vào tình trạng chờ , tiêu tốn RAM,CPU . . .Đi vào tình trạng treo hệ thống

Tấn công từ chối dịch vụ (Denial of Service)

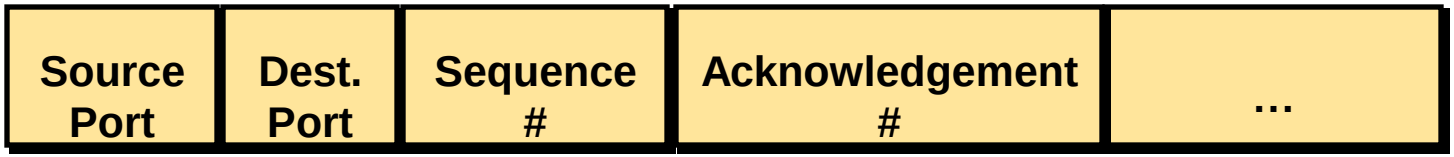


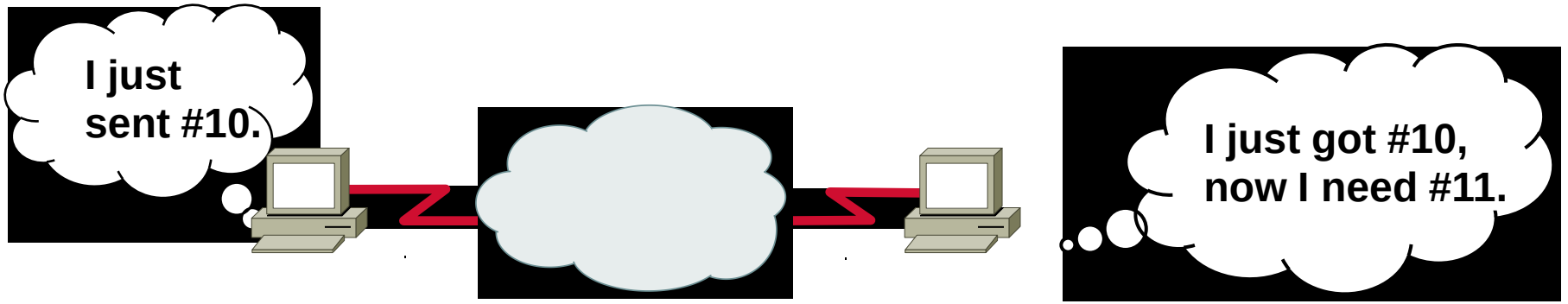
TCP Sequence and Acknowledgment Numbers

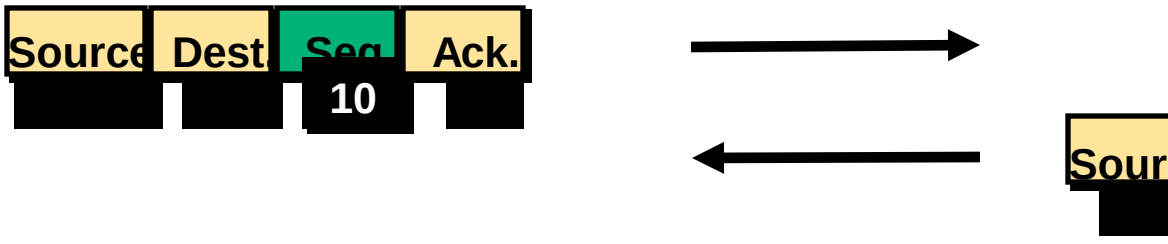




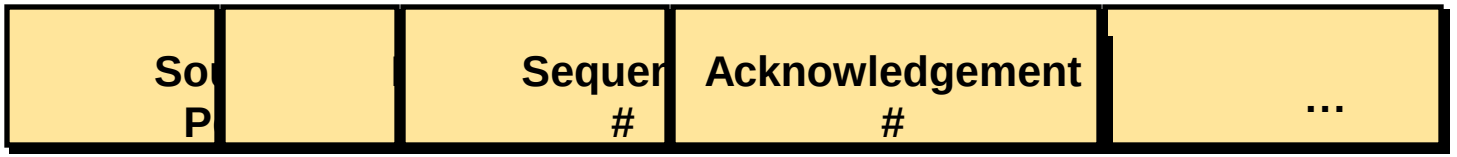
TCP Sequence and Acknowledgment Numbers

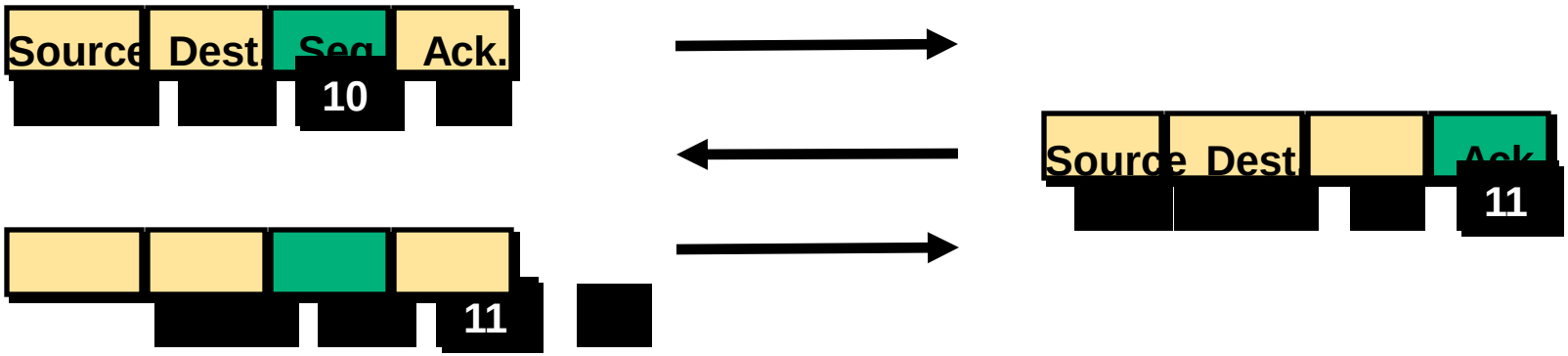






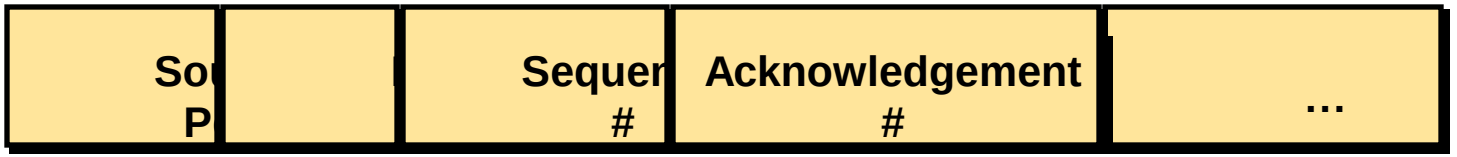
TCP Sequence and Acknowledgment Numbers

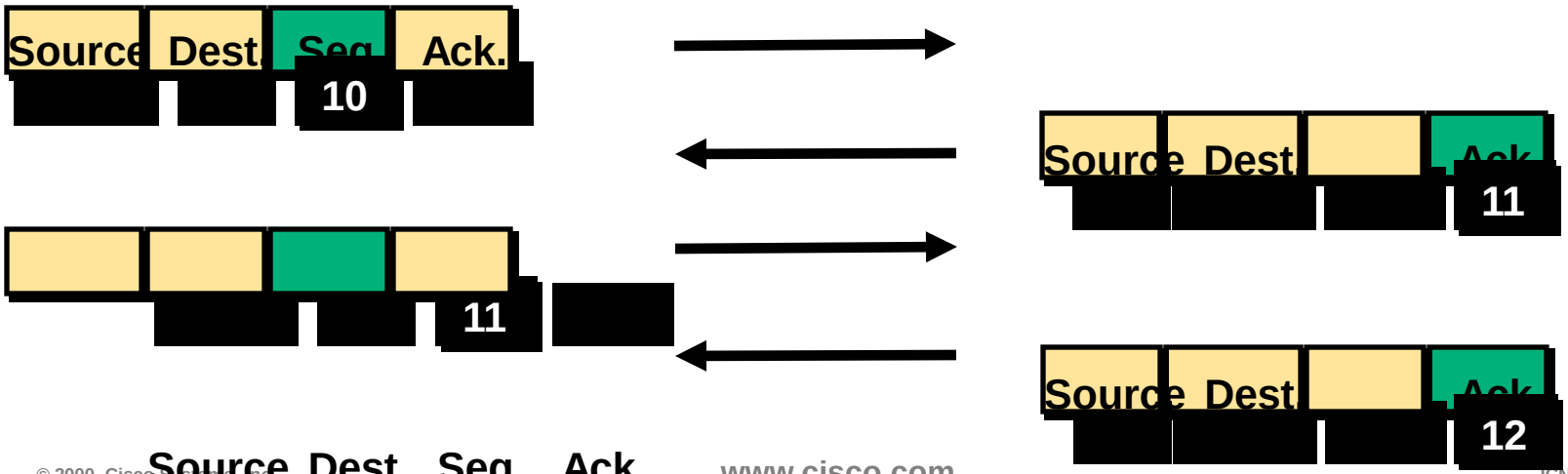




Source Dest. Seq. Ack.

TCP Sequence and Acknowledgment Numbers





Port ứng dụng

Dịch vụ chạy trên một host phải có một Port đăng ký để giao tiếp mới có thể xảy ra .

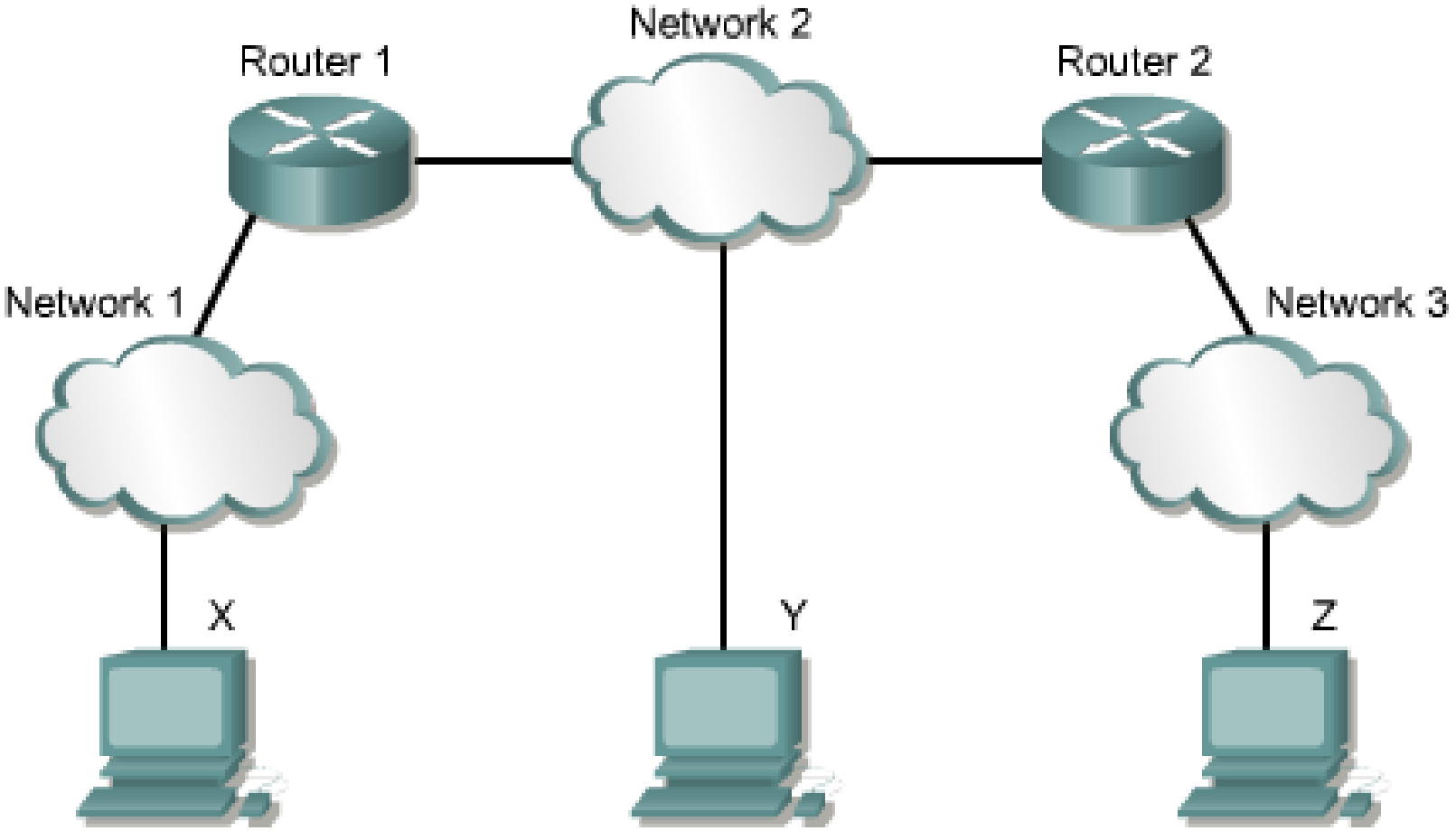
Một host ở xa kết nối đến một dịch vụ mong đợi mà dịch vụ này dùng transport layer và port

Trong TCP/IP gồm 2 giao thức hoạt động tầng transport là *TCPvà UDP và* một số *Port* chuẩn

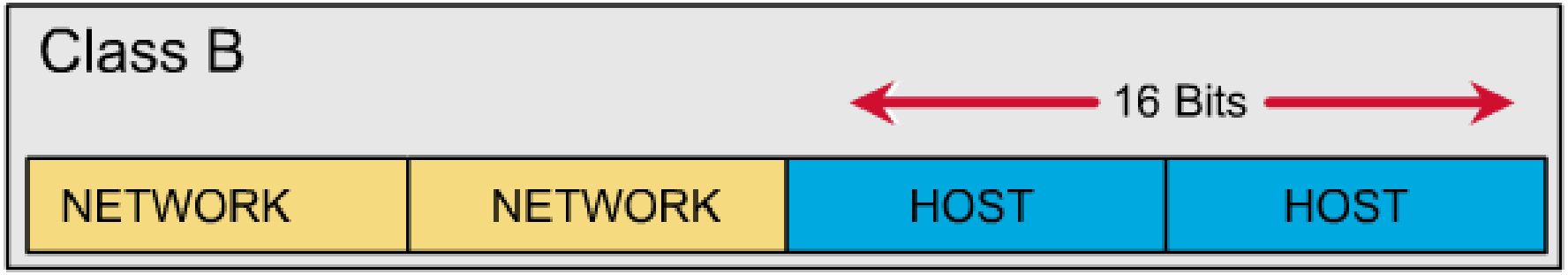
Port của Client

- Khi client kết nối đến một dịch vụ trên server, *source và destination port* phải được chỉ rỏ
- Trong TCp hoặc UDP đều có trường chứa source và destination port
- Khi client muốn kết nối đến một dịch vụ trên server nó tự tạo ra source port > 1024 và destination port là port chuẩn của ứng dụng như : web(80), Telnet(23), DNS(53) . . .

Internet architecture



IP address classes: Class B

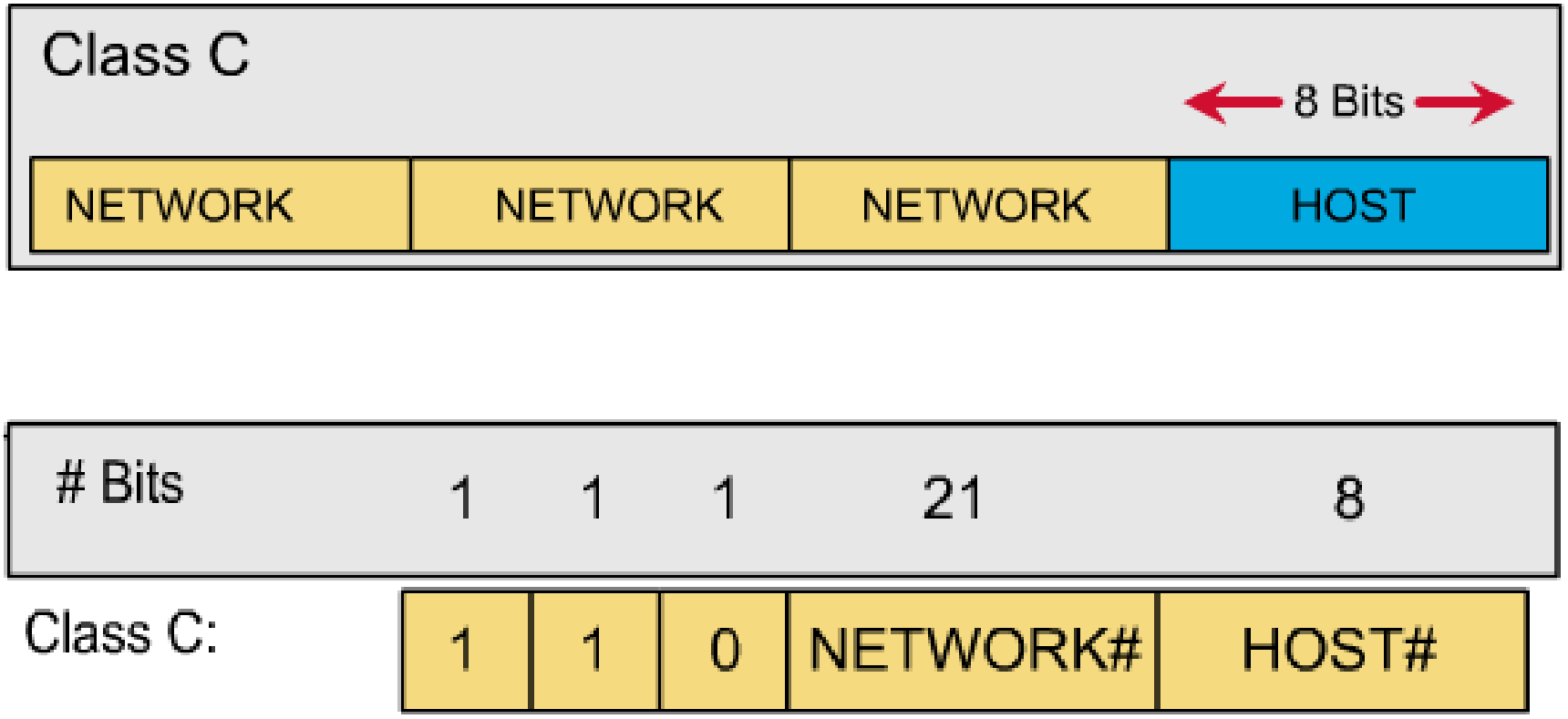


# Bits	1	1	14	16
Class B:	1	0	NETWORK#	HOST#

IP address classes: Class B

- The first 2 bits of a Class B address is always 1 0.
- The first two octets to identify the network part of the address.
- Possible network address from 128.0.0.0 to 191.255.0.0.
- The remaining two octets can be used for the host portion of the address.
- Class B network have up to 65.534 possible IP addresses.

IP address classes: Class C



IP address classes: Class C

- The first 3 bits of a Class C address is always 1 1 0.
- The first three octets to identify the network part of the address.
- Possible network address from 192.0.0.0 to 223.255.255.0.
- The remaining last octet can be used for the host portion of the address.
- Class C network have up to 254 possible IP addresses.

IP address classes: Summary

Class A : 1.0.0.0 - 126.0.0.0

Loopback network : 127.0.0.0

Class B : 128.0.0.0 - 191.255.0.0

Class C : 192.0.0.0 - 223.255.255.0

Class D, multicast :224.0.0.0 - 239.0.0.0

Class E, reserved :240.0.0.0 - 255.0.0.0

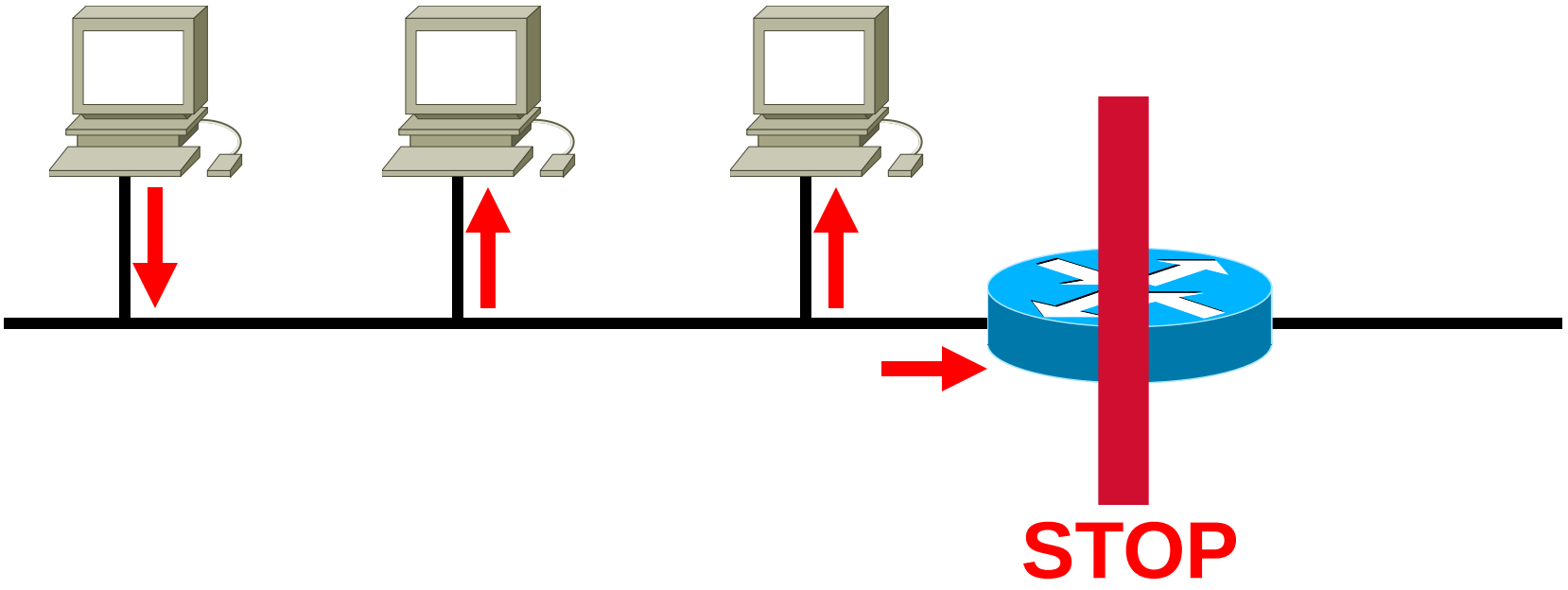
IPv4 adrressing

IP address class	IP address range (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)

Broadcast address

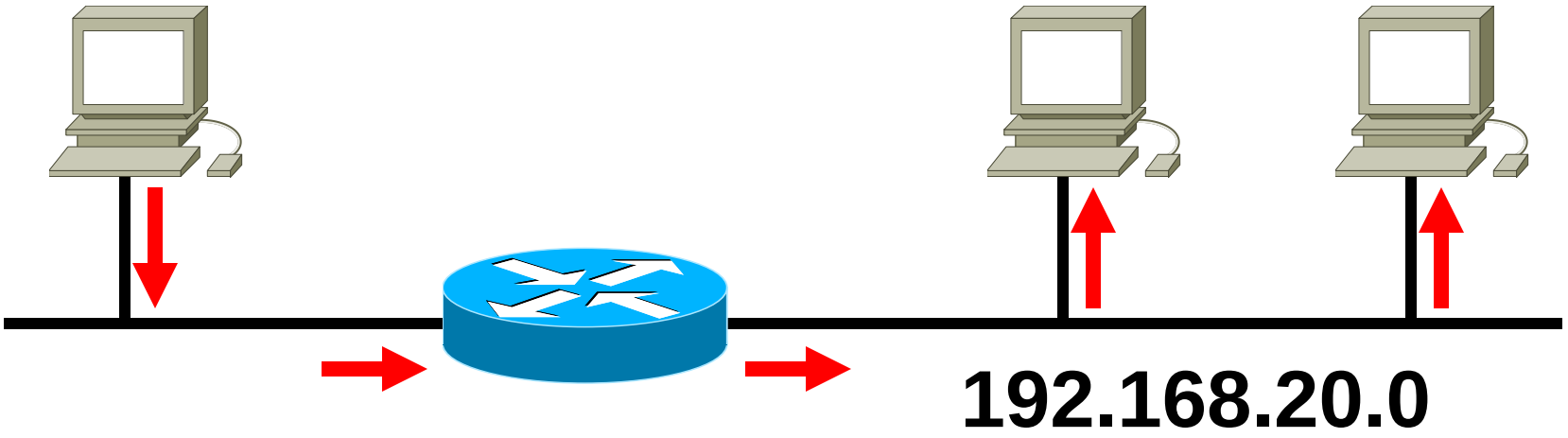
- Broadcast goes to every host with a particular network ID number.
- An IP address that ends with binary 1s in all host bits is reserved for the directed broadcast address.
- An IP address with binary 1s in all network bits and host bits is reserved for the local broadcast address.

Local broadcast address



255.255.255.255

Directed broadcast address



192.168.20.255

Broadcast address

Example: 172.16.20.200

172.16.20.200 is Class B address

Network portion: 172.16

Host portion: 20.200

Network address: 172.16.0.0

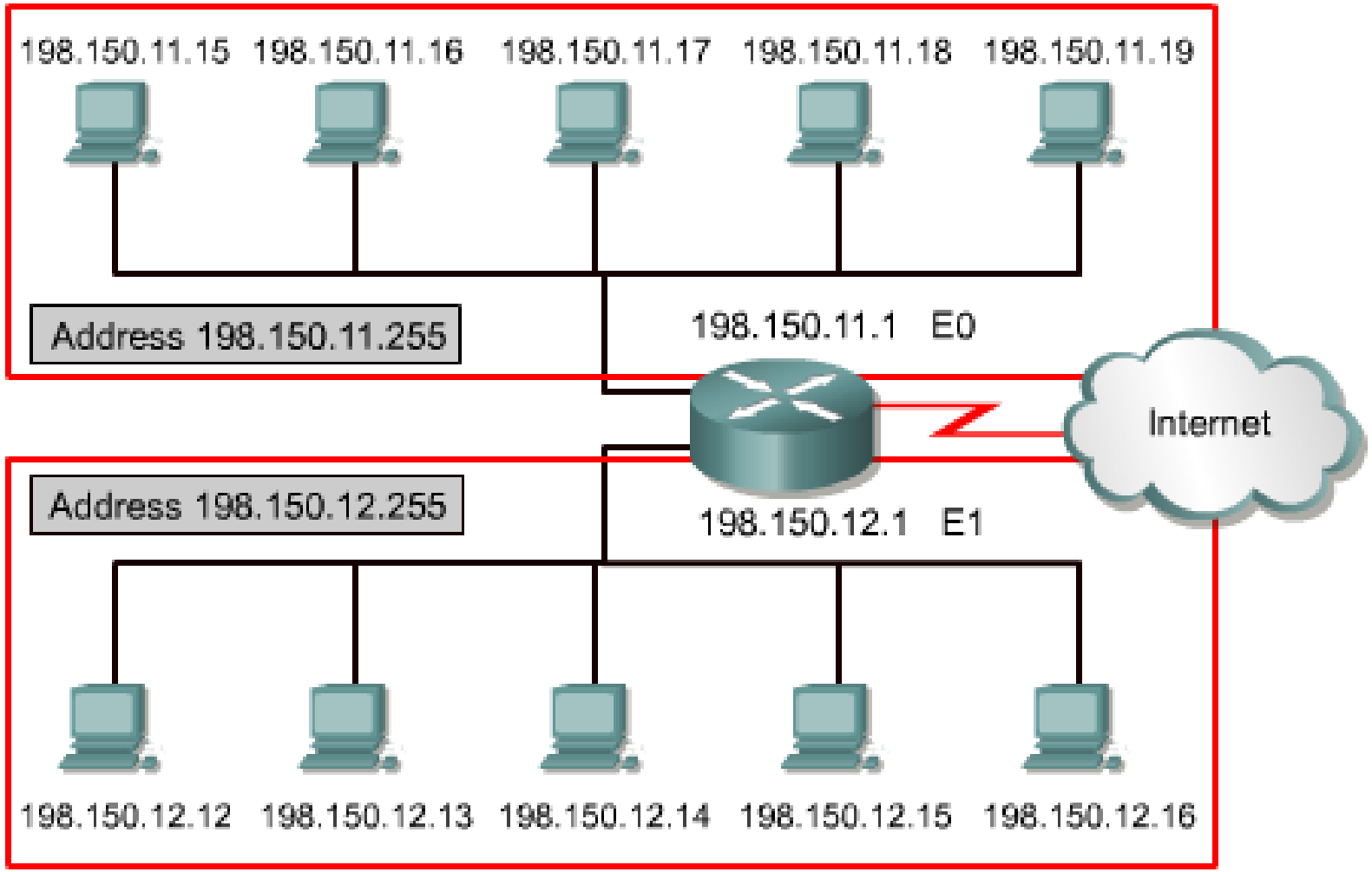
Broadcast address: 172.16.255.255

Private addresses

According to RFC-1918.

- Organizations make use of the private Internet address space for hosts that require IP connectivity within their enterprise network, but do not require external connections to the global Internet.
- Class A: 10.0.0.0.
- Class B: 172.16.0.0 - 172.31.0.0.
- Class C: 192.168.0.0 - 192.168.255.0.

Reserved IP addresses



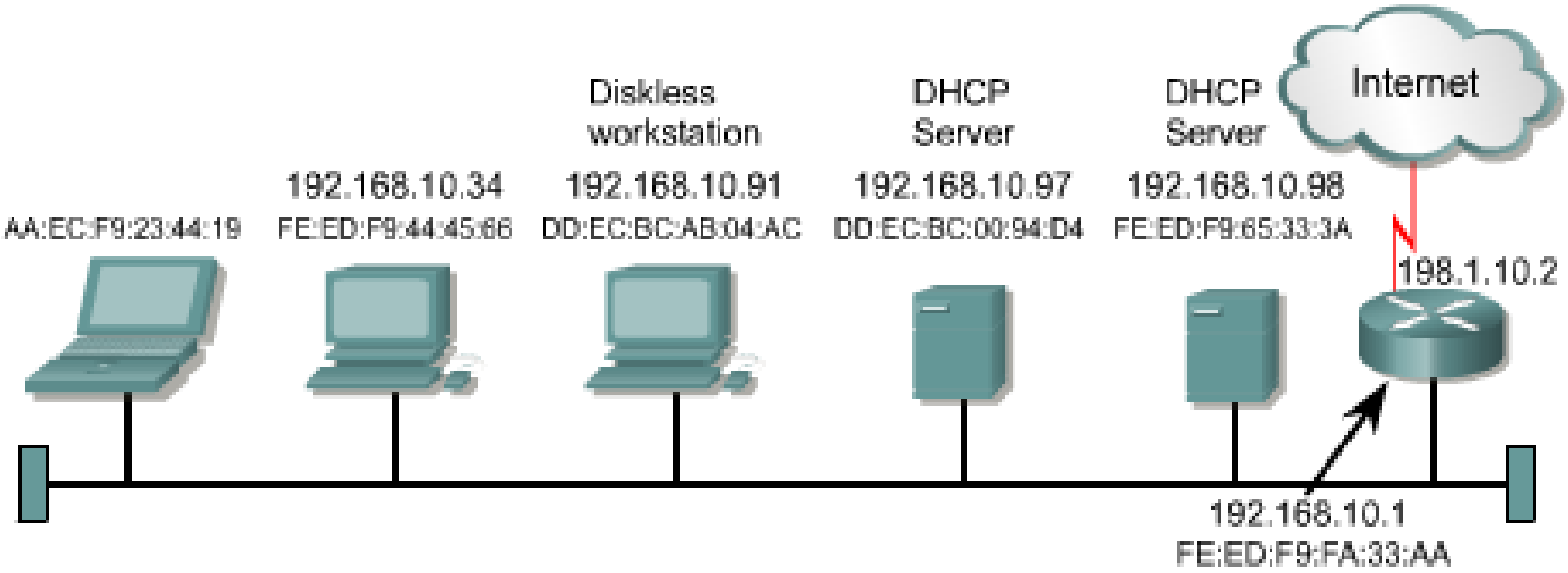
Private IP addresses

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

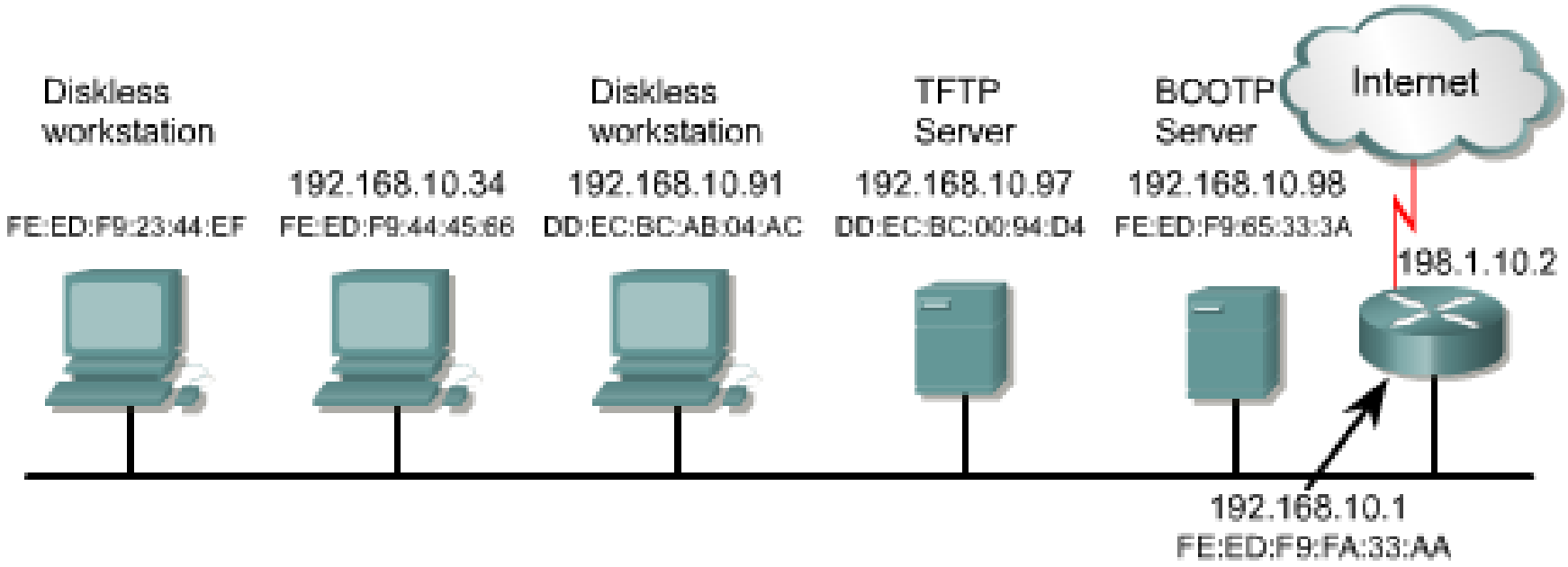
Obtaining an IP address

- Static assignment of an IP address
- RARP IP address assignment
- BOOTP IP address assignment
- DHCP IP address management
- Address Resolution Protocol (ARP)

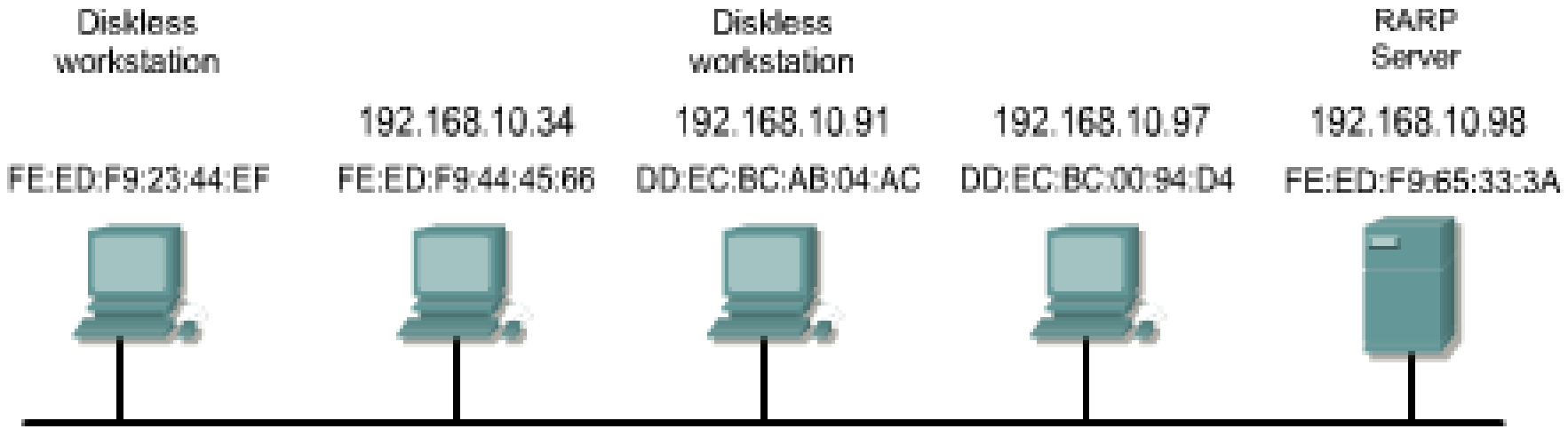
DHCP



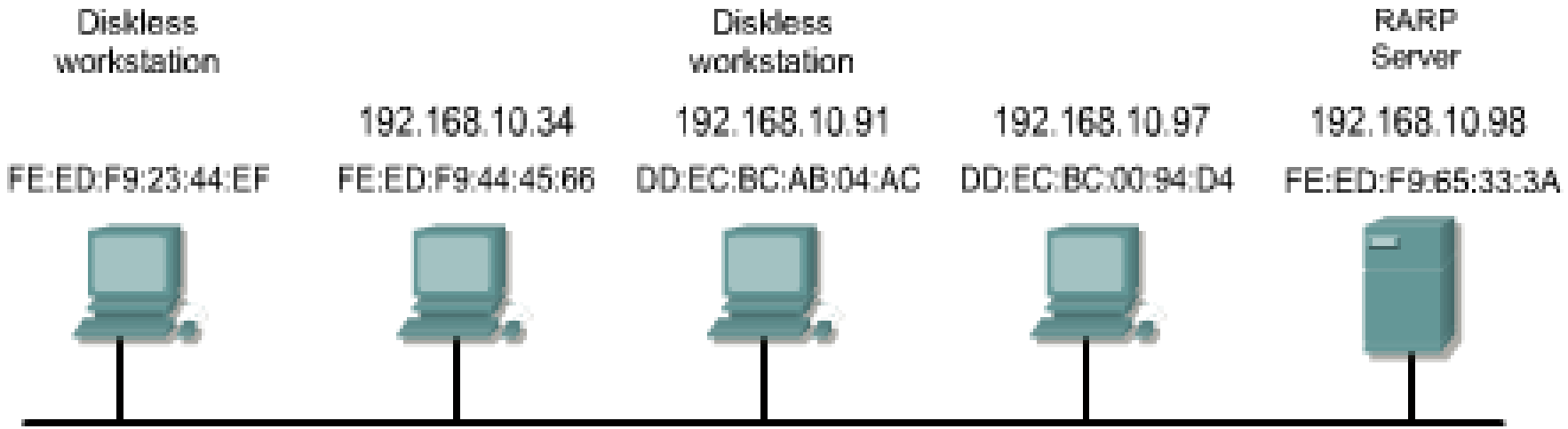
BOOTP



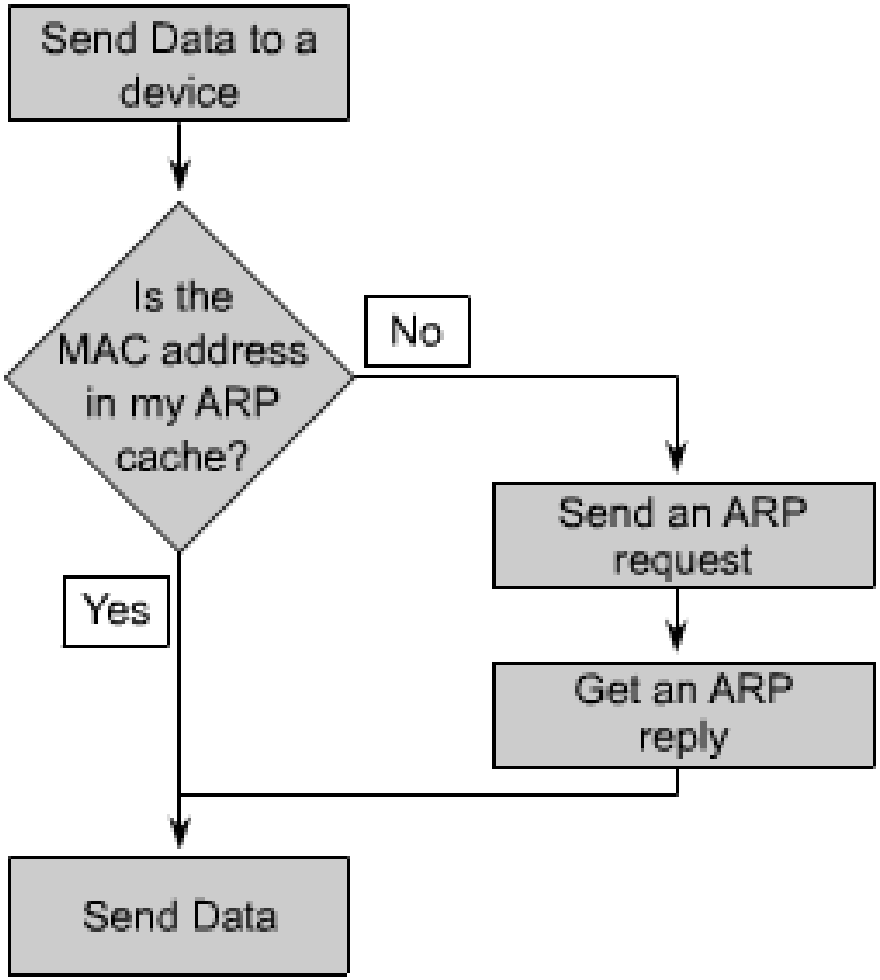
RARP



ARP



ARP Process



SUBNETTING AND CREATING A SUBNET

Cisco Systems

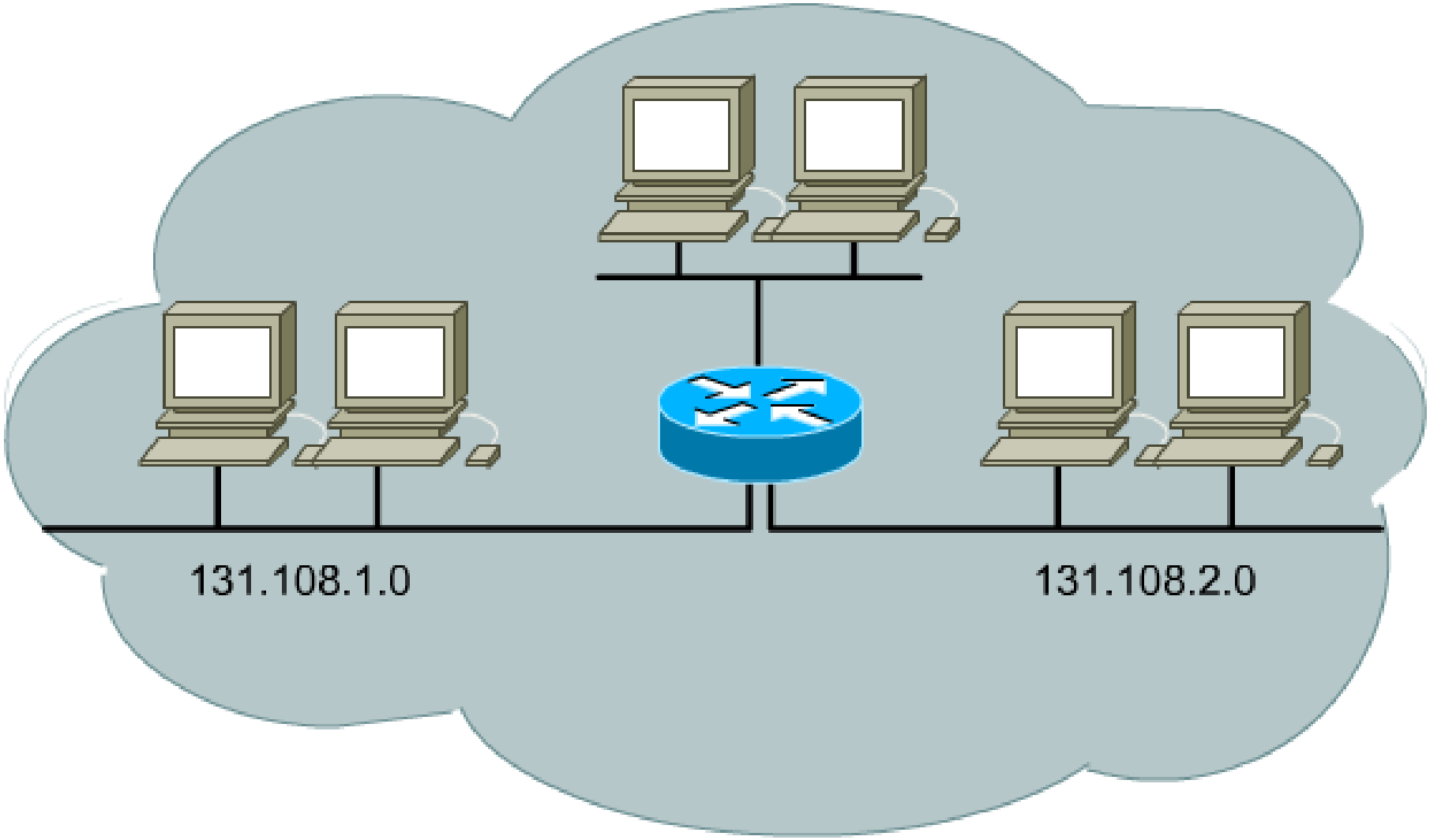
Why we need to divide network?

Network administrators sometimes need to divide networks, especially large ones, into smaller networks:

- **Reduce the size of a broadcast domain.**
- **Improve network security.**
- **Implement the hierarchical managements.**

So we need more network addresses for your network.

Divide network by three



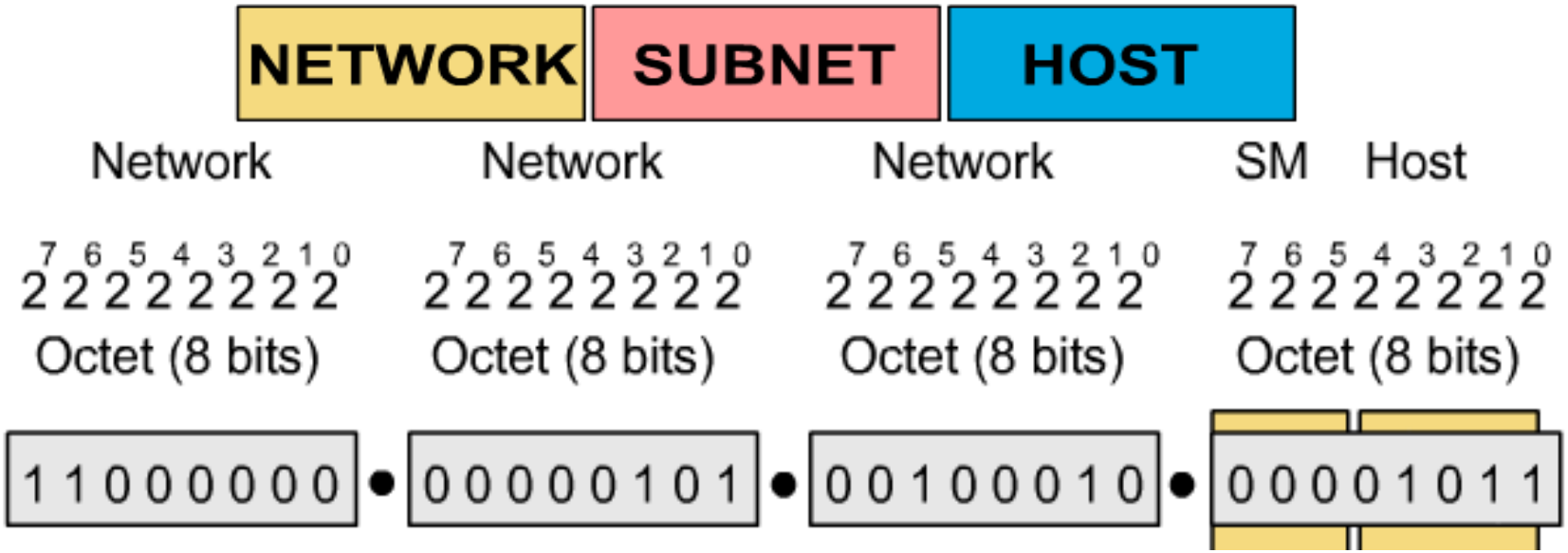
Subnetting

Subnetworks are smaller divisions of network.

- **Subnet addresses include the Class A, Class B, or Class C network portion, subnet field and a host field.**
- **To create a subnet address, a network administrator borrows bits from the original host portion and designates them as the subnet field.**
- **Subnet addresses are assigned locally, usually by a network administrator.**

Subnetting

SOLUTION: Create another section in the IP address called the subnet.



HOW???

By using a SUBNET MASK

Subnet mask

"Extended Network Prefix".

Determines which part of an IP address is the network field and which part is the host field.

- **32 bits long.**
- **Divided into four octets.**
- **All 1's use for Network and Subnet portions.**
- **All 0's use for Host portions.**

Network Default Subnet Mask

- Class A : 255.0.0.0
- Class B : 255.255.0.0
- Class C : 255.255.255.0

Default subnet mask: Example

IP address : 10.10.2.100

00001010.00001010.00000010.01100100.

Default SM : 255.0.0.0

11111111.0000000.00000000.00000000.

00001010.00001010.00000010.01100100.

Class A network:

- **8 bits for network portion.**
- **0 bits for subnet portion.**
- **24 bits for host portion.**

Subnet address: 10.0.0.0

Default subnet mask: Cont

IP address : 172.16.2.100

10101100.00010000.00000010.01100100.

Default SM : 255.255.0.0

11111111.11111111.00000000.00000000.

10101100.00010000.00000010.01100100.

Class B network:

- **16 bits for network portion.**
- **0 bits for subnet portion.**
- **16 bits for host portion.**

Subnet address: 172.16.0.0

Default subnet mask: Cont

IP address : 192.168.2.100

11000000.10101000.00000010.01100100.

Default SM : 255.255.255.0

11111111.11111111.11111111.00000000.

11000000.10101000.00000010.01100100.

Class C network:

- **24 bits for network portion.**
- **0 bits for subnet portion.**
- **8 bits for host portion.**

Subnet address: 192.168.2.0

Subnet mask: Example

IP address : 172.16.65.100

10101100.00010000.01000001.01100100.

Subnet Mask : 255.255.240.0

11111111.11111111.11110000.00000000.

10101100.00010000.01000001.01100100.

Class B network:

- **16 bits for network portion.**
- **4 bits for subnet portion.**
- **12 bits for host portion.**

Subnet address: 172.16.64.0

Boolean algebra review

Boolean operators:

- **AND.**
- **OR.**
- **NOT.**

AND operator

1 AND 1 = 1

1 AND 0 = 0

0 AND 1 = 0

0 AND 0 = 0

OR operator

1 OR 1 = 1

1 OR 0 = 1

0 OR 1 = 1

0 OR 0 = 0

NOT operator

NOT 1 = 0

NOT 0 = 1

Boolean algebra examples

1010 AND 0110 = 0010

1010 OR 0110 = 1110

Why we need to know Boolean ops?

IP Address	AND	Subnet Mask	=	Network and Subnet address

- Network layer performs the Boolean operations in order to find the network ID of a subnet
- Example:

  IP addr→ 172.16.65.100 AND 255.255.240.0 ←SM

  = 172.16.64.0 ← Subnetwork address

Subnetting example

Given network 172.16.0.0.

We need :

- **8 usable subnets**
- **Up to 1000 hosts on each subnet.**

Calculating a subnet

- Determine the class of network and default subnet mask.
- Determine how many bits to borrow.
- Determine the subnet mask and the actual number of subnets and hosts.
- Determine the ranges of host address for each subnet. Choose the subnets that you want to use.

Calculating a subnet: STEP 1

- Determine the Class of network

  Class B

- Determine the default subnet mask

  255.255.0.0

Calculating a subnet: STEP 2

- Number of subnets <= 2^n - 2 with n is number of bits that are borrowed.
- Number of hosts <= 2^m - 2 with m is number of bits that are remained from host bit *(m=host bit – n)*
- Determine how many bits to borrow from the host portion from requirement:
  - 8 subnets.
  - 1000 hosts on each subnet.

Calculating a subnet: STEP 2 (Cont.)

Choose n = 4:

- **Number of possible subnets is:**

$$2^4 - 2 = 14$$

- **Number of possible hosts on each subnet is:**

$$2^{(16-4)} - 2 = 4094$$

Other choice n = 5 , n = 6 ?

Calculating a subnet: STEP 2 (Cont.)

128	64	32	16	8	4	2	1	
1	0	0	0	0	0	0	0	= 128
1	1	0	0	0	0	0	0	= 192
1	1	1	0	0	0	0	0	= 224
1	1	1	1	0	0	0	0	= 240
1	1	1	1	1	0	0	0	= 248
1	1	1	1	1	1	0	0	= 252
1	1	1	1	1	1	1	0	= 254
1	1	1	1	1	1	1	1	= 255

The subnet mask: 255.255.240.0.

Calculating a subnet: STEP 3

Determine the subnets and the ranges of host address for each subnet. Including:

- **Sub-network addresses**
- **Range of usable IP addresses**
- **Sub-network broadcast addresses**

Calculating a subnet: STEP 3 (Cont.)

Determine the subnets from 4 borrowed bits from the host portion (last 2 bytes):

- 1st subnet: .0000**0000**.00000000
- 2nd subnet: .0001**0000**.00000000
- 3rd subnet: .0010**0000**.00000000

…

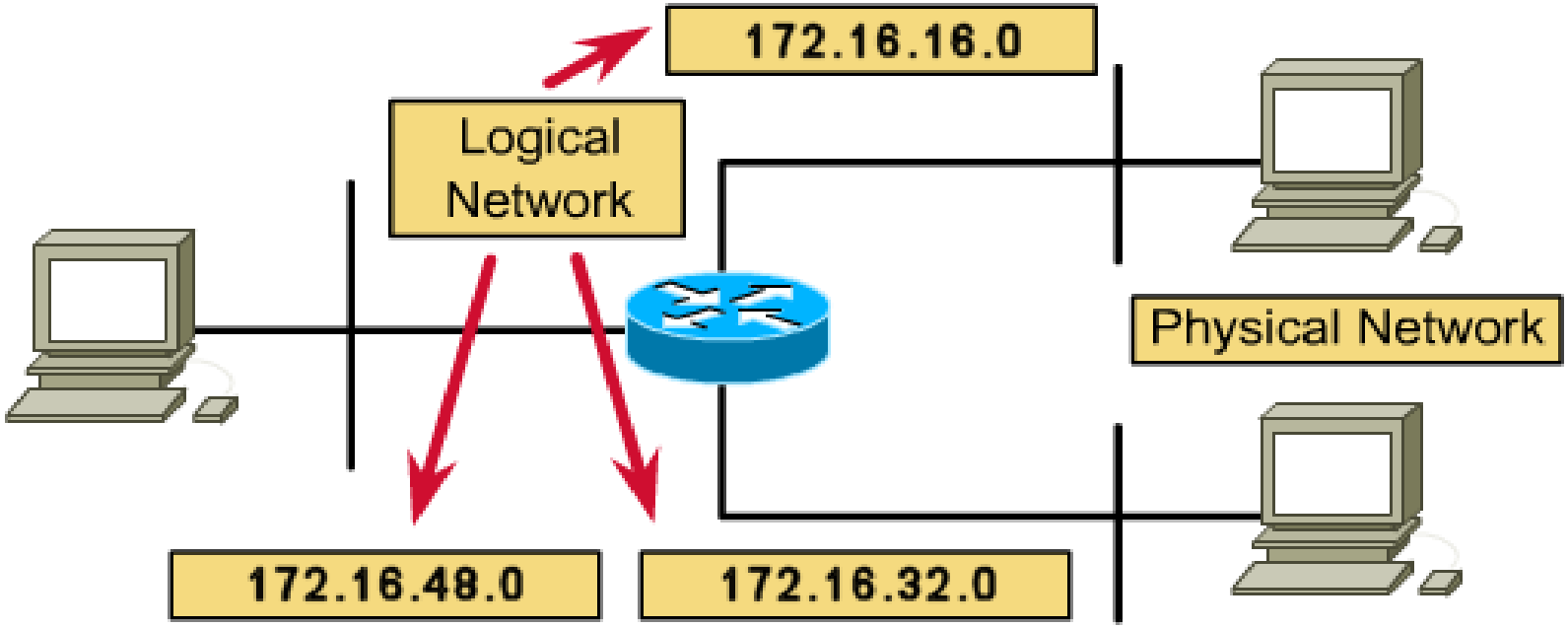
- 15th subnet: .1111**0000**.00000000

"Let subnet bits run"

Calculating a subnet: STEP 3 (Cont.)

No	Sub-network address	Possible host address	Broadcast address	Use ?
0	172.16.0.0	172.16.0.1 – 172.16.15.254	172.16.15.255	N
1	172.16.16.0	172.16.16.1 – 172.16.31.254	172.16.31.255	Y
2	172.16.32.0	172.16.32.1 – 172.16.47.254	172.16.47.255	Y
..	..	..	..	..
..	..	..	..	..
13	172.16.208.0	172.16.208.1 – 172.16.223.254	172.16.223.255	Y
14	172.16.224.0	172.16.224.1 – 172.16.239.254	172.16.239.255	Y
15	172.16.240.0	172.16.240.1 – 172.16.255.254	172.16.255.255	N

Calculating a subnet: STEP 3 (Cont.)



- Using subnets No.1 to No.8.
- Assign IP addresses to hosts and interfaces on each network. IP address configuration.

How many bits can I borrow?

The minimum bits you can borrow is:

2 bits.

The maximum bits you can borrow is:

A: 22 bits ~ 2^{22} - 2 = 4.194.302 subnets.

B: 14 bits ~ 2^{14} - 2 = 16.382 subnets.

C: 06 bits ~ 2^{06} - 2 = 62 subnets.

Addresses are loose by subnetting.

Number of Bits Borrowed	Number of Subnets Created	Number of Hosts Per Subnet	Total Number of Hosts	Percent Used
2	2	62	124	49%
3	6	30	180	71%
4	14	14	196	77%
5	30	6	180	71%
6	62	2	124	49%

- ***“Network administrator must strike a balance between the number of subnets required, the hosts per subnet that is acceptable, and the resulting waste of addresses”.***

Preparation for LAB

Chương I
HỌ GIAO THỨC TCP/IP

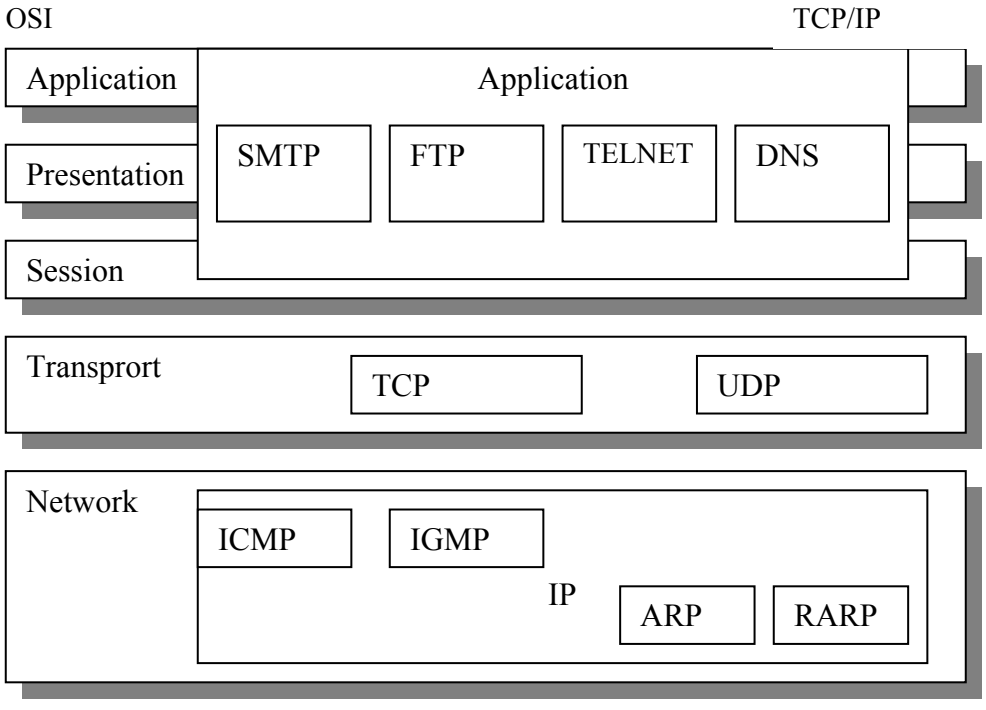
I.1.1. Họ giao thức TCP/IP

TCP/IP là một họ giao thức để cung cấp phương tiện truyền thông liên mạng và nó được cấu trúc theo kiểu phân cấp.

Khác với mô hình OSI/ISO tầng liên mạng sử dụng giao thức kết nối mạng "không liên kết" (connectionless) IP, tạo thành hạt nhân hoạt động của Internet. Cùng với các thuật toán định tuyến RIP, OSPF, BGP, tầng liên mạng IP cho phép kết nối một cách mềm dẻo và linh hoạt các loại mạng "vật lý" khác nhau như: Ethernet, Token Ring , X.25...

Giao thức trao đổi dữ liệu "có liên kết" (connection - oriented) TCP được sử dụng ở tầng vận chuyển để đảm bảo tính chính xác và tin cậy việc trao đổi dữ liệu dựa trên kiến trúc kết nối "không liên kết" ở tầng liên mạng IP.

Các giao thức hỗ trợ ứng dụng phổ biến như truy nhập từ xa (telnet), chuyển tệp (FTP), dịch vụ World Wide Web (HTTP), thư điện tử (SMTP), dịch vụ tên miền (DNS) ngày càng được cài đặt phổ biến như những bộ phận cấu thành của các hệ điều hành thông dụng như UNIX (và các hệ điều hành chuyên dụng cùng họ của các nhà cung cấp thiết bị tính toán như AIX của IBM, SINIX của Siemens, Digital UNIX của DEC), Windows9x/NT, Novell Netware,...



Hình 2.1 Mô hình OSI và mô hình kiến trúc của TCP/IP

Hình 13. Mô hình thanm chiếu TCP/IP với chuẩn OSI 7 lớp

Trong cấu trúc bốn lớp của TCP/IP, khi dữ liệu truyền từ lớp ứng dụng cho đến lớp vật lý, mỗi lớp đều cộng thêm vào phần điều khiển của mình để đảm bảo cho việc truyền dữ liệu được chính xác. Mỗi thông tin điều khiển này được gọi là một *header* và được đặt ở trước phần dữ liệu được truyền. Mỗi lớp xem tất cả các thông tin mà nó nhận được từ lớp trên là dữ liệu, và đặt phần thông tin điều khiển *header* của nó vào trước phần thông tin này. Việc cộng thêm vào các *header* ở mỗi lớp trong quá trình truyền tin được gọi là *encapsulation*. Quá trình nhận dữ liệu diễn ra theo chiều ngược lại: mỗi lớp sẽ tách ra phần *header* trước khi truyền dữ liệu lên lớp trên.
Mỗi lớp có một cấu trúc dữ liệu riêng, độc lập với cấu trúc dữ liệu được dùng ở lớp trên hay lớp dưới của nó. Sau đây là giải thích một số khái niệm thường gặp.
Stream là dòng số liệu được truyền trên cơ sở đơn vị số liệu là Byte.
Số liệu được trao đổi giữa các ứng dụng dùng TCP được gọi là *stream*, trong khi dùng UDP, chúng được gọi là *message*.
Mỗi gói số liệu TCP được gọi là *segment* còn UDP định nghĩa cấu trúc dữ liệu của nó là *packet*.

Lớp Internet xem tất cả các dữ liệu như là các khối và gọi là *datagram*. Bộ giao thức TCP/IP có thể dùng nhiều kiểu khác nhau của lớp mạng dưới cùng, mỗi loại có thể có một thuật ngữ khác nhau để truyền dữ liệu.

Phần lớn các mạng kết cấu phần dữ liệu truyền đi dưới dạng các *packets* hay là các *frames*.

Application	**Stream**
Transport	**Segment/datagram**
Internet	**Datagram**
Network Access	**Frame**

Cấu trúc dữ liệu tại các lớp của TCP/IP

Lớp truy nhập mạng

Network **A**ccess **L**ayer là lớp thấp nhất trong cấu trúc phân bậc của TCP/IP. Những giao thức ở lớp này cung cấp cho hệ thống phương thức để truyền dữ liệu trên các tầng vật lý khác nhau của mạng. Nó định nghĩa cách thức truyền các khối dữ liệu (datagram) IP. Các giao thức ở lớp này phải biết chi tiết các phần cấu trúc vật lý mạng ở dưới nó (bao gồm cấu trúc gói số liệu, cấu trúc địa chỉ...) để định dạng được chính xác các gói dữ liệu sẽ được truyền trong từng loại mạng cụ thể.

So sánh với cấu trúc OSI/OSI, lớp này của TCP/IP tương đương với hai lớp Datalink, và Physical.

Chức năng định dạng dữ liệu sẽ được truyền ở lớp này bao gồm việc nhúng các gói dữ liệu IP vào các *frame* sẽ được truyền trên mạng và việc ánh xạ các địa chỉ IP vào địa chỉ vật lý được dùng cho mạng.

Lớp liên mạng

Internet Layer là lớp ở ngay trên lớp Network Access trong cấu trúc phân lớp của TCP/IP. Internet Protocol là giao thức trung tâm của TCP/IP và là phần quan trọng nhất của lớp Internet. IP cung cấp các gói lưu chuyển cơ bản mà thông qua đó các mạng dùng TCP/IP được xây dựng.

Chức năng chính của - Giao thức liên mạng IP(v4)

Trong phần này trình bày về giao thức IPv4 (để cho thuận tiện ta viết IP có nghĩa là đề cập đến IPv4).

Mục đích chính của IP là cung cấp khả năng kết nối các mạng con thành liên mạng để truyền dữ liệu. IP cung cấp các chức năng chính sau:

- Định nghĩa cấu trúc các gói dữ liệu là đơn vị cơ sở cho việc truyền dữ liệu trên Internet.
- Định nghĩa phương thức đánh địa chỉ IP.
- Truyền dữ liệu giữa tầng vận chuyển và tầng mạng .
- Định tuyến để chuyển các gói dữ liệu trong mạng.
- Thực hiện việc phân mảnh và hợp nhất (fragmentation -reassembly) các gói dữ liệu và nhúng / tách chúng trong các gói dữ liệu ở tầng liên kết.

Định tuyến IP

Có hai loại định tuyến:

- Định tuyến trực tiếp: Định tuyến trực tiếp là việc xác định đường nối giữa hai trạm làm việc trong cùng một mạng vật lý.
- Định tuyến không trực tiếp. Định tuyến không trực tiếp là việc xác định đường nối giữa hai trạm làm việc không nằm trong cùng một mạng vật lý và vì vậy, việc truyền tin giữa chúng phải được thực hiện thông qua các trạm trung gian là các gateway.

Để kiểm tra xem trạm đích có nằm trên cùng mạng vật lý với trạm nguồn hay không, người gửi phải tách lấy phần địa chỉ mạng trong phần địa chỉ IP. Nếu hai địa chỉ này có địa chỉ mạng giống nhau thì datagram sẽ được truyền đi trực tiếp; ngược lại phải xác định một gateway, thông qua gateway này chuyển tiếp các datagram.

Khi một trạm muốn gửi các gói dữ liệu đến một trạm khác thì nó phải đóng gói datagram vào một khung (frame) và gửi các frame này đến gateway gần nhất. Khi một frame đến một gateway, phần datagram đã được đóng gói sẽ được tách ra và IP routing sẽ chọn gateway tiếp dọc theo đường dẫn đến

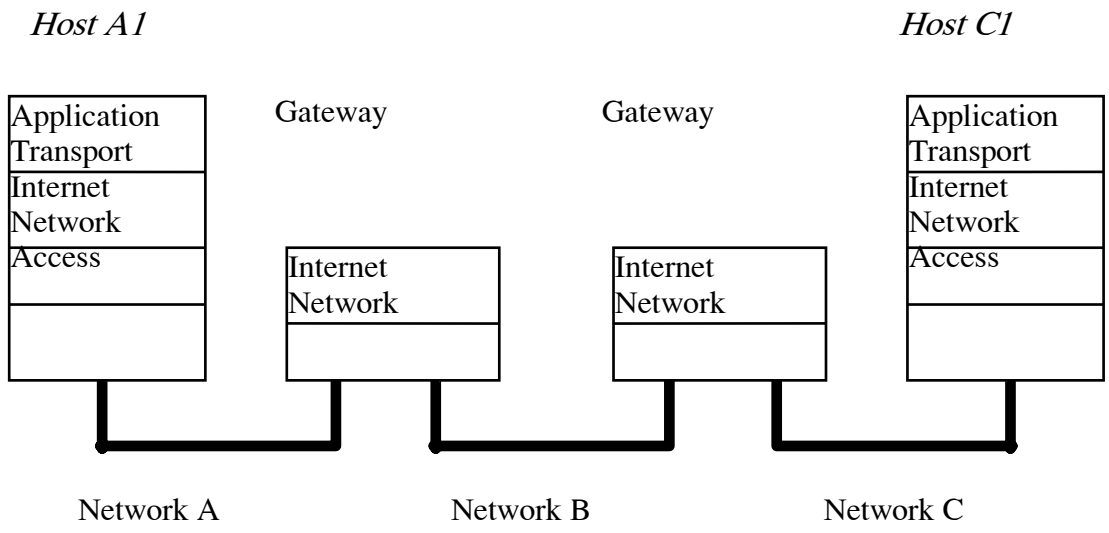
đích. Datagram sau đó lại được đóng gói vào một frame khác và gửi đến mạng vật lý để gửi đến gateway tiếp theo trên đường truyền và tiếp tục như thế cho đến khi datagram được truyền đến trạm đích.

Chiến lược định tuyến: Trong thuật ngữ truyền thống của TCP/IP chỉ có hai kiểu thiết bị, đó là các cổng truyền (gateway) và các trạm (host). Các cổng truyền có vai trò gửi các gói dữ liệu, còn các trạm thì không. Tuy nhiên khi một trạm được nối với nhiều mạng thì nó cũng có thể định hướng cho việc lưu chuyển các gói dữ liệu giữa các mạng và lúc này nó đóng vai trò hoàn toàn như một gateway.

Các trạm làm việc lưu chuyển các gói dữ liệu xuyên suốt qua cả bốn lớp, trong khi các cổng truyền chỉ chuyển các gói đến lớp Internet là nơi quyết định tuyến đường tiếp theo để chuyển tiếp các gói dữ liệu.

Các máy chỉ có thể truyền dữ liệu đến các máy khác nằm trên cùng một mạng vật lý. Các gói từ A1 cần chuyển cho C1 sẽ được hướng đến gateway G1 và G2. Trạm A1 đầu tiên sẽ truyền các gói đến gateway G1 thông qua mạng A. Sau đó G1 truyền tiếp đến G2 thông qua mạng B và cuối cùng G2 sẽ truyền các gói trực tiếp đến trạm C1, bởi vì chúng được nối trực tiếp với nhau thông qua mạng C. Trạm A1 không hề biết đến các gateway nằm ở sau G1. A1 gửi các gói số liệu cho các mạng B và C đến gateway cục bộ G1 và dựa vào gateway này để định hướng tiếp cho các gói dữ liệu đi đến đích. Theo cách này thì trạm C1 trước tiên sẽ gửi các gói của mình đến cho G2 và G2 sẽ gửi đi tiếp cho các trạm ở trên mạng A cũng như ở trên mạng B.

Hình vẽ sau mô tả việc dùng các gateway để gửi các gói dữ liệu:



H×nh 17. §Þnh tuyÕn gi÷a hai hÖ thèng

Việc phân mảnh các gói dữ liệu: Trong quá trình truyền dữ liệu, một gói dữ liệu (datagram) có thể được truyền đi thông qua nhiều mạng khác nhau. Một gói dữ liệu (datagram) nhận được từ một mạng nào đó có thể quá lớn để truyền đi trong gói đơn ở trên một mạng khác, bởi mỗi loại cấu trúc mạng cho phép một đơn vị truyền cực đại (Maximum Transmit Unit - MTU), khác nhau. Đây chính là kích thước lớn nhất của một gói mà chúng có thể truyền. Nếu như một gói dữ liệu nhận được từ một mạng nào đó mà lớn hơn MTU của một mạng khác thì nó cần được phân mảnh ra thành các gói nhỏ hơn, gọi là *fragment.* Quá trình này gọi là quá trình phân mảnh. Dạng của một *fragment* cũng giống như dạng của một gói dữ liệu thông thường. Từ thứ hai trong phần *header* chứa các thông tin để xác định mỗi *fragment* và cung cấp các thông tin để hợp nhất các fragment này lại thành các gói như ban đầu. Trường *identification* dùng để xác định *fragment* này là thuộc về gói dữ liệu nào.

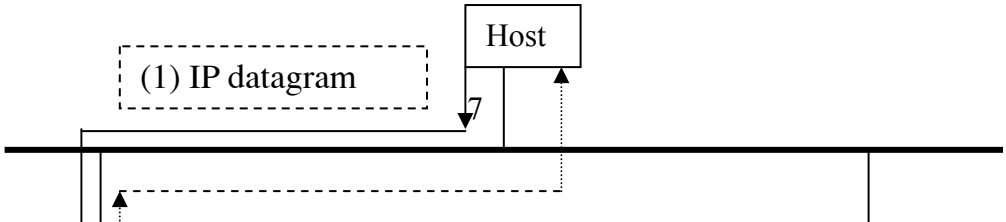
Giao thức ICMP

ICMP ((Internet Control Message Protocol) là một giao thức điều khiển của mức IP, được dùng để trao đổi các thông tin điều khiển dòng số liệu, thông báo lỗi và các thông tin trạng thái khác của bộ giao thức TCP/IP. Ví dụ:

- Điều khiển lưu lượng dữ liệu (Flow control): khi các gói dữ liệu đến quá nhanh, thiết bị đích hoặc thiết bị định tuyến ở giữa sẽ gửi một thông điệp ICMP trở lại thiết bị gửi, yêu cầu thiết bị gửi tạm thời ngừng việc gửi dữ liệu.
- Thông báo lỗi: trong trường hợp địa chỉ đích không tới được thì hệ thống sẽ gửi một thông báo lỗi "Destination Unreachable".
- Định hướng lại các tuyến đường: một thiết bị định tuyến sẽ gửi một thông điệp ICMP "định tuyến lại" (Redirect Router) để thông báo với một trạm là nên dùng thiết bị định tuyến khác để tới thiết bị đích. Thông điệp này có thể chỉ được dùng khi trạm nguồn ở trên cùng một mạng với cả hai thiết bị định tuyến.
- Kiểm tra các trạm ở xa: một trạm có thể gửi một thông điệp ICMP "Echo" để kiểm tra xem một trạm có hoạt động hay không.

Sau đây là mô tả một ứng dụng của giao thức ICMP thực hiện việc định tuyến lại (Redirect):

Ví dụ: giả sử host gửi một gói dữ liệu IP tới Router R1. Router R1 thực hiện việc quyết định tuyến vì R1 là router mặc định của host đó. R1 nhận gói dữ liệu và tìm trong bảng định tuyến và nó tìm thấy một tuyến tới R2. Khi R1 gửi gói dữ liệu tới R2 thì R1 phát hiện ra rằng nó đang gửi gói dữ liệu đó ra ngoài trên cùng một giao diện mà gói dữ liệu đó đã đến (là giao diện mạng LAN mà cả host và hai Router nối đến). Lúc này R1 sẽ gửi một thông báo ICMP Redirect Error tới host, thông báo cho host nên gửi các gói dữ liệu tiếp theo đến R2 thì tốt hơn.



Tác dụng của ICMP Redirect là để cho mọt host với nhận biết tối thiểu về định tuyến xây dựng lên một bảng định tuyến tốt hơn theo thời gian. Host đó có thể bắt đầu với một tuyến mặc định (có thể R1 hoặc R2 như ví dụ trên) và bất kỳ lần nào tuyến mặc định này được dùng với host đó đến R2 thì nó sẽ được Router mặc định gửi thông báo Redirect để cho phép host đó cập nhật bảng định tuyến của nó một cách phù hợp hơn.

I.6.2. Giao thức ARP và giao thức RARP

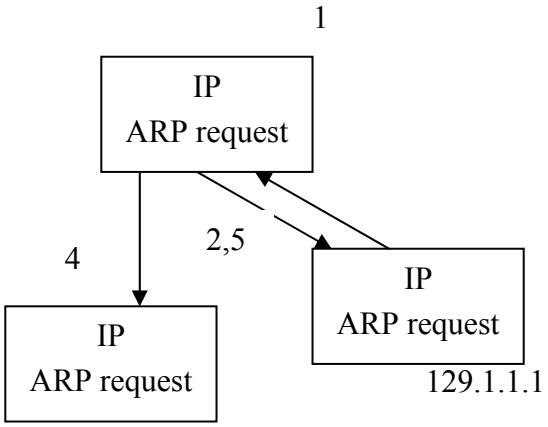
Địa chỉ IP được dùng để định danh các host và mạng ở tầng mạng của mô hình OSI, chúng không phải là các địa chỉ vật lý (hay địa chỉ MAC) của các trạm đó trên một mạng cục bộ (Ethernet, Token Ring,...). Trên một mạng cục bộ hai trạm chỉ có thể liên lạc với nhau nếu chúng biết địa chỉ vật lý của nhau. Như vậy vấn đề đặt ra là phải thực hiện ánh xạ giữa địa chỉ IP (32 bits) và địa chỉ vật lý (48 bits) của một trạm. Giao thức ARP (Address Resolution Protocol) đã được xây dựng để chuyển đổi từ địa chỉ IP sang địa chỉ vật lý khi cần thiết. Ngược lại, giao thức RARP (Reverse Address Resolution Protocol) được dùng để chuyển đổi địa chỉ vật lý sang địa chỉ IP. Các giao thức ARP và RARP không phải là bộ phận của IP mà IP sẽ dùng đến chúng khi cần.

Giao thức ARP

Giao thức TCP/IP sử dụng ARP để tìm địa chỉ vật lý của trạm đích. Ví dụ khi cần gửi một gói dữ liệu IP cho một hệ thống khác trên cùng một mạng vật lý Ethernet, hệ thông gửi cần biết địa chỉ Ethernet của hệ thống đích để tầng liên kết dữ liệu xây dựng khung gói dữ liệu.

Thông thường, mỗi hệ thống lưu giữ và cập nhật bảng thích ứng địa chỉ IP-MAC tại chỗ (còn được gọi là bảng ARP cache). Bảng thích ứng địa chỉ được cập nhật bởi người quản trị hệ thống hoặc tự động bởi giao thức ARP sau mỗi lần ánh xạ được một địa chỉ thích ứng mới.

Mỗi khi cần tìm thích ứng địa chỉ IP - MAC, có thể tìm địa chỉ MAC tương ứng với địa IP đó trước tiên trong bảng địa chỉ IP - MAC ở mỗi hệ thống. Nếu không tìm thấy, có thể sử dụng giao thức ARP để làm việc này. Trạm làm việc gửi yêu cầu ARP (ARP_Request) tìm thích ứng địa chỉ IP - MAC đến máy phục vụ ARP - server. Máy phục vụ ARP tìm trong bảng thích ứng địa chỉ IP - MAC của mình và trả lời bằng ARP_Response cho trạm làm việc. Nếu không, máy phục vụ chuyển tiếp yêu cầu nhận được dưới dạng quảng bá cho tất cả các trạm làm việc trong mạng. Trạm nào có trùng địa chỉ IP được yêu cầu sẽ trả lời với địa chỉ MAC của mình. Tóm lại tiến trình của ARP được mô tả như sau



Tiến trình ARP

1. IP yêu cầu địa chỉ MAC.
2. Tìm kiếm trong bảng ARP.
3. Nếu tìm thấy sẽ trả lại địa chỉ MAC.
4. Nếu không tìm thấy, tạo gói ARP yêu cầu và gửi tới tất cả các trạm.
5. Tuỳ theo gói dữ liệu trả lời, ARP cập nhật vào bảng ARP và gửi địa chỉ MAC đó cho IP.

Giao thức RARP

Reverse ARP (Reverse Address Resolution Protocol) là giao thức giải thích ứng địa chỉ AMC - IP. Quá trình này ngược lại với quá trình giải thích ứng địa chỉ IP - MAC mô tả ở trên, nghĩa là cho trước địa chỉ mức liên kết, tìm địa chỉ IP tương ứng.

Giao thức lớp chuyển tải (Transport Layer)

- Giao thức TCP ?

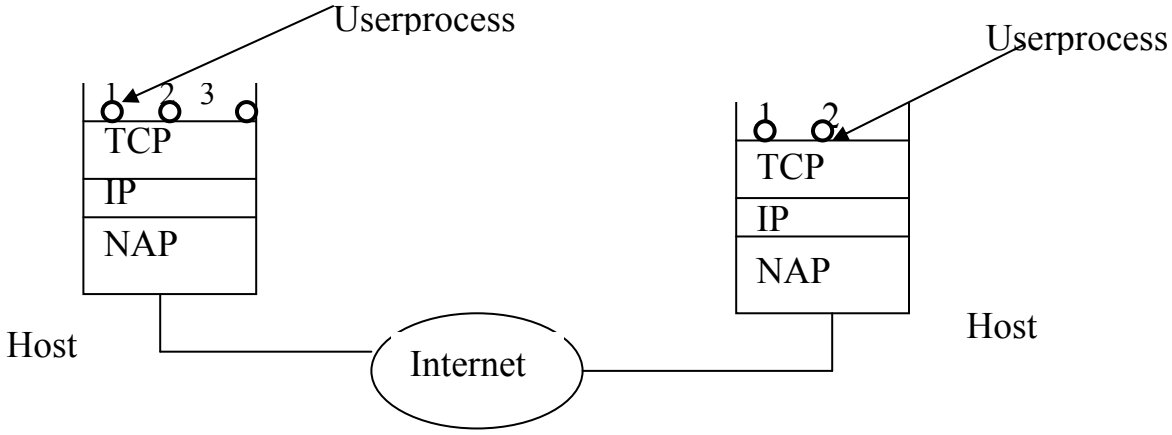
TCP (Transmission Control Protocol) là một giao thức "có liên kết" (connection - oriented), nghĩa là cần thiết lập liên kết (logic), giữa một cặp thực thể TCP trước khi chúng trao đổi dữ liệu với nhau.

TCP cung cấp khả năng truyền dữ liệu một cách an toàn giữa các máy trạm trong hệ thống các mạng. Nó cung cấp thêm các chức năng nhằm kiểm tra tính chính xác của dữ liệu khi đến và bao gồm cả việc gửi lại dữ liệu khi có lỗi xảy ra. TCP cung cấp các chức năng chính sau:

1. Thiết lập, duy trì, kết thúc liên kết giữa hai quá trình.
2. Phân phát gói tin một cách tin cậy.
3. Đánh số thứ tự (sequencing) các gói dữ liệu nhằm truyền dữ liệu một cách tin cậy.
4. Cho phép điều khiển lỗi.
5. Cung cấp khả năng đa kết nối với các quá trình khác nhau giữa trạm nguồn và trạm đích nhất định thông qua việc sử dụng các cổng.
6. Truyền dữ liệu sử dụng cơ chế song công (full-duplex).

Một tiến trình ứng dụng trong một host truy nhập vào các dịch vụ của TCP cung cấp thông qua một cổng (port) như sau:

Một cổng kết hợp với một địa chỉ IP tạo thành một socket duy nhất trong liên mạng. TCP được cung cấp nhờ một liên kết logic giữa một cặp socket. Một socket có thể tham gia nhiều liên kết với các socket ở xa khác nhau. Trước khi truyền dữ liệu giữa hai trạm cần phải thiết lập một liên kết TCP giữa chúng và khi kết thúc phiên truyền dữ liệu thì liên kết đó sẽ được giải phóng. Cũng giống như ở các giao thức khác, các thực thể ở tầng trên sử dụng TCP thông qua các hàm dịch vụ nguyên thuỷ (service primitives), hay còn gọi là các lời gọi hàm (function call).



NAP: Network Access Protocol

Cổng truy nhập dịch vụ TCP

- Thiết lập và kết thúc kết nối TCP

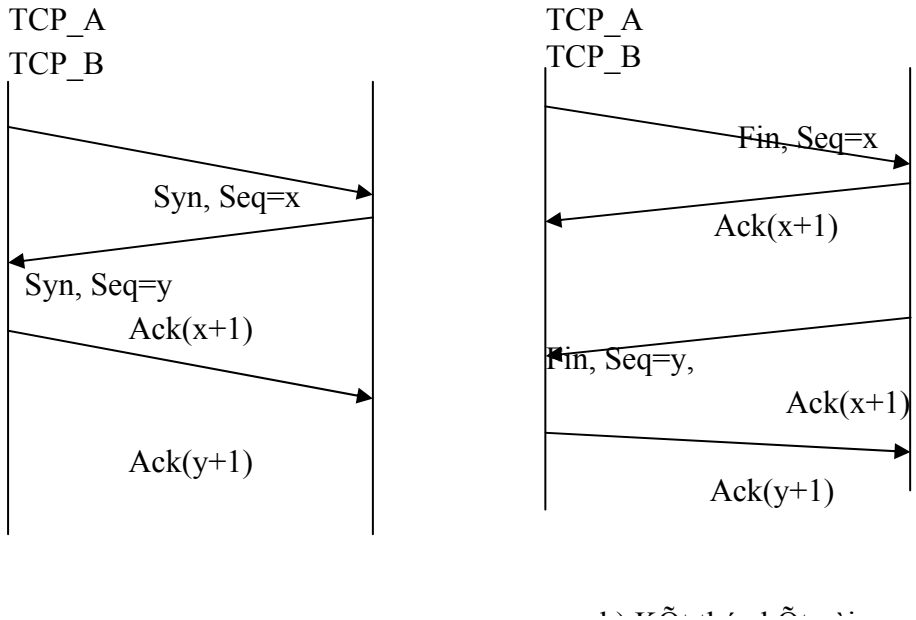
Thiết lập kết nối

Thiết lập kết nối TCP được thực hiện trên cơ sở phương thức bắt tay ba bước (Tree - way Handsake) hình 2.11. Yêu cầu kết nối luôn được tiến trình trạm khởi tạo, bằng cách gửi một gói TCP với cờ SYN=1 và chứa giá trị khởi tạo số tuần tự ISN của client. Giá trị ISN này là một số 4 byte không dấu và được tăng mỗi khi kết nối được yêu cầu (giá trị này quay về 0 khi nó tới giá trị 2^{32}). Trong thông điệp SYN này còn chứa số hiệu cổng TCP của phần mềm dịch vụ mà tiến trình trạm muốn kết nối (bước 1).

Mỗi thực thể kết nối TCP đều có một giá trị ISN mới số này được tăng theo thời gian. Vì một kết nối TCP có cùng số hiệu cổng và cùng địa chỉ IP được dùng lại nhiều lần, do đó việc thay đổi giá trị INS ngăn không cho các kết nối dùng lại các dữ liệu đã cũ (stale) vẫn còn được truyền từ một kết nối cũ và có cùng một địa chỉ kết nối.

Khi thực thể TCP của phần mềm dịch vụ nhận được thông điệp SYN, nó gửi lại gói SYN cùng giá trị ISN của nó và đặt cờ ACK=1 trong trường hợp sẵn sàng nhận kết nối. Thông điệp này còn chứa giá trị ISN của tiến trình trạm trong trường hợp số tuần tự thu để báo rằng thực thể dịch vụ đã nhận được giá trị ISN của tiến trình trạm (bước 2).

Tiến trình trạm trả lời lại gói SYN của thực thể dịch vụ bằng một thông báo trả lời ACK cuối cùng. Bằng cách này, các thực thể TCP trao đổi một cách tin cậy các giá trị ISN của nhau và có thể bắt đầu trao đổi dữ liệu. Không có thông điệp nào trong ba bước trên chứa bất kỳ dữ liệu gì; tất cả thông tin trao đổi đều nằm trong phần tiêu đề của thông điệp TCP (bước 3).



Quá trình kết nối theo 3 bước

Kết thúc kết nối

Khi có nhu cầu kết thúc kết nối, thực thể TCP, ví dụ cụ thể A gửi yêu cầu kết thúc kết nối với FIN=1. Vì kết nối TCP là song công (full-duplex) nên mặc dù nhận được yêu cầu kết thúc kết nối của A (A thông báo hết số liệu gửi) thực thể B vẫn có thể tiếp tục truyền số liệu cho đến khi B không còn số liệu để gửi và thông báo cho A bằng yêu cầu kết thúc kết nối với FIN=1 của mình. Khi thực thể TCP đã nhận được thông điệp FIN và sau khi đã gửi thông điệp FIN của chính mình, kết nối TCP thực sụ kết thúc.

Chương II

Công nghệ DataSocket

I. Giới thiệu về công nghệ DataSocket.

Xây dựng hệ thống đo lường và chuyển dữ liệu đo lường cùng các thuộc tính của chúng với tốc độ cao qua mạng Internet(TCP/IP) hiện là bài toán đuợc nhiều lĩnh vực quan tâm. Nó thường là môi trường thực nghiệm môi trường ảo, thường là hệ thống được xây dựng với các thiết bị đo ảo(VI), các hệ thống này thường xử dụng mô hình hệ thống đo lường phân tán kết hợp không chặt. Các hệ thống đo này cho phép dễ dàng cấu hình lại hệ thống, phối hợp các thành phần của hệ thống để thực hiện một phép đo yêu cầu.

Sự phát tán dữ liệu qua mang Internet(TCP/IP) có thể thục hiện theo các phương pháp sau:

- Mạng LAN.
- Mạng điện thoại công cộng(PSTN).
- xDSL.
- Wireless.
- Leased Line

Trên thế giới có nhiều hãng phát triển và trợ giúp các hệ thống này, nhất là hãng National Instruments, hãng này đã phát triển các điều khiển ActiveX dành cho thu thập dữ liệu đo lường, phát triển giao diện, xử lí dữ liệu và phát tán dữ liệu đo lường với tốc độ cao qua mạng Internet dựa trên cơ sở công nghệ DataSocket. Với các thư viện đó người sử dụng dễ dàng phát triển các ứng dụng đo lường và điều khiển qua mạng TCP/IP với sự tích hợp mạnh mẽ của công nghệ WEB, nó cho phép người sử dụng truy cập dữ liệu đo lường không khác gì truy cập các trang Web thông thường, đồng thời điều đó cũng cho cơ hội dễ dàng tích hợp dữ liệu đo lường với các công nghệ khác.

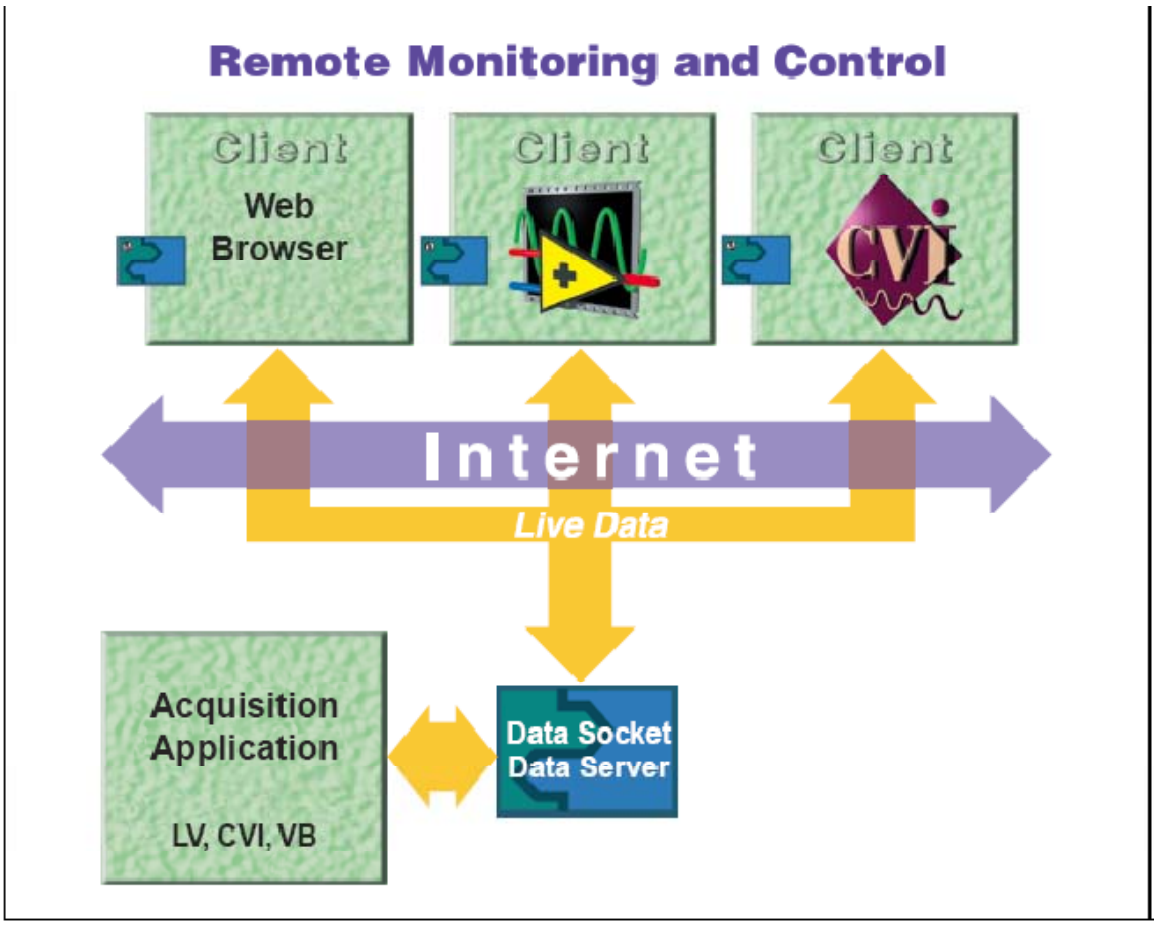
1. DataSocket là gì ?

DataSocket là một công nghệ lập trình mới dựa trên cơ sở chuẩn công nghiệp TCP/IP, để làm đơn giản hóa việc trao đổi dữ liệu giữa các ứng dụng

khác nhau trên một máy tính hoặc giữa những máy tính được kết nối với nhau qua mạng. DataSocket thực hiện một giao diện lập trình có hiệu năng cao và dễ sử dụng, cho phép thiết kế chia sẻ và phát sinh dữ liệu Online trong các hệ thống đo lường và tự động hóa.

Vì DataSocket là công cụ lập trình mới nên nó được sử dụng trong nhiều ứng dụng khác nhau như xây dựng mạng liên kết một cách thống nhất và hiệu quả giữa các phòng thí nghiệm của các cơ quan nghiên cứu, các trường đại học và các trung tâm đào tạo trong nước cũng như quốc tế với nhau. Từ đó tăng khả năng hợp tác nghiên cứu khoa học, chia sẻ tài nguyên, tiết kiệm đáng kể các thiết bị khoa học, hệ thống thí nghiệm đắt tiền hiện nay. Khi áp dụng công nghệ DataSocket chúng ta có thể khai thác được hết tính năng cũng như công suất của các hệ thống thí nghiệm đa năng nhưng lại đặt ở các trung tâm có vị trí xa nhau.

- **Các đặc trưng của DataSocket:**
    - Đọc và viết dữ liệu giữa nhiều đích và nhiều nguồn dữ liệu khác nhau.
    - Các nguồn và đích dữ liệu được chỉ ra thông qua các URL giống như truy cập trang WEB bằng trình duyệt WEB.
    - Giao diện lập trình đơn giản, độc lập cho phép truy cập tới các Server File, FTP, HTTP, OPC.
    - Giao thức truyền DataSocket(DSTP) là giao thức riêng của DataSocket, cho phép trao đổi dữ liệu với mọi kiểu dữ liệu thông qua các DataSocket Server, thậm trí cho phép truyền cả các Frame ảnh Online(Chuyển ảnh về dạng mảng và truyền) và tiến nói.
- **Mô hình phát tán dữ liệu dùng DataSocket**



2. Các thành phần của công nghệ DataSocket.

- Công nghệ DataSocket gồm 2 phần.
  - DataSocket API.
  - DataSocket Server.

2.1. DataSocket API.

- DataSocket API là một giao diện để giao tiếp với nhiều kiểu dữ liệu từ nhiều ngôn ngữ thông qua mạng.
- DataSocket API sử dụng như một điều khiển ActiveX, nó là một thư viện của LabWindows/CVI và cho phép nhúng vào các môi trường lập trình ứng dụng khác nhau như VB, VC^{++}, LabVIEW.
- **Cơ chế hoạt động của DataSocket API**: Tự động thực hiện chuyển dữ liệu đo lường thành một luồng các Byte mà có thể gửi được qua mạng đến địa chỉ đích, các ứng dụng DataSocket phía nhận sẽ chuyển

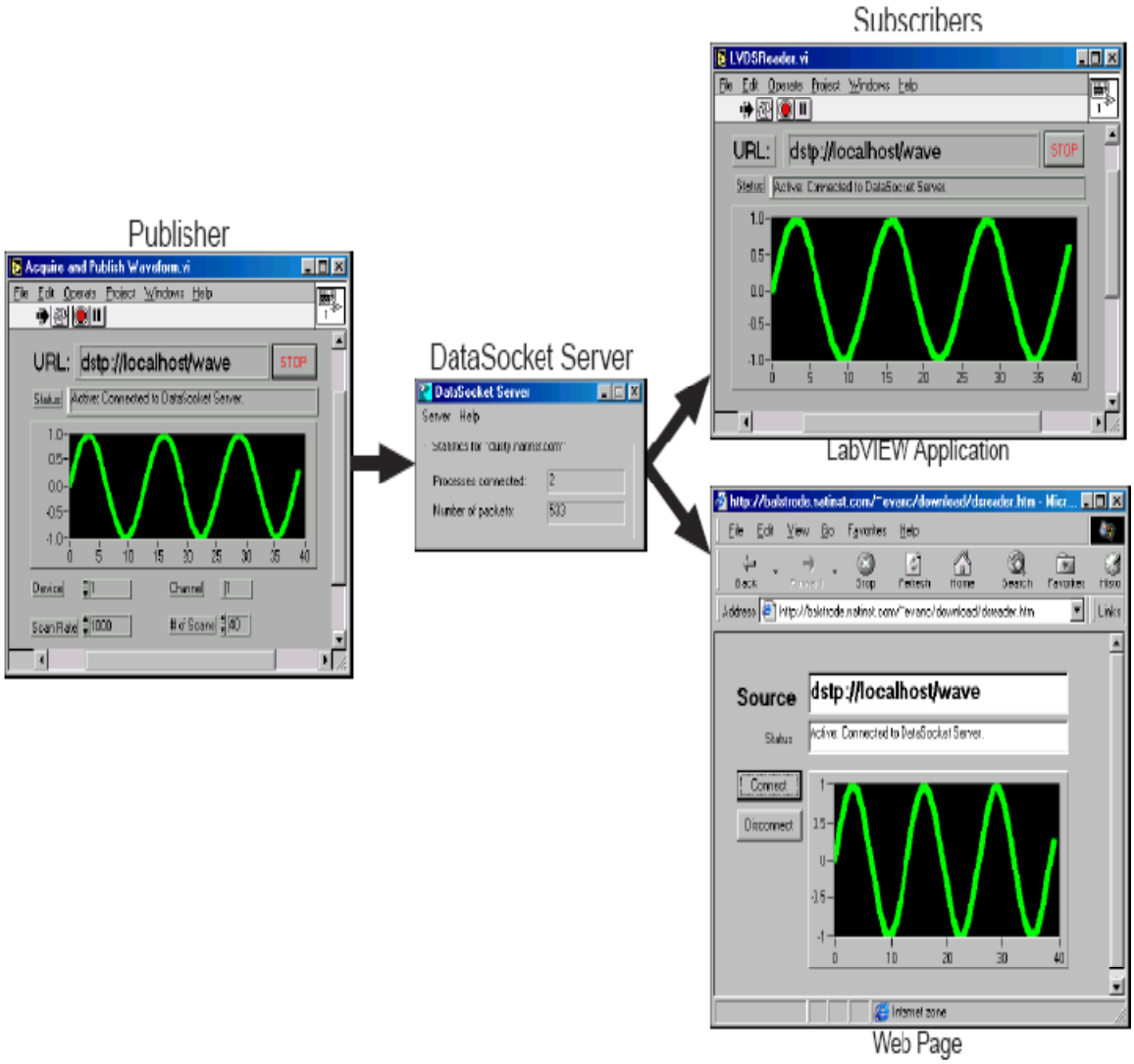
luồng Byte dữ liệu đó về dạng gốc của nó, sự chuyển đổi tự động này làm đơn giản vấn để phát triển các ứng dụng trên mạng.

- **DataSocket gồm 4 thao tác cơ bản**: Open(Mở), Read(Đọc), Write(Ghi), Close(Đóng), cho phép bạn mở kênh dữ liệu, đọc hoạc viết dữ liệu qua kênh đó và đóng kênh dữ liệu khi kết thúc. Ta có thể sử dụng DataSocket API cùng một số chương trình để đọc dữ liệu từ: HTTP Servers, FTP Servers, Local Files, DSTP Servers.

2.2. DataSocket Server.

- DataSocket là một Modul phần mềm độc lập với DataSocket API dùng để quảng bá dữ liệu đo Online qua mạng Internet tới các Client từ xa với tốc độ cao. DataSocket Server đơn giản hóa truyền thông qua Internet với giao thức lập trìnhTCP/IP, nó tự động quản lí các kết nối của Client.
- Hệ thống thực hiện phát tán dữ liệu qua mạng sử dụng DataSocket gồm 3 thành phần:
    + Bộ phận xuất bản dữ liệu(Publisher).
    + DataSocket Server
    + Bộ phận nhận dữ liệu(Subscriber).

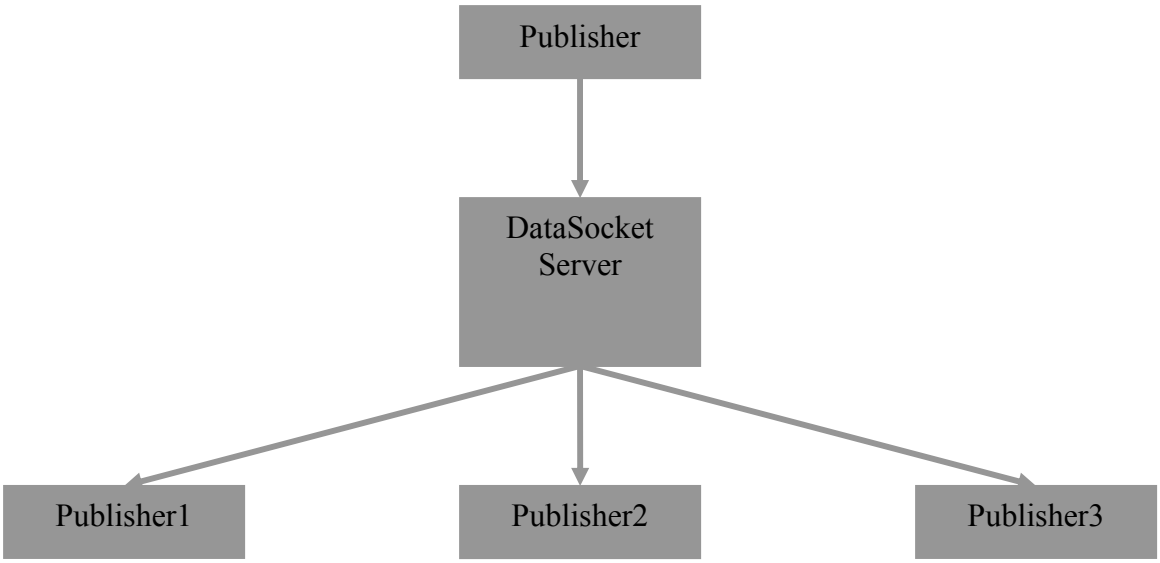
- Bộ phận xuất dữ liệu(Publisher): Sử dụng DataSocket API để viết dữ liệu thu được từ các ứng dụng thu nhập dữ liệu(Các thiết bị đo, các hệ thu thập dữ liệu...) tới Server.
- Các ứng dụng nhận dữ liệu(Subscriber): Sử dụng DataSocket API để đọc dữ liệu từ phía Server, cả ứng dụng xuất và ứng dụng nhận đều là các Client của DataSocket Server. Cả 3 thành phần để quảng bá dữ liệu có thể nằm trên cùng một máy hoạc trên các máy tính khác nhau.



- DataSocket Server:

+Khả năng chạy DataSocket Server trên các máy khác nhau cải tạo đáng kể hoạt động và khả năng an toàn của các hệ thống đo lường vì nó được cách li qua mạng máy tính.

+ DataSocket Server là một giải pháp dễ sử dụng.



- Hạt nhân cơ bản để phát triển ứng dụng trong các môi trường khác nhau của công nghệ DataSocket là các đối tượng ActiveX CWDataSocket và CWData.
- Đối tượng CWDataSocket là thành phần cho phép kết nối các nguồn dữ liệu khác nhau để đọc dữ liệu từ nguồn và viết dữ liệu tới đích. CWDataSocket lưu giữ dữ liệu trong các đối tượng CWData.

CWDataSocket		
Properties	Methods	Events
AccessMode ActualURL AutoConnect Data	AboutBox Connect ConnectTo Disconnect	OnDataUpdated OnStatusUpdated

DataUpdated LastError LastMessage Status StatusUpdated URL	SelectURL Update	

Các thuộc tính, phương pháp, sự kiện của CWDataSocket.

- Đối tượng CWData giữ các giá trị và thuộc tính gắn với các giá trị dữ liệu đó thông qua các thuộc tính Data.

CWData		
Properties	Methods	Events
Value	CopyFrom DeleteAttribute GetAttribute GetAttributeNames HasAttributeReset SetAttribute	

Các thuộc tính, phương pháp và sự kiện của CWData.

3. Giao thức DSTP

- Giao thức DSTP là giao thức quan trọng trong việc phát tán dữ liệu tốc độ cao qua mạng Internet.

- **<u>Các đặc điểm của giao thức DSTP.</u>**

- DSTP là một giao thức lớp ứng dụng để truyền dữ liệu tới nơi đọc dữ liệu từ một Server DSTP goi là DataSocket Server. DSTP được thực hiện trên nền TCP/IP và cung cấp truyền thông hướng kết nối giữa Server và Client. Trong đó phía Client sẽ duy trì phiên truyền thông với phía Server. Trong phiên truyền thông đó phía Server sẽ giữ các thông tin kết nối của phía Client. Máy Client cung cấp dữ liệu đo tới phía Server và được coi như những bộ xuất bản dữ liệu (Publisher)

hay là nơi viết dữ liệu. Còn phía nhân dữ liệu được coi là nhưng nơi đọc hay thành viên nhận dữ liệu(Subscribes => Thành viên thuê dữ liệu).

- Các thành phần.

Trong hệ thống DSTP gồm 3 thành phần:

- DataSocket Server.
- Nơi xuất bản dữ liệu.
- Nơi nhận dữ liệu.

Nơi xuất bản dữ liệu: Thu thập dữ liệu từ thiết bị thu thập dữ liệu cục bộ hoặc từ xa và gửi nó tới máy Server. Máy Server có thể được định trên cùng máy cục bộ hay trên máy từ xa qua mạng Internet.

Nơi nhận dữ liệu: Nhận dữ liệu từ máyServer. Đối với các ứng dụng phức tạp yêu cầu nhiều hơn một Publisher và nhiều hơn một Server.

- Nhận biết dữ liệu đo lường trên một Server

Sử dụng URL để nhận biết vị trí của dữ liệu đo đường trên Server. Một URL DSTP chỉ thị tên của Server DataSocket và đường dẫn truy cập đối với một mục dư liệu đo lường cụ thể. Các Publisher và Subscriber cần chỉ tới mục dữ liệu bởi cùng một URL.

- **<u>Cách sử dụng.</u>**

- Thiết lập một phiên làm việc

Người sử dụng tương tác với DSTP bằng cách sử dụng giao diện lập trình úng dụng DataSocket API trong một ứng dụng phần mềm National Instruments.

Để thiết lập một kết nối tới một DataSocket Server sử dụng API DataSocket người sử dụng chỉ cần viết tới URL, sau đó Client sẽ thông qua một vài bước để thiết lập một kết nối DSTP.

Các bước thực hiện kết nối DSTP:

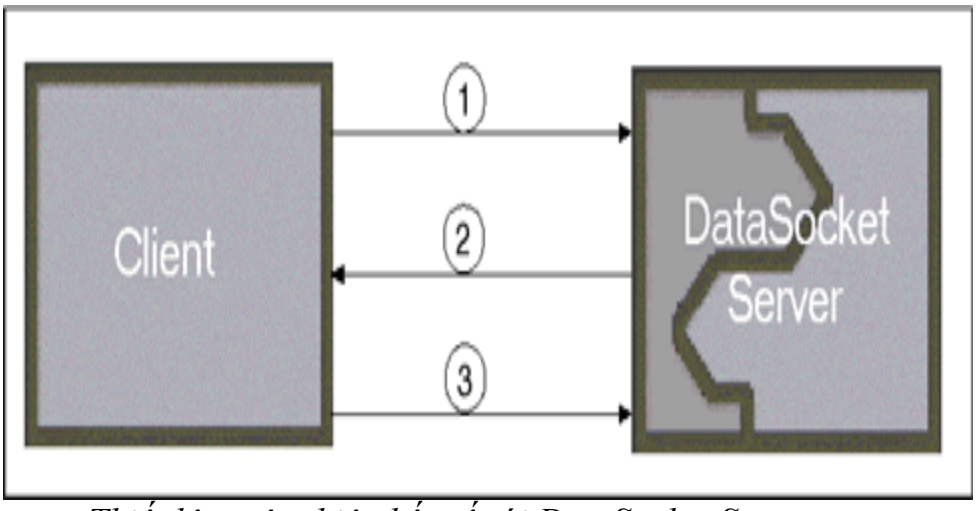
Bước 1: Để thiết lập một phiên với Server:

Client gửi thông điệp “Request to log on” tới Server. Yêu cầu cũng bao gồm số phiên bản DSTP đối với phiên kết nối.

Bước 2: Nếu Server chấp nhận kết nối nó sẽ gửi lại một thông báo để xác nhận số phiên bản DSTP.

Bước 3: Client gửi lại một yêu cầu để kết nối tới một URL cụ thể trên Server.

Các URL thể hiện dữ liệu đo lường cụ thể trên DataSocket Server được quy chiếu như là các mục DataSocket. DataSocket Server sử dụng cùng kết nối TCP đối với tất cả các mục tồn tại trong cùng không gian xử lý đó.



Thiết lập một phiên kết nố tới DataSocket Server.

- Truyền và nhận dữ liệu.

Để truyền dữ liệu tới Server , Client gửi toàn bộ phần đầu một thông điệp tới Server để viết dữ liệu được đóng gói trong thông điệp như là một giá trị mới đối với dữ liệu. Thông điệp cũng bao gồm URL nhậ dạng dữ liệu trên Server, để yêu cầu dữ liệu từ Server, Client gửi thông điệp yêu càu giá trị dữ liệu gần đây nhất. Thông điệp cũng gồm URL của dữ liệu yêu cầu từ Server. Client nhân dữ liệu tữ Server qua các yêu cầu dữ liệu rõ ràng hoạc qua

một kết nối cập nhật tự động, Server gửi dữ liệu cập nhật tới Subssribes được lập trình để cập nhật tự động ngay khi Server nhận giá trị mới từ Publisher.

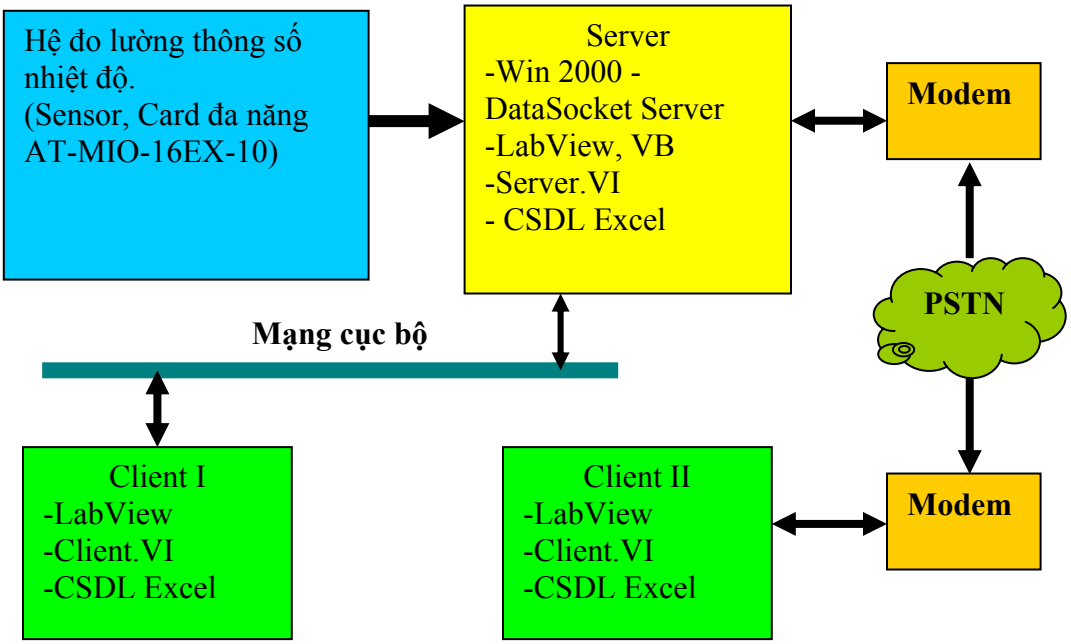
- Kết thúc một phiên.

Để kết thúc kết nối, Client gửi yêu cầu thôi kết nối tới Server.

Chương III

Thử nghiệm phát tán dữ liệu qua mạng TCP/IP sử dụng DataSocket

1. Mô hình hệ thử nghiệm



Hình 6. Mô hình hệ thống thử nghiệm

a) Mô tả phần cứng

Phần cứng thử nghiệm bao gồm:

**) Mạng LAN gồm 2 máy tính với cấu hình:*

+ CPU: P4-2GHz

+RAM: 256MB

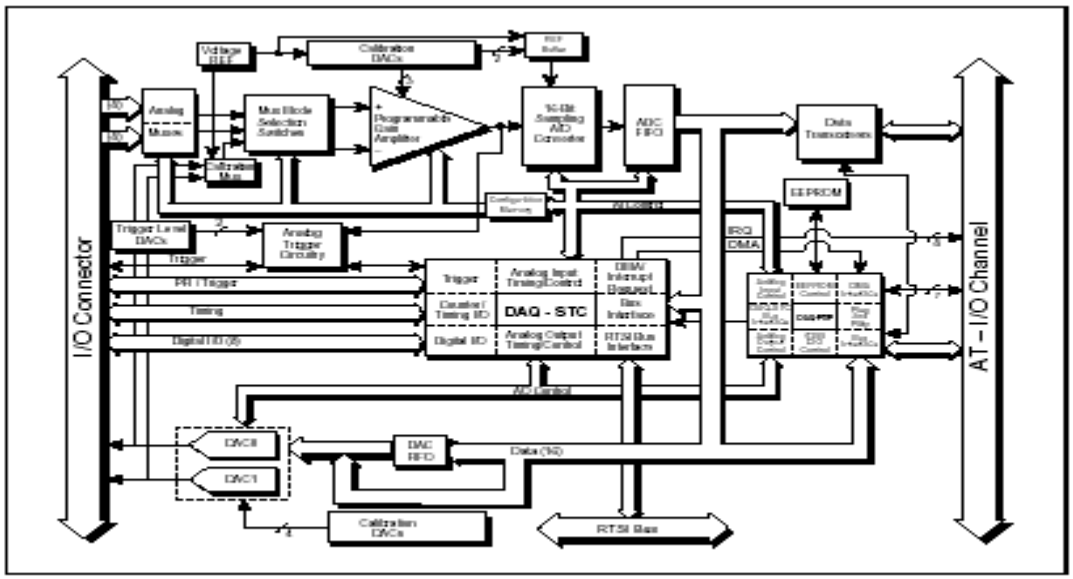
+HDD: 20GB

+Card mạng, cable mạng

+Monitor, keyboard, mouse

**) Card AT-MIO-16XE-10*

Đây là card đa năng, bao gồm: 16 kênh AI, 2 kênh AO, 8 kênh DIO và các mạch đếm, các mạch phát xung. Sơ đồ khối của card thể hiện như hình vẽ 7



Hình 7. Sơ đồ khối card AT-MIO-16XE-10

- Bộ ghép nối SCB-68 và cable
- Bộ chỉ thị tín hiệu số led

Bộ chỉ thị Led dùng để chỉ thị tín hiệu điều khiển số gửi từ máy từ xa qua mạng, gồm 8 kênh số DIO.

- Bộ chỉ thị tín hiệu hiện số

Bộ chỉ thị tín hiệu hiện số dùng chíp ICL7107 để chỉ thị số. Bộ này nhận tín hiệu Analog từ máy từ xa gửi đến máy cục bộ và được đưa qua bộ DAC tới bộ chỉ thị tín hiệu điện áp.

- Mạch đầu đo nhiệt độ PT100
- Sensor PT100

b) Mô tả phần mềm

Chương trình phần mềm gồm 2 chương trình: Chương trình server cài đặt trên máy chủ và chương trình client cài đặt trên các máy trạm. . Trong LabVIEW có các lệnh cho phép lập trình ứng dụng làm việc với DataSocket một cách thuận tiện, dễ dàng:

- Lệnh Select: Chọn URL DataSocket nguồn để đọc dữ liệu hoặc đích để viết dữ liệu

- Lệnh DataSocket Read: Đọc dữ liệu từ kết nối được chỉ bởi URL và trả về dữ liệu

-Lệnh DataSocket Write: Viết dữ liệu tới kết nối DataSocket được chỉ bởi URL

Các lệnh gộp dữ liệu và tách dữ liệu để truyền qua mạng. Dữ liệu qua mạng là loại dữ liệu không định kiểu Variant.

Lấy thuộc tính và giá trị gắn với dữ liệu kiểu Variant

Thay đổi hoặc tạo thuộc tính và giá trị đối với kiểu dữ liệu Variant

Chương trình sử dụng giao thức DSTP để phát tán dữ liệu qua mạng.

- Chương trình cài trên máy chủ Datasocket: Chương trình này gồm có các modul:

+Modul nhận tín hiệu vào Analog

+Modul nhận tín hiệu vào số

+Modul đưa tín hiệu ra Analog

+Modul đưa tín hiệu ra số

+Modul truyền phát tán dữ liệu đo lường qua mạng Internet

- Phần mềm này thực hiện các chức năng sau:

+ Khởi tạo các cổng, card AT-MIO-16XE-10

+ Khởi tạo DataSocket Server

+ Đặt server ở trạng thái nghe

+ Nhận tín hiệu tương tự từ các sensor, các thuộc tính của dữ liệu đo và chuyển qua mạng

+ Nhận tín hiệu số từ các mạch logic đầu vào và chuyển qua mạng

+ Nhận các tín hiệu điều khiển tương tự và số từ các client trên mạng và đưa ra chỉ thị hoặc điều khiển

+ Cho phép chuyển đổi giữa 2 chế độ Remote/Local

+ Tạo giao diện dễ dàng sử dụng

+ Cho phép dễ dàng chọn kênh và chọn thiết bị

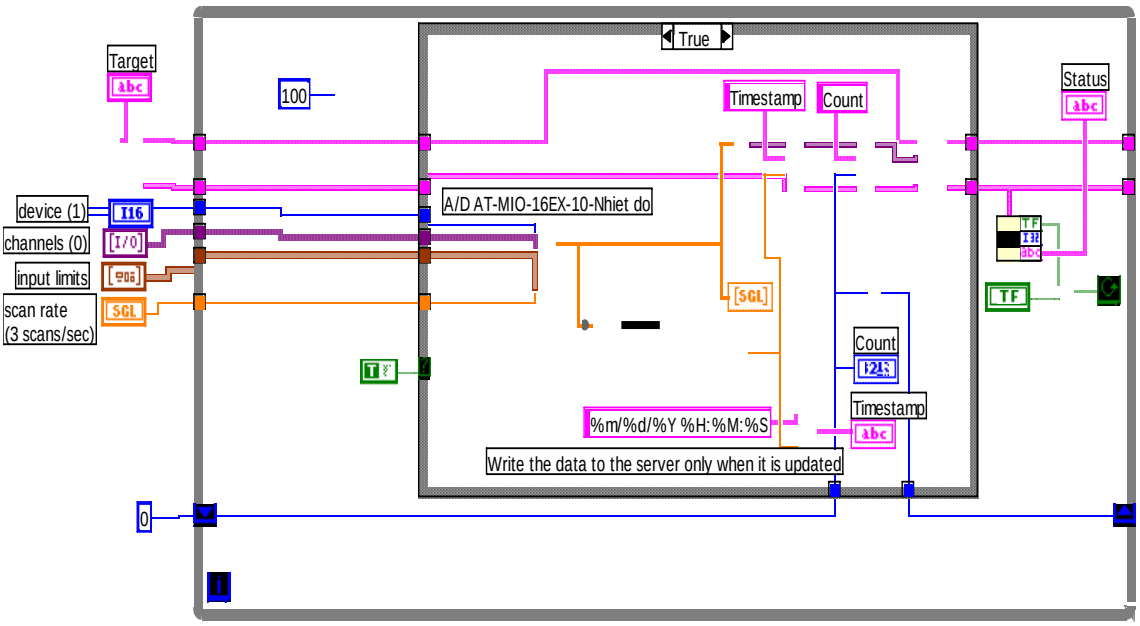
Hình 8 là giao diện người sử dụng để nhận dữ liệu từ xa và đưa ra điều khiển thiết bị qua card AT- MIO-16XE-10. Nó có 2 mode thực hiện:

- Mode cục bộ cho phép đưa ra tín hiệu điều khiển từ máy cục bộ
- Mode từ xa cho phép nhận tín hiệu điều khiển từ các client và đưa ra điều khiển thiết bị.

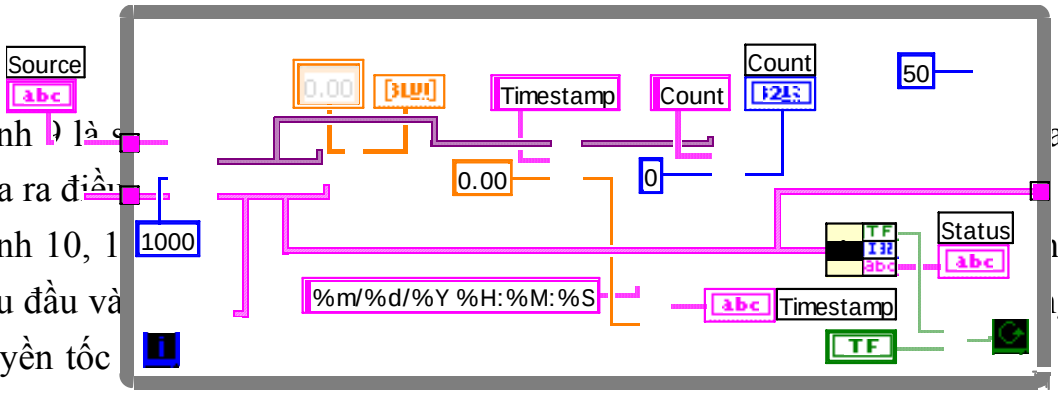
Trong giao diện này có các trường:

- Địa chỉ URL để đọc dữ liệu
- Hiển thị trạng thái kết nối
- Hiển thị đồ hoạ tín hiệu truền đến từ xa với các thuộc tính từ xa
- Trường chọn kênh tín hiệu tương tự, cho phép đến 2 kênh đưa tín hiệu ra
- Trường chọn thiết bị số
- Trường chọn kênh số
- Các nút điều khiển số để đưa ra tín hiệu số cục bộ trong mode cục bộ
- Các led chỉ thị tín hiệu số từ xa gửi tới trong mode từ xa
- Đồng hồ chỉ thị điện áp đưa ra điều khiển trong mode cục bộ hoặc từ xa

Hình x, x+1 là mã của2 modul chương trình chính server và client



Hình x



Hình 9 là s a và đưa ra điề Hình 10, 1 dữ liệu đầu và ghệ truyền tốc các trường sau:

- Trường chọn thiết bị
- Trường chọn kênh
- Trường đặt dải tín hiệu được phép nhận
- Trường chuyển đổi mode nhận tín hiệu từ thiết bị ngoài đưa vào hoặc mô phỏng(AI/Sample)
- Trường Target để nhập địa chỉ URL của đích sẽ viết dữ liệu phát tán.
- Trường Timestam để chỉ thị thuộc tính thời gian của dự liệu

Trường phát sinh tín hiệu số để phát tán qua mạng...

-Chương trình cài trên máy chủ Datasocket: Chương trình này gồm có các modul:

+ Modul nhận tín hiệu vào Analog

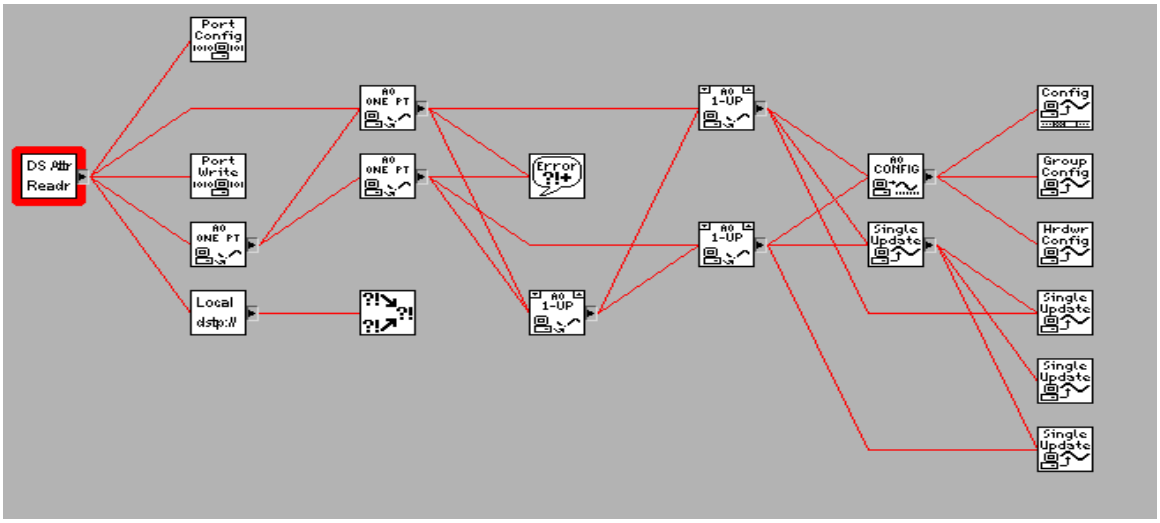
+ Modul nhận tín hiệu vào số

+ Modul đưa tín hiệu ra Analog

+ Modul đưa tín hiệu ra số

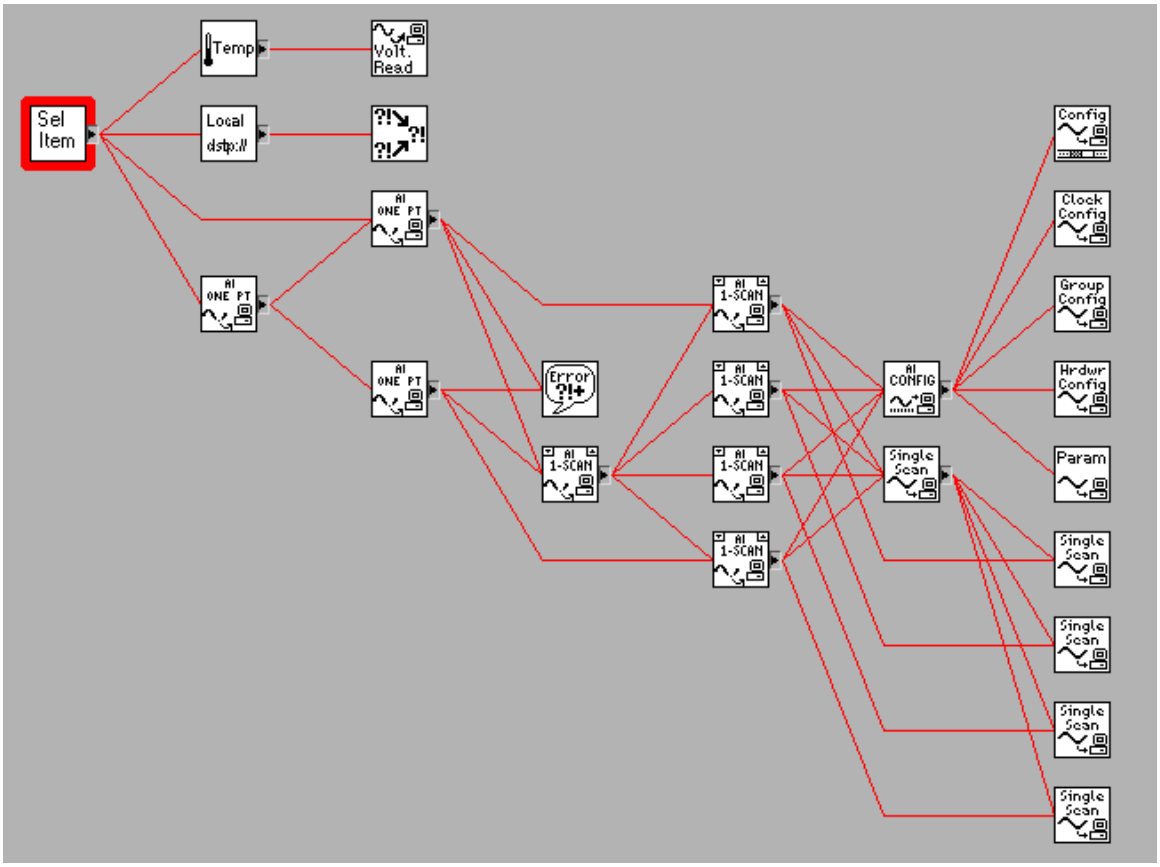
+ Modul truyền phát tán dữ liệu đo lường qua mạng Internet

Hình 8. Giao diện phần điều khiển Local/Remote



Hình 9. Lược đồ phân cấp của phần điều khiển Local/Remote

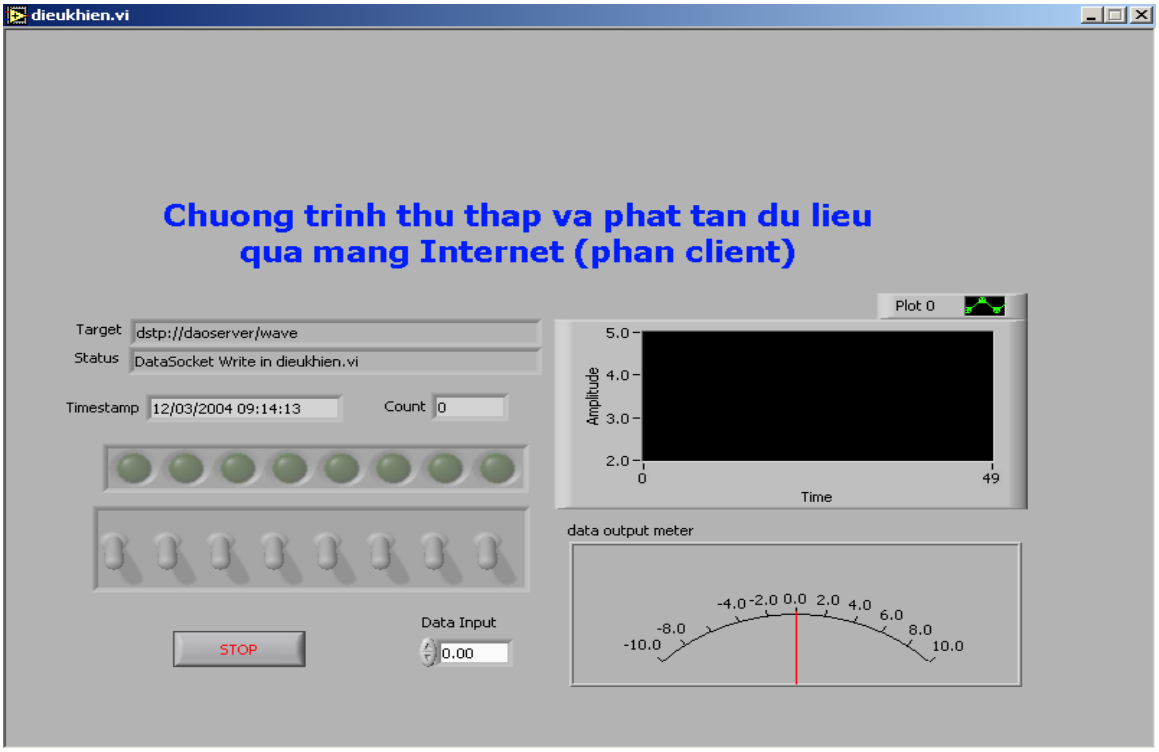
Hình 10. Giao diện phần phát tán dữ liệu đo lường qua Internet



Hình 11. Sơ đồ phân cấp của phần phát tán dữ liệu
đo lường qua mạng Internet

- Chương trình client: Chương trình client cài đặt trên các máy trên mạng, nó gồm các modul sau:
  + Modul nhận tín hiệu điều khiển
  + Modul gưi tín hiệu điều khiển đến server
  + Modul nhận dữ liệu đo lường gửi đến từ server
  + Modul nhận tín hiệu điều khiển từ người dùng và chuyển qua mạng tới server để điều khiển thiết bị trên máy từ xa
  + Hiển thị dữ liệu đồ hoạ.

  Giao diện của chương trình phần nhận tín hiệu phát tán và phần điều khiển từ xa thể hiện như hình 10, 11.



Hình 12. Giao diện chương trình trên máy client

Cả 2 phần mềm đều được viết bằng ngôn ngữ LABVIEW

2. Thử nghiệm và kết quả

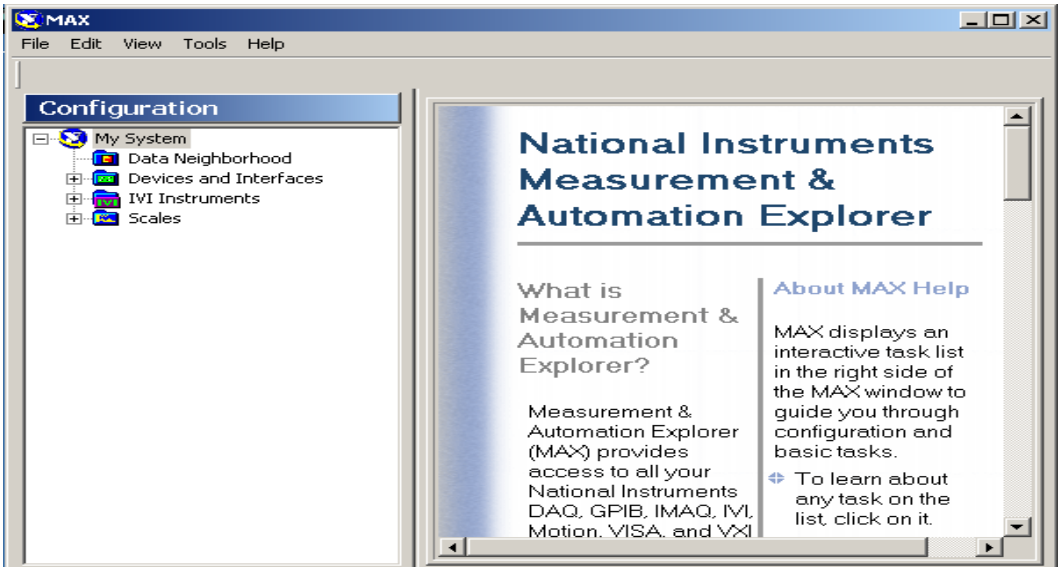
Trên sơ đồ thử nghiệm, máy chủ server có cài đặt modul phần mềm DataSocket Server, trình quản trị DataSocket Server, trình ứng dụng được xây dựng trong môi trường LabVIEW.

Hệ thống thử nghiệm gồm:

- Máy server cài Datasocket Server, Client 1 nối mạng cục bộ NT với máy Server
- Phần mềm LabVIEW được cài trên các máy tính.
- Trình ứng dụng Server.VI và Client.VI được phát triển trong môi trường LabVIEW.

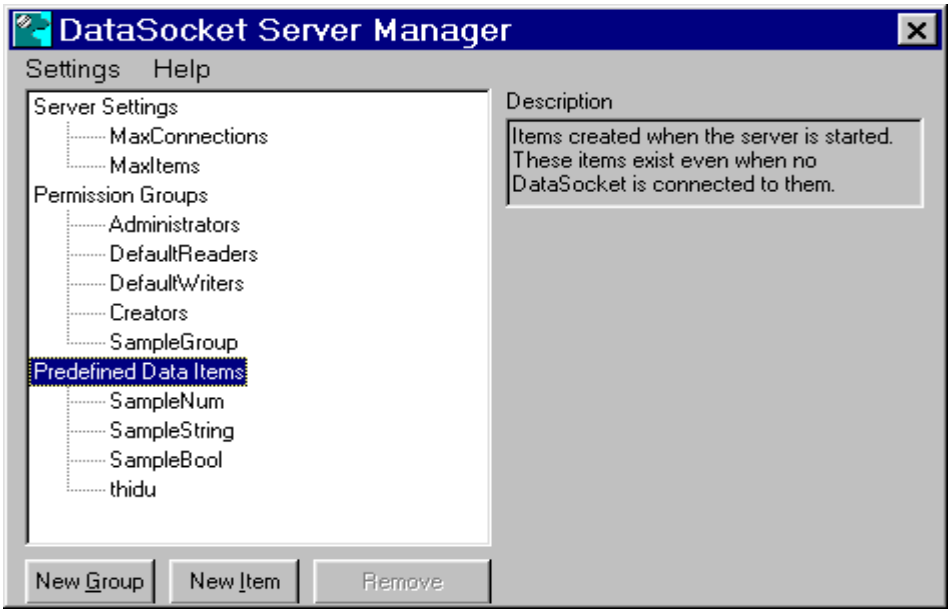
a) Cài đặt hệ thống

- Cài đặt LabView trên các hệ máy tính
- Dùng trình Measurement & Automation để khái báo thiết bị card AT-MIO-16XE-10 là loại card vào/ra đa chức năng với độ phân giải cao.



Hình 13. Trình Measurement & Automation

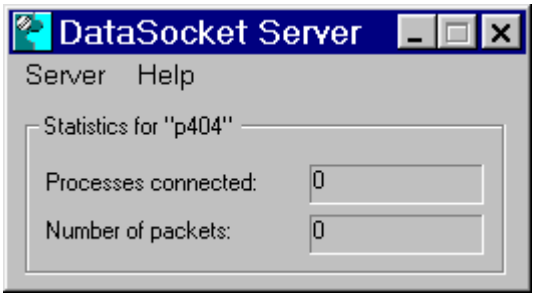
- Sử dụng trình DataSocket Server manager để khai báo các nhóm làm việc và định nghĩa các mục dữ liệu, tên các kết nối viết, đọc dữ liệu, số kết nối cực đại. Trình quản trị DataSocket Server cho phép tới hàng nghìn kết



nối thực hiện đồng thời.

Hình 13. Trình quản trị DataSocket Server

- Khởi tạo DataSocket Server, màn hình có dạng sau:



Hình 14. Màn hình kiểm thị DataSocket Server

- Khởi tạo chương trình ứng dụng phát triển trong môi trường LabVIEW Server.VI và client.VI để thu thập dữ liệu qua card ADC/DAC, đồng thời được viết qua kết nối DataSocket Server.

Thực nghiệm đã thực hiện các công việc sau:

- Phát tán dữ liệu đọc đầu vào hoặc dữ liệu mẫu cả loại tương tự với các thược tính của dữ liệu và số qua mạng Internet và chỉ thị các kết quả trên máy tính từ xa.
- Thực hiện gửi tín hiệu điều khiển từ xa tới máy để đưa ra điều khiển thiết bị cả loại số và tương tự.
- Thử nghiệm các vấn đề quản trị hệ thống phát tán dữ liệu qua mạng với trình quản trị DataSocket Server.

b) Kết quả

Hệ thống thử nghiệm cả phần cứng và phần mềm đã làm việc tốt, tin cậy và đáp ứng đựôc yêu cầu đặt ra của bài toán. Qua thử nghiệm hệ thống

hoàn toàn có thể triể khai ra diện rộng trong thực tế, nhất là trong lĩnh vực giáo dục đào tạo và sản xuất.

Chương IV

Một số kết luận đối với việc khai thác công nghệ DataSocket

Trong đồ án này, chúng tôi đã tiến hành thử nghiệm việc thu thập và phát tán dữ liệu đo lường và điều khiển qua mạng Internet với tốc độ cao và sử dụng hệ thống thiết bị đo lường ảo. Đây là một hệ thống kết hợp không chặt chẽ, nó cho phép thực hiện các phép đo và cấu hình hệ thống đo lường phân tan một cách mềm dẻo.

Đồ án đã xây dựng hệ thử nghiệm, xây dựng phần mềm làm việc trong môi trường LabVIEW đã cho kết quả thử nghiệm tốt và có thể triển khai được trong thực tế.

Đồ án sẽ được tiếp tục phát triển và thử nghiệm với các loại dữ liệu hình ảnh, âm thanh và thử nghiệm với môi trường WEB. Hệ thống đo lường sẽ cho phép bất kể người sử dụng nào thao tác trên mạng truy cập lấy dữ liệu đo lường và quan sát không khác gì truy cập trang WEB bình thường với sự tích hợp của các dịch vụ khác nhau và đa phương tiện.

Kết quả thử nghiệm thu được cho phép đánh giá việc sử dụng công nghệ DataSocket của hãng National Instruments, là một hãng nổi tiêng thế giới về các thiết bị và các hệ thống đo lường điều khiển, cho phép phát triển các phần mềm hoàn chỉnh để ứng dụng trong dạy học, ứng dụng trong liên kết các trung tâm thí nghiệm, các phòng thí nghiệm nói riêng và cho các ứng dụng truyền dữ liệu qua mạng Internet với tốc độ cao nói chung, nhằm tăng cường hợp tác nghiên cứu khoa học, chia sẻ dữ liệu, chia sẻ thiết bị thí nghiệm, mở các dịch vụ tiến hành cho thuê thiết bị thí nghiệm từ xa và khả năng tiến hành thí nghiêm từ xa. Trên cơ sở đó giảm được chi phí thiết bị, nâng cao hiệu suất sử dụng thiết bị. Nhất là ở điều kiện Việt nam chúng ta, kinh phí đầu tư thiết bị khoa học cho nghiên cứu khoa học, cho các trường Đại học , các trung tâm dạy nghề còn hạn chế mà các trung tâm đó lại cách xa nhau về địa lý.

Không những vậy, công nghệ DataSocket còn cho phép phát triển các hệ thống đo lường và điều khiển từ xa trong công nghiệp qua mạng với giao thức TCP/IP một cách dễ dàng, thuận tiện với một sự đa dạng các kiểu dữ liệu, kể cả dạng dữ liệu ảnh và âm thanh. Điều này cho phép nhiều người, nhiều lĩnh vực không chuyên nghiệp tin học vẫn có thể dễ dàng phát triển được ứng dụng của riêng mình qua mạng.

Với công nghệ này, tương lai gần đây chúng tôi sẽ tiến hành thử nghiệm chuyên sâu và hoàn chỉnh hệ thống liên kết giữa một số phòng thí nghiệm thuộc các trung tâm khoa học, và các trung tâm thí nghiệm của các Trường Đại học cũng như ứng dụng rộng rãi trong giảng dạy với hệ thống trợ giúp công nghệ đa phương tiện(MultiMedia).

Tài liệu tham khảo

[1] DataSocket Technical Overview, National Instrument Corparation, 1998

[2]CWDataSocket Overview, National Instrument Corparation,2000

[3]Using LabVIEW with TCP /IP and UDP, National Instrument Corparation,2000

[4] Using DDE in LabVIEW, National Instrument Corparation,2000

[5] LabVIEW User Manual , National Instruments Corparation,2000

[6] LabVIEW Measurements Manual, National Instruments Corparation,2000

[7] Gary –Johnson, LabVIEW Graphical Programming(Practical Application in Instrumentation and Control), McGraw-Hill Companies, Inc, 1997

Bài tiểu luận mạng máy tính

15.11.2010

- Giao thức TCP/IP được phát triển từ mạng ARPANET và Internet và được dùng như giao thức mạng và vận chuyển trên mạng Internet .

- TCP(Transmission Control Protocol) là giao thức thuộc tầng vận chuyển và IP(Internet Protocol) là giao thức thuộc tầng mạng của mô hình OSI.

- Họ giao thức TCP/IP hiện nay là giao thức được sử dụng rộng rãi nhất để liên kết các máy tính và các mạng .

- Giao thức IP là một giao thức kiểu không liên kết (*connectionless*) có nghĩa là không cần có giai đoạn thiết lập liên kết trước khi truyền dữ liệu .

- Nhiệm vụ chính của giao thức IP là cung cấp khả năng kết nối các mạng con thành liên kết mạng để truyền dữ liệu ,vai trò của IP là vai trò của giao thức tầng mạng trong mô hình OSI

- Mục đích của địa chỉ IP là để định danh duy nhất cho một máy tính bất kỳ trên liên mạng.

- Địa chỉ IP gồm 2 phần : địa chỉ mạng (*netid*) và địa chỉ máy (*hostid*).

- Mỗi địa chỉ IP có độ dài 32 bits được tách thành 4 vùng (*mỗi vùng 1 byte*), có thể biểu thị dưới dạng thập phân , bát phân , thập lục phân hay nhị phân .Cách viết phổ biến nhất là dùng ký pháp thập phân có dấu chấm (*dotted decimal notation*) để tách các vùng .

- **Mạng lớp A**:địa chỉ mạng (*netid*) là 1 byte và địa chỉ host (*hostid*) là 3 byte.

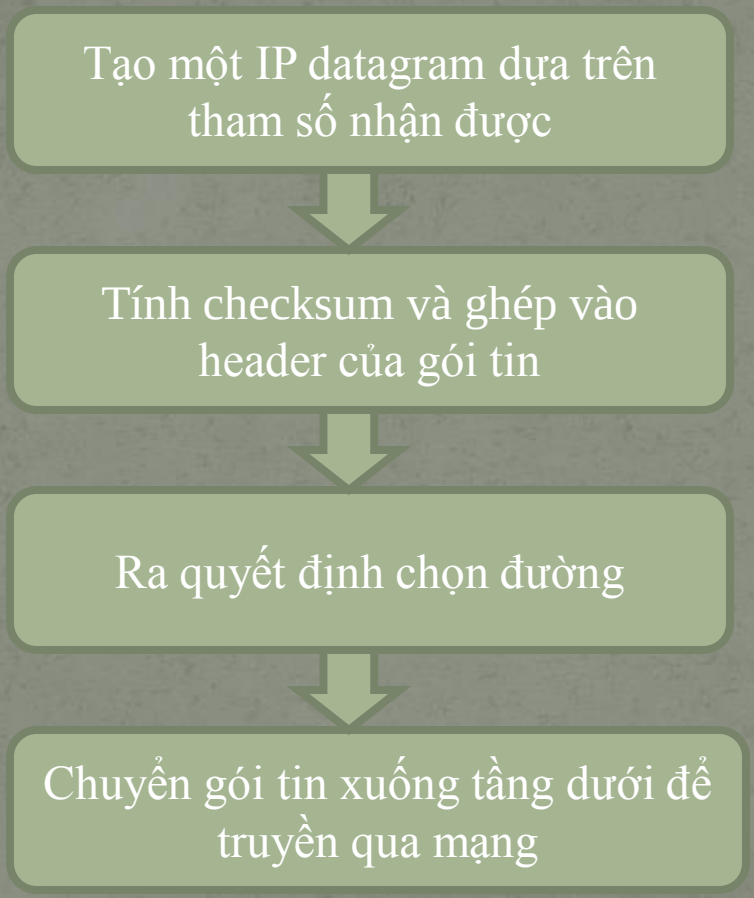
- **Mạng lớp B**:địa chỉ mạng (*netid*) là 2 byte và địa chỉ host (*hostid*) là 2 byte.

- **Mạng lớp C**:địa chỉ mạng (*netid*) là 3 byte và địa chỉ host (*hostid*) là 1 byte.

Hình 1

Ví dụ 1

[redacted]


Tạo một IP datagram dựa trên tham số nhận được
Tính checksum và ghép vào header của gói tin
Ra quyết định chọn đường
Chuyển gói tin xuống tầng dưới để truyền qua mạng

Tính checksum

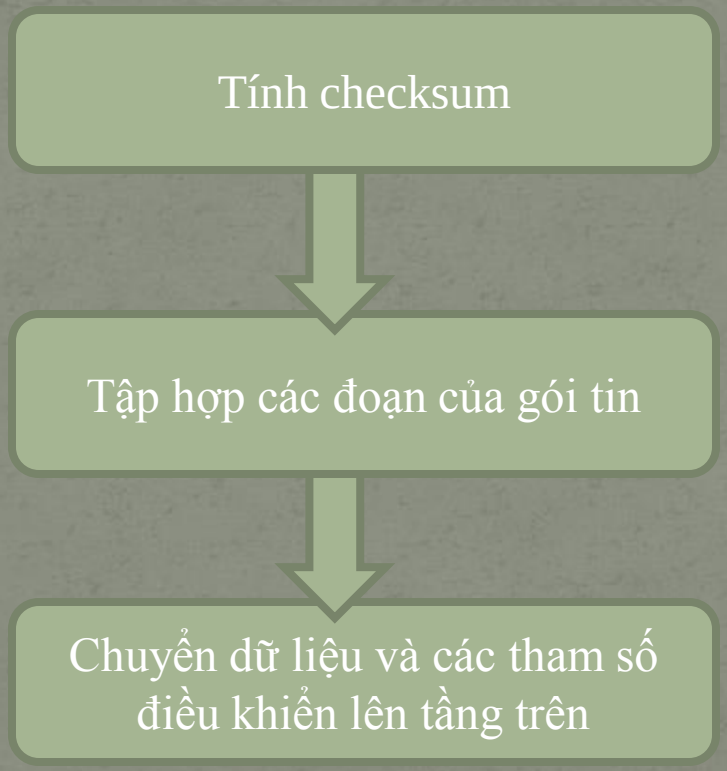
Giảm giá trị tham số Time to Live

Ra quyết định chọn đường

Phân loại gói tin

Kiến tạo lại IP header

Chuyển Datagram xuống tầng dưới


Tính checksum
Tập hợp các đoạn của gói tin
Chuyển dữ liệu và các tham số điều khiển lên tầng trên

- TCP là một giao thức "**có liên kết** " (*connection-oriented*), nghĩa là cần phải thiết lập liên kết giữa hai thực thể TCP trước khi chúng trao đổi dữ liệu với nhau .

- Một tiến trình ứng dụng trong máy tính truy nhập vào các dịch vụ của giao thức TCP thông qua một cổng (*port*) của TCP .Số hiệu cổng TCP được thể hiện bởi 2 bytes.

- Một cổng TCP kết hợp với địa chỉ IP tạo thành một đầu nối TCP/IP (*socket*) duy nhất trong liên mạng .

Bị động

- Người sử dụng yêu cầu TCP chờ đợi một yêu cầu liên kết gửi đến từ xa thông qua một đầu nối TCP/IP (*tại chỗ*).Người sử dụng dùng hàm passive Open có khai báo cổng TCP và các thông số khác (*mức ưu tiên , mức an toàn*)

Chủ động

- Người ta sử dụng yêu cầu TCP mở một liên kết với một đầu nối TCP/IP ở xa . Liên kết sẽ được xác lập nếu có một hàm Passive Open tương ứng đã được thực hiện tại đầu nối TCP/IP ở xa đó .

- Dữ liệu được gửi xuống TCP theo các khối (*block*). Khi nhận được một khối dữ liệu , TCP sẽ lưu trong bộ đệm (*buffer*).

- Nếu cờ PUSH được dựng thì toàn bộ dữ liệu trong bộ đệm được gửi , kể cả khối dữ liệu mới đến sẽ được gửi đi .

- Ngược lại cờ PUSH không được dựng thì dữ liệu được giữ lại trong bộ đệm và sẽ gửi đi khi có cơ hội thích hợp

- Ở trạm đích dữ liệu sẽ được TCP lưu trong bộ đệm gắn với mỗi liên kết .
- Nếu dữ liệu được đánh dấu với một cờ PUSH thì toàn bộ dữ liệu trong bộ đệm (kể cả các dữ liệu được lưu trữ từ trước) sẽ được chuyển lên cho người sử dụng .
- Còn nếu dữ liệu đến không được đánh dấu với cờ PUSH thì TCP chờ tới khi thích hợp mới chuyển dữ liệu với mục tiêu tăng hiệu quả hệ thống.

Hàm Close

- Yêu cầu đóng liên kết một cách bình thường .
- Khi nhận được một hàm Close TCP sẽ truyền đi tất cả dữ liệu còn trong bộ đệm thông báo rằng nó đóng liên kết .

Hàm Abort

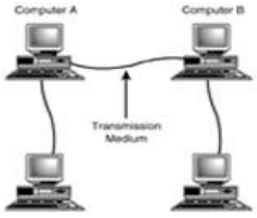
- Người sử dụng có thể đóng một liên kết bất kỳ và sẽ không chấp nhận dữ liệu qua liên kết đó nữa .
- Do đó dữ liệu có thể bị mất đi khi đang được truyền đi .TCP báo cho TCP ở biết rằng liên kết đã được hủy bỏ và TCP ở xa sẽ thông báo cho người sử dụng của mình.

Khái Niệm Về TCP/IP

K/N: TCP/IP là một hệ thống giao thức - một tập hợp các giao thức hỗ trợ việc lưu truyền trên mạng

Các giao thức TCP/IP có vai trò xác định quá trình liên lạc trong mạng và quan trọng hơn cả là định nghĩa "hình dáng" của một đơn vị dữ liệu và những thông tin chứa trong nó để máy tính đích có thể dịch thông tin một cách chính xác

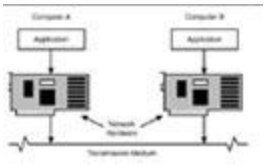
Một hệ thống mạng là tập hợp của nhiều máy tính hoặc các thiết bị tương tự, chúng có thể liên lạc với nhau thông qua một trung gian truyền tải, như ở hình 1.1.



Hình 1.1 - Một mạng cục bộ điển hình.

Trong phạm vi một hệ thống mạng, các yêu cầu và dữ liệu từ một máy tính được chuyển qua bộ phận trung gian (có thể là dây cáp mạng hoặc đường điện thoại) tới một máy tính khác. Trong hình 1.1, máy tính A phải có khả năng gửi thông tin hoặc yêu cầu tới máy tính B. Máy tính B phải hiểu được thông điệp của máy tính A và đáp lại bằng cách gửi hồi âm cho máy tính A.

Một máy tính tương tác với thế giới thông qua một hoặc nhiều ứng dụng. Những ứng dụng này thực hiện các nhiệm vụ cụ thể và quản lý dữ liệu ra và vào. Nếu máy tính đó là một phần của hệ thống mạng, thì một trong số các ứng dụng trên sẽ có thể giao tiếp với các ứng dụng trên các máy tính khác thuộc cùng hệ thống mạng. Bộ giao thức mạng là một hệ thống các quy định chung giúp xác định quá trình truyền dữ liệu phức tạp. Dữ liệu đi từ ứng dụng trên máy này, qua phần cứng về mạng của máy, tới bộ phận trung gian và đến nơi nhận, thông qua phần cứng của máy tính đích rồi tới ứng dụng. (Xem hình 1.2).



Hình 1.2 - Vai trò của một bộ giao thức mạng.

Các giao thức TCP/IP có vai trò xác định quá trình liên lạc trong mạng và quan trọng hơn cả là định nghĩa "hình dáng" của một đơn vị dữ liệu và những thông tin chứa trong nó để máy tính đích có thể dịch thông tin một cách chính xác. TCP/IP và các giao thức liên quan tạo ra một hệ thống hoàn chỉnh quản lý quá trình dữ liệu được xử lý, chuyển và nhận trên một mạng sử dụng TCP/IP. Một hệ thống các giao thức liên quan, chẳng hạn như TCP/IP, được gọi là bộ giao thức.

Căn bản về TCP/IP

TCP/IP (Transmission Control Protocol/Internet Protocol) là một bộ protocols (giao thức) được thiết kế để đạt hai mục tiêu chính:

1. Cho phép truyền thông qua các đường dây của mạng rộng (Wide Area Network - WAN).
2. Cho phép truyền thông giữa các môi trường đa dạng.

Do đó hiểu được cái gốc của các protocols nầy giúp ta hiểu đuộc sự quan trọng của chúng trong các mạng ngày nay.

Lịch sử của TCP/IP

Vào cuối thập niên 1960, cơ quan **Advanced Research Projects Agency (DARPA)** của bộ Quốc Phòng Mỹ thực hiện nhiều loạt thí nghiệm để gởi các kiện hàng dữ kiện đi lại mọi hướng (packet-switching) trên mạng. Hai mục tiêu chính của công tác nầy là:

1. Triển khai một mạng để giúp các trung tâm nghiên cứu chia sẽ các thông tin.

2. Triển khai một mạng để nối chặt chẽ các địa điểm quốc phòng trong trường hợp Mỹ bị tấn công bằng vũ khí nguyên tử.

Kết quả là bộ TCP/IP. Sau nầy **Internet Society** (Hội Internet) dùng một nhóm tư vấn mang tên **The Internet Architecture Board (IAB)** (Ban Kiến trúc Internet) để trông coi việc làm cho TCP/IP càng ngày càng hay hơn. Mỗi khi ai có sáng kiến kỹ thuật gì muốn đề nghị với Ban thì người ta xin Ban đăng lên và thông báo cho những ai quan tâm có ý kiến. Bản thông báo ấy được gọi là **Request for Comments (RFC)** (Yêu cầu cho biết ý kiến). Nếu đa số các guru về TCP/IP thấy hay thì có thể lần lần đề nghị ấy đuợc cho vào TCP/IP.

Những TCP/IP protocols và các công cụ

Như ta biết, truyền thông giữa hàng triệu computers trên Internet xãy ra được nhờ có TCP/IP protocol, một cách giao thức trên mạng rất thông dụng trong vòng các computers chạy Unix trước đây. Vì nó rất tiện dụng nên Microsoft đã dùng TCP/IP làm giao thức chính cho mạng Windows2000. TCP/IP là tập hợp của nhiều protocols, mà trong số đó có các Protocols chánh sau đây:

- **TCP (Transmission Control Protocol):** Chuyên việc nối các hosts lại và bảo đảm việc giao hàng (messages) vì nó vừa dùng sự xác nhận hàng đến (**Acknowledgement**) giống như thư bảo đảm, vừa kiểm xem kiện hàng có bị hư hại không bằng cách dùng **CRC (Cyclic Redundant Check)** , giống như có đóng khằng chỗ mở kiện hàng.
- **IP (Internet Protocol):** Lo về địa chỉ và chuyển hàng đi đúng hướng, đến nơi, đến chốn.
- **SMTP (Simple Mail Transfer Protocol):** Chuyên việc giao Email.
- **FTP (File Transfer Protocol):** Chuyên việc gởi File (upload/download) giữa các hosts.
- **SNMP (Simple Network Management Protocol):** Dùng cho các programs quản lý mạng để user có thể quản lý mạng từ xa.

- **UDP (User Datagram Protocol):** Chuyên giao các bọc nhỏ (packets) của một kiện hàng. Nó nhanh hơn TCP ví không có sự kiểm tra hay sửa lỗi. Ngược lại, nó không bảo đảm việc giao hàng.

Là Network Administrator ta nên làm quen với các công cụ chuẩn để làm việc với TCP/IP như:

- **File Transfer Protocol (FTP):** Để thử upload/download files giữa các hosts.
- **Telnet:** Cho ta Terminal Emulation (giả làm một Terminal) để nói chuyện với một Host chạy program Telnet Server.
- **Packet Internet Groper (Ping):** Dùng để thử TCP/IP configurations và connections.
- **IPCONFIG:** Để kiểm TCP/IP configuration của local host.
- **NSLOOKUP:** Dùng line command để đọc các records trong DNS (Domain Name System) database.
- **TRACERT:** Để display các khúc đường (route) dùng giữa hai hosts.

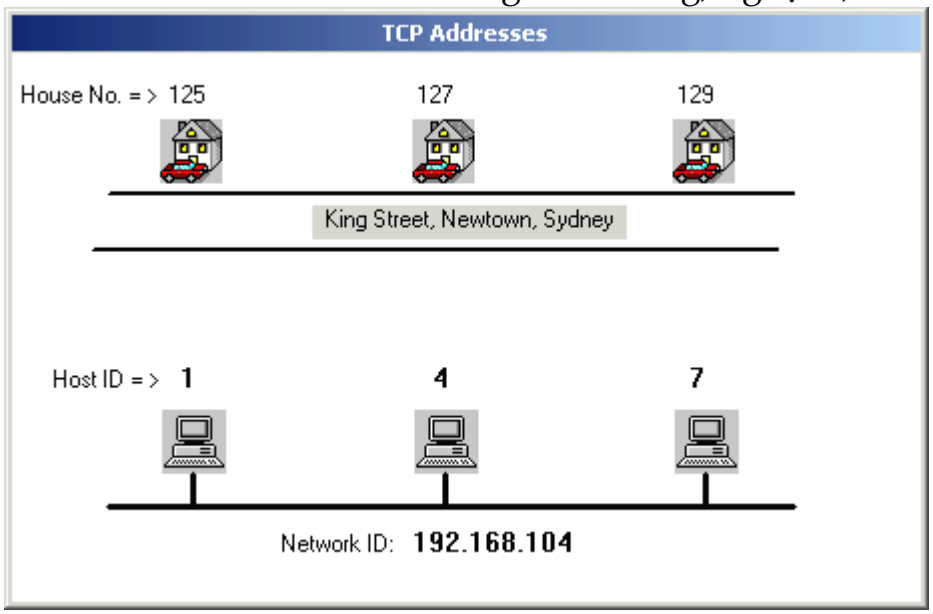
Địa chỉ TCP

Mỗi computer trên LAN/Internet phải có một địa chỉ TCP độc đáo (unique). Một địa chỉ TCP gồm có 32 bits, chia làm 4 nhóm gọi là **Octet** (có 8 bits, tức là 1 Byte dữ kiện) và được viết dưới dạng:
11000000 . 01101010 . 00000011 . 11001000
Mặc dầu trên đây là các con số mà computers thấy, nhưng đó không phải là các con số mà con người suy nghĩ. Do đó người ta thường viết nó dưới dạng gọi là **dotted decimal** (số thập phân với dấu chấm) như sau:
192.100.3.200.
Vì địa chỉ TCP như thế rất khó nhớ nên người ta quy ước dùng các tên dễ nhớ hơn như www.yahoo.com, www.vps.org, .v.v.. rồi nhờ những chỗ đặc biệt trên mạng, gọi là **Domain Name Server (DNS)** đổi các user friendly names nầy ra các địa chỉ TCP để làm việc.

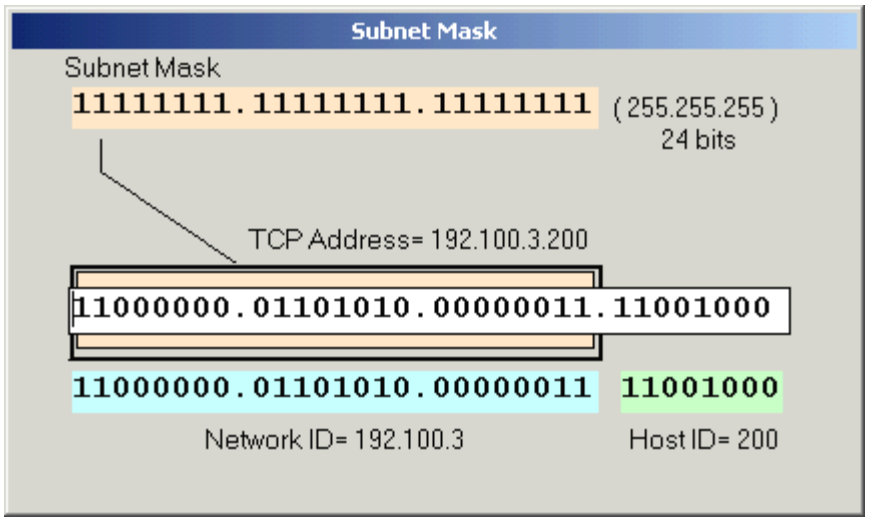
Để việc trao đổi các messages giữa các hosts trên mạng có hiệu năng, người ta thường gom các Hosts lại thành từng nhóm, gọi là Network. Mỗi Network được cho một NetworkID. Do đó mỗi địa chỉ TCP được chia ra làm hai phần:

- **Network ID** (hay Network Address): Dùng để chuyển các messages đến đúng Network (còn gọi là Subnet hay **Segment**.
- **Host ID** (hay Host Address):

Thí dụ như ba địa chỉ TCP **192.168.104.1, 192.168.104.4, 192.168.104.7** có cùng Network ID **192.168.104**.
Một Subnet của các computers giống như một con đường của những căn nhà, mỗi căn nhà có một con số để phân biệt nhưng địa chỉ của tất cả các căn nhà đều có chung tên đường, ngoại ô, thành phố .v.v. .



Con số bits , đếm từ trái qua phải, của địa chỉ TCP để dùng cho Network ID được gọi là **Subnet Mask**. Ta có thể dùng 8, 16, 24, 25 bits .v.v.. tùy ý, nhưng phải nói cho system biết ta dùng bao nhiêu bits để nó có thể tính ra phần nào trong 32 bits là của NetworkID, phần nào là của HostID.



Để biết thêm về Subnet xin hãy đọc bài Subnet Mask.

Các địa chỉ TCP được chỉ định cho mỗi Host không thay đổi nầy được gọi là **Static Address**. Khi ta dial-up Internet để connect qua **ISP (Internet Service Provider)**, computer của ta thường được ISP phát cho một địa chỉ TCP để dùng tạm trong thời gian máy ta connect trong lúc ấy. Lần tới, ta dial-up Internet sẽ được ISP cấp cho một địa chỉ TCP khác, một trong những địa chỉ TCP mà ISP đã được cơ quan đăng ký địa chỉ TCP của thế giới cung cấp.

Như thế, mỗi lần ta dùng Internet thì computer của chúng ta là một host trong mạng Internet TCP/IP của toàn thế giới. Computer ta có thể truyền thông với các hosts khác và ngược lại, người ta cũng có thể thấy và tò mò dòm ngó những gì trong computer chúng ta trong khả năng của TCP/IP. Tức là, hể mở cửa làm ăn thì coi chừng ngoại lai lén vào.

Khi tất cả các computer trên mạng dùng cho Internet được giới hạn trong vòng một cơ quan, tổ chức hay tập đoàn thì ta gọi nó là **Intranet**. Thường thường các computers trong Intranet nằm trên cùng một Local Area Network (LAN), các message được gởi đi lại với vận tốc cao (10Mbits/sec - 100Mbits/sec). Ngay cả khi một công ty có hai, ba địa điểm cách nhau, các đuờng dây viễn thông liên kết cũng có vận tốc tối thiểu là 128Kbits/sec.

Đã gọi là Intranet thì ta muốn dịch vụ Internet chỉ dành cho nội bộ và người ngoài kkông thể nào tò mò thấy được.

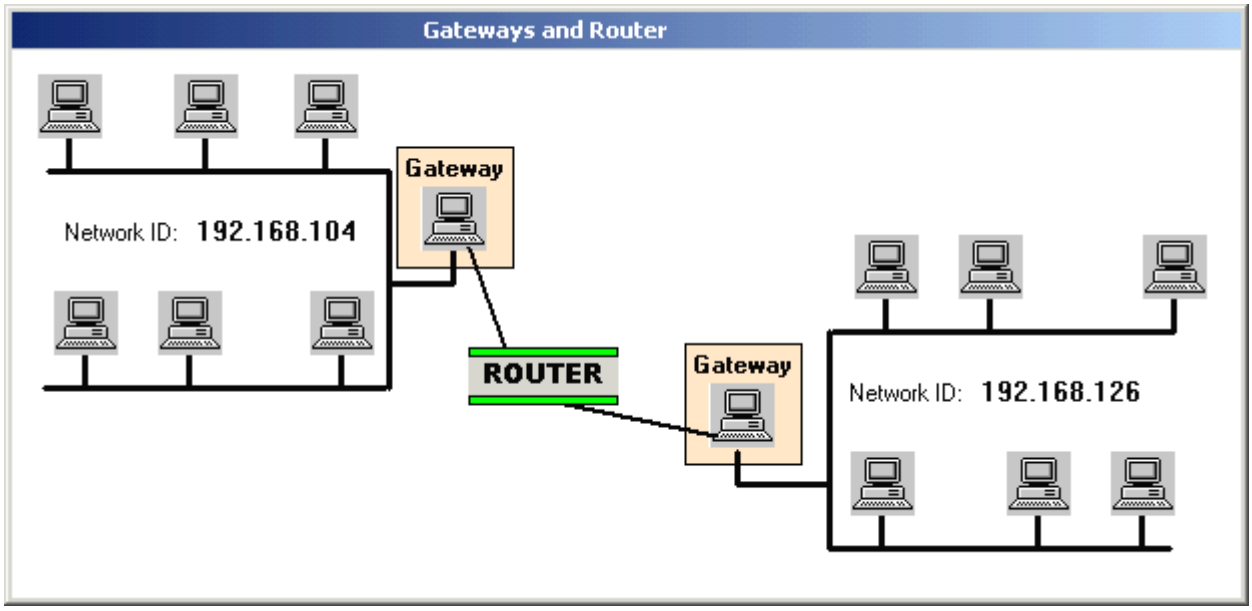
Gateway, Router và Firewall

Nếu ta không có ý định nối Network của mình với Internet bên ngoài hay Network TCP/IP nào khác thì không có gì phải lo và ở trong vòng Network riêng tư của ta, ta có thể cấp các địa chỉ TCP thoải mái.

Như đã nói ở trên, địa chỉ TCP của tất cả mọi hosts trong một Network đầu có cùng một NetworkID. Bên trong một Network, messages được gởi đi giữa các hosts rất nhanh. Nếu muốn gởi messages từ một Network nầy qua một Network khác thì phải qua một host có vị trí đặc biệt trong cùng Network gọi là **Gateway** (cổng liên hệ bên ngoài). Tỷ như một lá thư từ Đồng Tháp muốn đi ngoại quốc thì phải qua Gateway ở Thành phố HCM. Tương tợ như vậy, ở Network bên kia cũng có một Gateway để đón nhận message từ Gateway bên nầy.

Để chuyển messages giữa hai Networks ta cần phải có một dụng cụ đặc biệt, hardware hay software (một hộp hay một program), gọi là **Router** (phát âm là **rau-tơ** trong tiếng Việt).

Router là dụng cụ giúp cho hai Networks truyền thông nhau. Nó giống như một thông dịch viên vậy, có thể nói chuyện với cả hai bên. Đối với mỗi Network, Router hoạt động như thể nó là một host trong Network ấy. Hình dưới đây minh họa cách dùng Gateways và Router để nối hai Networks lại với nhau:



Trong hình trên, nếu cả hai Gateways thật ra là hai Network cards nằm trên cùng một computers chạy MSWindows2000 Server, ta có thể dùng software để làm nhiệm vụ của Router. Như thế ta khỏi phải mua một hộp Router.

Firewall (bức tường lửa) là từ dùng để nói đến phương tiện ta dùng để kiểm soát chặt chẽ sự đi lại của các messages. Ta dùng Firewall để ngăn ngừa kẻ lạ xâm phạm vào khu vực mạng TCP/IP của cơ quan ta. Như ta đã thấy, Router có thể đảm nhiệm công tác ấy. Vấn đề là nếu ta gắt gao quá thì sự đi lại rất giới hạn và không tiện lợi cho công việc làm ăn. Ngược lại, nếu ta dễ dãi quá thì không còn an toàn gì cả.

Phân chia giai cấp A,B,C

Như đã giải thích ở trên, Subnet Mask cho biết bao nhiêu bits đầu của địa chỉ TCP được dùng làm NetworkID, còn các bits còn lại là HostID. Để biểu diễn một Subnet Mask dùng 24 bits cho một NetworkID, ta có thể viết **135.100.3.200/24**. Đa số các NetworkID ta thường gặp dùng 24 bit Subnet Mask. Nhưng thật ra, người ta phân chia giai cấp các địa chỉ TCP ra làm các Classes A, B và C.

Các địa chỉ của **Class A** dùng Octet thứ nhất. Có điều người ta không

dùng bit thứ nhất, nó luôn luôn bằng 0. Do đó toàn bộ Internet chỉ có **127** Class A Networks. Dù địa chỉ **127** là một địa chỉ Class A, ta không thể dùng nó được vì nó được **reserved** (dành riêng) để thử Loopback (**Loopback Testing**) . Mỗi Class A Network có trên 16 triệu (2 lũy thừa 24) hosts. Khỏi phải nói, bây giờ ta không thể xin một Class A Network được nữa, vì các Đại Sư Huynh đã dành hết rồi. Trong số các công ty lớn ấy có General Electric, IBM, Apple, Xerox, và Đại học Columbia.

Các Networks thuộc **Class B** bắt đầu với Octet thứ nhất có values trong range **128 đến 191**. Trong Class B ta dùng 2 Octets đầu cho NetwordID. Do đó ta chỉ có 16,384 Class B Networks, mỗi Network có 65,534 (2 lũy thừa 16)hosts. Tất cả các Networks Class B đều đã bị người ta xí hết rồi. Trong số các công ty ấy có Microsoft và Exxon.

Sau cùng là **Class C** Networks bắt đầu với Octet thứ nhất có values trong range **192 đến 223** và dùng 3 Octets đầu tiên để biểu diễn NetworkID. Như thế ta có khoảng 2 triệu Class C Networks, nhưng mỗi Network chỉ có thể support 254 hosts (HostID=1 cho đến 254), HostID=255 được reserved cho Loopback testing, HostID=0 thì bất hợp lệ. Tin mừng cho chúng ta là mình còn xin một Class C network được.

Các loại Servers

Có ba thứ dịch vụ ta thường dùng nhất trên Internet. Đó là Surfing the Web (chu du ta bà thế giới từ trang Web nầy đến trang Web khác), Email và download File bằng cách dùng FTP (File Transfer Protocol).
Cho mỗi thứ dịch vụ ta dùng ở đầu kia phải có một Server (một program phục vụ) - do đó tùy theo ta đang connect với chỗ nào ở thới điểm ấy, tại chỗ cung cấp dịch vụ phải có Web server, Mail Server hay FTP Server để đáp ứng request (thỉnh cầu) của bạn.
Bạn hỏi nếu một Computer trên Internet chạy cả 3 loại Servers nói

trên thì làm sao phân biệt message nào là cho Server nào khi chúng đến cùng một địa chỉ TCP. Xin trả lời là ngoài địa chỉ TCP ra, mỗi computer còn có nhiều **Ports**, để khi ta nối với Server trên một computer ta còn cho biết Port number. Thí dụ cho Web (**WWW**) thì dùng **Port 80**, cho **FTP** thì dùng **Port 21** , .v.v.. Cách dùng các Port numbers giống giống như dùng tên của các cá nhân sống trong cùng một căn nhà khi gởi thư cho họ. Ngoài địa chỉ của căn nhà ta còn nói rõ là thư ấy cho cha, mẹ hay người con nào.
Hơn nữa, mỗi loại message còn dùng một protocol khác nhau, nên ta có thể Surf the Net, gởi/nhận Email và download/upload files cùng một lúc trên một đường dây điện thoại mà không sợ lẫn lộn. Bạn có thể tưởng tượng TCP/IP như cái protocol căn bản của Internet, rồi nằm lên phía trên là những protocols khác. Cũng giống như trong mạng bưu chính, xe hàng là căn bản của việc chuyên chở, nhưng kích thước các kiện hàng theo chuẩn lớn, nhỏ giúp người ta phân biệt các loại hàng hóa khác nhau.

MH/MĐ: MẠNG CĂN BẢN

- Bài 1: GIỚI THIỆU VỀ MẠNG MÁY TÍNH
- Bài 2: CHUẨN MẠNG VÀ MÔ HÌNH OSI
- Bài 3: GIAO THỨC TCP/IP VÀ IP ADDRESS V.4
- Bài 4: KỸ THUẬT MẠNG CỤC BỘ LAN
- Bài 5: QUẢN TRỊ TÀI KHOẢN CỤC BỘ VÀ TÀI NGUYÊN MẠNG
- Bài 6: CÔNG NGHỆ MẠNG WIRELESS LAN VÀ ADSL
- Bài 7: CHẨN ĐOÁN VÀ XỬ LÝ SỰ CỐ MẠNG
- ÔN TẬP
- BÁO CÁO ĐỒ ÁN
- THI CUỐI MÔN

TRƯỜNG CAO ĐẲNG NGHỀ CNTT iSPACE Website: http://www.ispace.edu.vn

BÀI 3: Giao thức TCP/IP và IP Address V.4

Chuẩn hóa quá trình trao đổi thông tin, dữ liệu giữa các máy tính. Định dạng cấu trúc dữ liệu và phương thức khi truyền.

- Giới thiệu TCP/IP
- Bộ giao thức TCP/IP
- Một số giao thức khác
- IP Address V.4
- Một số giao thức khác
- Xử lý một số sự cố thông dụng

iSpace

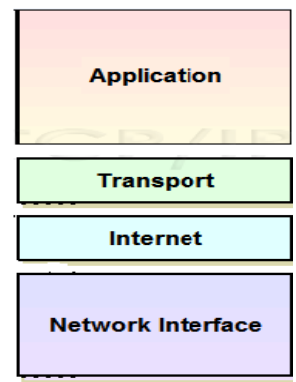
MỤC TIÊU BÀI HỌC

- Hiểu được Mô hình và chức năng các tầng của TCP/IP.
- Biết các giao thức phổ biến và các khái niệm về Port và Socket
- Hiểu được tiến trình trao đổi dữ liệu của các máy tính.
- Biểu diễn được địa chỉ IP V4
- Xử lý các sự cố kết nối mạng TCP/IP

Giới thiệu TCP/IP

TCP/IP là bộ giao thức chuẩn giúp các hệ thống (platforms) khác nhau truyền thông với nhau, là giao thức chuẩn của truyền thông Internet.

- **Mô hình kiến trúc của TCP/IP**
  - TCP/IP là chuẩn Internet
  - Được phát triển bởi US DoD (United States Department of Defense).
  - Làm việc độc lập với phần cứng mạng
  - Mô hình TCP/IP có 4 lớp : Application, Transport, Internet, Network Access

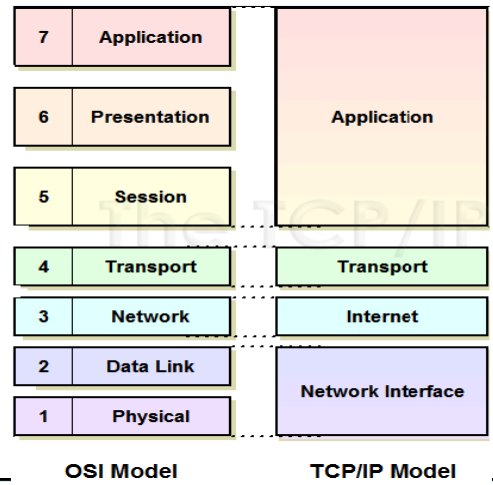


TCP/IP Model

Giới thiệu TCP/IP

Mô hình kiến trúc của TCP/IP

- Tương quan mô hình OSI và mô hình TCP/IP (So sánh OSI và TCP/IP)



Bộ giao thức TCP/IP

Bộ giao thức TCP/IP gồm 4 tầng, mỗi tầng trong mô hình TCP/IP có một chức năng riêng biệt.

Chức năng các lớp trong mô hình TCP/IP

- Application Layer (tầng ứng dụng)
  - Hỗ trợ ứng dụng cho các giao thức tầng Host to Host
  - Cung cấp giao diện người sử dụng
  - Các giao thức gồm:
    - HTTP(HyperText Transfer Protocol)
    - FTP (File Transfer Protocol)
    - Telnet
    - SMTP(Simple Mail Transfer Protocol
    - POP3

Bộ giao thức TCP/IP

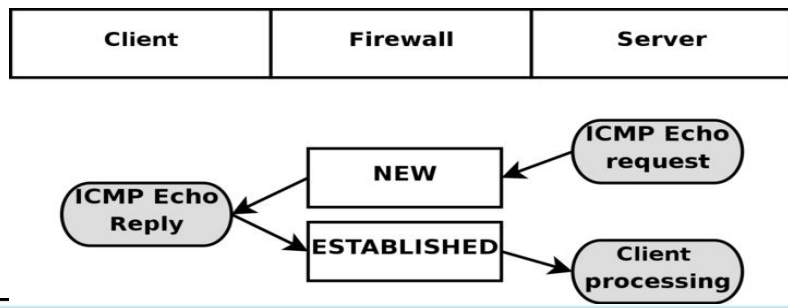
- **Chức năng các lớp trong mô hình TCP/IP**
  - Transport Layer (Host to Host-tầng vận chuyển)
    - Thực hiện kết nối giữa 2 máy trên mạng theo 2 giao thức
      - Giao thức điều khiển trao đổi dữ liệu TCP (Transmission Control Protocol)
      - Giao thức dữ liệu người dùng UDP (User Datagram Protocol)

TCP	UDP
Reliable	Unreliable
Connection-oriented	Connectionless
Segment retransmission and flow control through windowing	No windowing or retransmission
Segment sequencing	No sequencing
Acknowledge segments	No acknowledgement

TRƯỜNG CAO ĐẲNG NGHỀ CNTT iSPACE Website: http://www.ispace.edu.vn

Bộ giao thức TCP/IP

- **Chức năng các lớp trong mô hình TCP/IP**
  - Internet layer (tầng mạng)
    - IP(Internet Protocol) : Giao thức vận chuyển
    - RIP(Route Information Protocol): Tìm đường
    - ICMP : Ping (kiểm tra nối mạng)
    - ARP(Address Resolution Protocol): phân giải dia chi vật lý



iSPACE

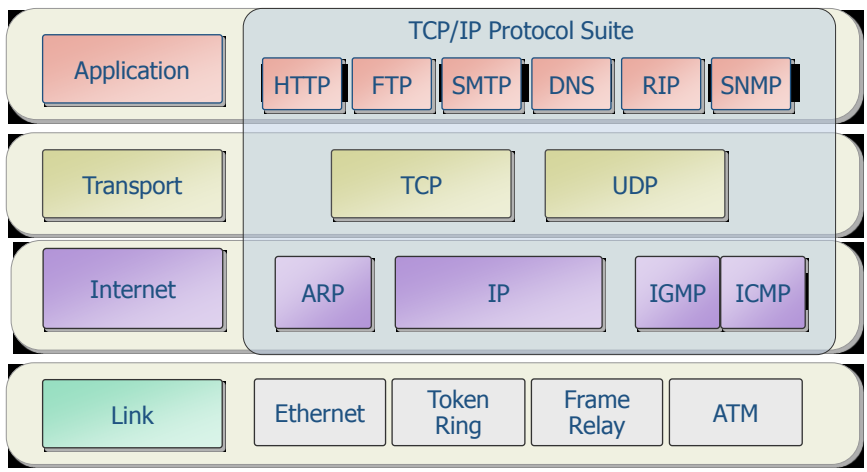
Bộ giao thức TCP/IP

- **Chức năng các lớp trong mô hình TCP/IP**
  - Network Interface Layer (tầng truy nhập mạng)
    - Tầng này nắm giữ những định dạng dữ liệu và truyền dữ liệu đến cable
    - Cung cấp các phương tiện kết nối vật lý:
      - Cable
      - Bộ chuyển đổi (Transceiver)
      - Card mạng (Nic)
    - Giao thức kết nối, giao thức truy nhập đường truyền (CSMA/CD, token ring, token bus,…)
    - Cung cấp các dịch vụ cho tầng Internet, phân đoạn dữ liệu thành các khung

TRƯỜNG CAO ĐẲNG NGHỀ CNTT iSPACE Website: http://www.ispace.edu.vn

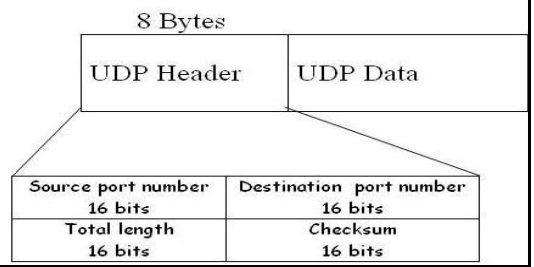
Bộ giao thức TCP/IP

- **Chức năng các lớp trong mô hình TCP/IP**



Bộ giao thức TCP/IP

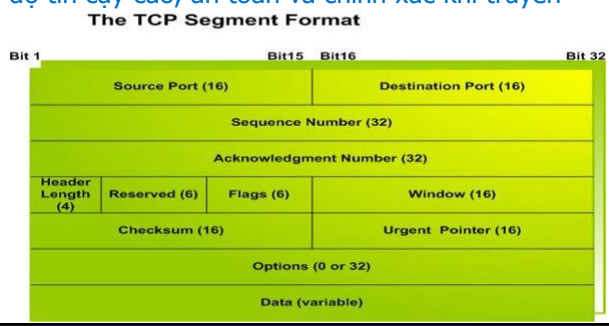
- **Một số giao thức chính**
  - Giao thức gói tin người dùng UDP (User Datagram Protocol)
    - UDP là giao thức không liên kết (Connectionless)
    - Không có độ tin cậy cao, không có cơ chế xác nhận ACK
    - Phù hợp các ứng dụng yêu cầu xử lý nhanh
      - Giao thức SNMP (Simple Network Management Protocol)
      - Voip ứng dụng UDP



TRƯỜNG CAO ĐẲNG NGHỀ CNTT iSPACE Website: http://www.ispace.edu.vn

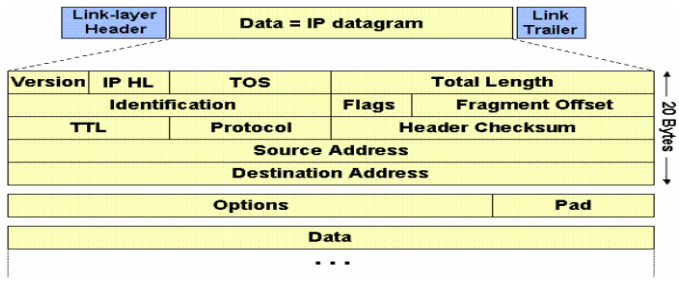
Bộ giao thức TCP/IP

- **Một số giao thức chính**
  - Giao thức điều khiển truyền TCP (Transmission Control Protocol)
    - TCP là giao thức hướng liên kết (Connection Oriented)
    - Thực thể TCP phát và thực thể TCP thu thương lượng để thiết lập 1 kết nối logic tạm thời
    - Có độ tin cậy cao, an toàn và chính xác khi truyền



Bộ giao thức TCP/IP

- **Một số giao thức chính**
  - Giao thức mạng IP (Internet Protocol)
    - IP (Internet protocol) là giao thức không liên kết
    - Truyền dữ liệu với phương thức chuyển mạch gói IP datagram
    - Định địa chỉ và chọn đường
    - IP định tuyến các gói tin bằng cách sử dụng các bảng định tuyến động



TRƯỜNG CAO ĐẲNG NGHỀ CNTT iSPACE Website: http://www.ispace.edu.vn

Bộ giao thức TCP/IP

- **Một số giao thức chính**
  - Giao thức thông báo điều khiển mạng ICMP (Internet Control Message Protocol)
    - ICMP là giao thức điều khiển ở tầng IP, sử dụng để trao đổi các thông tin điều khiển dòng dữ liệu
      - Điều khiển lưu lượng (Flow control)
      - Thông báo lỗi
      - Định dạng lại các tuyến (Ridirect router)
      - Kiểm tra các trạm ở xa
    - Các loại thông điệp ICMP
      - Thông điệp truy vấn
      - Thông điệp thông báo lỗi

sPace

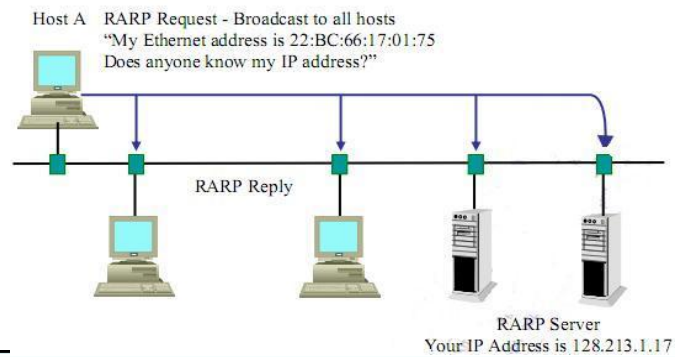
Bộ giao thức TCP/IP

- **Một số giao thức chính**
  - Giao thức phân giải địa chỉ ARP (Address Resolution Protocol)
    - IP yêu cầu địa chỉ MAC
    - Tìm kiếm trong bảng ARP
    - Nếu tìm thấy sẽ trả lại địa chỉ MAC
    - Nếu không tìm thấy, tạo gói ARP yêu cầu gởi tới tất cả các trạm
    - Tùy theo gói tin trả lời, ARP cập nhật vào bảng ARP và gởi địa MAC cho IP

TRƯỜNG CAO ĐẲNG NGHỀ CNTT iSPACE Website: http://www.ispace.edu.vn

Bộ giao thức TCP/IP

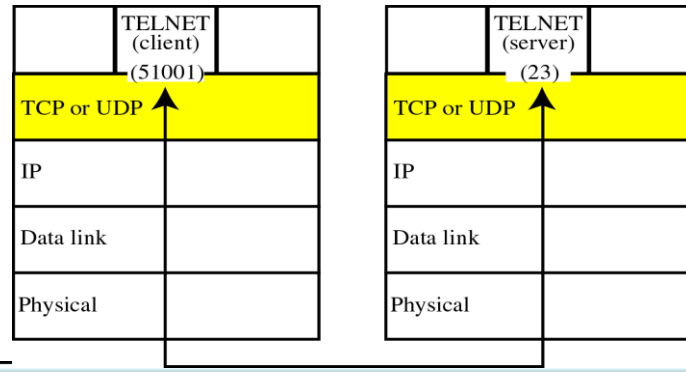
- **Một số giao thức chính**
  - Giao thức phân giải địa chỉ ngược RARP (Reverse Address Resolution Protocol)
    - Quá trình này ngược lại với quá trình ARP
    - RARP phát hiện địa chỉ IP khi biết địa chỉ MAC



iSPACE

Bộ giao thức TCP/IP

- **Ports**
  - Giá trị port được biểu diễn 2 byte(16 bits : 0 to 65535)
    - Well Known Ports : 0 - 1023.
    - Registered Ports : 1024 - 49151
    - Dynamic and/or Private Ports : 49152 - 65535



Bộ giao thức TCP/IP

- **Địa chỉ MAC**
  - Địa chỉ vật lý (Physical Address) của thiết bị mạng.
  - 6 bytes, 48 bits, gồm 12 ký số hệ Hecxa.
  - 6 ký số đầu để nhận diện nhà sản xuất.
  - 6 ký số sau nhận diện thiết bị phần cứng của mỗi nhà Sản Xuất.
  - Hoạt động ở lớp Data Link của mô hình OSI.

A5-0C-D3-1B-05-46

Một số giao thức khác

Ngoài bộ giao thức TCP/IP, còn một số bộ giao thức khác do các hãng phát triển cho hệ thống mạng LAN của mình

- **Internetwork Packet Exchange/Sequenced Packet Exchange (IPX/SPX)**
  - Được công ty Novell thiết kế sử dụng cho các sản phẩm mạng của chính hãng
  - SPX hoạt động trên tầng transport của mô hình OSI, bảo đảm độ tin cậy của liên kết truyền thông từ mút đến mút.

TRƯỜNG CAO ĐẲNG NGHỀ CNTT iSPACE Website: http://www.ispace.edu.vn

Một số giao thức khác

- **AppleTalk**
  - Do hãng Apple computer phát triển cho họ máy tính cá nhân Macintosh
  - Giao thức Apple được phát triển trên tầng vật lý của Ethernet và Token Ring.
    - Các vùng tối đa trên một phân mạng: Phase 1 là 1, phase 2 là 255
    - Các node tối đa trên mỗi mạng: Phase 1 là 254, phase 2 khoảng 16 triệu
    - Địa chỉ động dựa trên các giao thức truy nhập
    - Định tuyến Spit-horizon

IP Address V.4

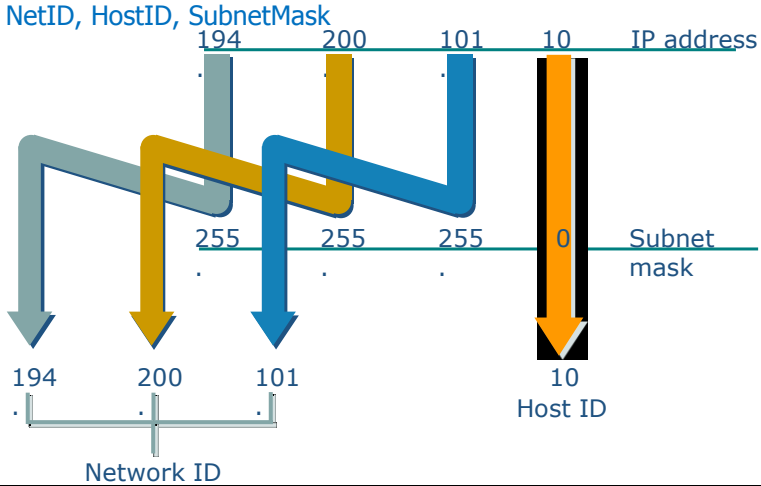
Các hệ thống máy tính trên mạng LAN và Internet liên lạc với nhau qua địa chỉ IP. Địa chi IP đang sử dụng là IP Address v.4

- **Địa chỉ IP (IPV4)**
  - Địa chỉ IP v.4 là một số 32 bít, được chia làm 4 Octects(4 bytes), cách nhau = "."

  - Class Bit: Xác định IP thuộc lớp nào(A, B, C, D, E)
  - Net ID: định danh địa chỉ mạng
  - HostID: định địa chỉ IP của một host cụ thể.

IP Address V.4

- Địa chỉ IP (IPV4)
  - NetID, HostID, SubnetMask



IP Address V.4

Địa chỉ IP (IPV4)

- Các lớp địa chỉ IPv4

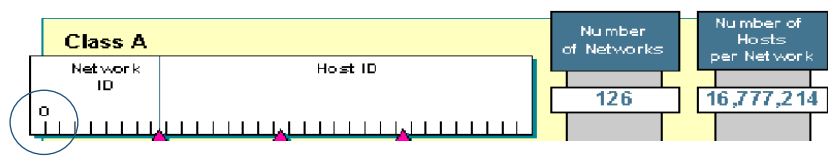
Lớp	#Số bit net_id	#Số bit host_id	Bắt đầu bằng bit	Giá trị byte đầu	Subnet mask
A	8	24	0	1-126	255.0.0.0
B	16	16	10	128-191	255.255.0.0
C	24	8	110	192-223	255.255.255.0
D	Reserved for multicast		1110	224-239	N/A
E	Reserved for R & D		11110-11111	240-255	N/A

TRƯỜNG CAO ĐẲNG NGHỀ CNTT iSPACE Website: http://www.ispace.edu.vn

IP Address V.4

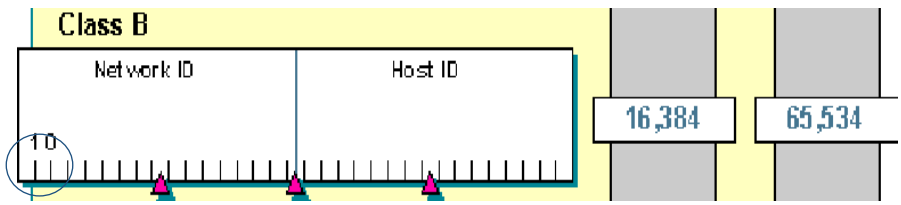
Địa chỉ IP (IPV4)

- IP lớp A
  - 1-Bit cao nhất của byte 1 để nhận biết lớp A và có giá trị là 0
  - Số Net ID :
    - Chiếm 1 byte giá trị, $2^{8-1} = 27 = 128$
    - Trừ 2 giá trị đặc biệt (0 : cục bộ mạng, 127 mạng loopback) : 126
  - Số Host ID :
    - Chiếm 3 bytes giá trị, $2^{24} - 2 = 16.777.216$



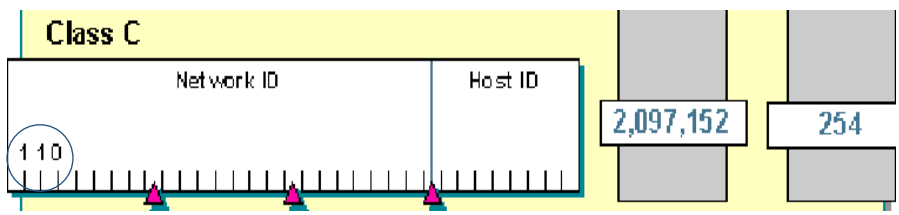
IP Address V.4

- **Địa chỉ IP (IPV4)**
  - IP lớp B
    - 2-bit cao nhất của byte 1 để nhận biết lớp B và có giá trị là 10.
    - Số Net ID :
      - Chiếm 2 bytes giá trị, $2^{16-2} = 2^{14} =$ *16.384*
    - Số Host ID :
      - Chiếm 2 bytes giá trị, $2^{16}-2=65.536$



IP Address V.4

- **Địa chỉ IP (IPV4)**
  - IP lớp C
    - 3-bit cao nhất để nhận biết lớp C và có giá trị là 110.
    - Số Net ID :
      - Chiếm 3 bytes giá trị, $2^{24-3} = 2^{21} =$ ***2.097.152***
    - Số Host ID :
      - Chiếm 1 bytes giá trị, $2^{8}-2= 256$



IP Address V.4

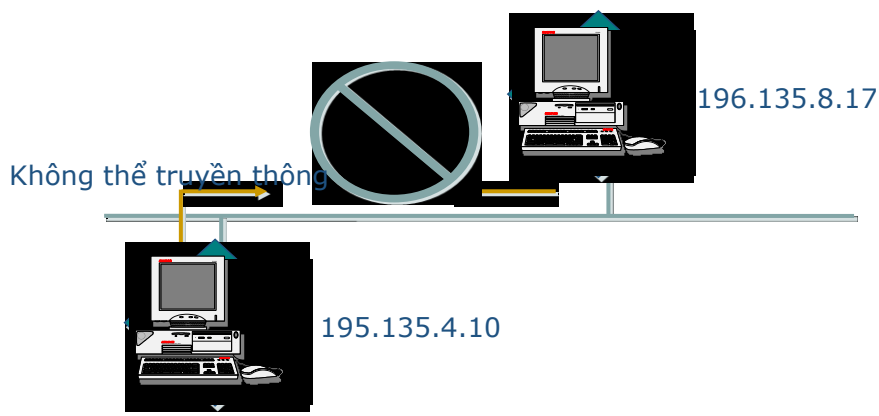
Địa chỉ IP (IPV4)

- Mặt nạ mạng (Subnet Mask)

	1st byte	2nd byte	3rd byte	4th byte
Địa chỉ IP 145.98.20.5	10010001	01100010	0001 0100	00000101
Subnet mask 255.255.0.0	11111111	11111111	0000 0000	00000000
Địa chỉ mạng	10010001 or 145 .	01100010 98 .	0000 0000 0 .	00000000 0

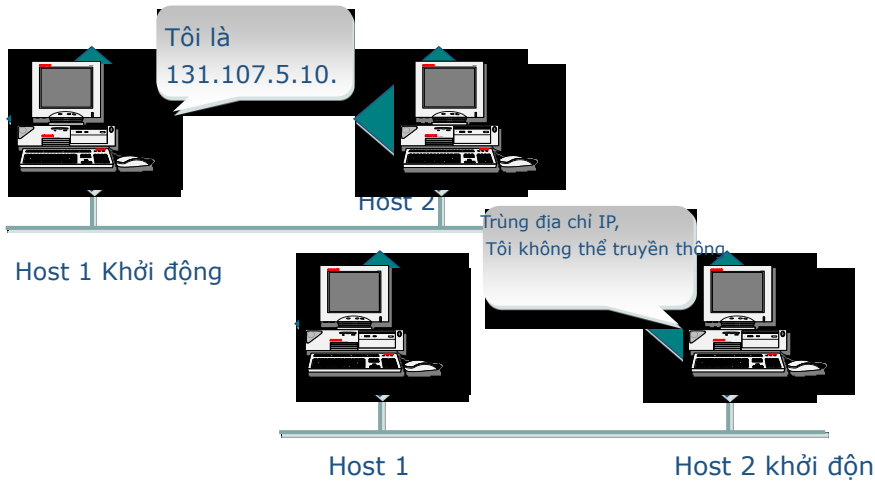
Một số sự cố thông dụng

Thiết lập địa chỉ mạng không đúng



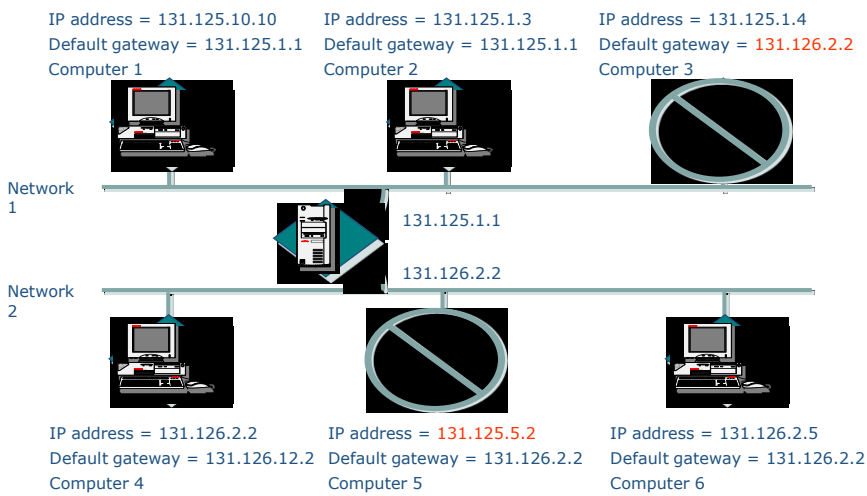
Một số sự cố thông dụng

Trùng địa chỉ IP



Một số sự cố thông dụng

Sai địa chỉ mạng



TÓM LƯỢC BÀI HỌC

- Cấu trúc và chức năng mô hình TCP/IP
- Địa chỉ IP V.4
- Cấu hình địa chỉ IP v.4
- **Kết luận**
  - Giúp sinh viên hiểu được, bộ giao thức trao đổi trên hệ thống mạng LAN và Internet
  - Nắm được địa chỉ IP v.4
  - Triển khai địa chỉ IP cho hệ thống mạng LAN cho doanh nghiệp.
  - Sử dụng địa chỉ IP đúng với hệ thống mạng LAN của doanh nghiệp

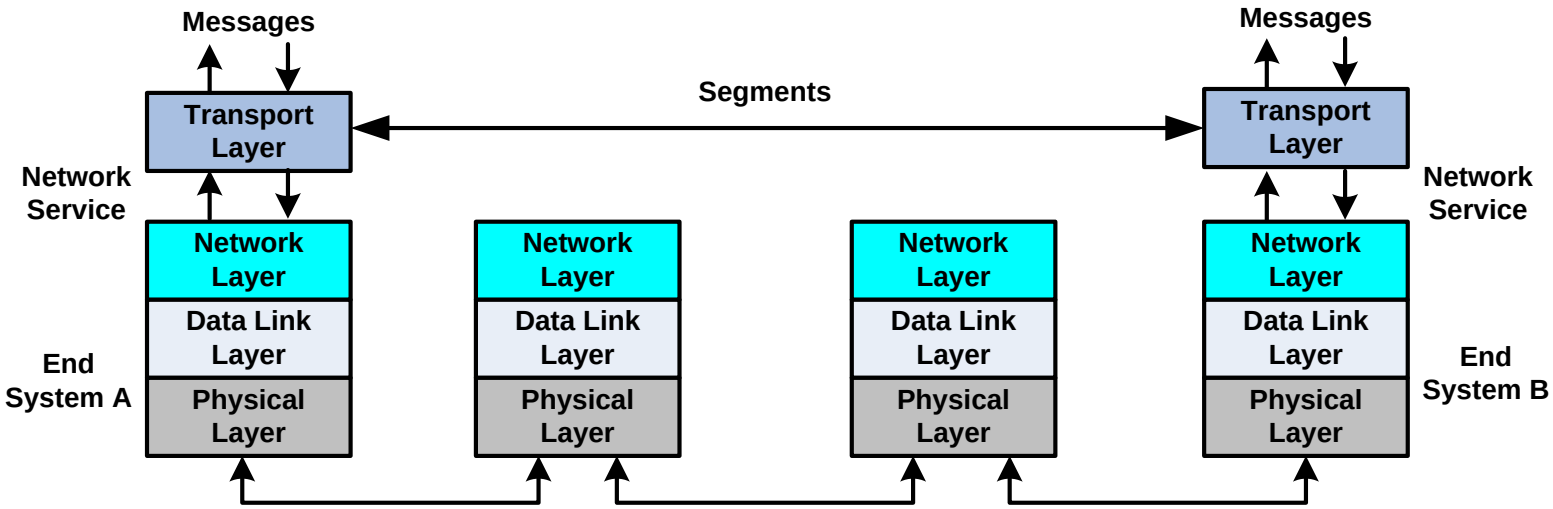
HỎI - ĐÁP

Q & A

Tổng quan về mạng Internet và giao thức TCP/IP

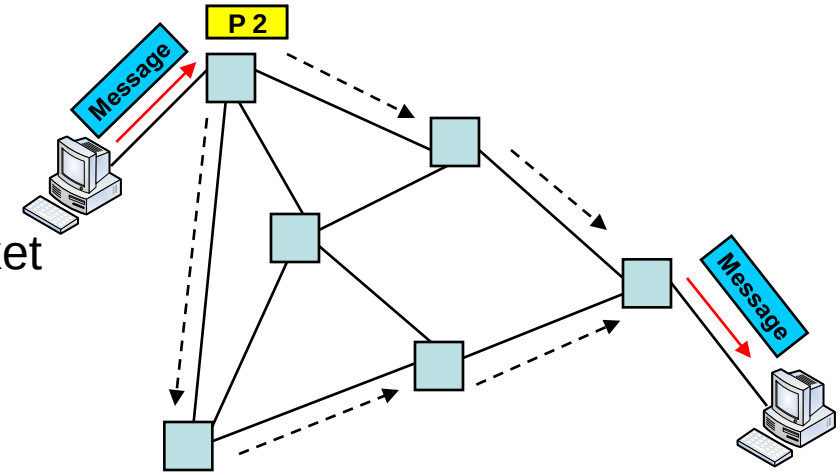
- Datagram và Virtual Circuits (VC)
- Routing trong mạng chuyển mạch gói
- Shortest path routing
- Giao thức IP
  - ✓Internet protocol
  - ✓ ARP, ICMP
  - ✓ Internet routing protocols
  - ✓ DHCP, NAT, mobile IP
- Giao thức TCP và UDP
  - ✓UDP
  - ✓TCP

- **Mạng chuyển mạch gói (packet switching network)**
  - ✓ **Vấn đề của Network layer**
    - o Cần có các phần tử mạng phân tán: switch and router
    - o Large scale: nhiều user (con người & thiết bị truyền thông)
    - o Địa chỉ hóa và định tuyến (addressing & routing)
  - ✓ **Dịch vụ mạng cho tầng transport layer: connection-oriented, connectionless, best-effort**

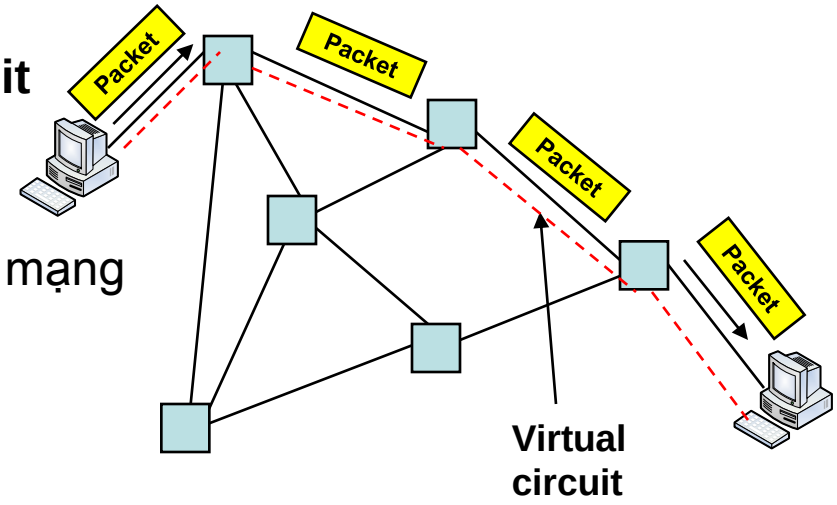


- ✓ **Chức năng của Network layer**
  - o **Routing**: Cơ chế định tuyến cho các gói tin trong mạng
  - o **Forwarding**: chuyển tiếp các gói tin qua các thiết bị mạng
  - o **Priority & scheduling**: xác định trật tự truyền các gói tin trong mạng
  - o Congestion control, segmentation & reassembly, security (tùy chọn)
- ✓ **Datagram và Virtual Circuit (VC)**
  - o **Chuyển mạch gói**
    - ▪ Truyền thông tin qua các packet (gói tin)
    - ▪ Khả năng có trễ ngẫu nhiên và mất packet
    - ▪ Mỗi ứng dụng có yêu cầu truyền tin khác nhau

- ✓ **Mạng chuyển mạch gói**
  - o Truyền các gói tin giữa các user
  - o Đường truyền và chuyển mạch gói (router)
  - o Chế độ làm việc
    - ▪ Connectionless
    - ▪ Virtual circuit
- ✓ **Packet switching – datagram**
  - o Message chia thành các packet
  - o Địa chỉ nguồn và đích đặt trong packet header
  - o Packet có thể đến đích không theo trật tự

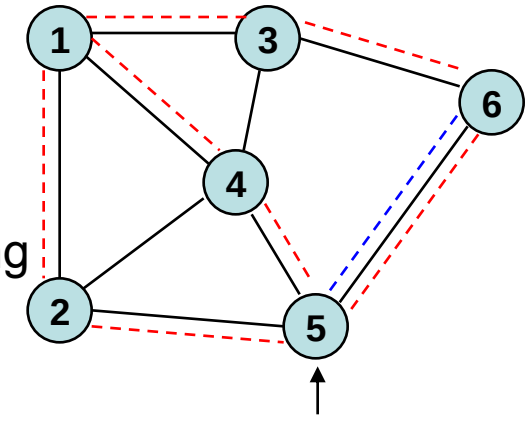


- ✓ **Routing table trong mạng chuyển mạch gói**
  - o Các tuyến được xác định từ bảng định tuyến
  - o Xác định chặng tiếp theo (**next hop**) đi tới đích qua output port
  - o Kích thước bảng định tuyến tăng theo địa chỉ đích
  - o Ví dụ: Internet routing
- ✓ **Packet switching – Virtual circuit**
  - o Giai đoạn thiết lập liên kết (call set-up phase): xác định con trỏ theo đường dẫn trong mạng
  - o Các packets trong kết nối đi theo cùng đường dẫn
  - o Có thể thay đổi bitrate, delay

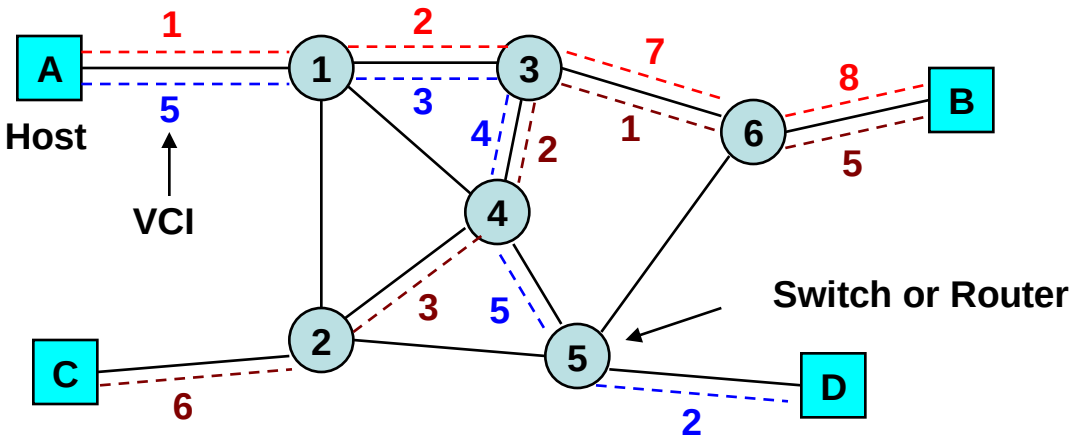


- ✓ **Thiết lập liên kết**
  - o Thông tin báo hiệu (signaling message) xác định liên kết và các bảng khởi tạo (setup table) trong các chuyển mạch
  - o Các liên kết được xác định nhờ **virtual circuit identifier** (VCI)
  - o Khi setup table được thiết lập, packet được truyền trên đường dẫn
- ✓ **Virtual circuit forwarding tables (VC FT)**
  - o Đầu vào của mỗi chuyển mạch gói có **FT**
  - o Tìm VCI tương ứng cho incoming packet
  - o Xác định đầu ra tới **next hop** và thêm VCI tương ứng cho đường link
  - o VC FT có thể mang thông tin về mức ưu tiên của packet, v.v…

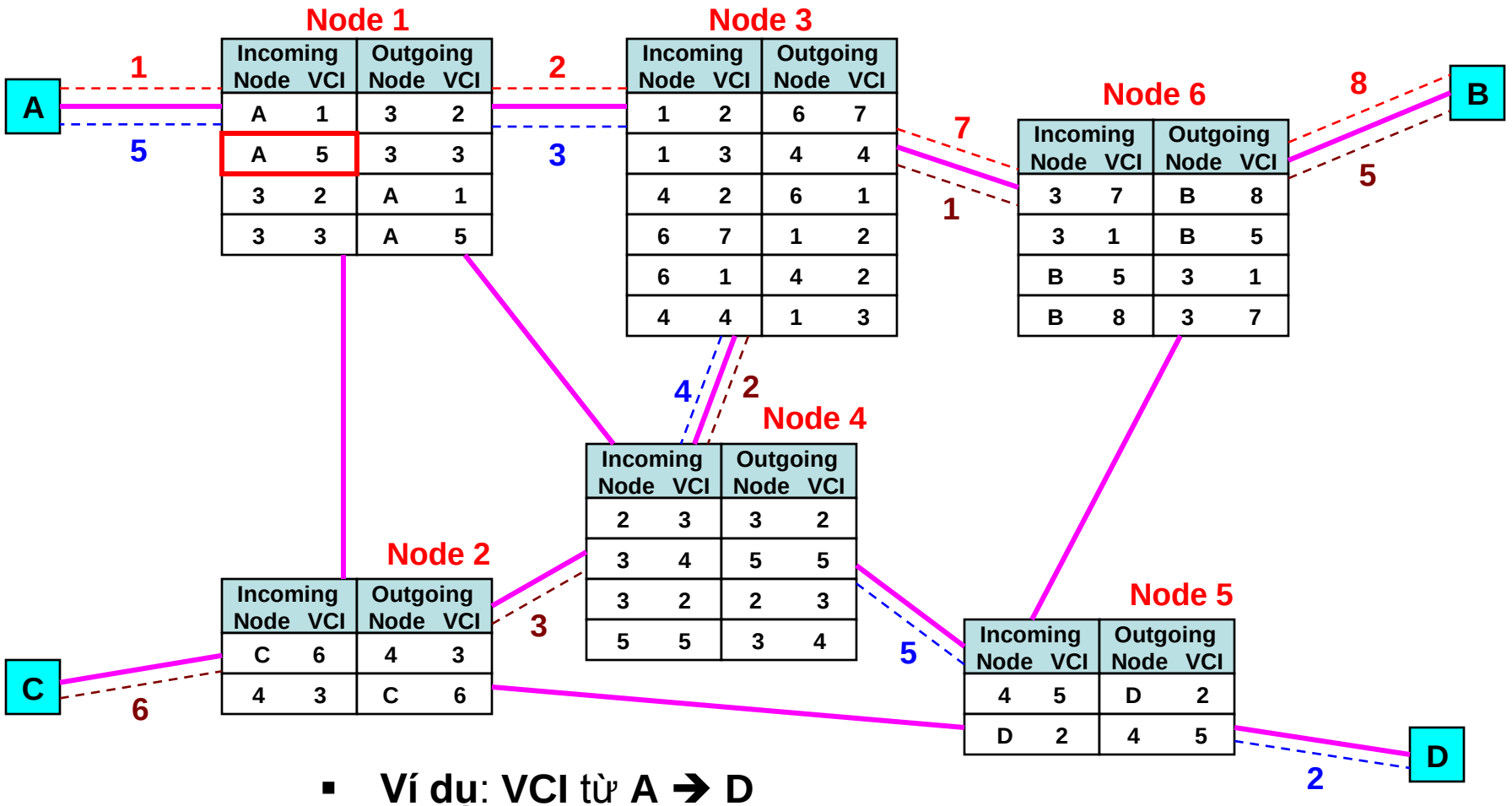
- ✓ **Định tuyến trong mạng chuyển mạch gói**
  - o Có thể có 3 tuyến từ node 1 tới node 6: 1-3-6, 1-4-5-6, 1-2-5-6
  - o Tuyến nào tối ưu nhất? : Min delay, min hop, max BW, min cost
  - o Thuật toán định tuyến
    - ▪ Truyền nhanh và chính xác
    - ▪ Thích ứng với thay đổi của cấu hình mạng (link & node failure)
    - ▪ Thích ứng với sự thay đổi lưu lượng mạng từ nguồn đến đích
  - o Centralized vs distributed routing, static vs dynamic routing
- ✓ **Tạo bảng định tuyến (routing table - RT)**
  - o Cần có thông tin về trạng thái link
  - o Sử dụng thuật toán định tuyến để thông báo trạng thái link: broadcast, flooding
  - o Tính toán tuyến theo thông tin:
    - ▪ Single metric, multiple metric
    - ▪ Single route, alternate route



- ✓ **Định tuyến trong Virtual-circuit (VC) packet network**
  - o Tuyến được xác lập khi khởi tạo liên kết
  - o Các bảng định tuyến trong các switch thực hiện chuyển tiếp packet theo tuyến đã được xác lập



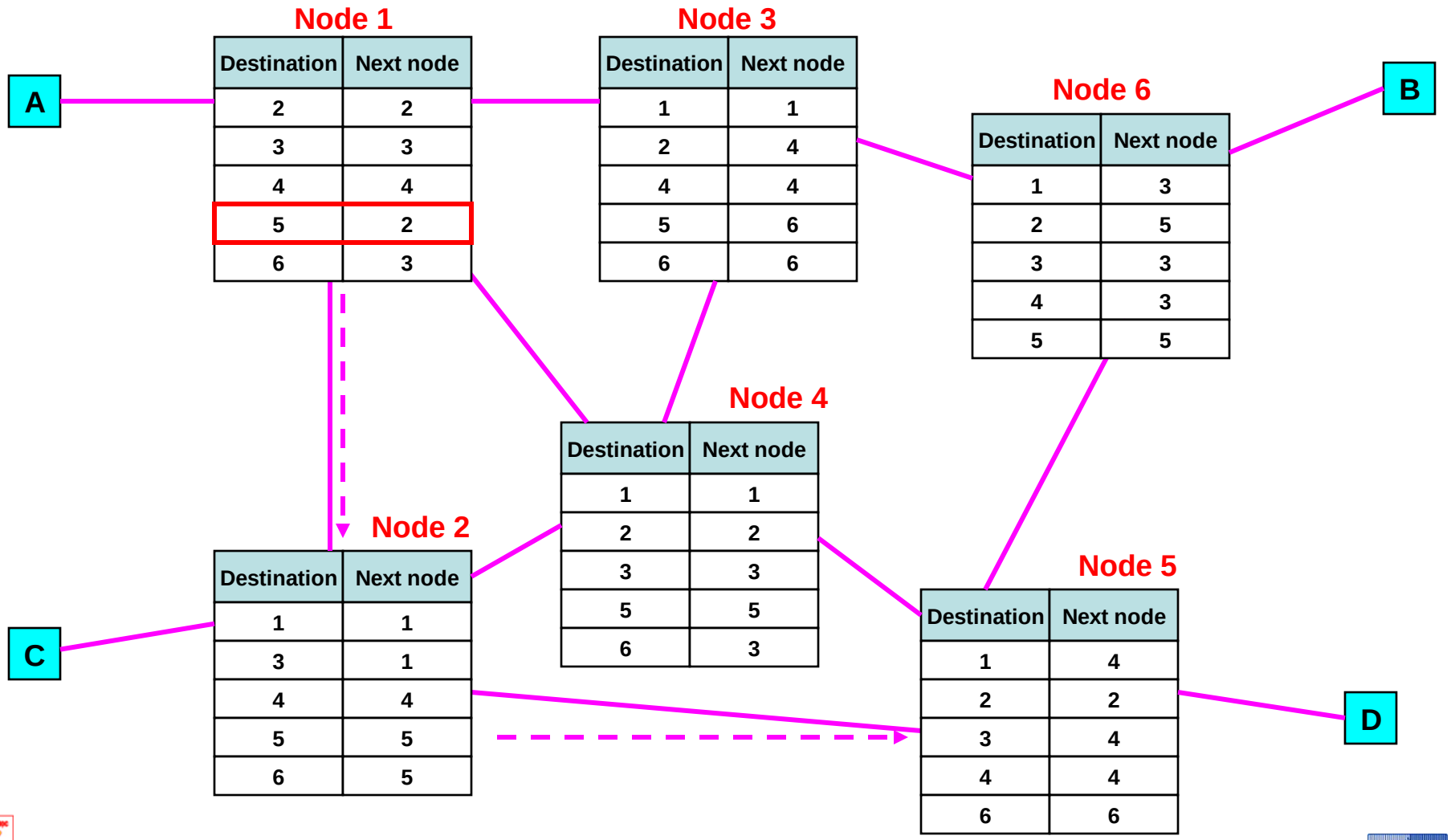
o **RT** trong **VC packet network**



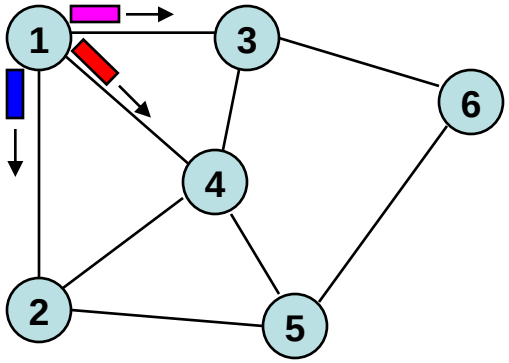
- **Ví dụ**: VCI từ **A ➔ D**

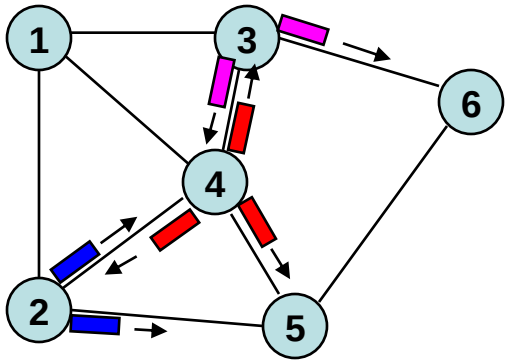
Từ **A & VCI 5 ➔ 3 & VCI 3 ➔ 4 & VCI 4 ➔5 & VCI 5 ➔ D & VCI 2**

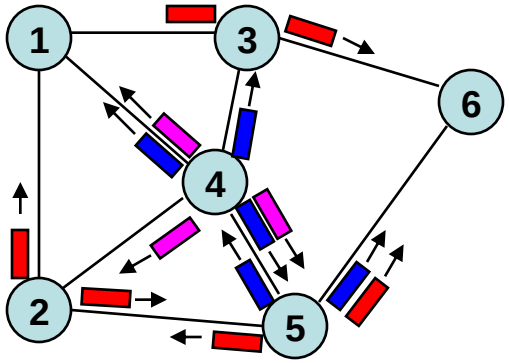
o **RT** trong **Datagram packet network**



- **Định tuyến (routing) trong mạng chuyển mạch gói**
  - ✓ **Định tuyến đặc biệt: flooding và deflection**
    - o Flooding
      - Gửi gói tin tới tất cả các node trong mạng: Không cần bảng định tuyến, sử dụng kiểu quảng bá để gửi các packet tới các nút mạng
      - Limited-flooding:
        - ❖Time-to-live cho mỗi gói tin: giới hạn số chặng chuyển tiếp
        - ❖ Trạm nguồn điền số thứ tự cho mỗi packet

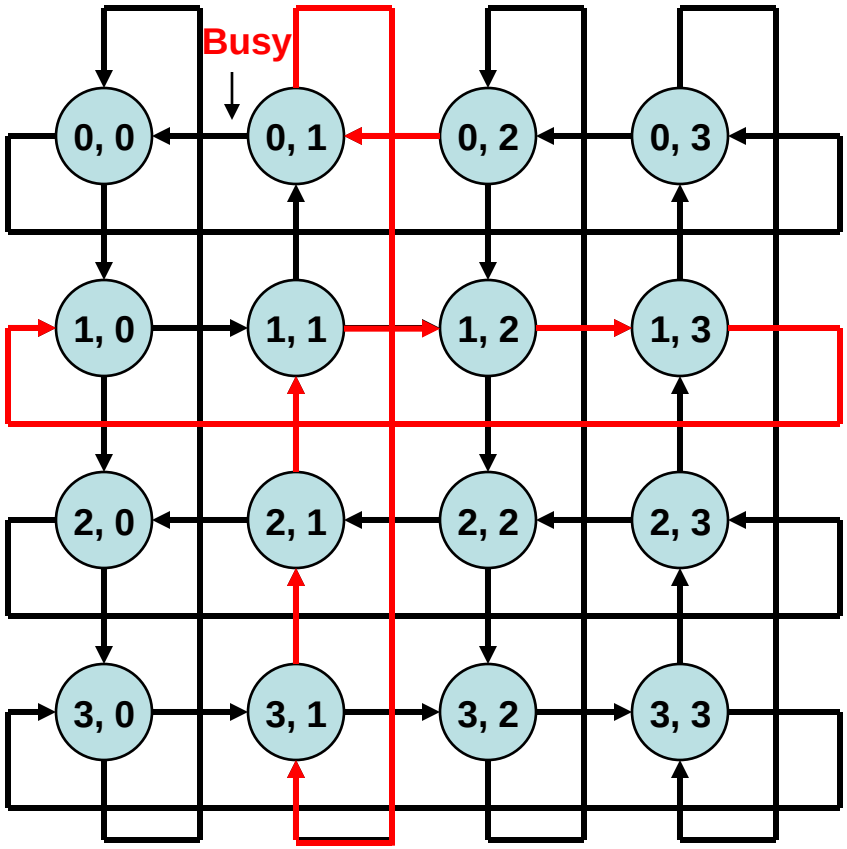




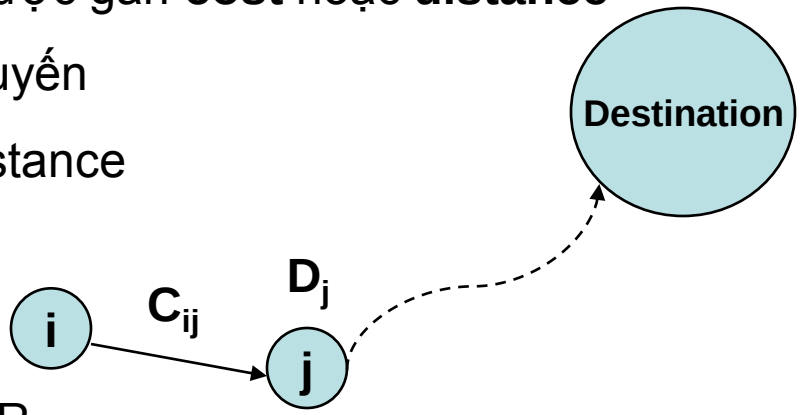


- Deflection routing
  - Network chuyển tiếp các packet tới các cổng (port) xác định
  - Nếu port này busy, packet sẽ được chuyển hướng tới port khác

Node (0, 2) ➔ (1, 0)



- **Shortest path routing**
  - ✓ Shortest path & routing
    - o Có nhiều tuyến kết nối giữa nguồn và đích
    - o Định tuyến: chọn tuyến kết nối ngắn nhất (**shortest path - SP**) thực hiện phiên truyền dẫn
    - o Mỗi tuyến kết nối giữa 2 node được gắn **cost** hoặc **distance**
  - ✓ Routing metrics: Tiêu chí đánh giá tuyến
    - o **Path length**: Tổng cost hoặc distance
    - o Các tiêu chí:
      - ▪ Đếm số chặng (hop count)
      - ▪ Reliability, link reliability, BER
      - ▪ Delay
      - ▪ Bandwidth
      - ▪ Load



Nếu D_j là khoảng cách ngắn nhất tới đích từ node **i**, và nếu node **j** liền kề nằm trên **SP** ➔ $D_i = C_{ij} + D_j$

- ✓ Các phương án
  - o Distance vector protocol (**DVP**)
    - ▪ Các node kề nhau trao đổi thông tin về khoảng cách đi tới đích
    - ▪ Xác định chặng tiếp theo (**next hop - NH**) tới địa chỉ đích
    - ▪ Thuật toán Bellman-Ford **SP** (phân tán)
  - o Link state protocol (**LSP**)
    - ▪ Thông tin về link state được gửi tới tất cả các router (flooding)
    - ▪ Router có thông tin đầy đủ về cấu hình mạng
    - ▪ **SP** và **NH** được tính toán
    - ▪ Thuật toán Dijkstra **SP** (tập trung)
- ✓ **Distance vector (DV)**: Vector khoảng cách
  - o **Routing table** (**RT**) cho mỗi địa chỉ đích: **next-node (NN),** distance
  - o Tổng hợp **RT**: Các node lân cận trao đổi **RT,** xác định next hope

✓ Bellman-Ford algorithm

1. ***Initialization***
   - Khoảng cách từ node **d** tới chính nó: $\mathbf{D_d = 0}$
   - Khoảng cách từ node **i** bất kỳ tới **d:** $\mathbf{D_i = \infty}$, $\mathbf{i \neq d}$
   - Node tiếp theo chưa được xác định: $\mathbf{n_i = -1, i \neq d}$
2. ***Send step***
   - Cập nhật **DV** cho các node kề bên qua đường link trực tiếp
3. ***Receive step***
   - Tại node i, tìm **NH** có khoảng cách ngắn nhất tới d
     - $\mathbf{D_i(d) = Min_j\{C_{ij} + D_j\}, i \neq j}$
     - Thay cặp giá trị cũ $\mathbf{(n_i, D_i(d))}$ bằng giá trị mới $\mathbf{(n_i^*, D_j^*(d))}$ nếu tìm được **NN** mới
   - Quay lại bước 2 cho đến khi không còn thay đổi thêm nữa

Iteration	Node 1	Node 2	Node 3	Node 4	Node 5
Initial	(-1, ∞)	(-1, ∞)	(-1, ∞)	(-1, ∞)	(-1, ∞)

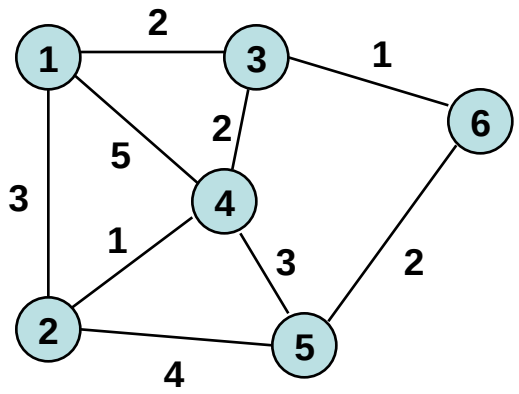
Node 2 ➔ Node 6

- **2-1-3-6**: 3 + 2 + 1 = 6
- **2-4-3-6**: 1+ 2 + 1 = 4
- **2-5-6**: 4 + 2 = 6

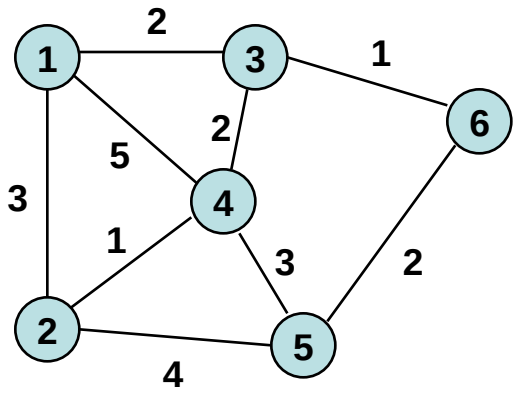
Đường nào ngắn nhất?

(n, D_i)

- **n: NN đi tới đích**
- **D_i**: khoảng cách ngắn nhất từ node i tới đích

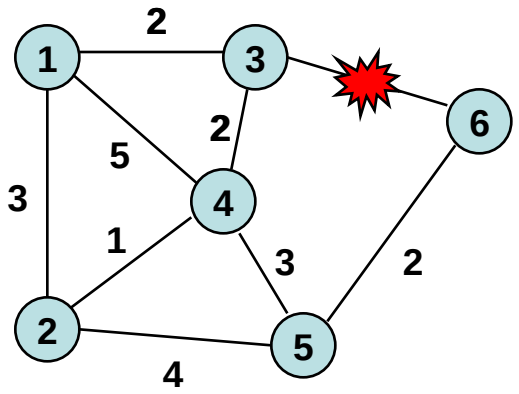


Iteration	Node 1	Node 2	Node 3	Node 4	Node 5
Initial	(-1, ∞)	(-1, ∞)	(-1, ∞)	(-1, ∞)	(-1, ∞)
1	(-1, ∞)	(-1, ∞)	(6, 1)	(-1, ∞)	(6, 2)
2	(3, 3)	(5, 6)	(6, 1)	(3, 3)	(6, 2)
3	(3, 3)	(4, 4)	(6, 1)	(3, 3)	(6, 2)
4	(3, 3)	(4, 4)	(6, 1)	(3, 3)	(6, 2)



o Khi có lỗi mạng

Iteration	Node 1	Node 2	Node 3	Node 4	Node 5
	(3, 3)	(4, 4)	(6, 1)	(3, 3)	(6, 2)
Update 1	(3, 3)	(4, 4)	(4, 5)	(3, 3)	(6, 2)
Update 2	(3, 7)	(4, 4)	(4, 5)	(5, 5)	(6, 2)
Update 3	(3, 7)	(4, 6)	(4, 7)	(5, 5)	(6, 2)
Update 4	(2, 9)	(4, 6)	(4, 7)	(5, 5)	(6, 2)
Update 5	(2, 9)	(4, 6)	(4, 7)	(5, 5)	(6, 2)

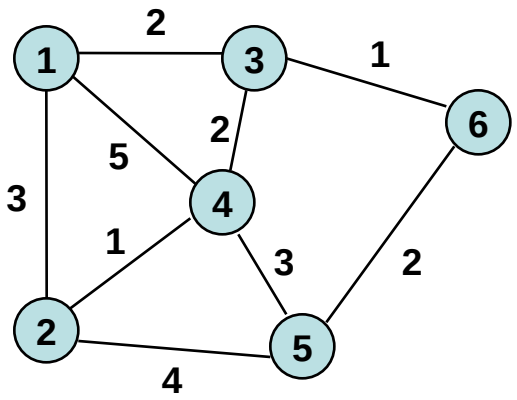


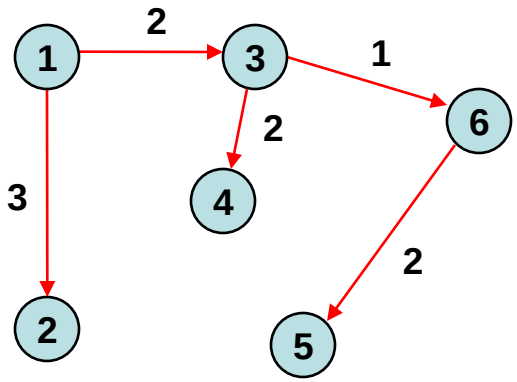
- **Link-state algorithm**
  - ✓ Quá trình 2 giai đoạn
    - o Mỗi node nguồn được nhận bản đồ **(map)** của tất cả các node khác và **link-state** của mạng
    - o Tìm **SP** trên bản đồ từ node nguồn tới tất cả các node đích
  - ✓ Quảng bá thông tin về **link-state**
    - o Mỗi node **i** trong mạng gửi quảng bá tới từng node mạng:
      - **ID** của node liền kề: $\mathbf{N_i}$ = tập hợp của các node liền kề node **i**
      - Khoảng cách tới node liền kề của nó $\{\mathbf{C_{ij}} \mid \mathbf{j} \in \mathbf{N_i}\}$

- ✓ **Dijstra algorithm:** tìm **SP** theo thứ tự
  - o **N**: tập hợp các node đã tìm thấy **SP**
  - o **Initialization** (*Bắt đầu với node nguồn **s***)
    - **N = {s}, D_s = 0**: Khoảng cách từ node **s** tới chính nó bằng 0
    - **D_j = C_{sj}, j ≠ s**: Khoảng cách tới node liền kề kết nối trực tiếp
  - o **Step A** (*Tìm node **i** gần nhất*)
    - Tìm node **i** ∈ **N** sao cho **D_i = min D_j** với **j** ∈ **N**
    - Cập nhật node **i** vào tập hợp **N**
    - Nếu **N** chứa tất cả các node, **STOP**
  - o **Step B** (*cập nhật **minimum cost***)
    - Với mỗi node **j** ∈ **N**, tính **D_j = min (D_j, D_i + C_{ij})**
    - Quay lại **step A**

✓ Thực hiện thuật toán Dijkstra

o **Ví dụ**: Tìm **SP** cho **Node 1**





Iteration	N	D_2	D_3	D_4	D_5	D_6
Initial	{1}	3	2	5	∞	∞
1	{1, 3}	3	2	4	∞	3
2	{1, 2, 3}	3	2	4	7	3
3	{1, 2, 3, 6}	3	2	4	5	3
4	{1, 2, 3, 4, 6}	3	2	4	5	3
5	{1, 2, 3, 4, 5, 6}	3	2	4	5	3

o **RT** của **node 1**

Destination	Next node	Cost
2	2	3
3	3	2
4	3	4
5	3	5
6	3	3

o Khi có link bị hỏng

- Router thiết lập khoảng cách của link về ∞ và gửi thông báo cập nhật sử dụng phương pháp *flooding*
- Tất cả các router sẽ tính toán và cập nhật **SP**

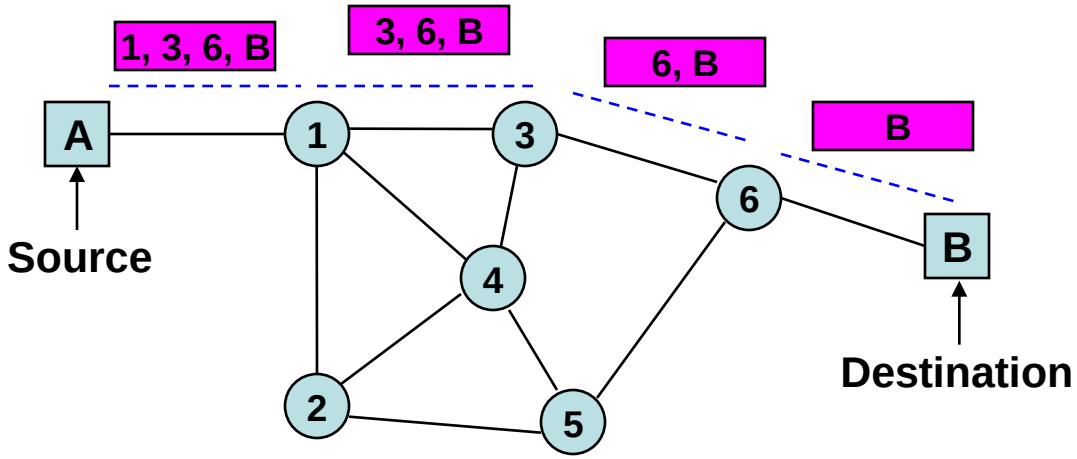
o Vấn đề thông báo cập nhật link cost

- Gắn số thứ tự cho mỗi thông báo về cập nhật link cost
- Kiểm tra mỗi thông báo đến. Nếu là thông báo mới, cập nhật và gửi quảng bá. Nếu là thông báo cũ, gửi lại theo link đến

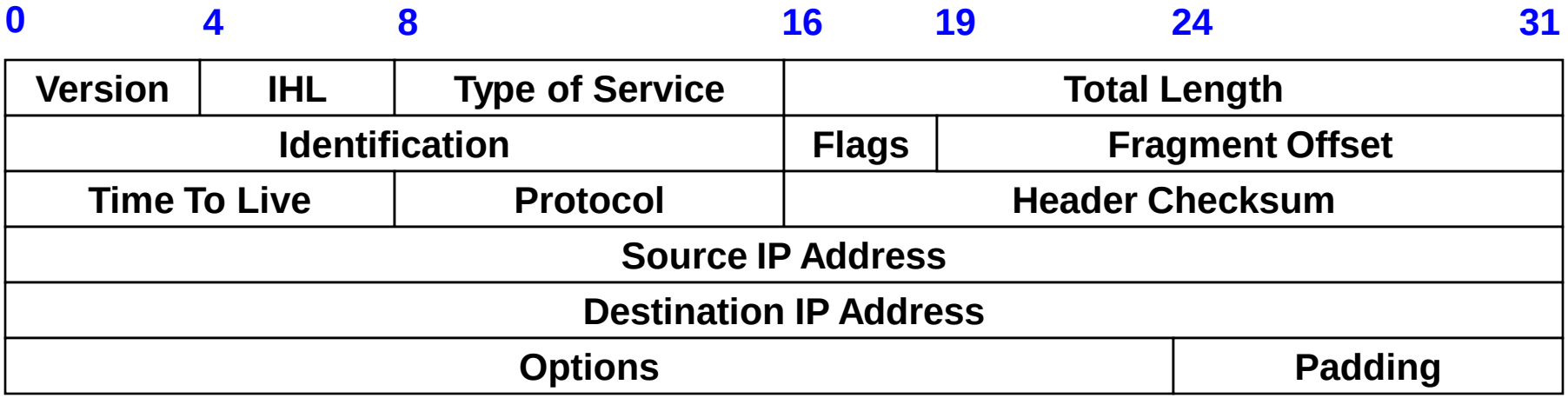
✓ **Source routing**

o Source chỉ định tuyến cho các packet

- **Strict**: Source chỉ định tất cả các node cho packet
- **Loose**: Chỉ một phần các node được chỉ định



- **Internet protocol (IP)**
  - ✓ IP packet header: tối đa 20 byte, trường option không quá 40 byte



0	4	8	16	19	24	31
Version	IHL	Type of Service	Total Length			
Identification			Flags	Fragment Offset		
Time To Live		Protocol	Header Checksum			
Source IP Address						
Destination IP Address						
Options					Padding	

- **Version**: **IPv4**
- **Internet Header Length (IHL)**: Độ dài IP header tính theo 32 bit/word
- **Type of Service (ToS)**: Mức ưu tiên cho packet tại mỗi router.
- **Total Length**: Số byte các IP packet, bao gồm header và data (**< 65536**)
- **Identification, Flags, Fragment Offset**: Sử dụng trong **fragmentation** và **reassembly**

0	4	8	16	19	24	31
Version	IHL	Type of Service	Total Length			
Identification			Flags	Fragment Offset		
Time To Live		Protocol	Header Checksum			
Source IP Address						
Destination IP Address						
Options					Padding	

- **Time To Live (TTL)**: Số chặng tối đa cho mỗi packet được phép đi qua
  - ❖ Qua mỗi **router** trên đường tới đích, **TTL** giảm 1 đơn vị
  - ❖ Nếu **TTL** đạt giá trị 0 trước khi tới đích, **router** hủy IP packet, gửi thông báo lỗi tới nguồn
- **Protocol**: Báo cho layer phía trên IP data trong packet tại đích
  - ❖ **TCP (6), UDP (17), ICMP (1)**
- **Header Checksum**: Kiểm tra tính chính xác của IP header nhận được
- **Source IP, Destination IP address**: Địa chỉ IP của tram nguồn và đích

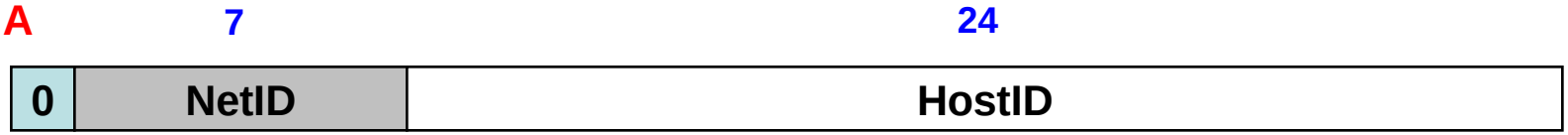
0	4	8	16	19	24	31
Version	IHL	Type of Service	Total Length			
Identification			Flags	Fragment Offset		
Time To Live		Protocol	Header Checksum			
Source IP Address						
Destination IP Address						
Options					Padding	

- **Option**: có độ dài thay đổi, cho phép packet yêu cầu một số tùy chọn đặc biệt - mức bảo mật, timestamp cho packet tại mỗi router
- **Padding**: đảm bảo header la số nguyên lần các từ 32 bit

- ✓ Xử lý IP header
  - o Kiểm tra độ chính xác của IP header thông qua tính toán **Header Checksum**, đồng thời kiểm tra tính hợp lệ của các trường trong header (IP version, length, …)
  - o Xác định chặng tiếp theo sử dụng bảng định tuyến
  - o Cập nhật các trường cần thiết: TTL, header checksum, …
- ✓ Phương pháp địa chỉ hóa IP
  - o Mỗi trạm có địa chỉ IP 32 bit duy nhất: **NetID, hostID**
  - o **NetID** là duy nhất, được sử dụng trong định tuyến, được quản lý bởi
    - ▪ American Registry for Internet Numbers (**ARIN**)
    - ▪ Reseaux IP Europeens (**RIPE**)
    - ▪ Asia Pacific Network Information Center (**APNIC**)
  - o Mỗi liên kết vật lý sử dụng địa chỉ vật lý duy nhất; **multi-home host**
  - o Biểu diễn trong hệ 10 cho mỗi octet (**VD**: 128.10.1.2)

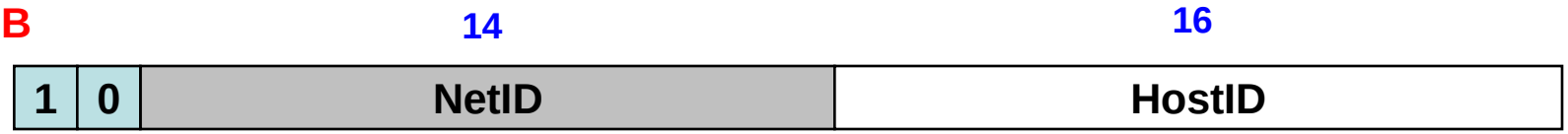
✓ Phân lớp địa chỉ IP

Class A



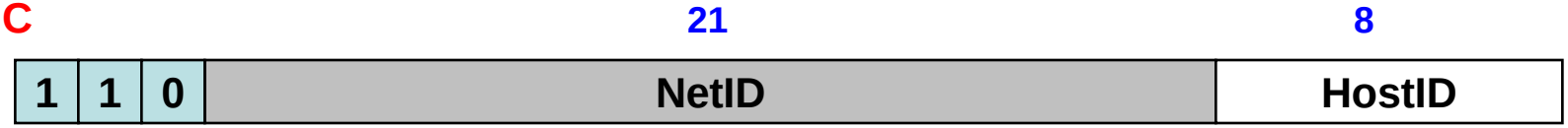
- Tối đa **126 mạng** với tối đa **16 triệu host / mạng**: **1.0.0.0** đến **127.255.255.255**

Class B



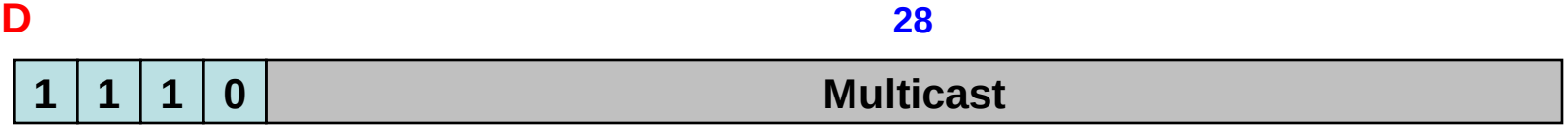
- Tối đa **16382 mạng** với tối đa **64000 host / mạng**: **128.0.0.0 : 191.255.255.255**

Class C



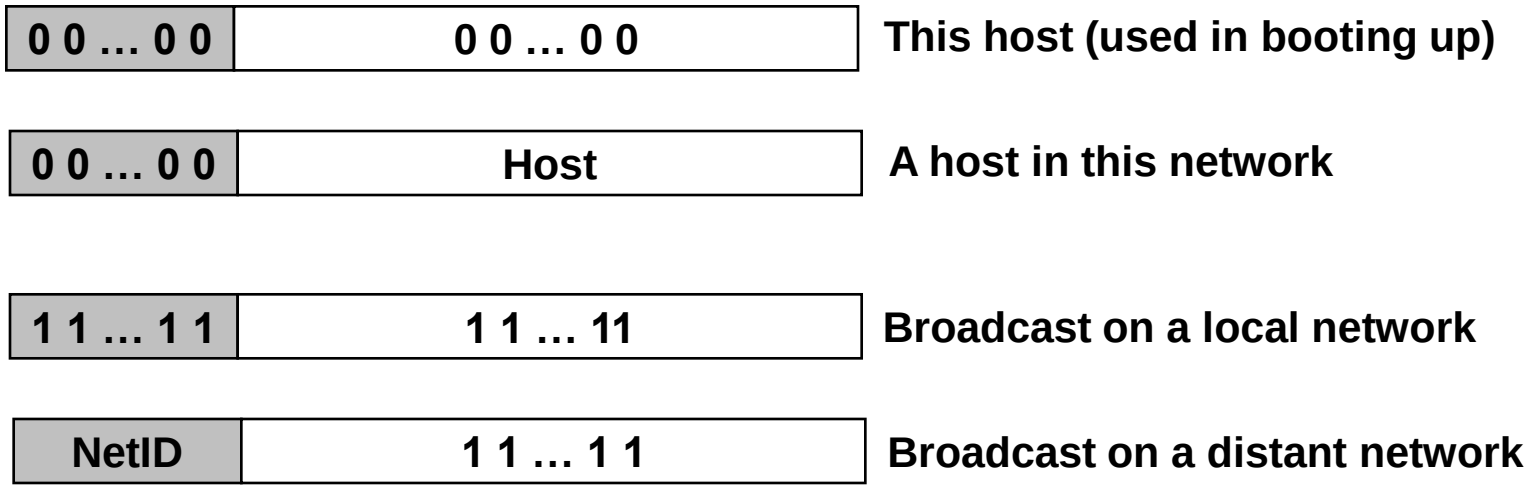
- Tối đa **2 triệu mạng** với tối đa **254 host / mạng**: **192.0.0.0 : 223.255.255.255**

Class D



- Tối đa **250 triệu multicast group**: **224.0.0.0** : **239.255.255.255**

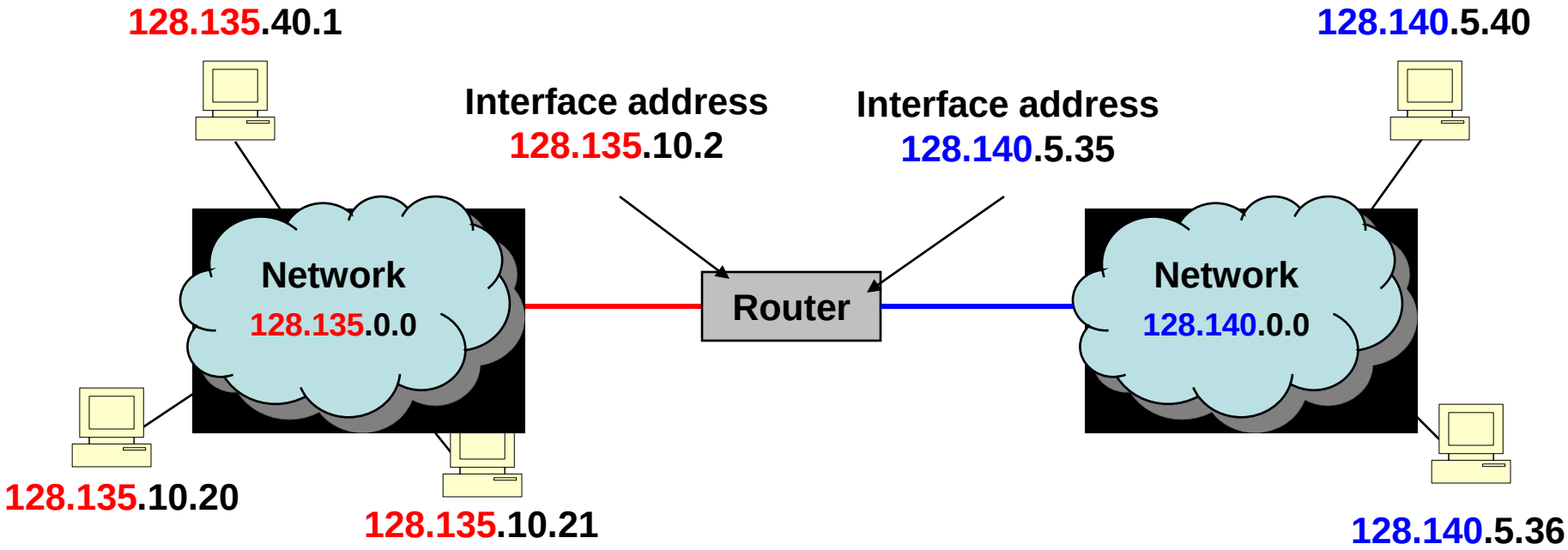
✓ Một số địa chỉ IP đặc biệt



✓ Địa chỉ IP đặc biệt dùng trong mạng riêng (**private IP address**)

- o Router trong mạng chung từ chối packet với các địa chỉ IP này
- o **Range 1: 10.0.0.0 – 10.255.255.255**
- o **Range 2: 172.16.0.0 – 172.31. 255.255**
- o **Range 3: 192.168.0.0 – 192.168.255.255 (Home LAN)**
- o **Network Address Translation (NAT):** chuyển đổi IP riêng và toàn cầu

✓ Ví dụ: **IP addressing**

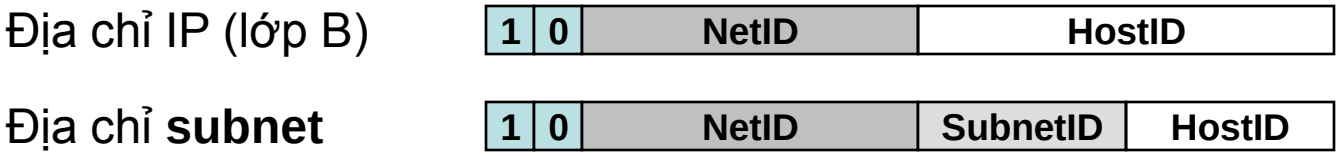


o **HostID = all 0**: tham chiếu tới mạng được chỉ ra bởi **NetID**

o **HostID = all 1**: truyền quảng bá packet trong mạng với **NetID**

✓ Địa chỉ hóa mạng con (**Subnet addressing - SA**)

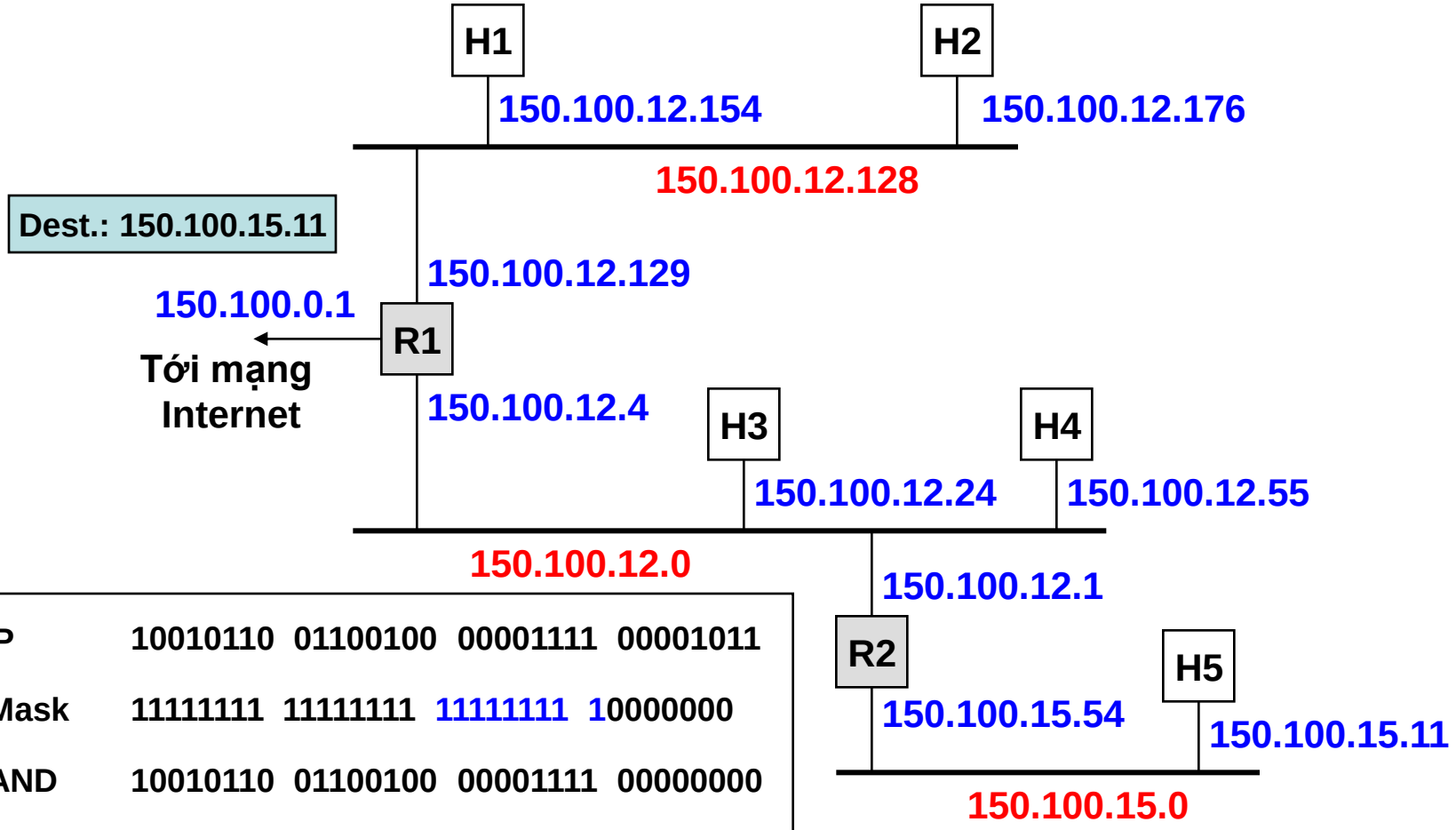
- o **SA** sử dụng cấu trúc mạng ở mức thấp hơn trong mạng hiện tại
- o Trong suốt đối với mạng ở bên ngoài
- o Đơn giản hóa việc quản lý nhiều mạng LAN
- o Mặt nạ (**masking**): sử dụng để xác định số mạng con **(subnet)**



✓**Ví dụ**: 1 tổ chức có địa chỉ IP lớp B với netID: **150.100.0.0** (16 bit hostID)

- o Tạo các mạng con có tối đa **100 host/subnet**
  - **7 bit:** vừa đủ cho mỗi **subnet** đạt số host yêu cầu
  - **16 – 7 = 9 bit:** subnetID
- o Áp dụng **subnet mask** cho địa chỉ IP để tìm mạng con tương ứng
- o Ví dụ: Tìm **subnet** cho địa chỉ IP **150.100.12.176**
- o **Địa chỉ IP**: **10010110 01100100 00001100 10110000**
- o **Mask**: **11111111 11111111 11111111 10000000**
- o **AND**: **10010110 01100100 00001100 10000000**
- o **Subnet:** **150.100.12.128**
- o **Broadcast subnet: 150.100.12.255**
- o Các host kết nối vào subnet: **150.100.12.129 – 150.100.12.254**
- o Các router chỉ sử dụng đia chỉ **subnet** bên trong tổ chức này

✓ **Ví dụ**: Giả sử **9 bit subnetID** và **7 bit hostID**

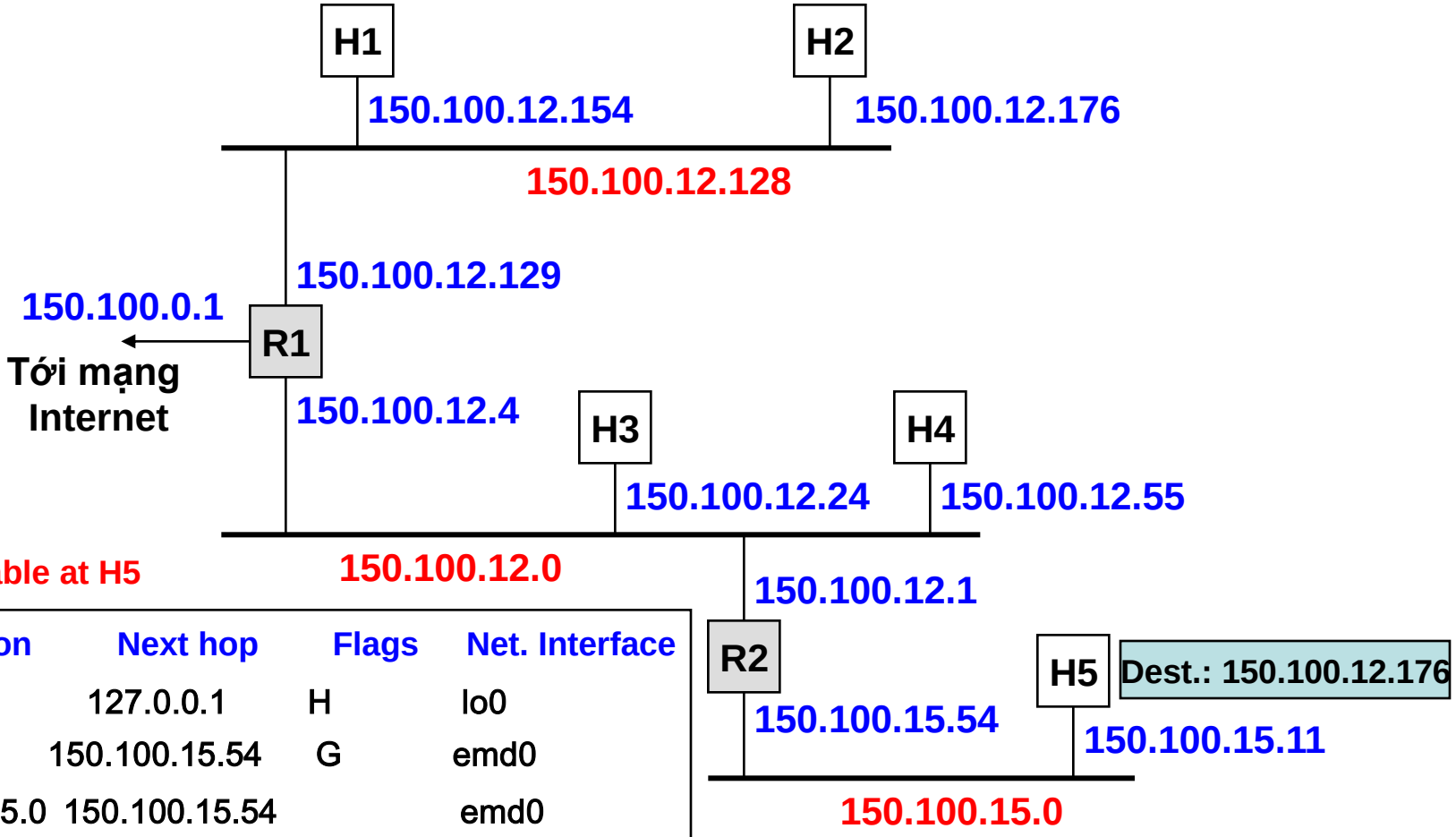


IP	10010110	01100100	00001111	00001011
Mask	11111111	11111111	11111111	10000000
AND	10010110	01100100	00001111	00000000
Subnet	150.100.15.0			

✓ **Định tuyến với subnetwork**

- o IP layer trong host và router lưu giữ routing table (**RT**)
- o **Host**: tham chiếu **RT**
  - Nếu host đích cùng mạng, gửi packet trực tiếp tới host đích sử dụng giao diện mạng tương ứng
  - Nếu không cùng mạng, gửi packet gián tiếp qua **default router**
- o **Router**: Kiểm tra địa chỉ IP của packet nhận được
  - Nếu không biết IP đích, tham chiếu **RT** và xác định **next hop**
- o **Routing table**
  - Mỗi dòng trong **RT** chứa: **Dest. IP , next-hop router IP, subnet mask, phy. address, network interface, statistics, flag**
  - Flag
    - ❖ **H = 1/0**: định tuyến tới host/network
    - ❖ **G = 1/0**: định tuyến tới router (gateway)/trực tiếp

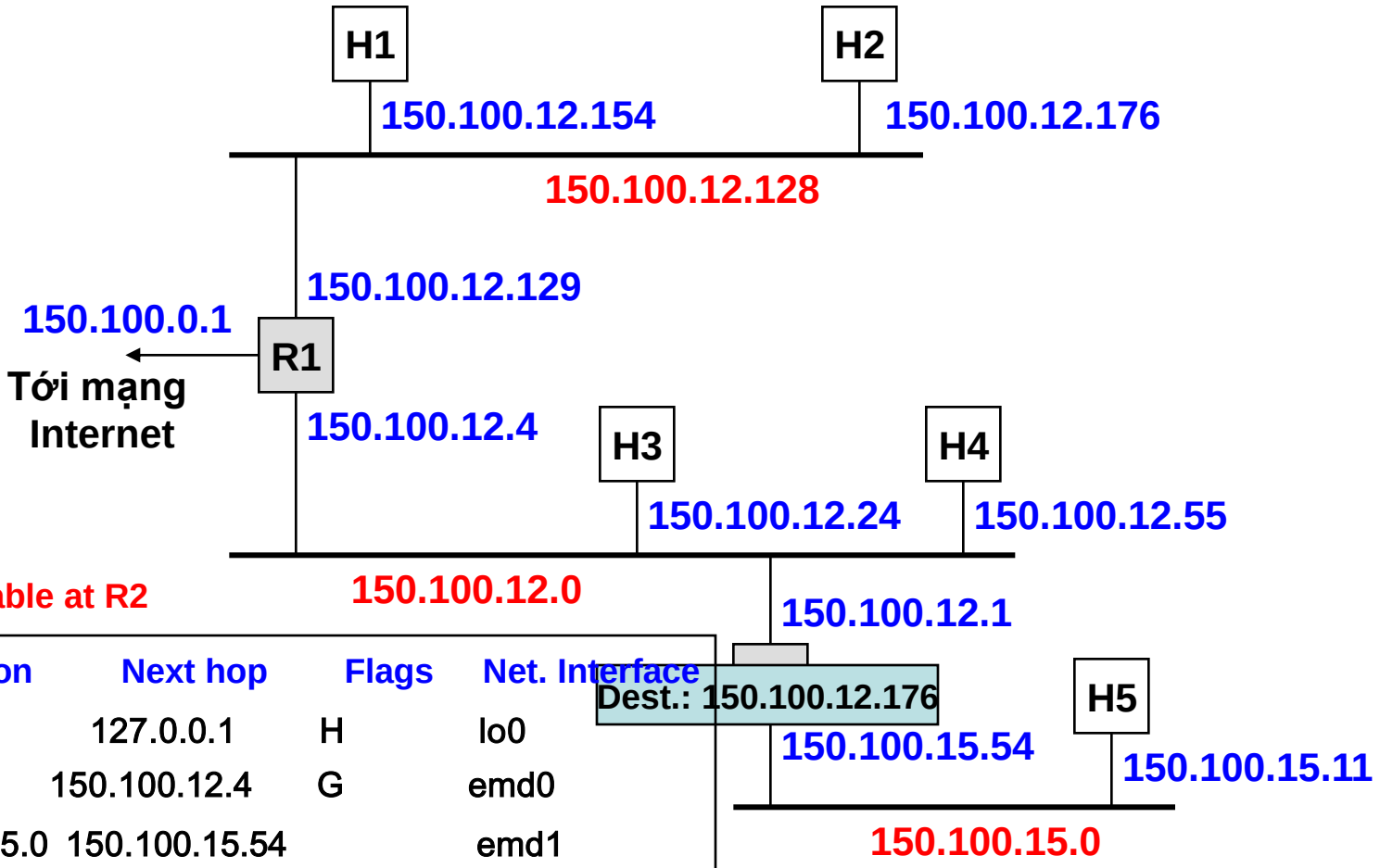
o **Ví dụ**: **Host 5 → Host 2**



Routing table at H5

Destination	Next hop	Flags	Net. Interface
127.0.0.1	127.0.0.1	H	lo0
default	150.100.15.54	G	emd0
150.100.15.0	150.100.15.54		emd0

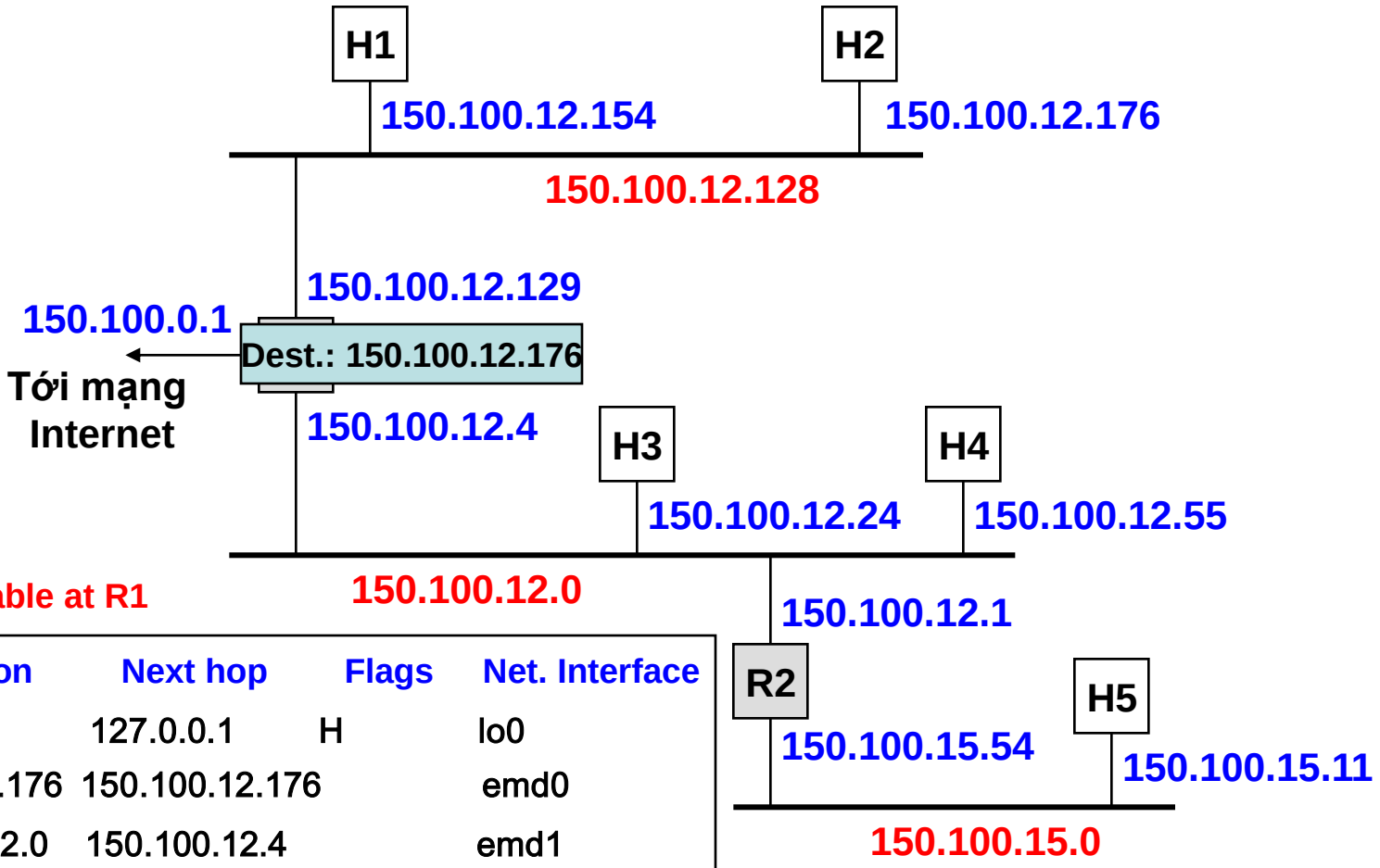
o **Ví dụ**: **Host 5 → Host 2**



Routing table at R2

Destination	Next hop	Flags	Net. Interface
127.0.0.1	127.0.0.1	H	lo0
default	150.100.12.4	G	emd0
150.100.15.0	150.100.15.54		emd1
150.100.12.0	150.100.12.1		emd0

o **Ví dụ**: **Host 5 → Host 2**



Routing table at R1

Destination	Next hop	Flags	Net. Interface
127.0.0.1	127.0.0.1	H	lo0
155.100.12.176	150.100.12.176		emd0
150.100.12.0	150.100.12.4		emd1
150.100.15.0	150.100.12.1	G	emd1

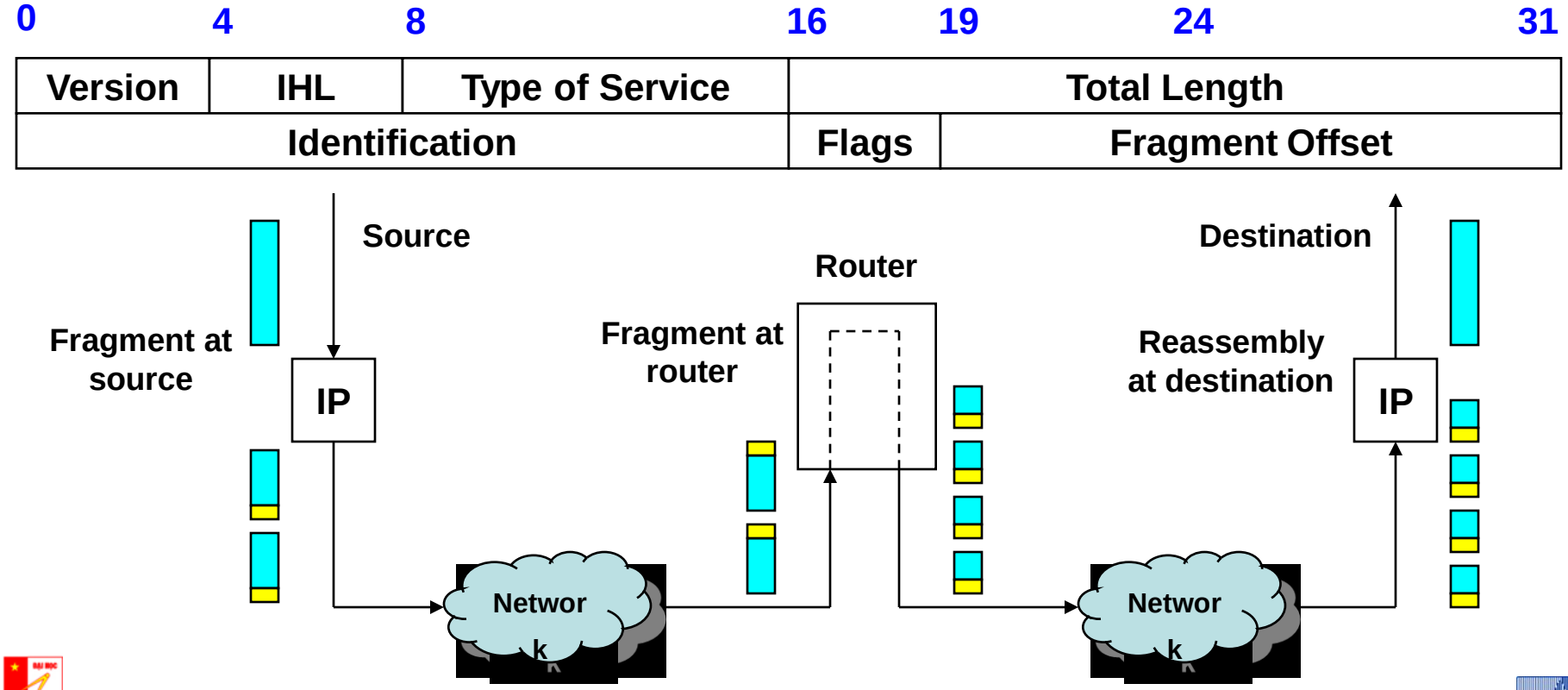
- ✓ Vấn đề địa chỉ IP
  - o 1900: 2 vấn đề nảy sinh
    - ▪ Hết các dải địa chỉ IP
    - ▪ Bảng định tuyến IP phát triển cồng kềnh
  - o Giải pháp tạm thời
    - ▪ Subnetting
    - ▪ Classless Interdomain Routing (**CIDR**)
    - ▪ Network Address Translation (**NAT**)
  - o Giải pháp lâu dài: **IPv6**

✓ **CIDR** và **supernetting**

- o Địa chỉ IP lớp A, B, C không mềm dẻo
- o **CIDR**: NetID với số bit bất kỳ
- o Ví dụ: 205.100.0.0/22
  - 22: số bit trong mask – 255.255.252.0
- o CIDR định tuyến sử dụng **prefix** của địa chỉ IP, không để ý tới class
  - Bảng định tuyến: **<IP address, network mask>**
  - Do độ dài prefix thay đổi, từ bảng định tuyến phải xác định prefix dài nhất trùng nhau
- o **Supernetting**: CIDR sử dụng kỹ thuật supernetting, cho phép 1 địa chỉ IP đại diện cho 1 nhóm địa chỉ IP (lớp A, B, C)
- o Ví dụ: CIDR sử dụng địa chỉ IP 205.100.0.0/22 đại diện cho 4 địa chỉ IP phân lớp C (205.100.0.0, 205.100.1.0, 205.100.2.0, 205.100.3.0)

✓ **Fragmentation** và **reassembly**

- o **Identification**:nhận biết kiểu gói tin
- o **Flag (3 bit)**: **Unused**, **MF,** (more fragment), **DF** (don't fragment)
- o **Fragment offset**: vị trí **fragment** trong **packet** (đơn vị **8 byte**)



✓ Ví dụ: Packet được truyền qua mạng với **Max. Trasfer Unit** (**MTU**)
MTU = **576** byte**, header** = **20** byte**, data** = **1484** byte

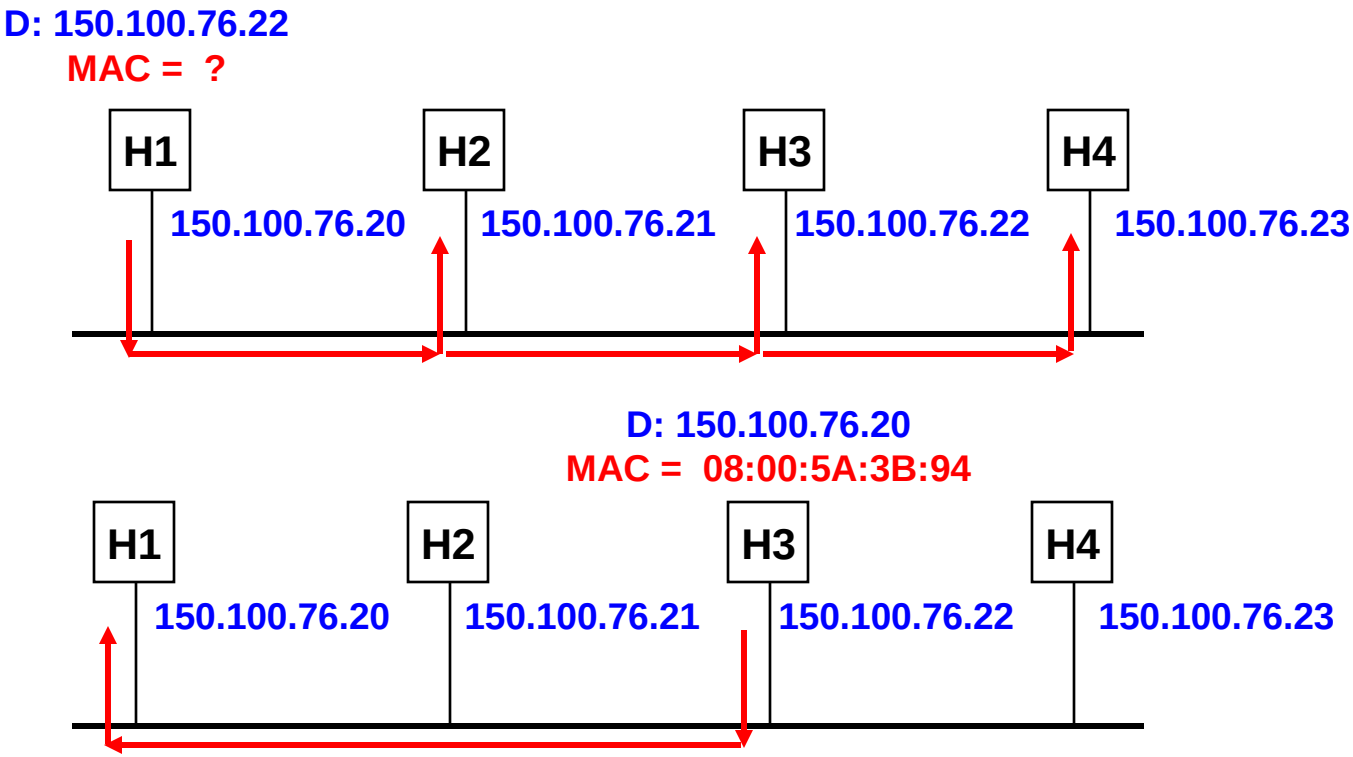
- o **Max. data length/frament**: **576 – 20** = **556** byte
- o Chọn max. data length **= 552** (số nguyên lần của 8)

	Total Length	ID	MF	Fragment offset
Original packet	1504	x	0	0
Fragment 1	572	x	1	0
Fragment 2	572	x	1	69
Fragment 3	400	x	0	138

✓ Address Resolution Protocol (ARP)

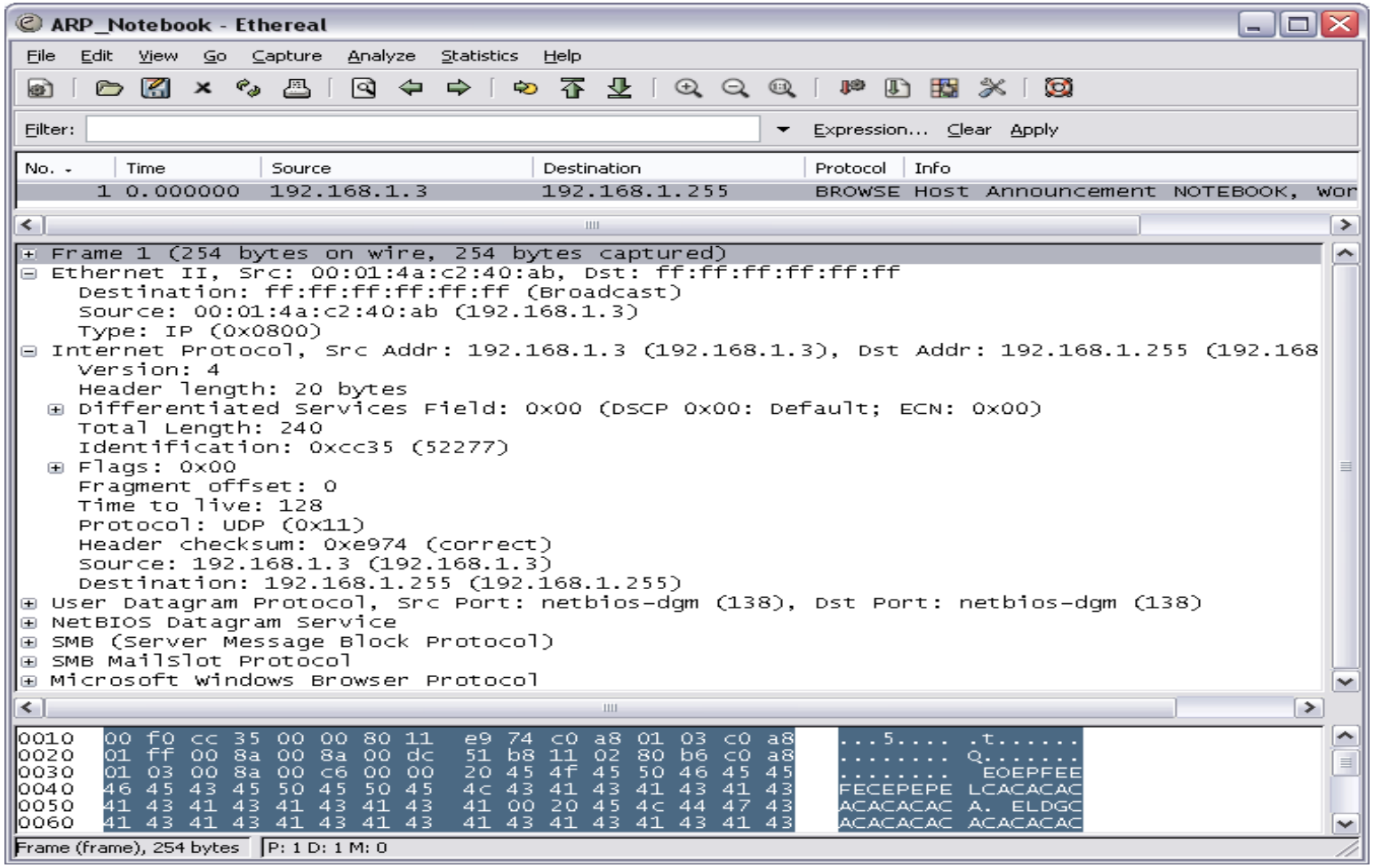
o Địa chỉ IP sử dụng để phân biệt host, nhưng được truyền trên đường truyền vật lý sử dụng địa chỉ MAC (ví dụ trong Ethernet)

o **ARP**: Ánh xạ địa chỉ IP ➔ vật lý



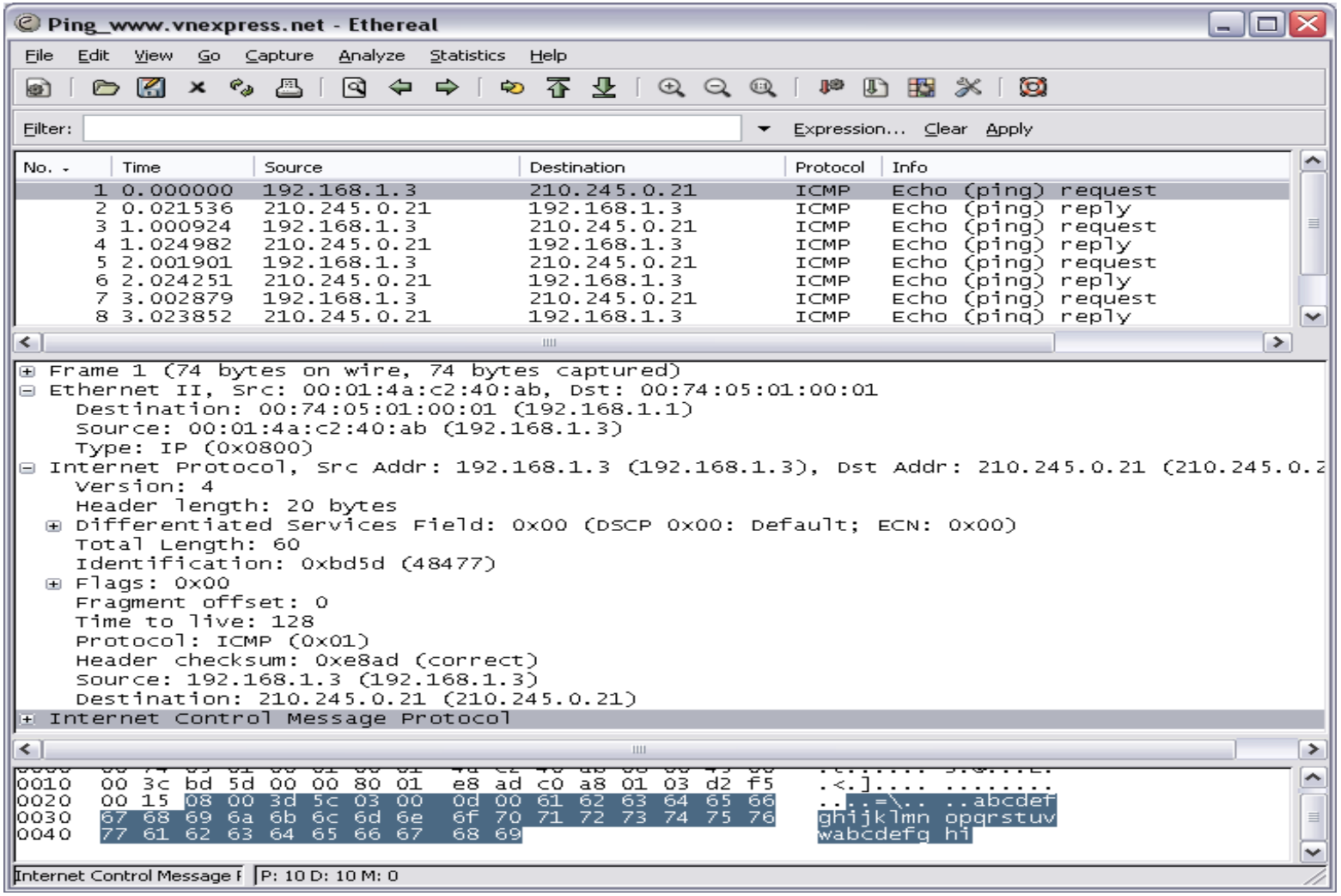
o **Reverse ARP (RARP)**: Nhận địa chỉ IP từ server (**bootstraped**)

```
Interface: 0.0.0.0 --- 0x2
  Internet Address      Physical Address      Type
  206.38.190.192        00-01-4a-c2-40-ab     static

Interface: 192.168.1.3 --- 0x3
  Internet Address      Physical Address      Type
  192.168.1.1           00-01-4a-c2-40-ab     static
```



✓ **Internet Control Message Protocol (ICMP)**

- o Được đóng gói trong IP packet (protocol type = 1)
- o Xử lý các thông báo điều khiển và lỗi
- o Nếu router không gửi được packet, gửi **ICMP "host unreachable"** đến sender
- o Nếu router nhận được packet lẽ ra cần phải gửi tới một router khác, nó gửi **ICMP "redirect"** tới **sender** để thay đổi bảng định tuyến
- o **ICMP "router discovery"** cho phép 1 host tìm hiểu về các router trong mạng, khởi động và cập nhật bảng định tuyến
- o **ICMP echo request** (type = 0) và **reply** (type = 0): sử dụng trong **ping**

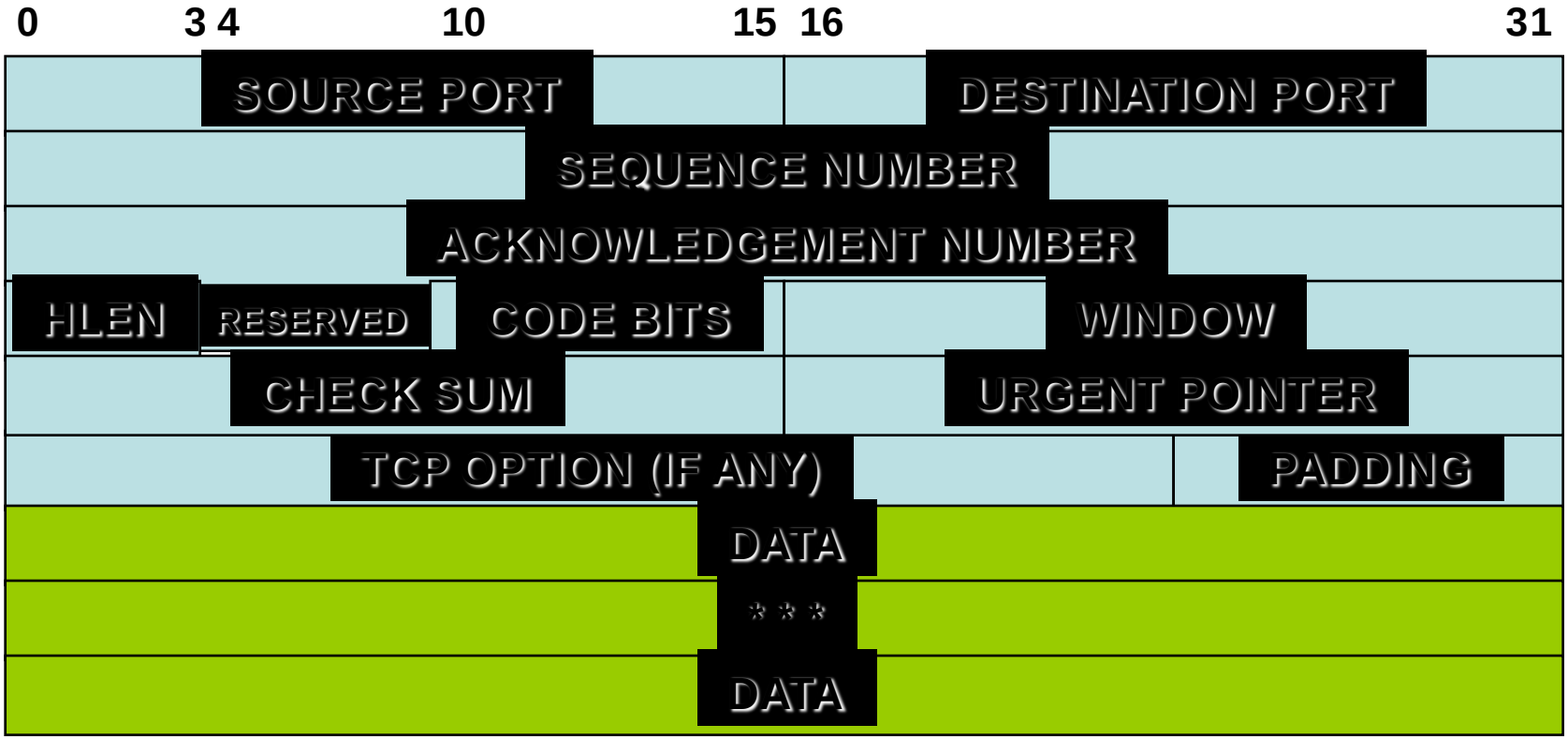

Ping_www.vnexpress.net - Ethereal
File Edit View Go Capture Analyze Statistics Help
Filter: Expression... Clear Apply
No. Time Source Destination Protocol Info
1 0.000000 192.168.1.3 210.245.0.21 ICMP Echo (ping) request
2 0.021536 210.245.0.21 192.168.1.3 ICMP Echo (ping) reply
3 1.000924 192.168.1.3 210.245.0.21 ICMP Echo (ping) request
4 1.024982 210.245.0.21 192.168.1.3 ICMP Echo (ping) reply
5 2.001901 192.168.1.3 210.245.0.21 ICMP Echo (ping) request
6 2.024251 210.245.0.21 192.168.1.3 ICMP Echo (ping) reply
7 3.002879 192.168.1.3 210.245.0.21 ICMP Echo (ping) request
8 3.023852 210.245.0.21 192.168.1.3 ICMP Echo (ping) reply
Frame 1 (74 bytes on wire, 74 bytes captured)
Ethernet II, Src: 00:01:4a:c2:40:ab, Dst: 00:74:05:01:00:01
Destination: 00:74:05:01:00:01 (192.168.1.1)
Source: 00:01:4a:c2:40:ab (192.168.1.3)
Type: IP (0x0800)
Internet Protocol, Src Addr: 192.168.1.3 (192.168.1.3), Dst Addr: 210.245.0.21 (210.245.0.2
Version: 4
Header length: 20 bytes
Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
Total Length: 60
Identification: 0xbd5d (48477)
Flags: 0x00
Fragment offset: 0
Time to live: 128
Protocol: ICMP (0x01)
Header checksum: 0xe8ad (correct)
Source: 192.168.1.3 (192.168.1.3)
Destination: 210.245.0.21 (210.245.0.21)
Internet Control Message Protocol
0010 00 3c bd 5d 00 00 80 01 e8 ad c0 a8 01 03 d2 f5 .<.]....
0020 00 15 08 00 3d 5c 03 00 0d 00 61 62 63 64 65 66 ....=\.. ..abcdef
0030 67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76 ghijklmn opqrstuv
0040 77 61 62 63 64 65 66 67 68 69 wabcdefg hi
Internet Control Message P
P: 10 D: 10 M: 0

- ✓ **Một số giao thức của tầng IP**
  - o Routing information protocol (**RIP**)
  - o Open shortest Path First (**OSPF**)
  - o Border Gateway Protocol (**BGP**), Exterior Gateway Protocol (**EGP**)
  - o Dynamic Host Configuration Protocol (**DHCP**), Network Address Translation (**NAT**), Mobile IP
- ✓ **Transport Control Protocol** (**TCP**) và **User Data Protocol** (**UDP**)
  - o TCP Reliable Stream Service
  - o TCP Protocol
  - o TCP Connection Management
  - o TCP Error/Flow/Congestion Control
  - o UDP

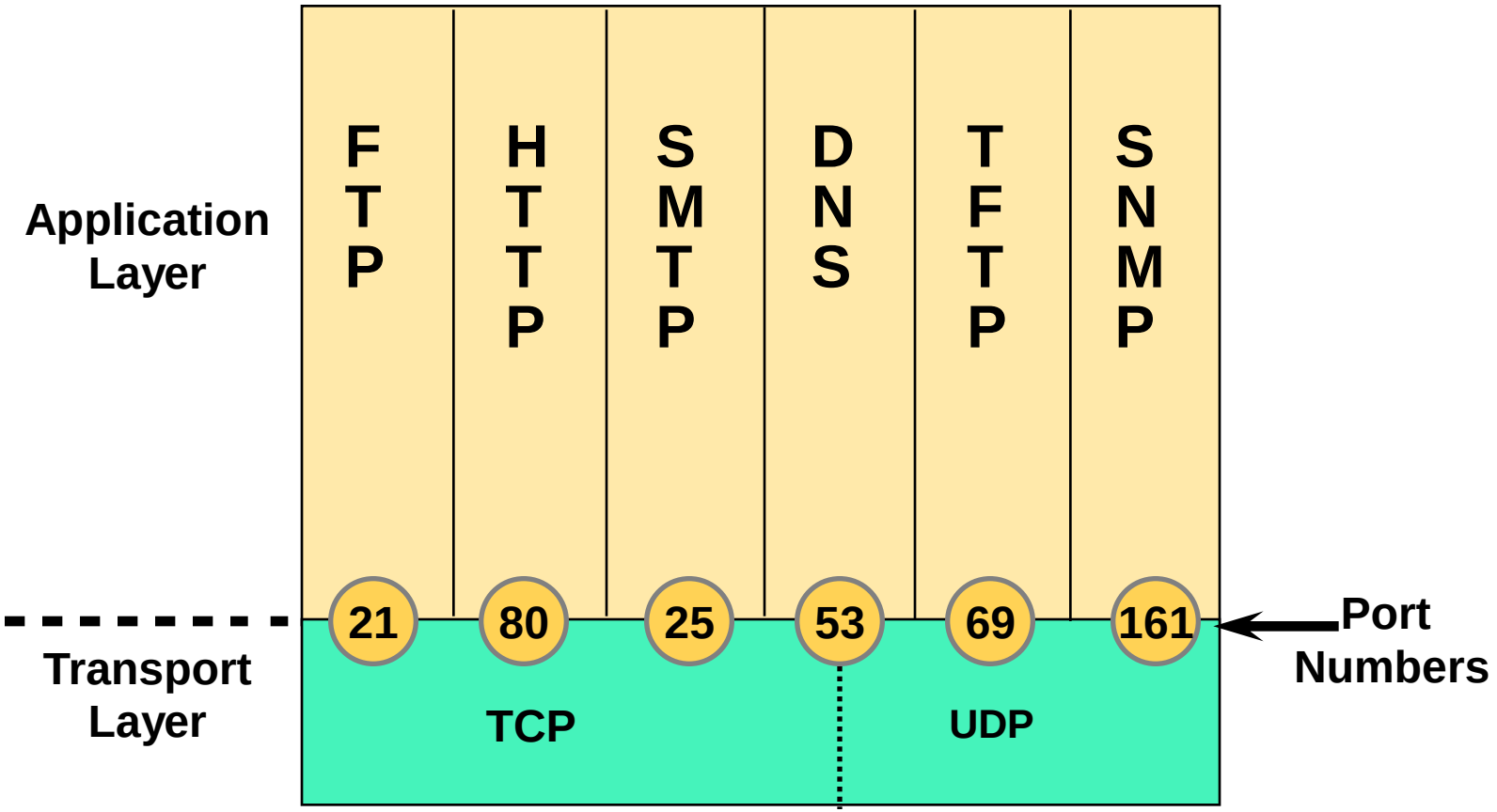
o **Transmission Control Protocol**

- TCP cung cấp dịch vụ chuyển giao thông tin có kết nối (connection - oriented)
- Bao gồm việc kiểm tra và sửa lỗi.
- TCP cung cấp dịch vụ tin cậy với cơ chế gọi là "Positive Ackowledgment with Retransmission" (PAR). Đơn giản là trạm nguồn tiếp tục gửi thông tin đi cho tới khi nó nhận đ- ợc thông báo dữ liệu đã đ- ợc nhận chính xác tại trạm đích.

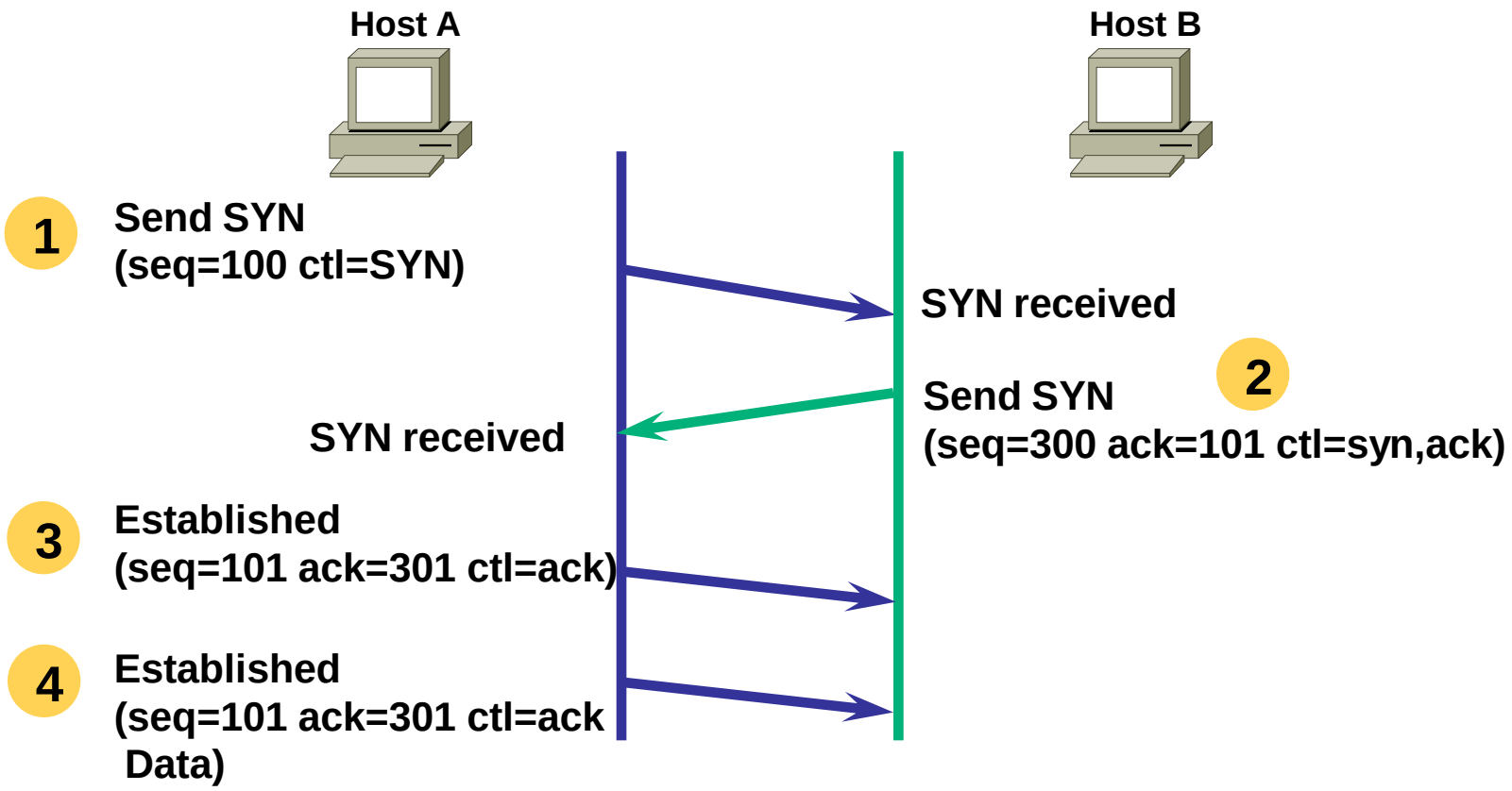
Cấu trúc segment TCP



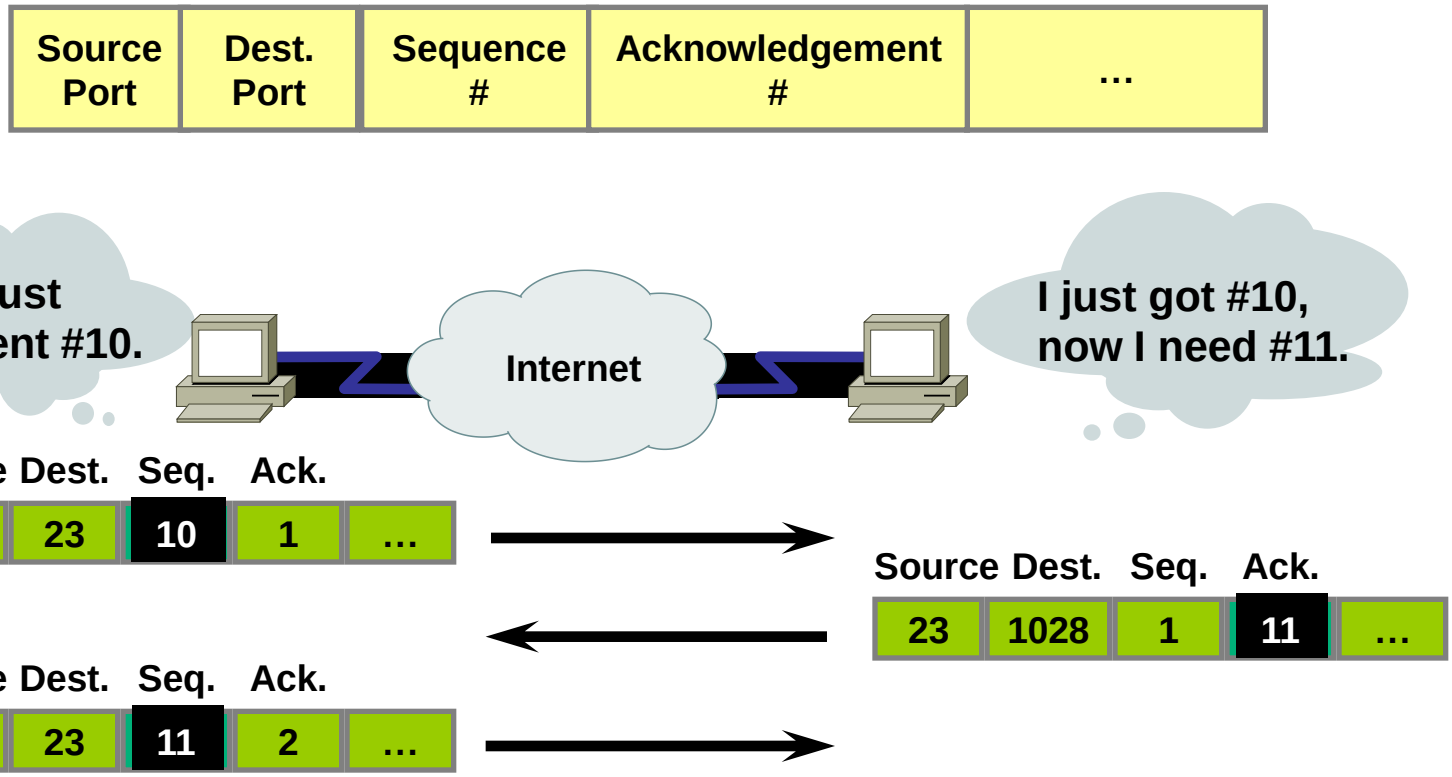
Port number, phân kenh và dồn kênh



TCP Handshake/Open Connection

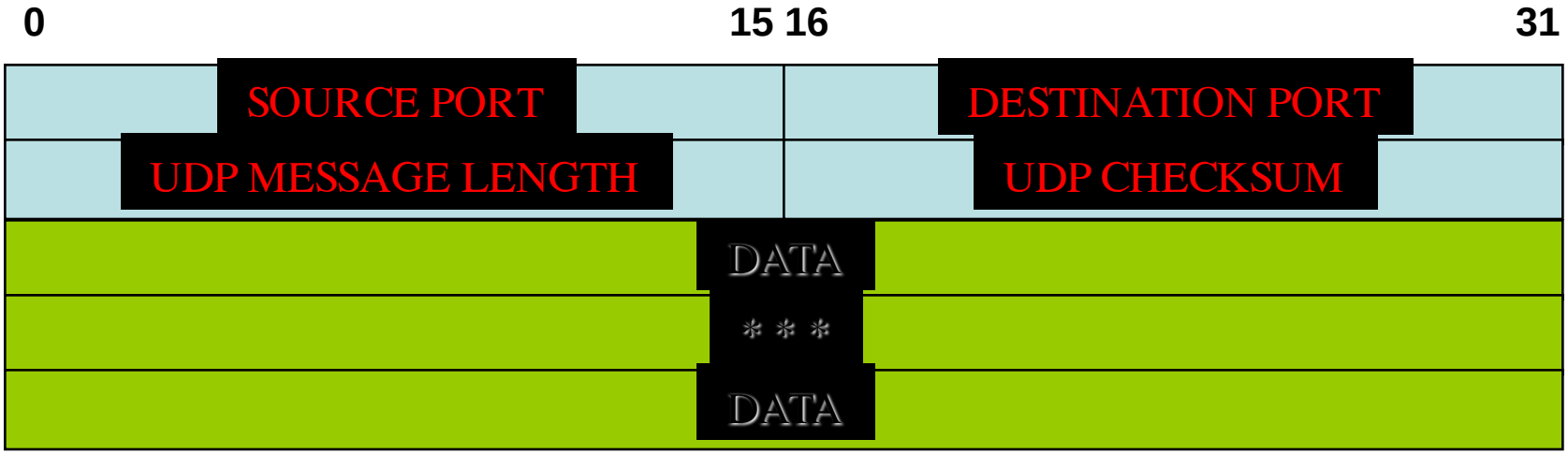


TCP Sequence and Acknowledgment Numbers



o User data protocol

- UDP cho phép ch- ơng trình ứng dụng truy cập trực tiếp đến gói tin của dịch vụ chuyển giao giống nh- dịch vụ mà giao thức IP cung cấp.
- Nó cho phép ứng dụng trao đổi thông tin qua mạng với ít thông tin điều khiển nhất.
- UDP là giao thức không kết nối, kém tin cậy vì nó không có cơ chế kiểm tra tính đúng đắn của dữ liệu truyền.



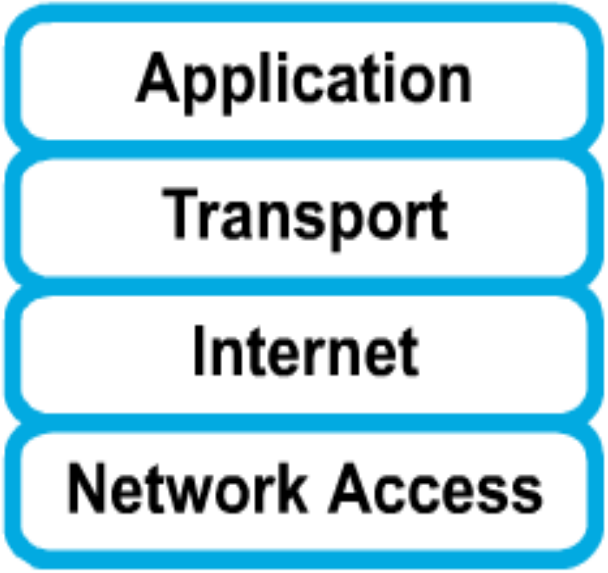
GIAO THỨC TCP/IP
HỆ THỐNG ĐỊA CHỈ IP

NHỮNG CHỦ ĐỀ CHÍNH

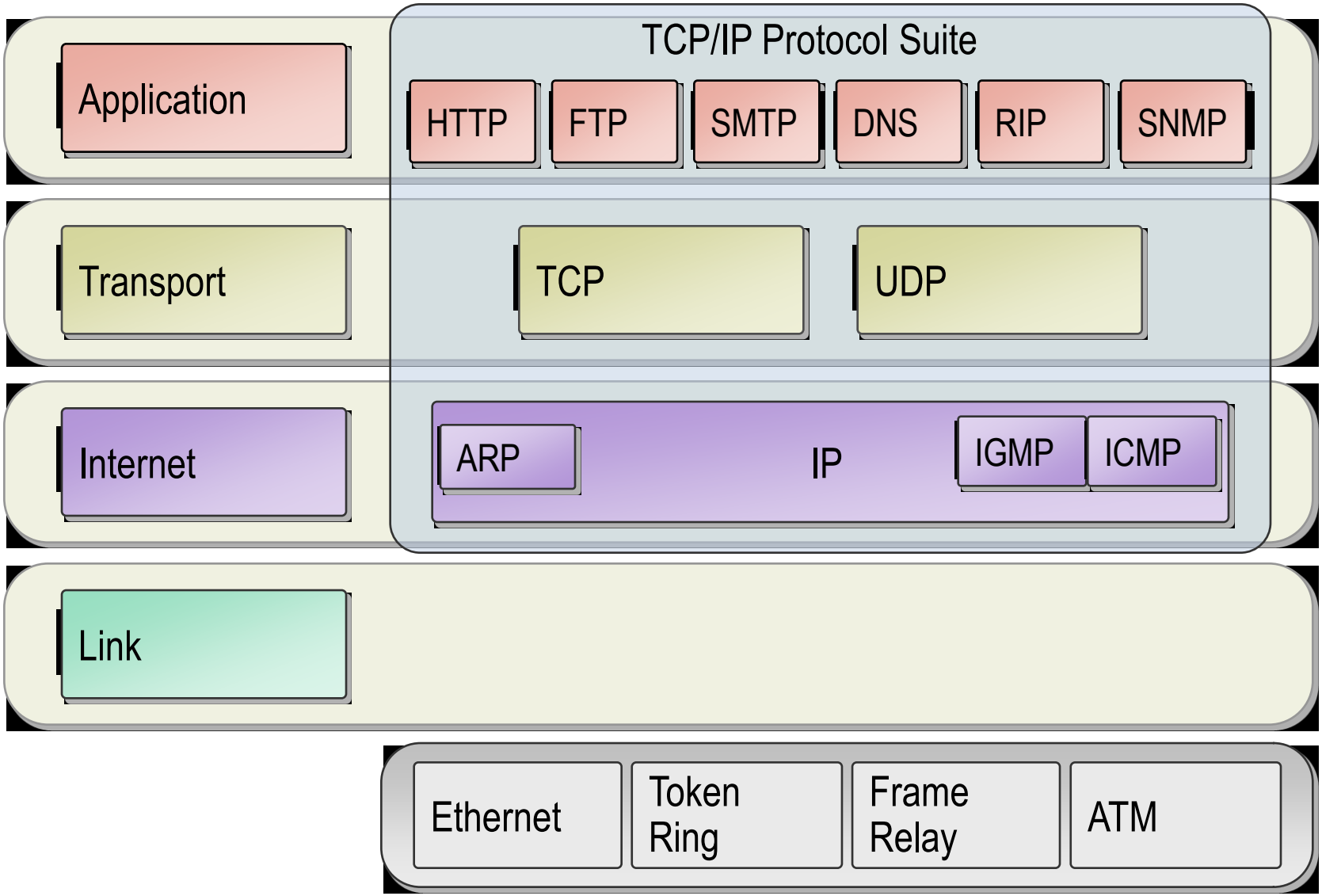
- Giới thiệu TCP/IP
- Kiến trúc TCP/IP
- Hệ thống địa chỉ
- Những công cụ TCP/IP

GiỚI THIỆU TCP/IP

- TCP/IP là bộ giao thức chuẩn giúp các hệ thống (platforms) khác nhau truyền thông với nhau, là giao thức chuẩn của truyền thông Internet.
- TCP/IP có 2 giao thức chính TCP và IP :
  - TCP đảm nhiệm chuyển data giữa hai hệ thống.
  - IP đảm nhiệm tìm đường chuyển các gói đi.

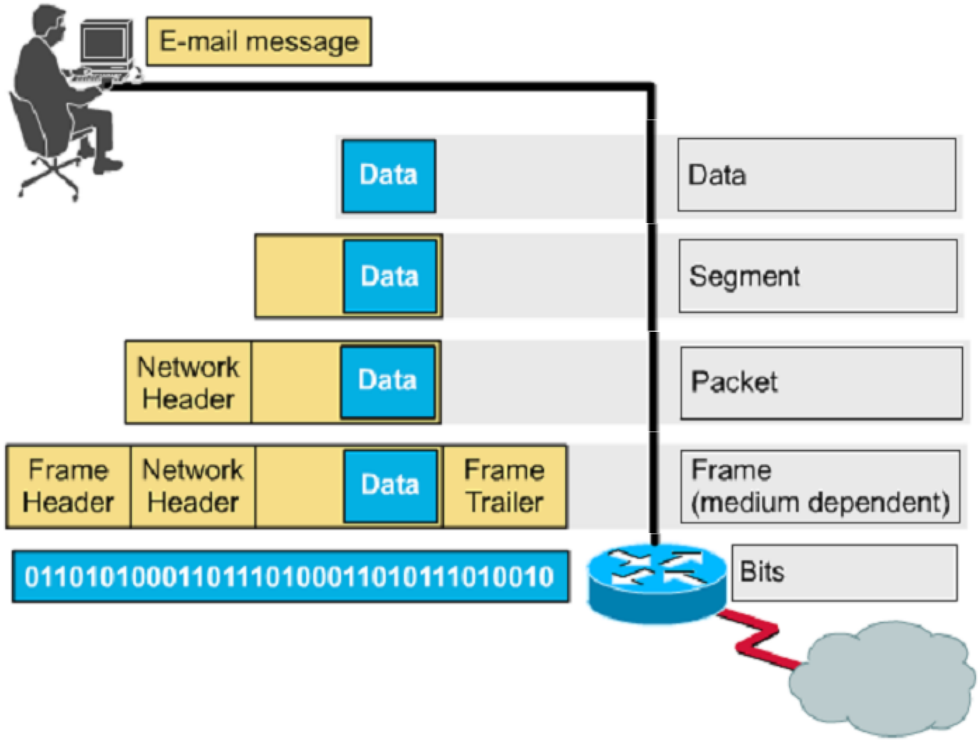


MÔ HÌNH TCP/IP



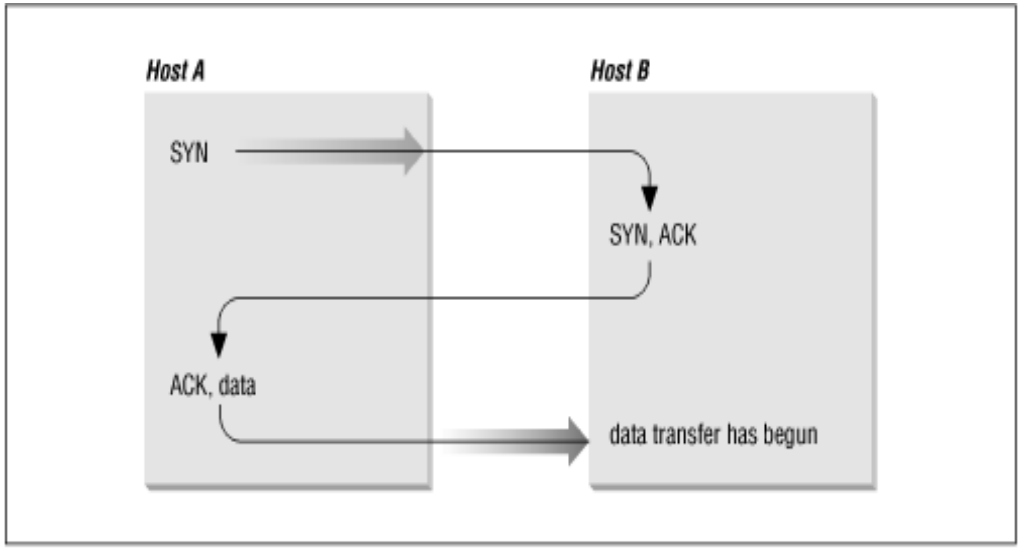
ĐÓNG GÓI TCP/IP

- Đơn vị data được tạo trên lớp Application gọi là *Message*.
- TCP, UDP (Transport) tạo ra đơn vị data gọi là *Segment* hay *User Datagram*.
- IP (Internet) tạo ra đơn vị data gọi là *IP Datagram*.
- Đơn vị data được tạo trên lớp Link gọi là *Frame*.



KiẾN TRÚC TCP/IP

- *Application* : Xác thực, nén, các dịch vụ người dùng.
- *Transport* : TCP đảm nhiệm chuyển data giữa hai hệ thống và cung cấp truy xuất mạng cho Application.
- *Internet* : Packet routing.
- *Link* : Giao diện Kernel OS/device driver với mạng.



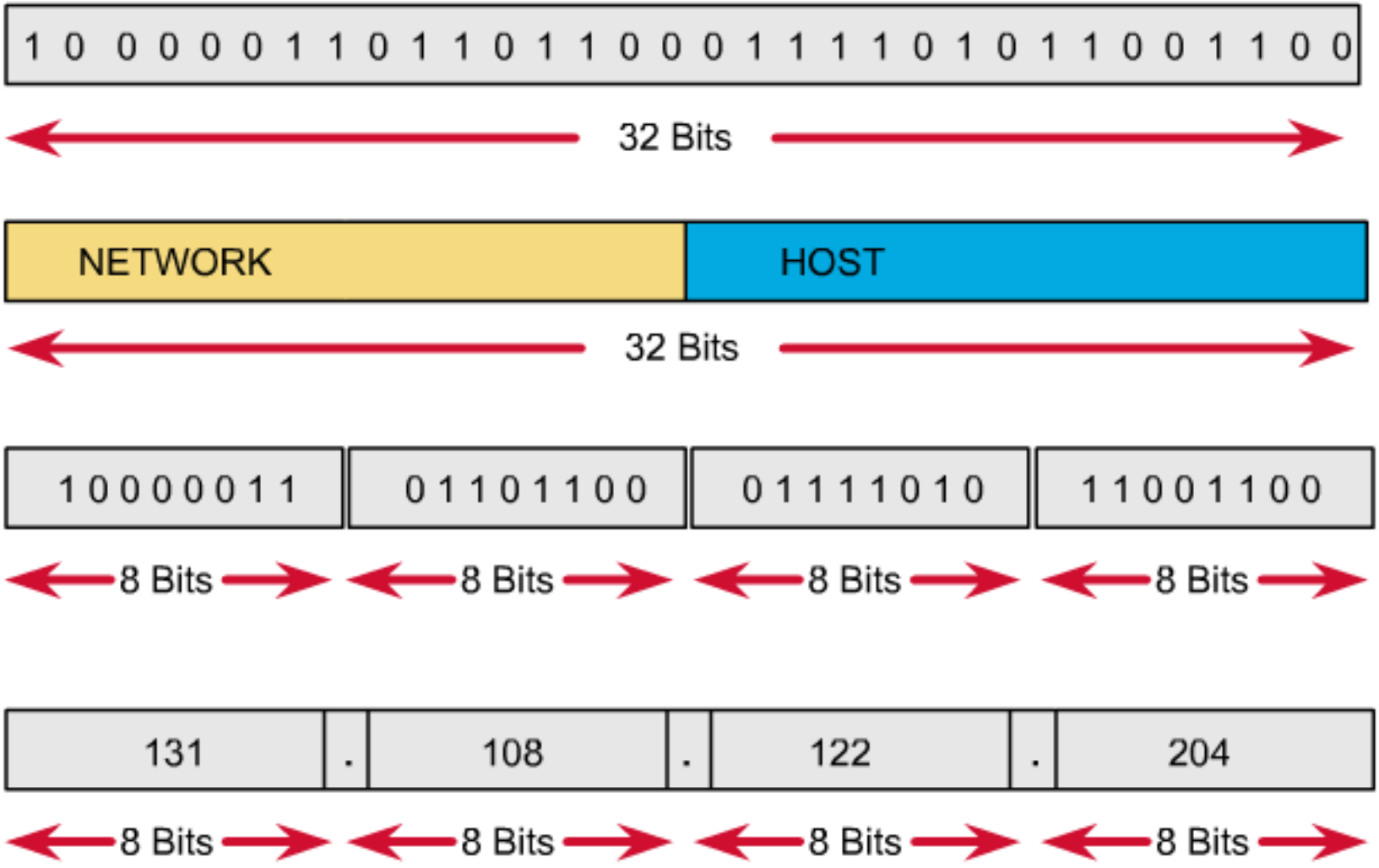
IP ADDRESSING

Binary : 11000000.10101000.000000001.00001000 and 11000000.10101000.00000001.00001001

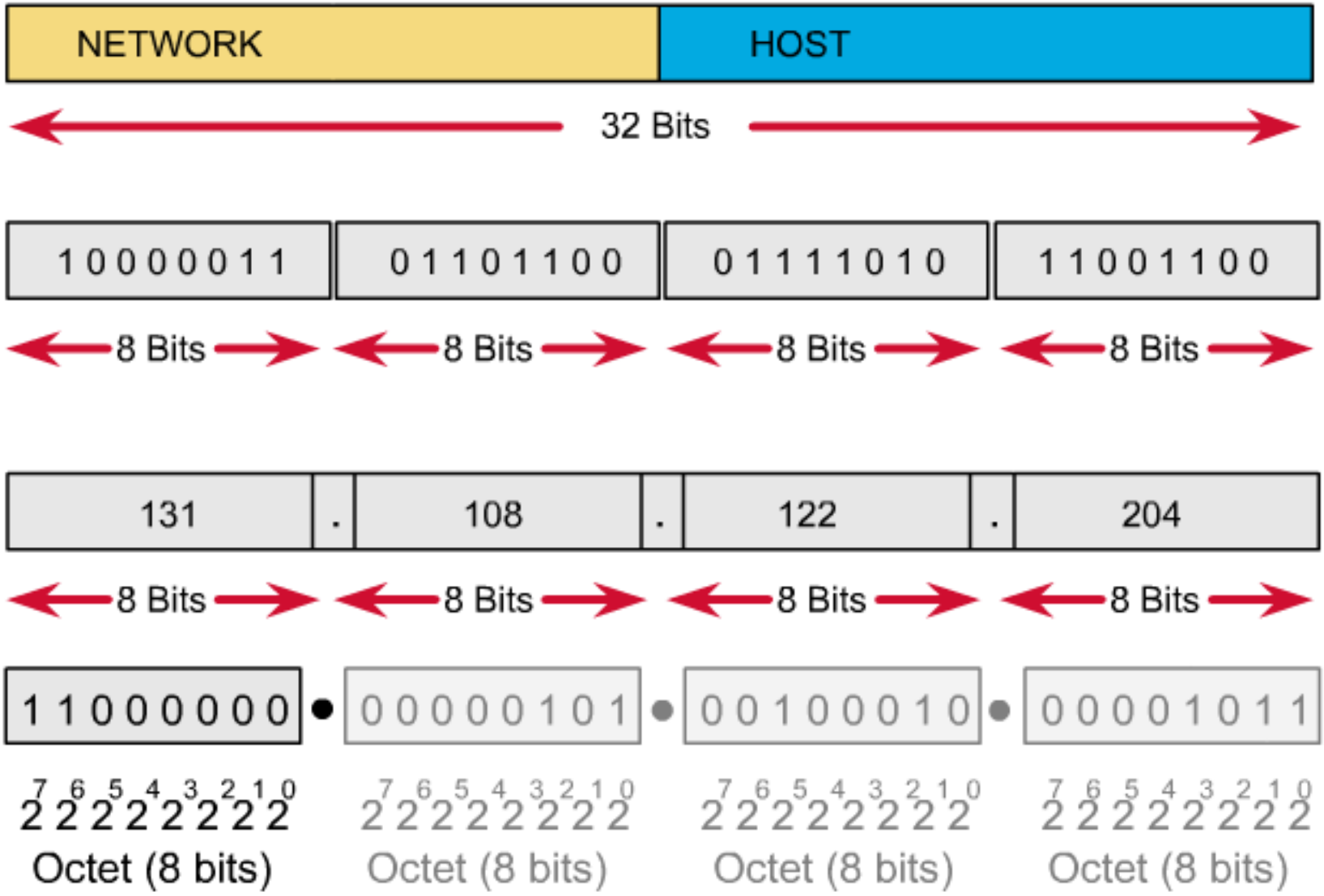
Decimal : 192.168.1.8 and 192.168.1.9

Both the binary and decimal numbers represent the same values, but it is much easier to see with the dotted decimal values. This is one of the common problems found in working directly with binary numbers. The long strings of repeated ones and zeros make transposition and omission errors more likely.

IP ADDRESS FORMAT



BINARY AND DECIMAL CONVERSION



FAST CONVERSION

$2^{(7)}$	$2^{(6)}$	$2^{(5)}$	$2^{(4)}$	$2^{(3)}$	$2^{(2)}$	$2^{(1)}$	$2^{(0)}$
128	64	32	16	8	4	2	1

Exercise: DEC – BIN

203

Exercise: DEC – BIN

203

128 64 32 16 8 4 2 1

Exercise: BIN – DEC

10100010

Exercise: BIN – DEC

10100010

162

Exercise: BIN – DEC

A :10101010
B :10110110
C :10111010
D :00111010

Exercise: BIN – DEC

A :10101010	170
B :10110110	182
C :10111010	186
D :00111010	58

NETWORK ADDRESS

- Địa chỉ mạng: là địa chỉ mà Host ID chỉ chứa toàn bit 0

192.168.1.0

HOST ADDRESS

- Địa chỉ host: là địa chỉ mà phần HostID vừa tồn tại bit 0 và vừa tồn tại bit 1

192.168.1.1

SUBNET MASK

Địa chỉ netmask(mặt nạ mạng) là địa chỉ mà phần bit ở NetID toàn là bit 1 và phần bit ở HostID toàn là bit 0

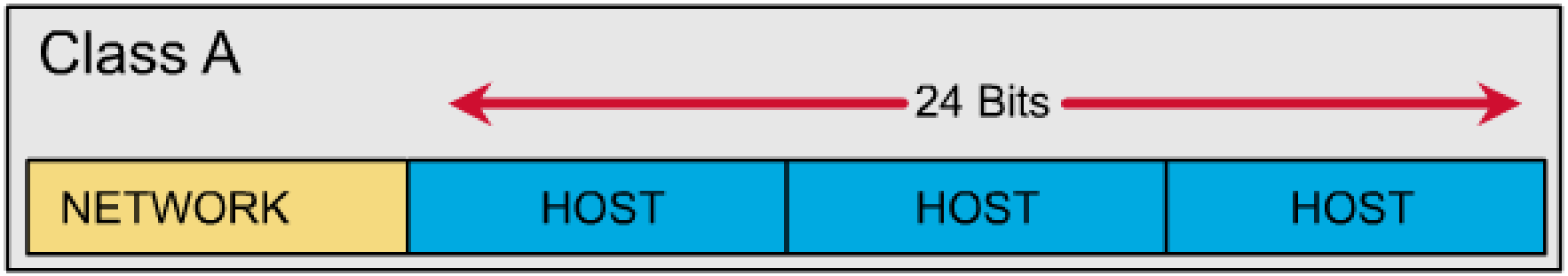
255.255.255.0

BROADCAST ADDRESS

- Địa chỉ broadcast: là địa chỉ mà phần HostID chứa toàn bit 1

192.168.1.255

IP ADDRESS: CLASS A

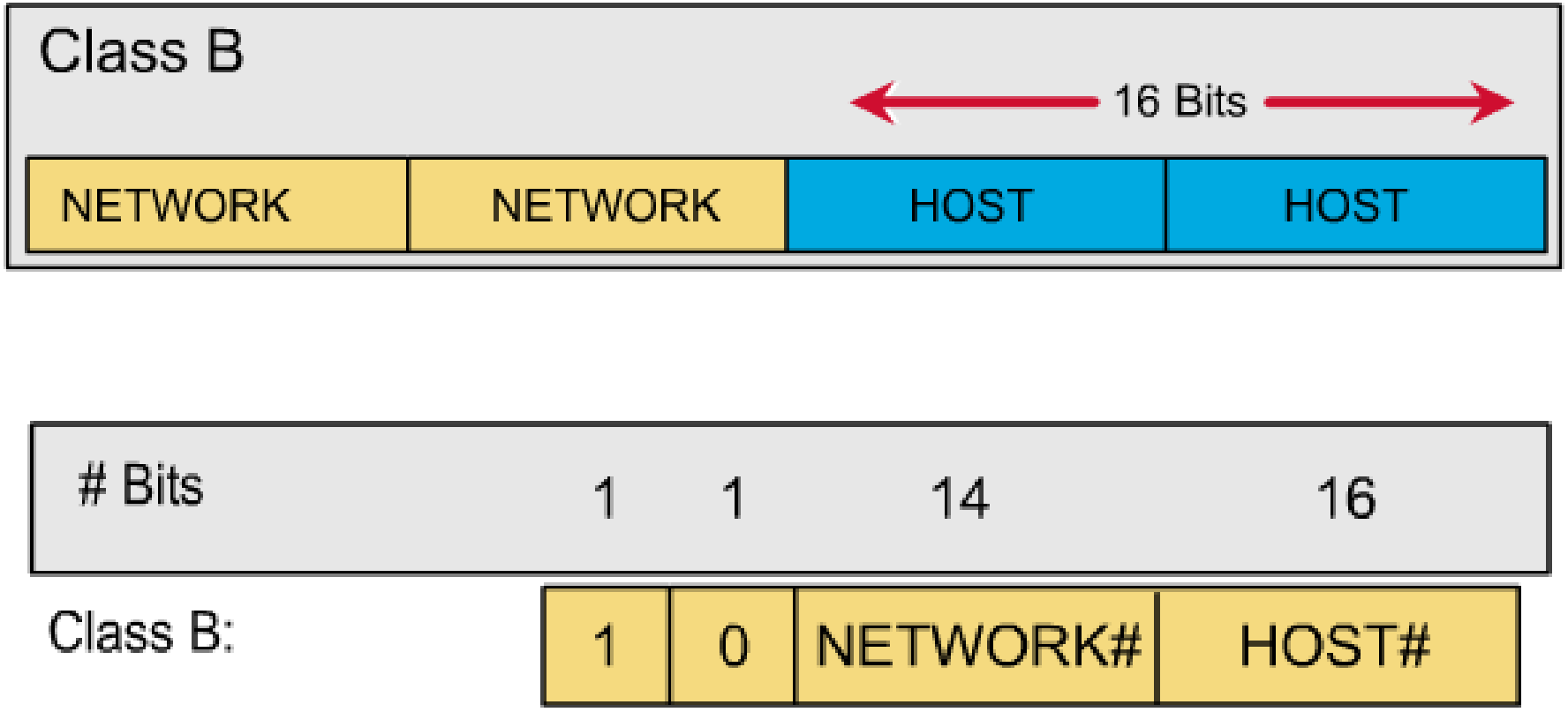


# Bits	1	7	24
Class A:	0	NETWORK#	HOST#

IP ADDRESS: CLASS A

- Bit đầu tiên của class A luôn là 0.
- Dùng 8 bit để sử dụng cho NetID.
- Dãy địa chỉ mạng có thể bắt đầu từ 1.0.0.0 đến 127.0.0.0
- Sử dụng 3 octet làm phần HostID.
- Mỗi Network ở class A có 16,777,214 địa chỉ Host.

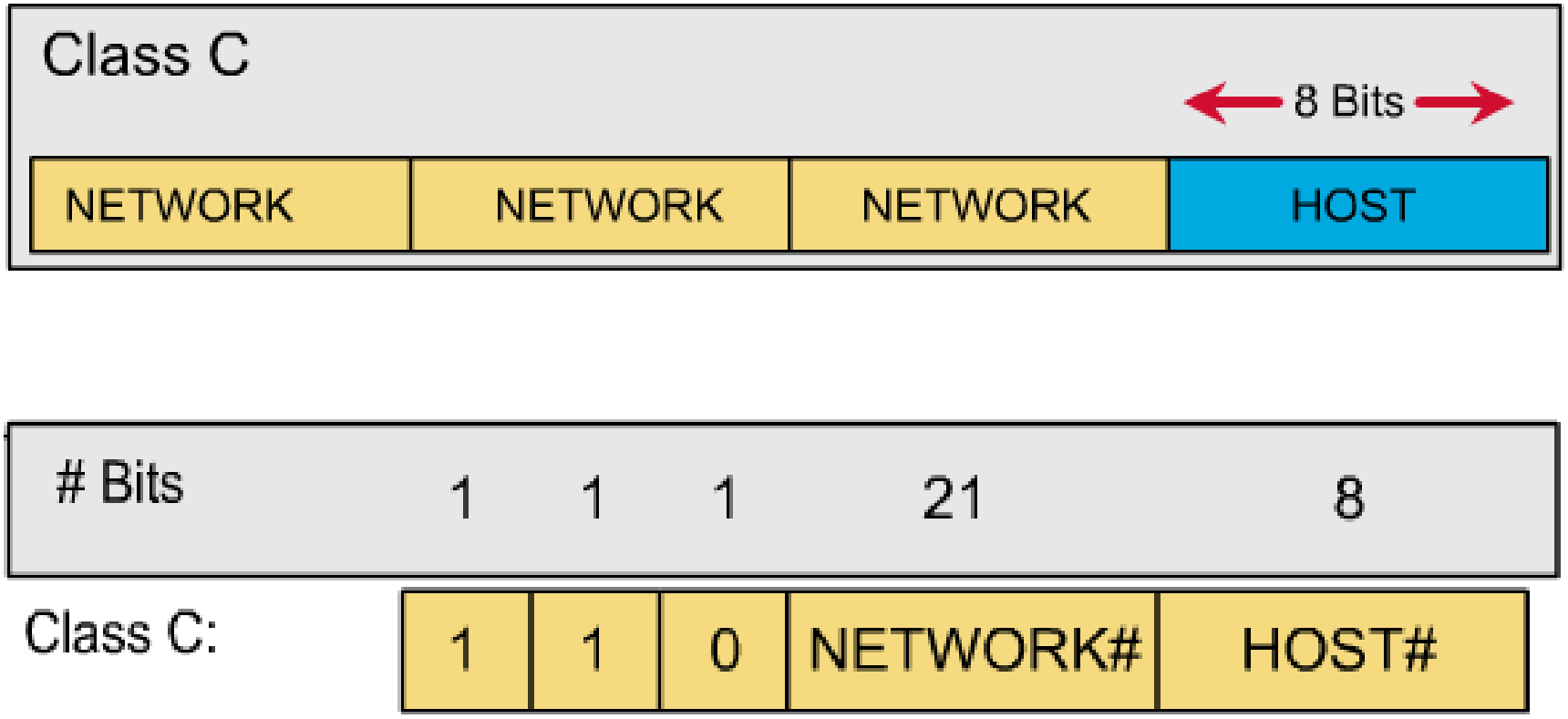
IP ADDRESS: CLASS B



IP ADDRESS: CLASS B

- 2 bit đầu tiên của class B luôn là 10.
- 2 octect đầu tiên được sử dụng làm NetID.
- Dãy địa chỉ mạng có thể bắt đầu từ 128.0.0.0 đến 191.255.0.0
- Sử dụng 2 octet làm phần HostID.
- Mỗi Network ở class B có 65534 địa chỉ Host.
- 255.255.0.0

IP ADDRESS: CLASS C



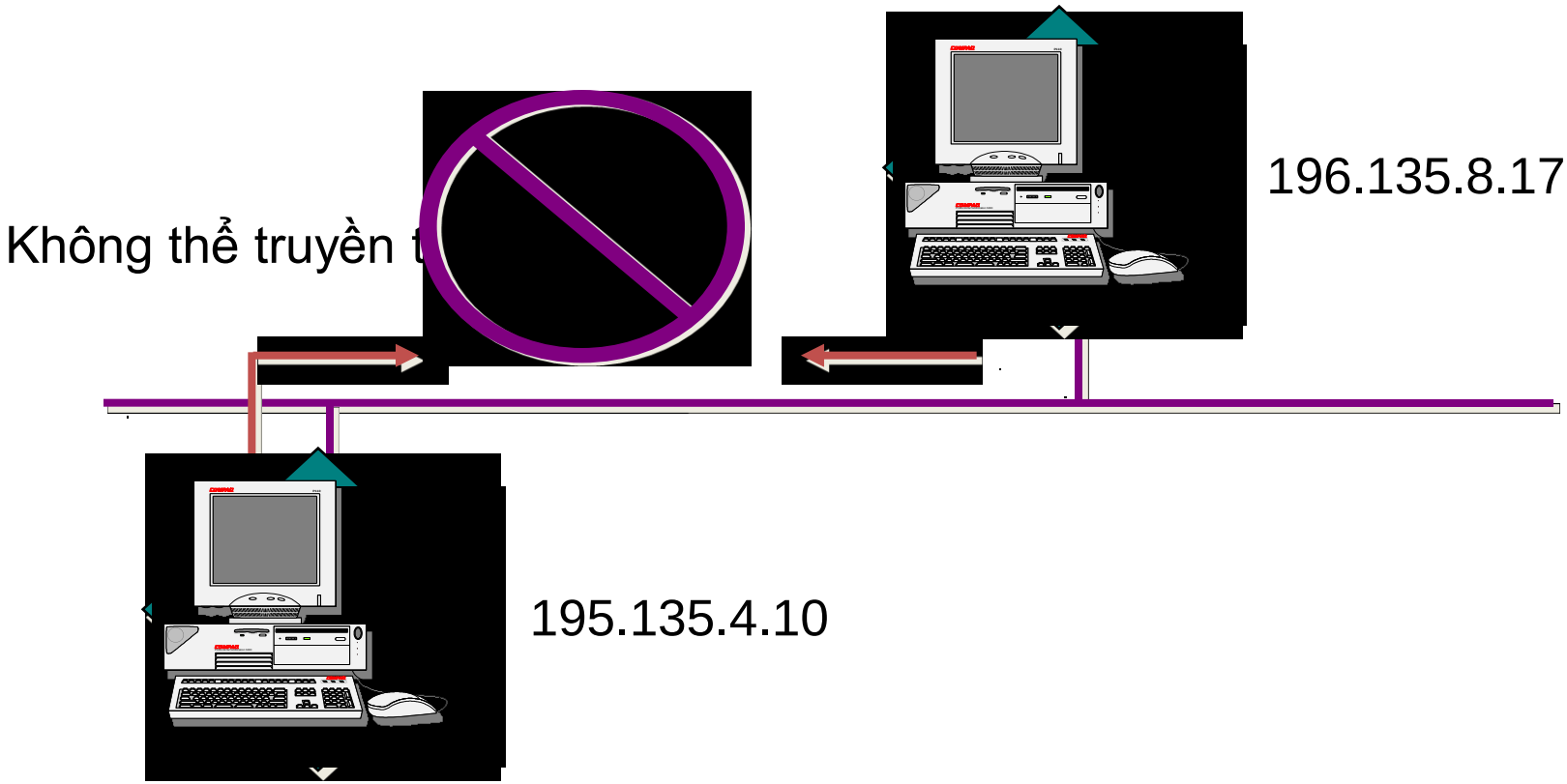
IP ADDRESS: CLASS C

- 3 bit đầu tiên của class C luôn là 110.
- 3 octect đầu tiên được sử dụng làm NetID.
- Dãy địa chỉ mạng có thể bắt đầu từ 192.0.0.0 đến 223.255.255.0
- Sử dụng 1 octet cuối làm phần HostID.
- Mỗi Network ở class C có 254 địa chỉ Host.

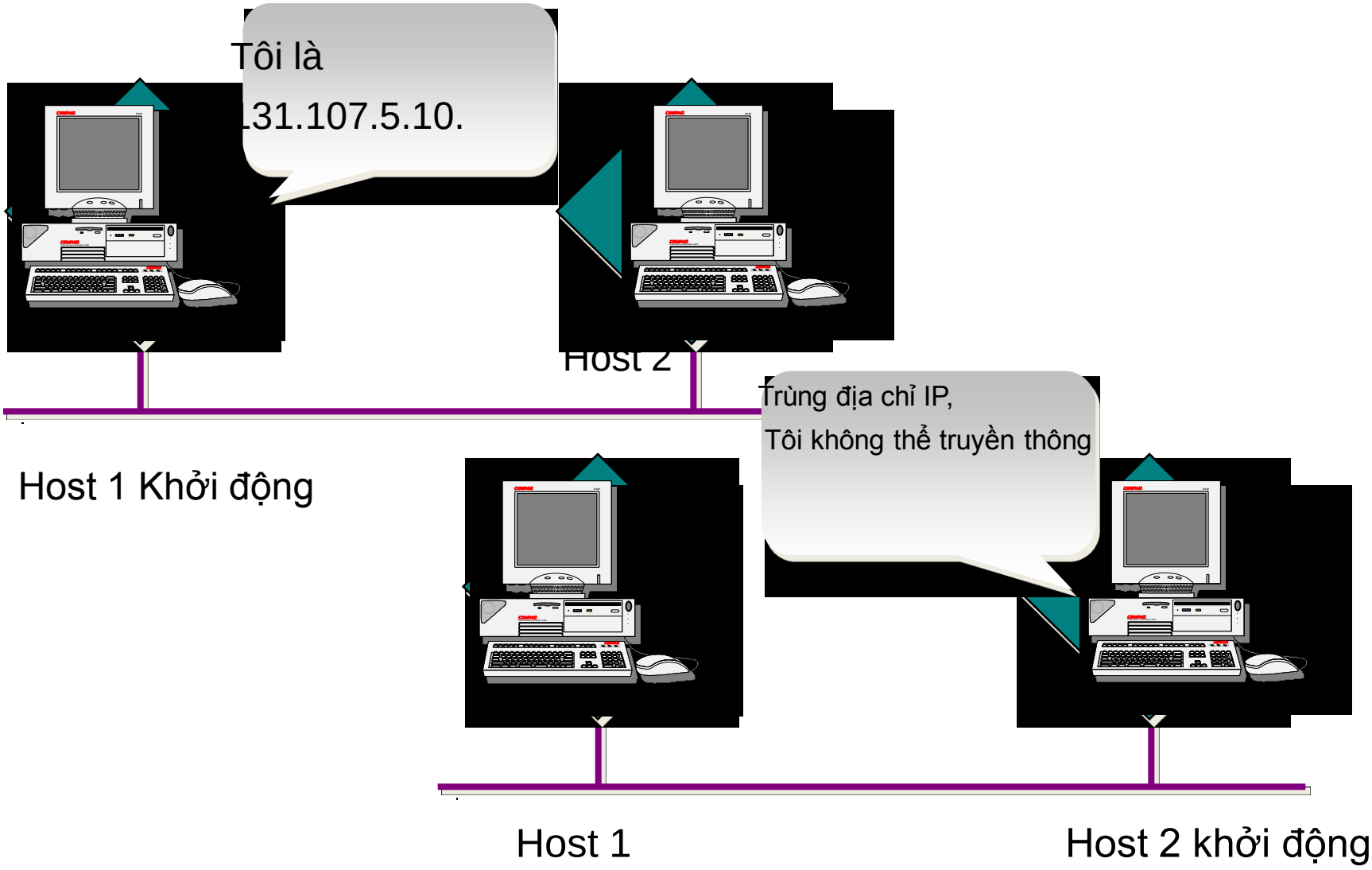
IP ADDRESS SUMMARY

- Class A : 1.0.0.0 - 126.0.0.0
- Loopback network : 127.0.0.0
- Class B : 128.0.0.0 - 191.255.0.0
- Class C : 192.0.0.0 - 223.255.255.0
- Class D, multicast :224.0.0.0 - 239.0.0.0
- Class E, reserved :240.0.0.0 - 255.0.0.0

THIẾT LẬP ĐỊA CHỈ MẠNG KHÔNG ĐÚNG



TRÙNG ĐỊA CHỈ IP

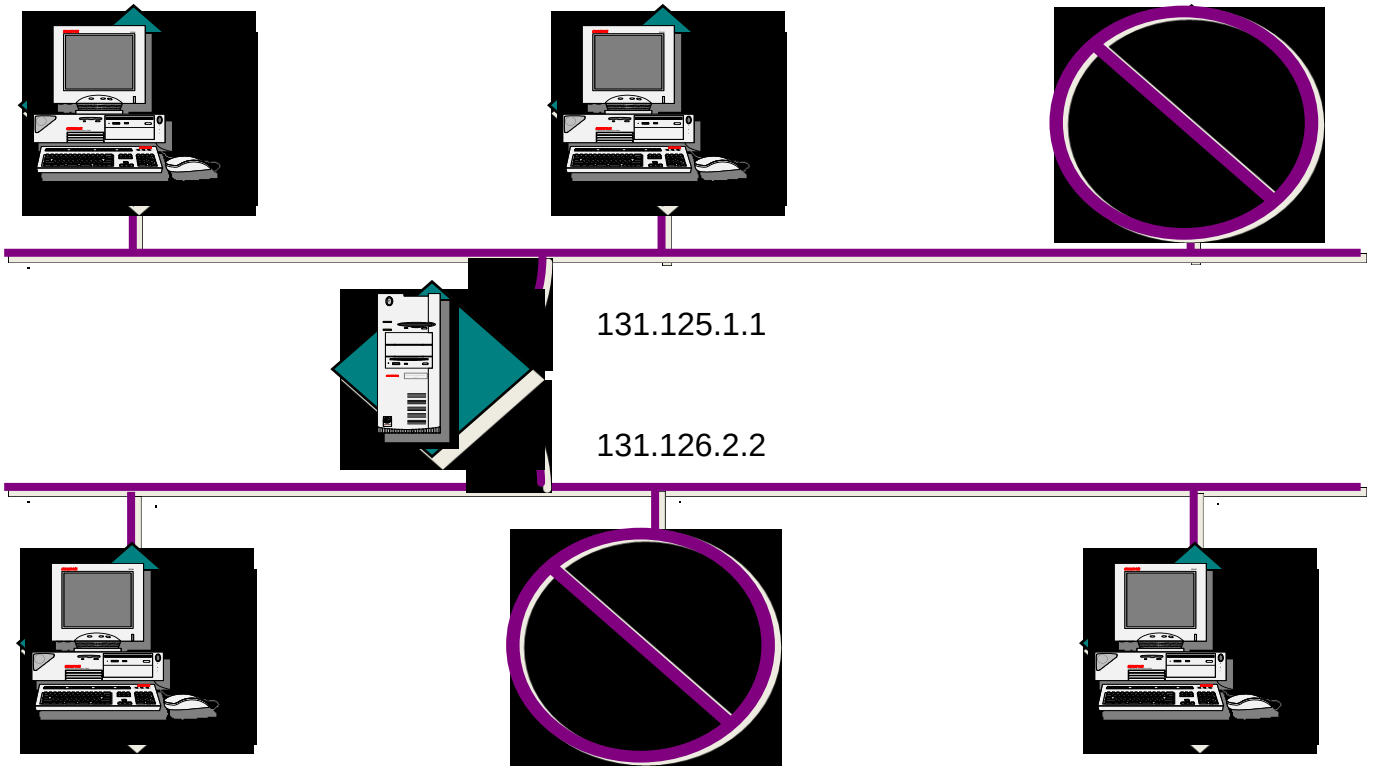


SAI ĐỊA CHỈ?

IP address = 131.125.10.10
Default gateway = 131.125.1.1
Computer 1

IP address = 131.125.1.3
Default gateway = 131.125.1.1
Computer 2

IP address = 131.125.1.4
Default gateway = 131.126.2.2
Computer 3



IP address = 131.126.2.2
Default gateway = 131.126.12.2
Computer 4

IP address = 131.125.5.2
Default gateway = 131.126.2.2
Computer 5

IP address = 131.126.2.5
Default gateway = 131.126.2.2
Computer 6

PRIVATE ADDRESS

- Theo chuẩn RFC-1918.
- Class A: 10.0.0.0.
- Class B: 172.16.0.0 - 172.31.0.0.
- Class C: 192.168.0.0 - 192.168.255.0.

TẠI SAO CẦN PHẢI CHIA MẠNG CON

- Chia mạng mặc nhiên thành mạng nhỏ hơn.
- Phù hợp với mô hình mạng hiện tại của Công ty.
- Giảm traffic, cô lập mạng khi cần thiết.
- Phải đặt bộ định tuyến(Router) giữa các mạng con này.
- Phương pháp :
  - Lấy các bits cao nhất của Phần Host cho phần Netwok.
  - Tính các NetIDs và các HostIDs.

110nnnnn	nnnnnnnn	nnnnnnnn	*nn*hhhhhh

CÁC PHÉP TOÁN TRÊN BIT

- Boolean operators:
  - –AND.
  - –OR.
  - –NOT.

AND OPERATOR

- 1 AND 1 = 1
- 1 AND 0 = 0
- 0 AND 1 = 0
- 0 AND 0 = 0

OR OPERATOR

- 1 OR 1 = 1
- 1 OR 0 = 1
- 0 OR 1 = 1
- 0 OR 0 = 0

NOT OPERATOR

- NOT 1 = 0
- NOT 0 = 1

EXAMPLE

- 1010 AND 0110 = 0010
- 1010 OR 0110 = 1110

TẠI SAO CHÚNG TA CẦN PHẢI BIẾT PHÉP TOÁN TRÊN BIT?

IP Address	AND	Subnet Mask	=	Network and Subnet address

- Example:

  IP addr→ 192.168.1.1 AND 255.255.255.0 ←SM

  = 192.168.1.0 ← Network address

THỰC HÀNH CHIA MẠNG CON

- Cho network 172.16.0.0.
- Yêu cầu:

☞ Chia ra 8 mạng con

☞ Mỗi mạng con có trên 1000 địa chỉ host

BƯỚC 1

- Xác định địa chỉ 172.16.0.0 thuộc về

Class B

- Địa chỉ subnet mask mặc định sẽ là

255.255.0.0

BƯỚC 2

- Số subnets <= 2^n - 2 với n là số bit ta mượn.
- Số hosts <= 2^m - 2 với m là số bit còn lại sau khi mượn n bt*(m=host bit – n)*
- Quyết định xem cần mượn bao nhiêu bit ở phần HostID để thỏa
  - 8 subnets.
  - Trên 1000 hosts cho mỗi subnet.

BƯỚC 2

- Chọn n = 4:
  - Số subnet:

$$2^4 - 2 = 14$$

  - Số lượng host:

$$2^{(16-4)} - 2 = 4094$$

- *Hoặc n = 5 , n = 6 ?*

BƯỚC 2
XÁC ĐỊNH SUBNET MASK

128	64	32	16	8	4	2	1	
1	0	0	0	0	0	0	0	= 128
1	1	0	0	0	0	0	0	= 192
1	1	1	0	0	0	0	0	= 224
1	1	1	1	0	0	0	0	= 240
1	1	1	1	1	0	0	0	= 248
1	1	1	1	1	1	0	0	= 252
1	1	1	1	1	1	1	0	= 254
1	1	1	1	1	1	1	1	= 255

The subnet mask: 255.255.240.0.

BƯỚC 3

- Tính ra các subnets, số lượng host cho mỗi subnet. Bao gồm:
  - Địa chị mạng
  - Dãy địa chỉ host
  - Địa chỉ broadcast

BƯỚC 3

- Quyết định mượn 4 bit, dãy địa chỉ mạng sẽ là: (chỉ khảo sát 2 bytes cuối)
- 1st subnet: .00000000.00000000
- 2nd subnet: .00010000.00000000
- 3rd subnet: .00100000.00000000
- …

15th subnet: .11110000.00000000

BƯỚC 3

No	Subnet Address	Host address range	Broadcast address	Use ?
0	172.16.0.0	172.16.0.1 – 172.16.15.254	172.16.15.255	N
1	172.16.16.0	172.16.16.1 – 172.16.31.254	172.16.31.255	Y
2	172.16.32.0	172.16.32.1 – 172.16.47.254	172.16.47.255	Y
..	..	..	..	..
..	..	..	..	..
13	172.16.208.0	172.16.208.1 – 172.16.223.254	172.16.223.255	Y
14	172.16.224.0	172.16.224.1 – 172.16.239.254	172.16.239.255	Y
15	172.16.240.0	172.16.240.1 – 172.16.255.254	172.16.255.255	N

TA CÓ THỂ MƯỢN TỐI ĐA BAO NHIÊU BIT?

- Số bit tối thiêu có thể mượn là:

2 bits.

- Số bit tối đa có thể mượn là:

A: 22 bits ~ 2^{22} - 2 = 4.194.302 subnets.

B: 14 bits ~ 2^{14} - 2 = 16.382 subnets.

C: 06 bits ~ 2^{06} - 2 = 62 subnets.

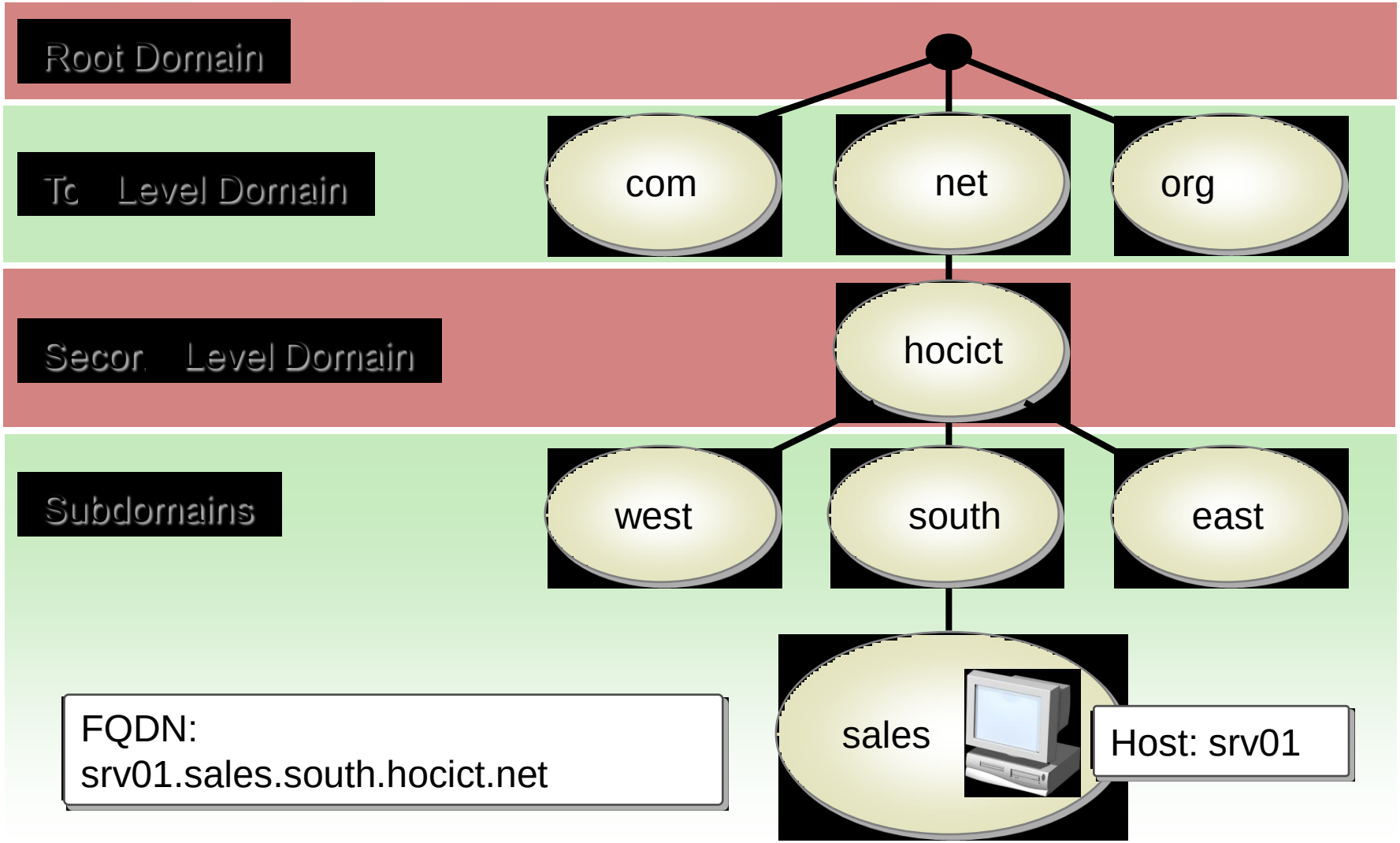
CLASS C

Number of Bits Borrowed	Number of Subnets Created	Number of Hosts Per Subnet	Total Number of Hosts	Percent Used
2	2	62	124	49%
3	6	30	180	71%
4	14	14	196	77%
5	30	6	180	71%
6	62	2	124	49%

DNS

- IP Address là số khó nhớ, nên hệ thống mẫu tự có tính gợi nhớ cao được tạo ra gọi là DNS.
- Cấu trúc DNS :
  - Gồm nhiều phần cách nhau dấu “.”
  - Có ít nhất 2 phần : Second-Level.top-level
  - DNS <=255 kí tự, mỗi phần <=63 kí tự
- Top-Level :
  - 3 kí tự : Com, Edu, Gov, Mil, Org, Net
  - 2 kí tự : Vn, Th, Tw, Sg, Jp, Fr, It, Uk, Ca..
- Ví dụ : Địa chỉ Host www.hocict.net

PHÂN CẤP DNS


Root Domain
Tc Level Domain
Secor Level Domain
Subdomains
com
net
org
hocict
west
south
east
sales
Host: srv01
FQDN:
srv01.sales.south.hocict.net

FULL QUALITY DOMAIN NAME

Examples:

FQDN

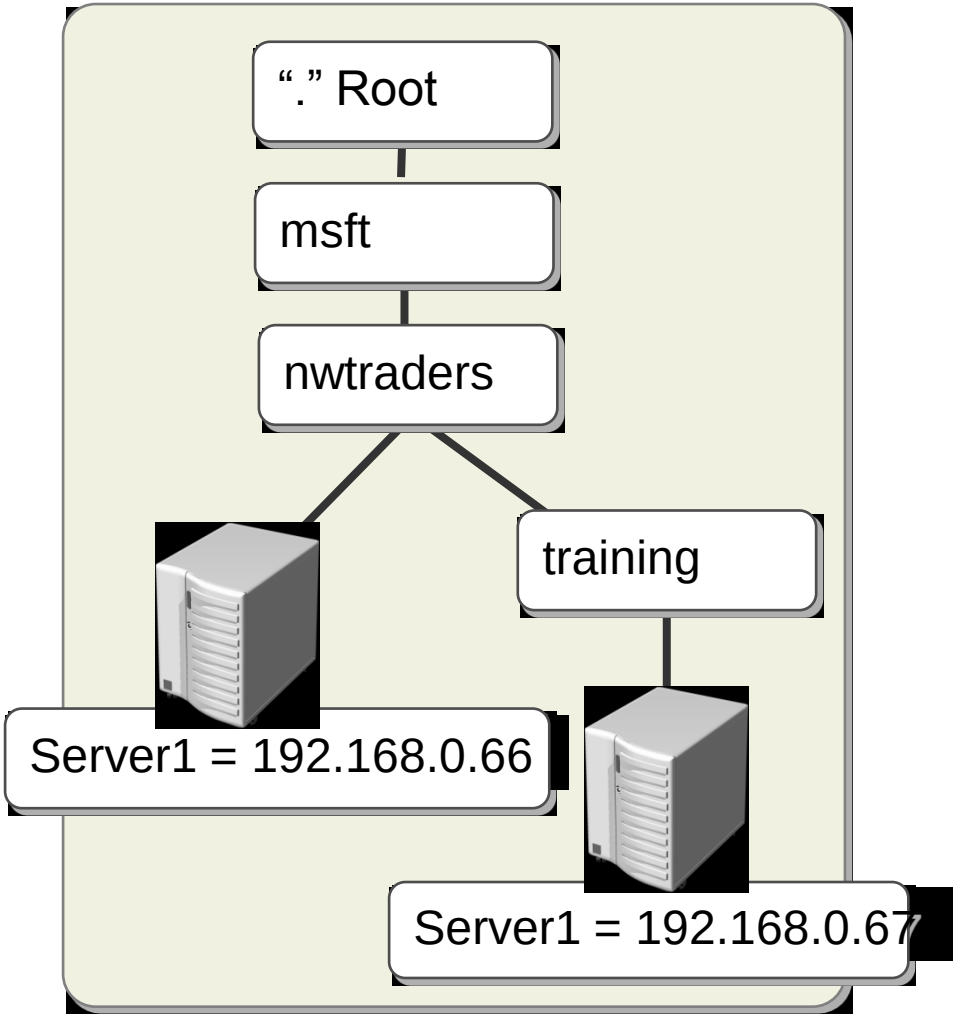
server1.nwtraders.msft.

Host Name DNS Suffix

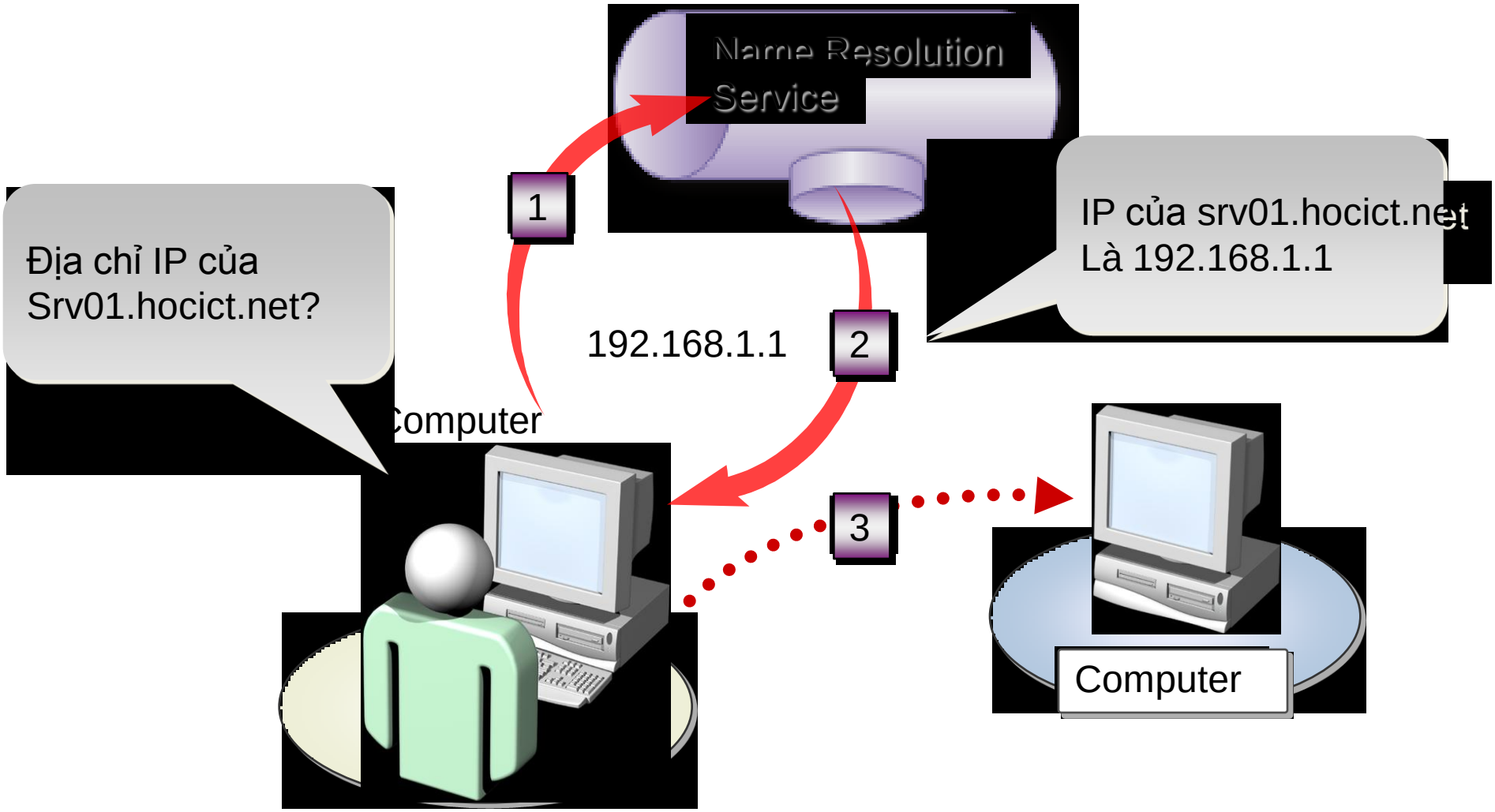
FQDN

server1.training.nwtraders.msft.

Host Name DNS Suffix

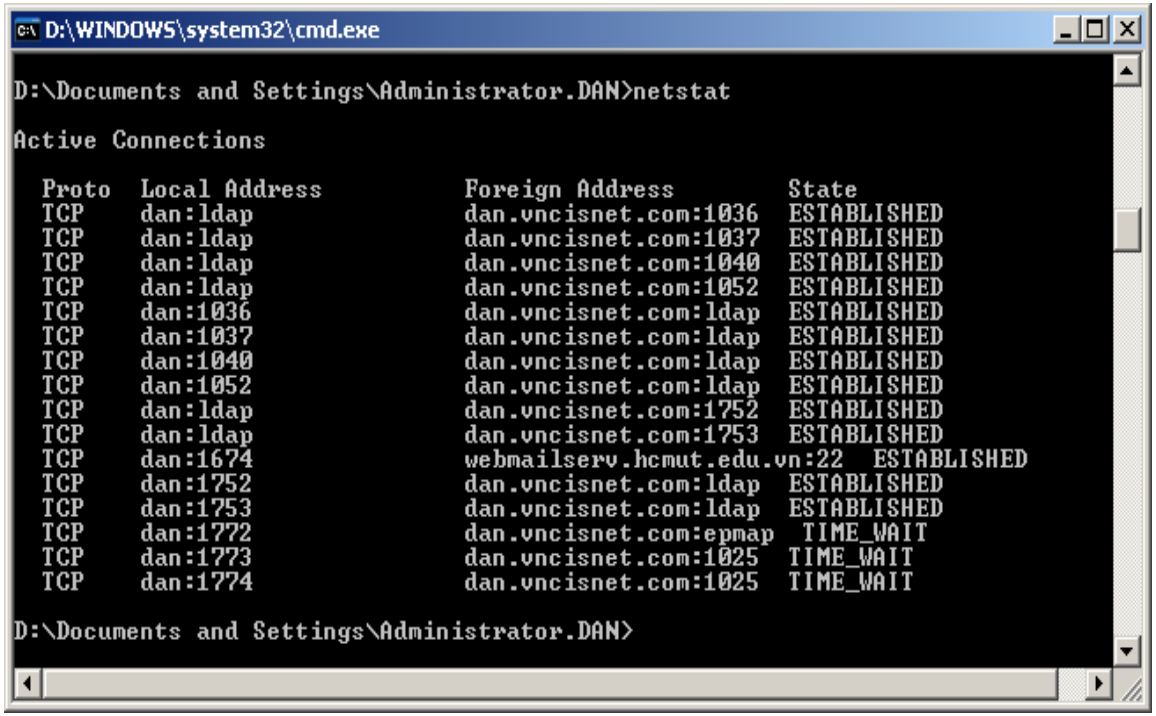


PHÂN GIẢI TÊN MIỀN SANG ĐỊA CHỈ IP



TIỆN ÍCH TCP/IP

- ARP
- netstat
- *nbtstat*
- FTP
- Ping
- Ipconfig
- Tracert
- Telnet
- Nslookup
- …



QUESTION & ANSWER

Tổng quan về giao thức TCP/IP

Mail:

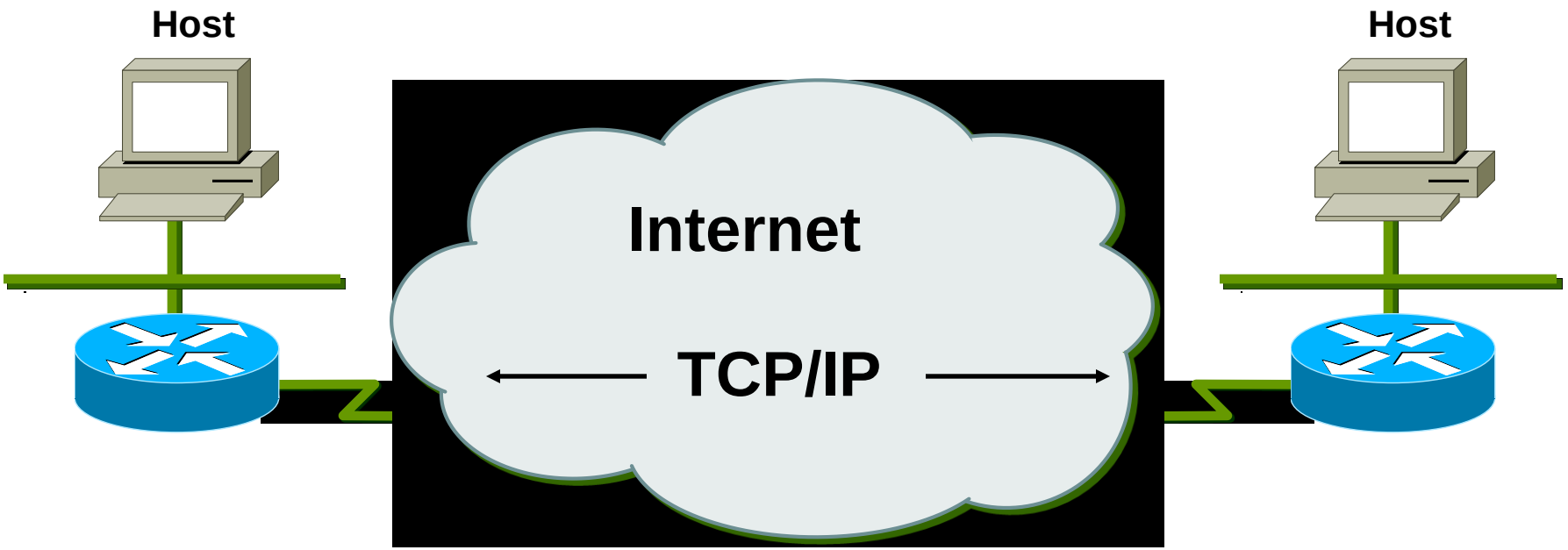
lephuc@ptithcm.edu.vn

http://is.ptithcm.edu.vn/~lephuc

Nội dung

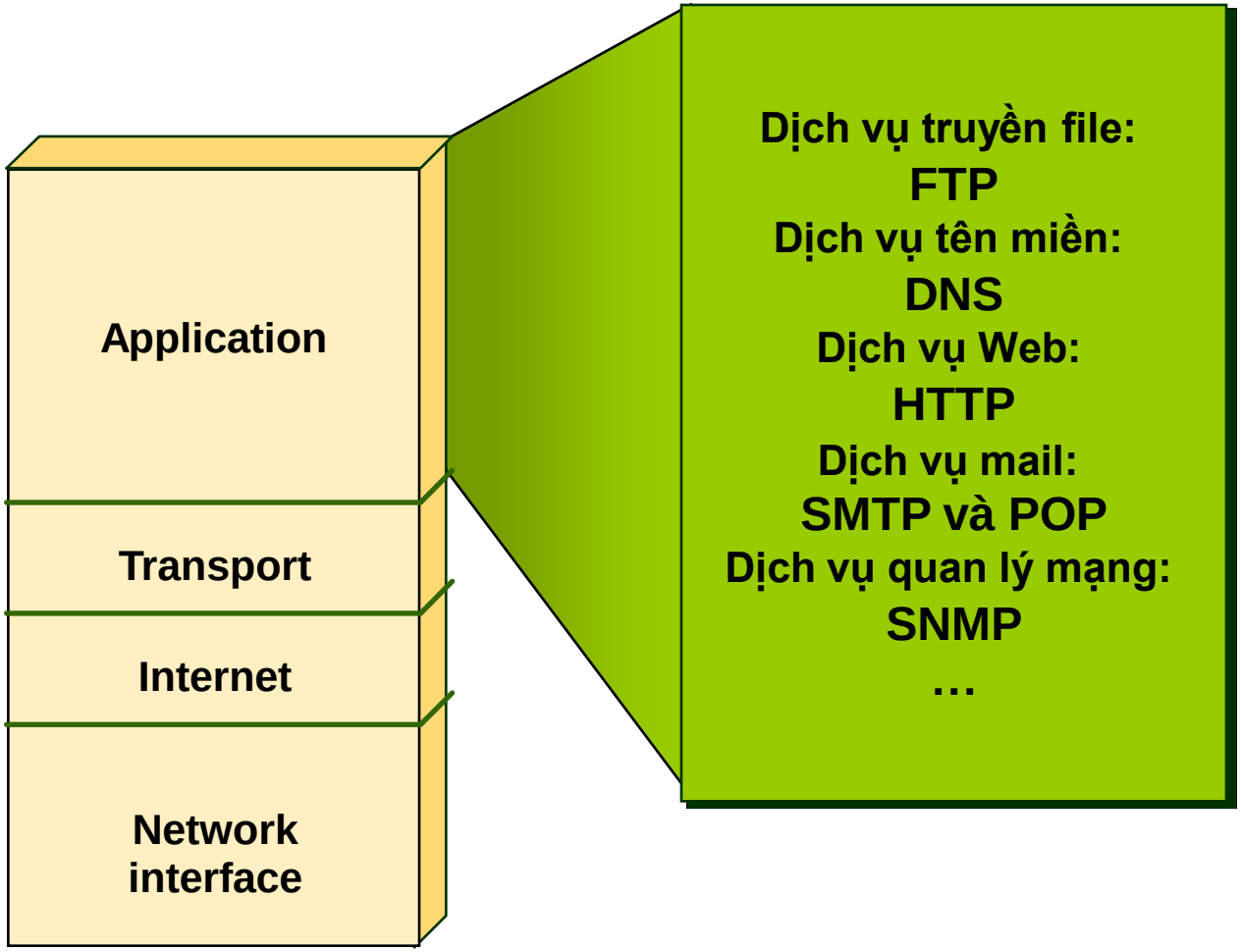
- Các thành phần của TCP/IP
- Giao thức IP
- Giao thức TCP
- Giao thức UDP
- Các giao thức phụ trợ (ARP, ICMP)
- Sơ đồ chuyển đổi trạng thái TCP

TCP/IP trên mạng Internet

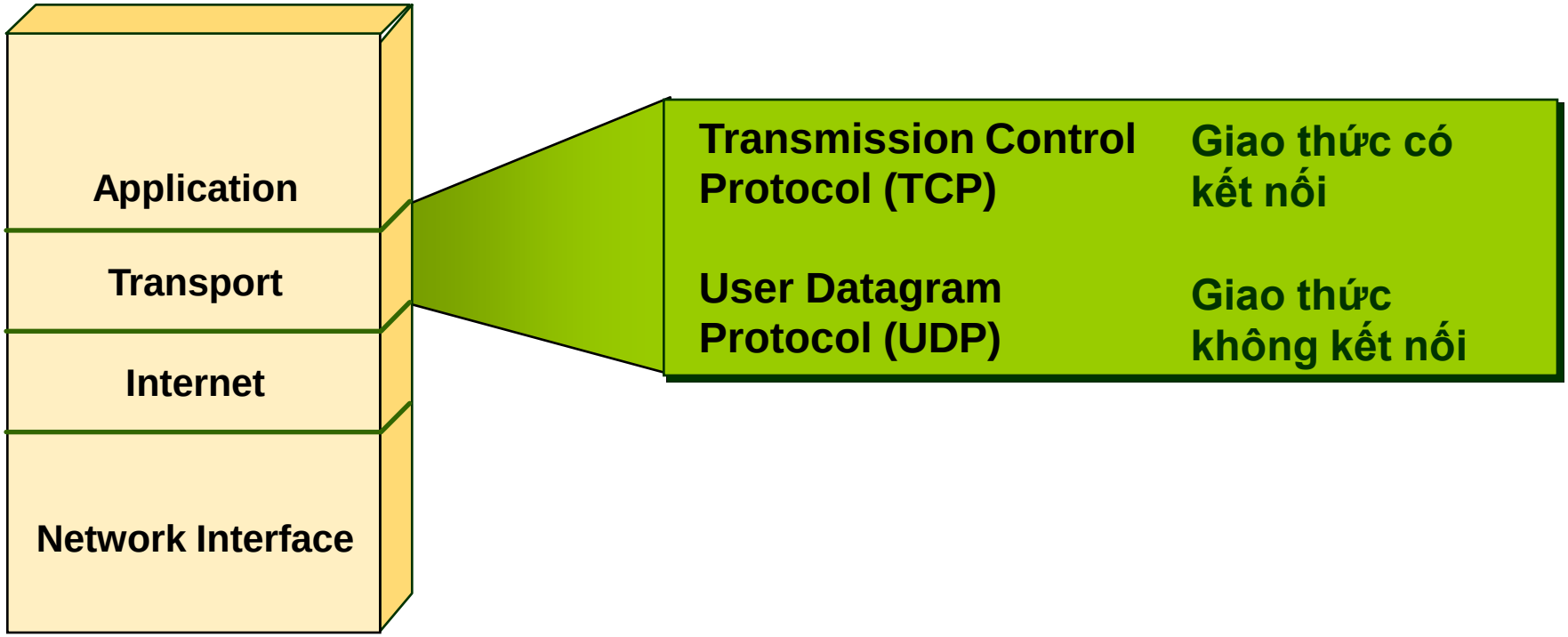


•Là giao thức truyền thống, bắt buộc đối với mạng Internet.

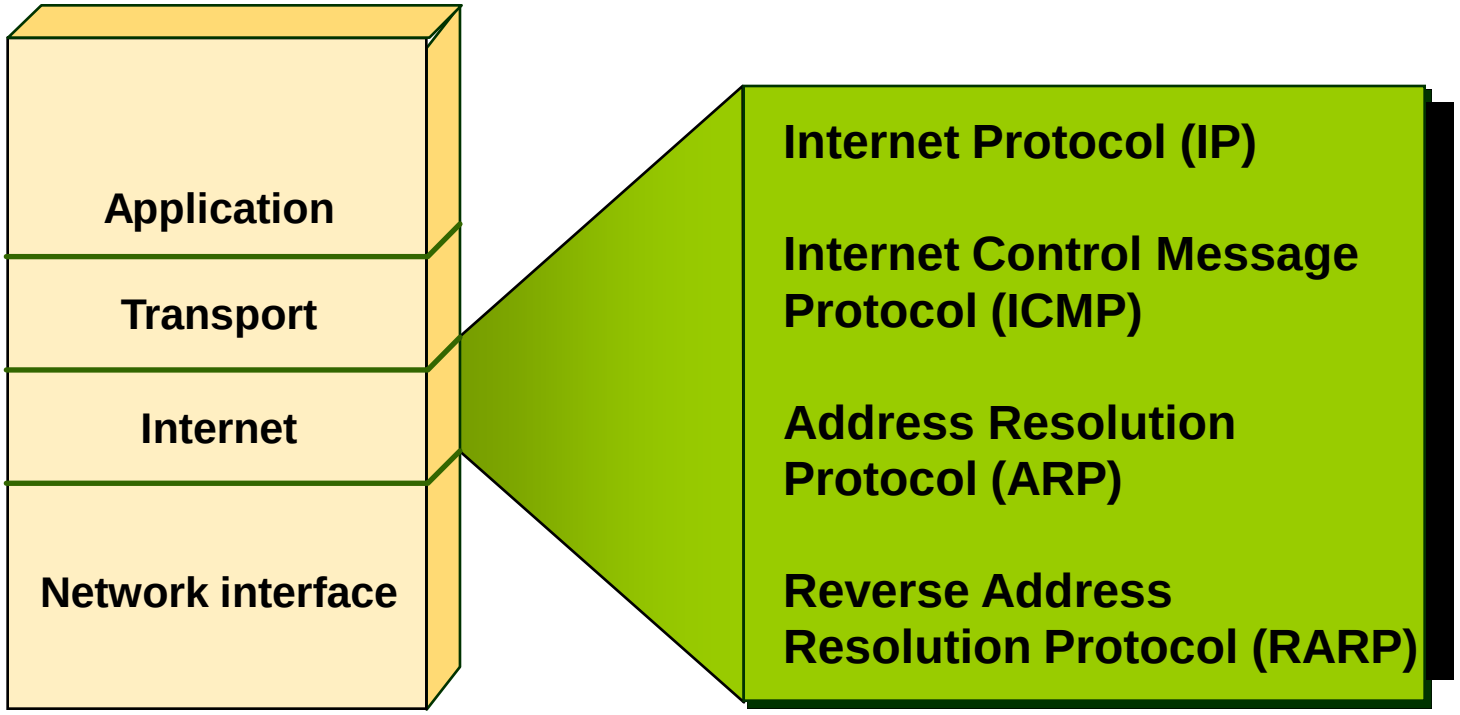
Lớp ứng dụng của TCP/IP



Lớp Vận chuyển trong TCP/IP



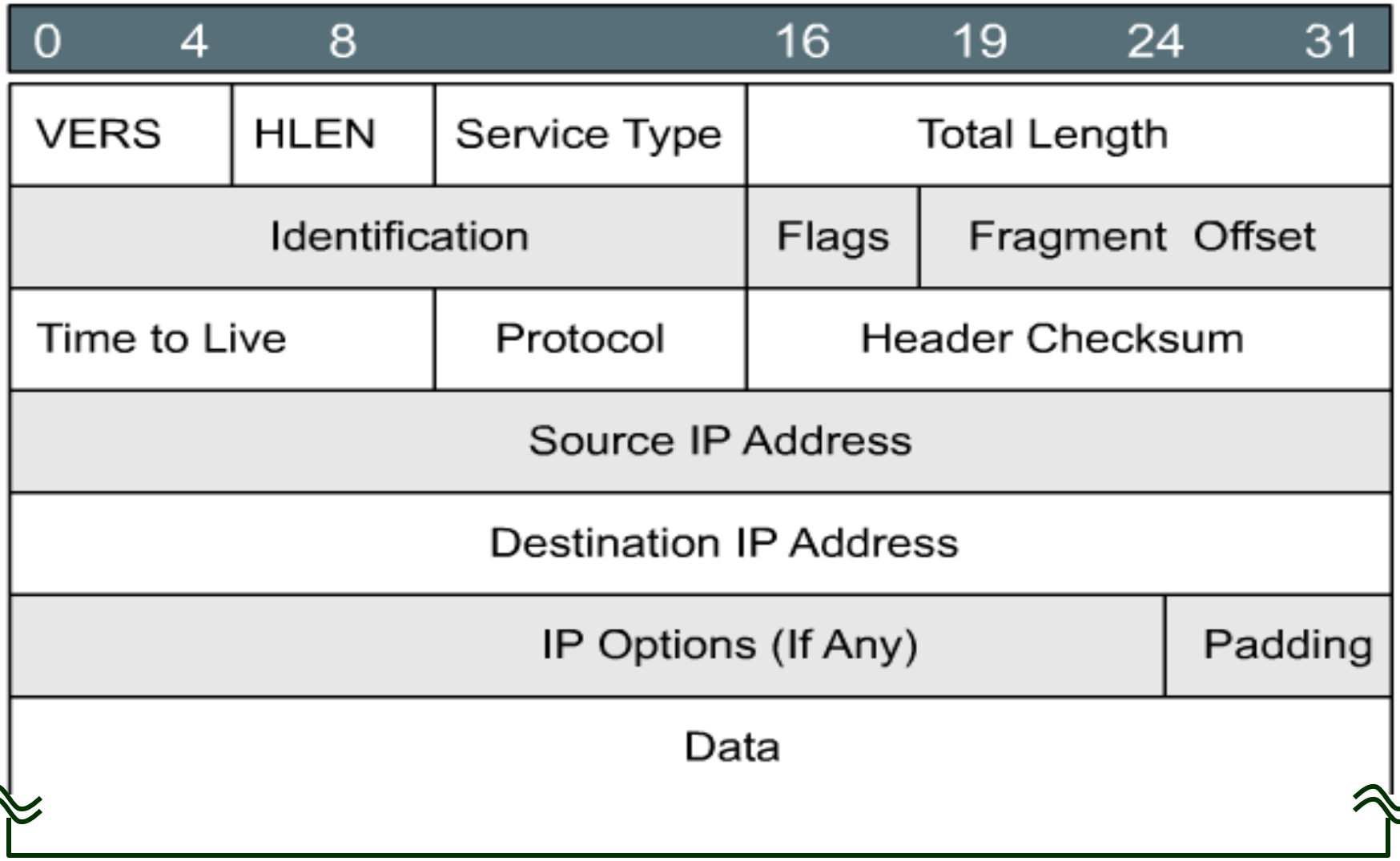
Lớp Liên kết mạng trong TCP/IP



Giao thức IP

- IP là giao thức không có kết nối (connectionless protocol), do đó không có thủ tục thiết lập và giải tỏa kết nối.
- IP không kiểm tra lỗi trên phần dữ liệu, do đó giao thức lớp trên (TCP) phải thực hiện chức năng này.
- Chức năng cơ bản của IP là tìm đường chuyển gói dữ liệu đến đúng nơi nhận

Cấu trúc gói dữ liệu IP



Giao thức ICMP

- Kiểm tra kết nối (echo request)
  - `Reply from …:` *Kết nối hoạt động tốt*
  - `Request timeout:` *Kết nối không tồn tại*
  - `Destination … unreachable:` *Định tuyến sai*
- Dò đường đi (Route tracing)
  - *Gói dữ liệu đi qua những router nào để đến đích?*

Giao thức ICMP

Kiểm tra kết nối

```
C:\Documents and Settings\IBM>ping www.google.com

Pinging www.l.google.com [72.14.235.147] with 32 bytes of data:

Reply from 72.14.235.147: bytes=32 time=91ms TTL=244
Reply from 72.14.235.147: bytes=32 time=88ms TTL=244
Reply from 72.14.235.147: bytes=32 time=94ms TTL=244
Reply from 72.14.235.147: bytes=32 time=84ms TTL=244

Ping statistics for 72.14.235.147:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 84ms, Maximum = 94ms, Average = 89ms
```

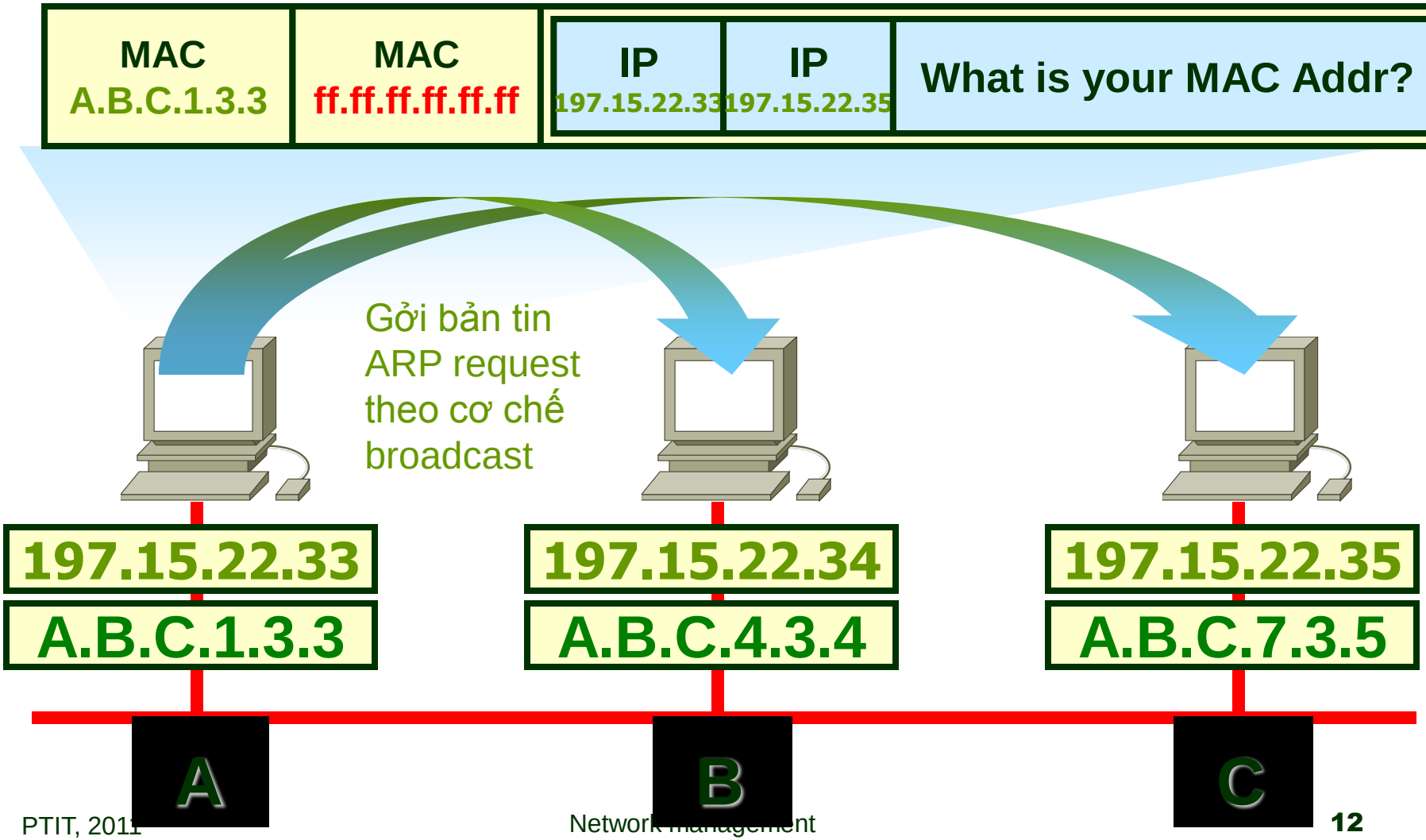
Dò đường đi

```
C:\Documents and Settings\IBM>tracert www.tuoitre.com.vn

Tracing route to www.tuoitre.com.vn [203.162.163.35]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  192.168.10.254
  2     3 ms     2 ms     2 ms  203.162.100.33
  3     8 ms    11 ms     8 ms  203.162.143.73
  4    19 ms    19 ms    19 ms  localhost [123.30.120.41]
  5     9 ms    12 ms    11 ms  localhost [123.30.120.22]
  6    19 ms    19 ms     9 ms  www.tuoitre.com.vn [203.162.163.35]

Trace complete.
```

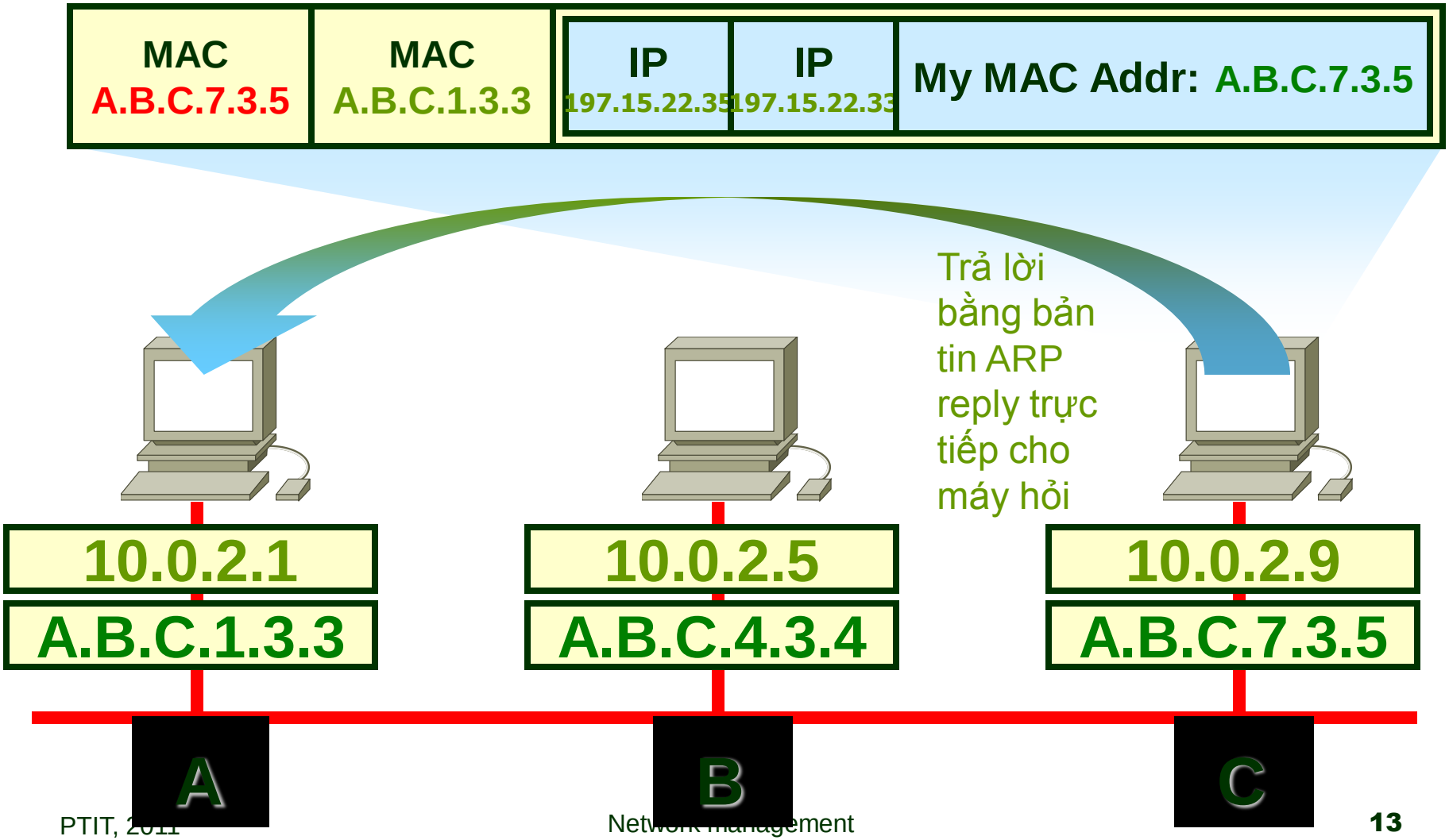
Giao thức ARP

- Tìm địa chỉ vật lý (MAC address) của máy đích trong mạng nội bộ (local network) khi biết địa chỉ IP.
- Hoạt động theo cơ chế broadcast.
- Lưu lại kết quả truy vấn cho các lần gởi kế tiếp.

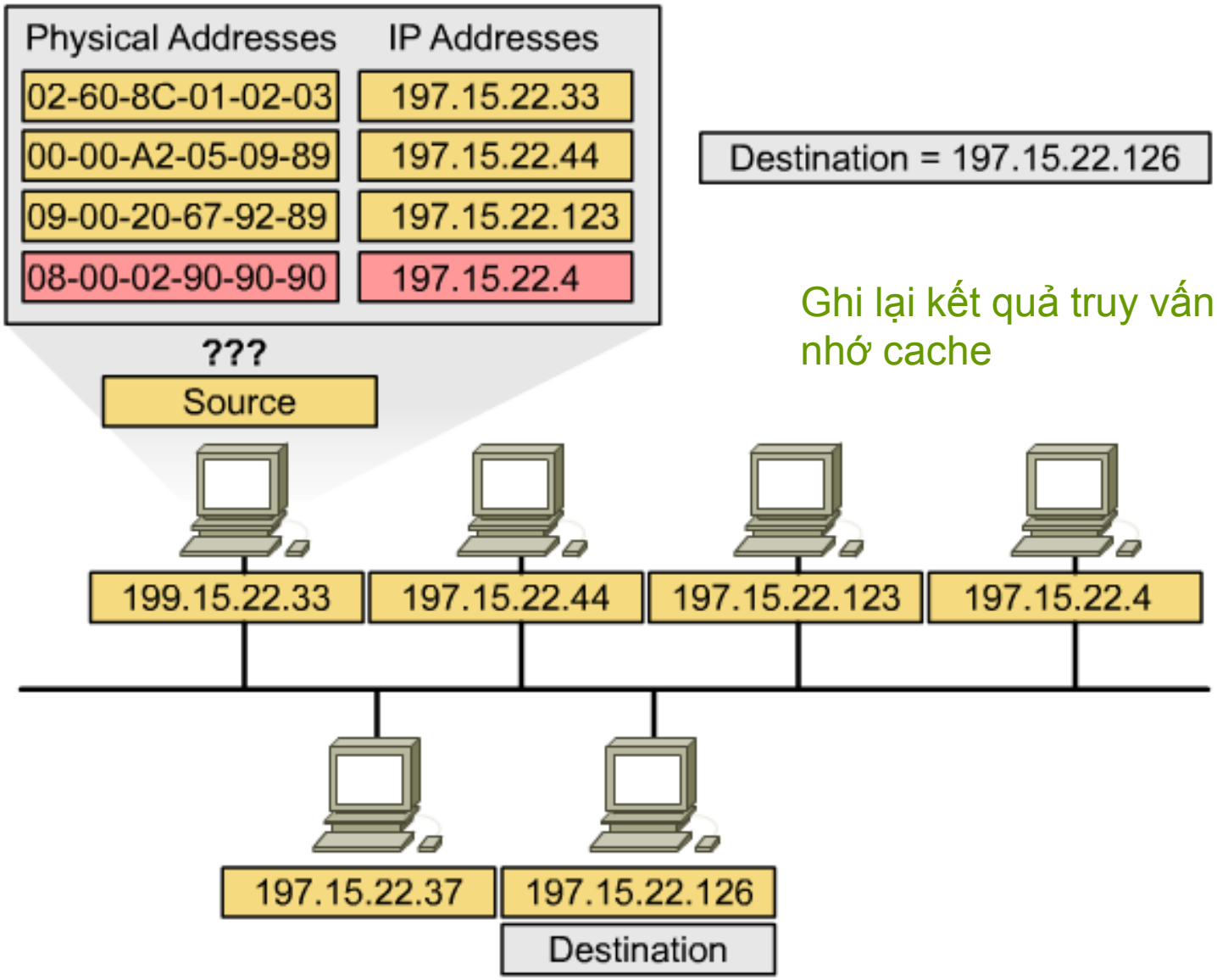
Giao thức ARP: ARP request



Giao thức ARP: ARP reply

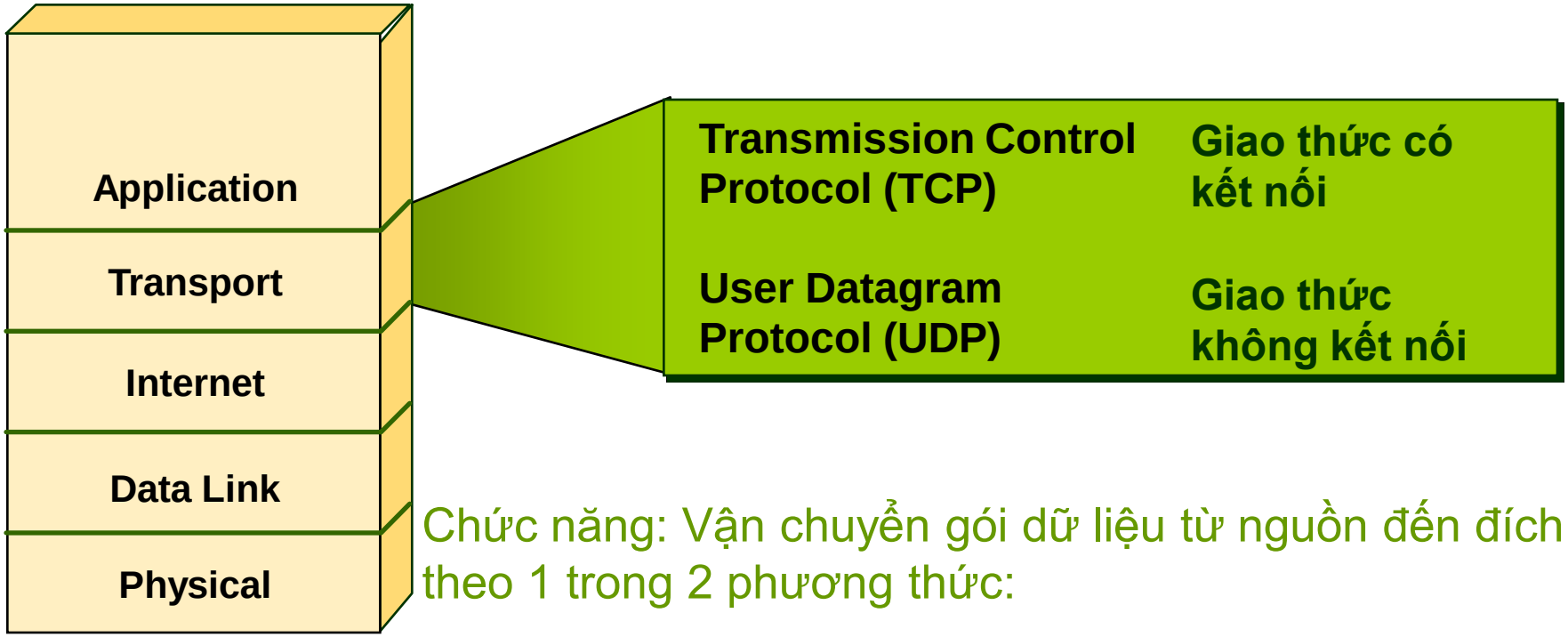


Giao thức ARP: Caching



Ghi lại kết quả truy vấn trong bộ nhớ cache

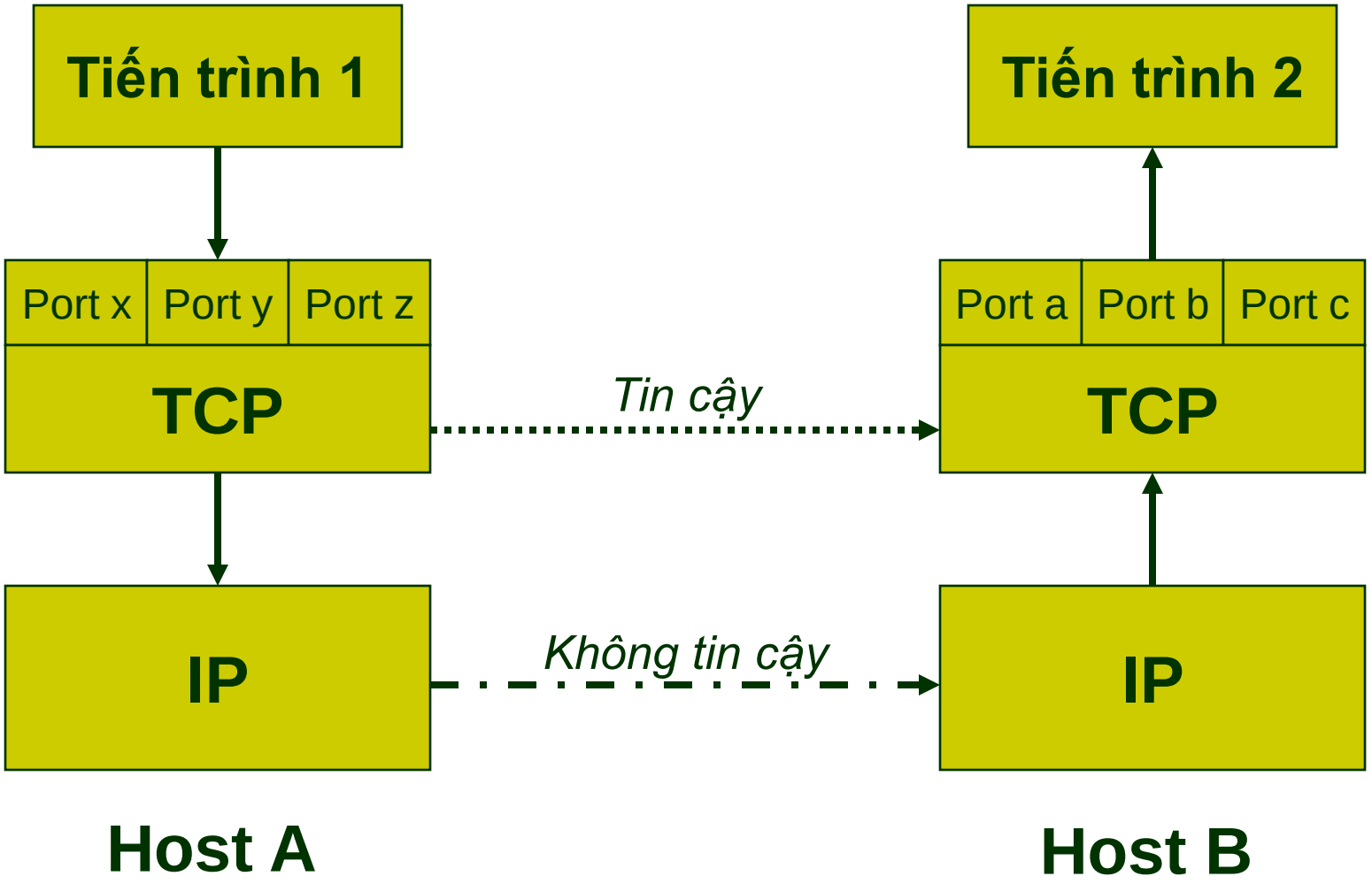
Lớp Vận chuyển trong TCP/IP



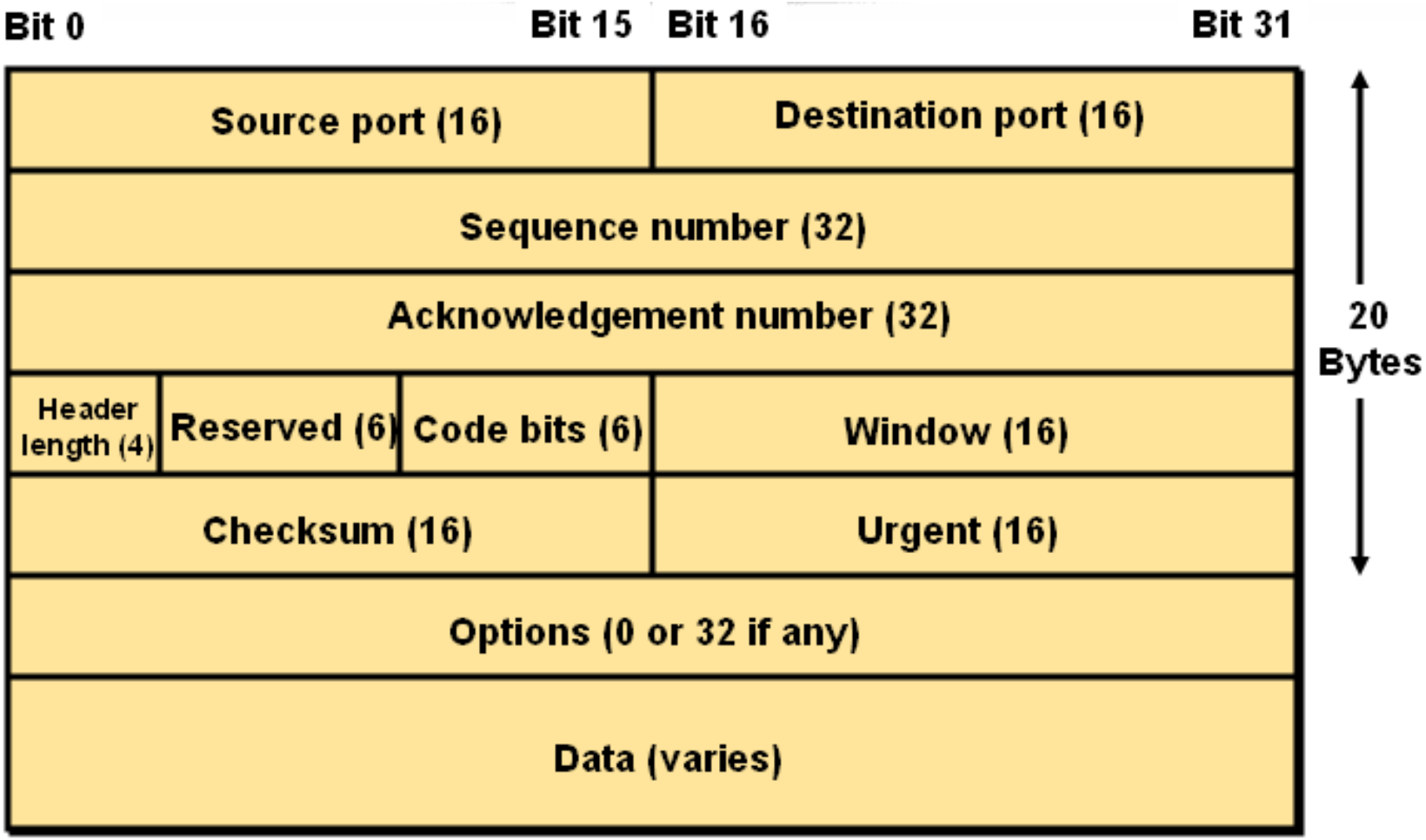
Chức năng: Vận chuyển gói dữ liệu từ nguồn đến đích theo 1 trong 2 phương thức:

- ***Tin cậy (TCP):*** Có sửa lỗi và điều khiển
- ***Không tin cậy (UDP):*** Không sửa lỗi, không điều khiển

Giao thức TCP

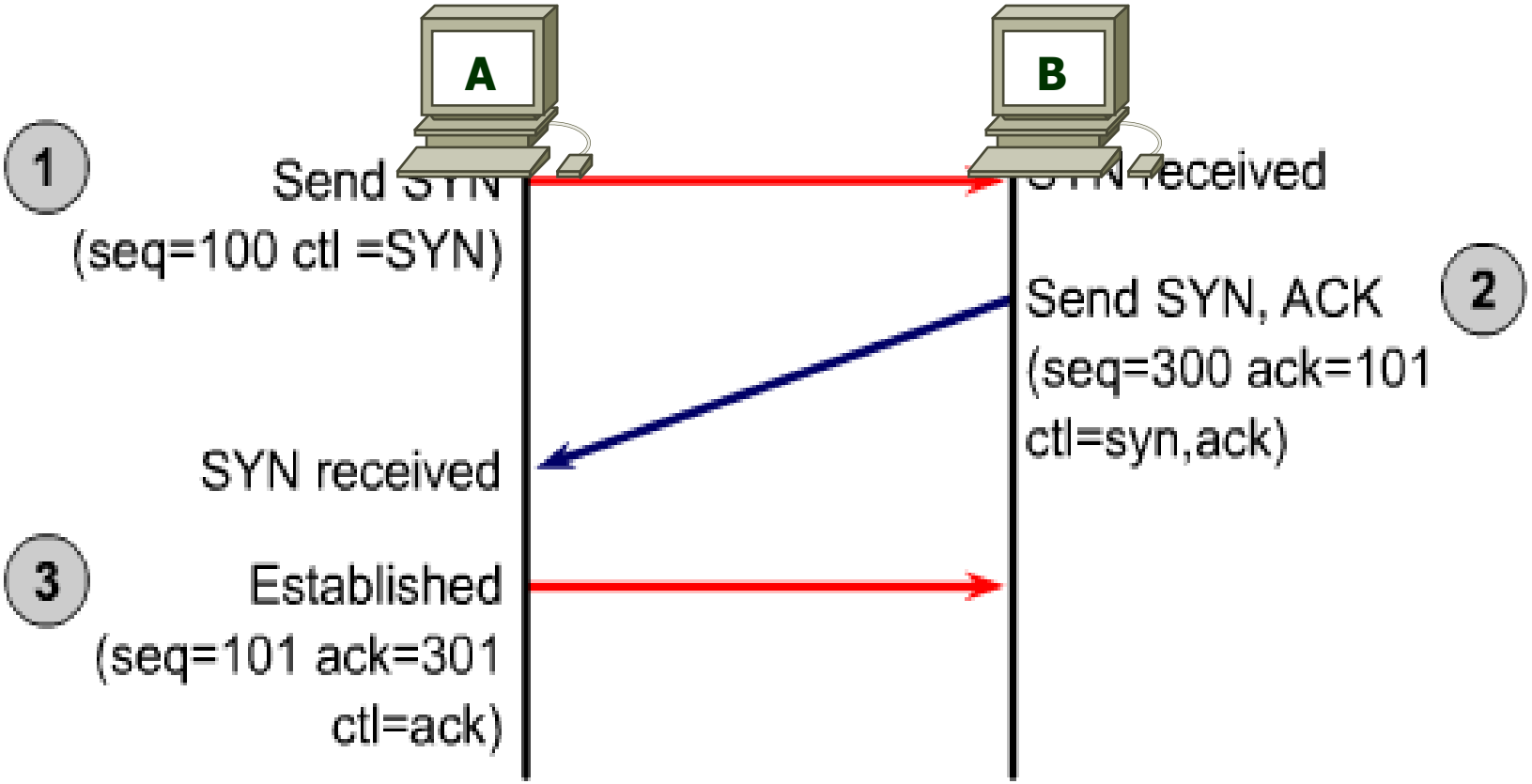


Cấu trúc gói dữ liệu TCP

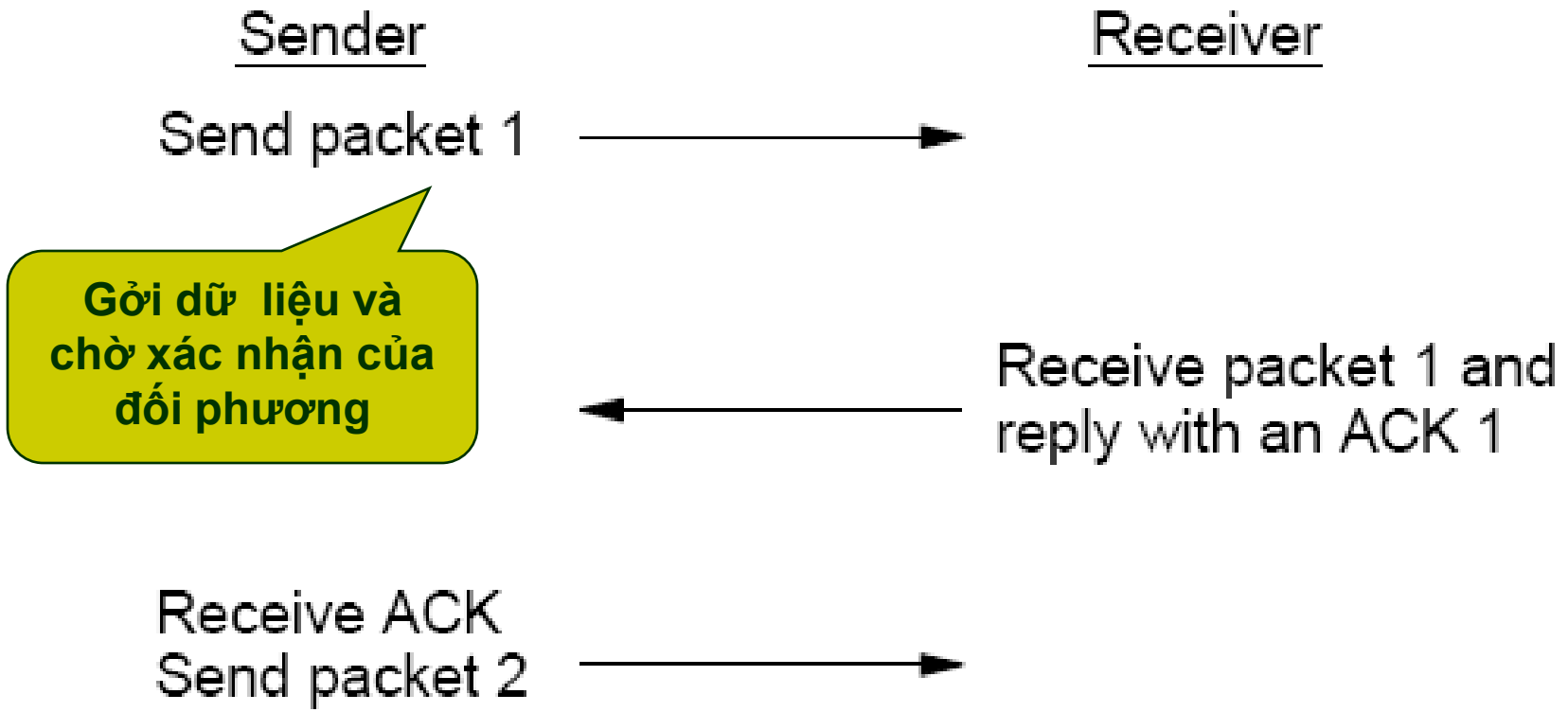


Thủ tục thiết lập kết nối TCP

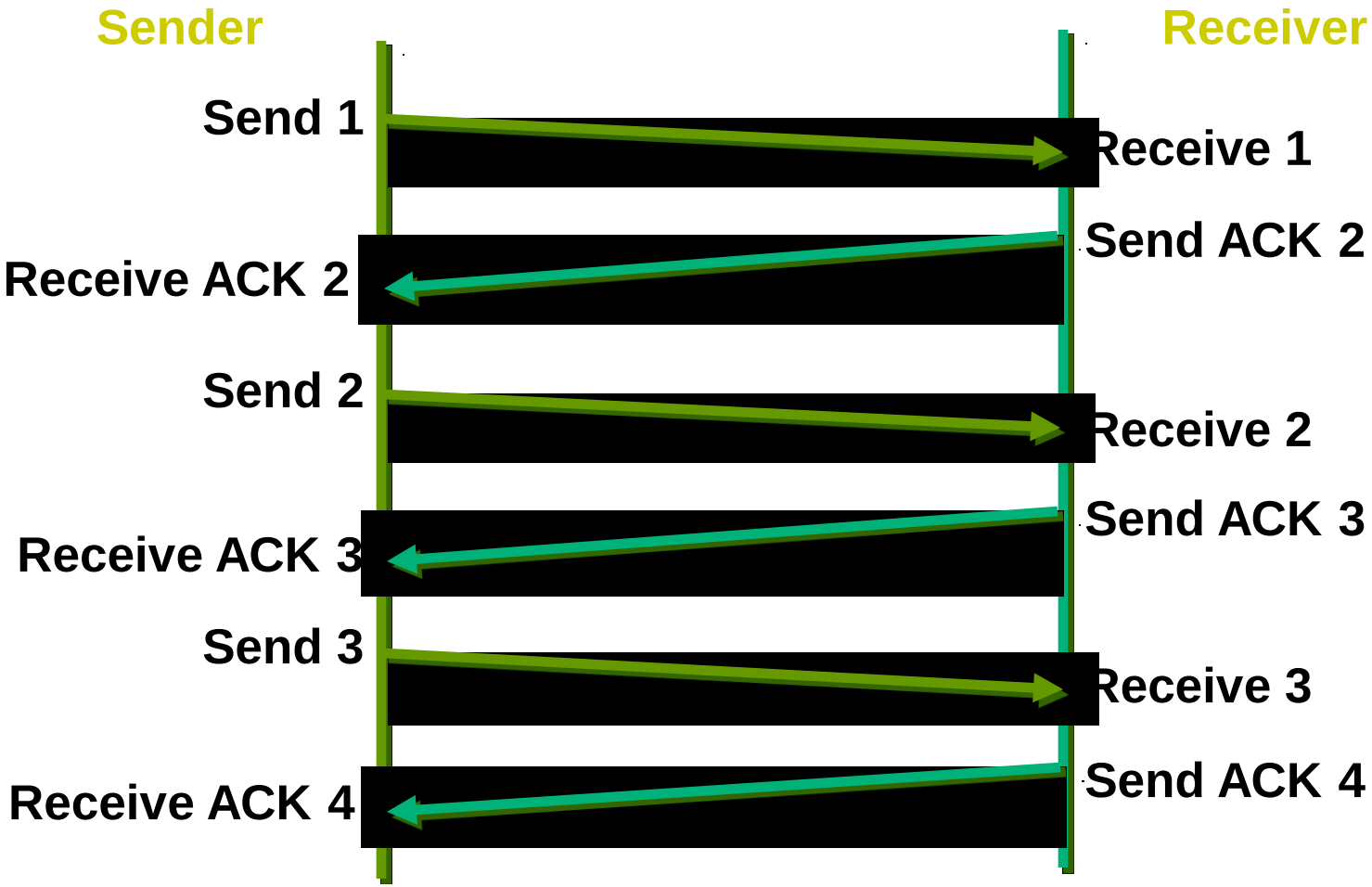
Thủ tục bắt tay 3 chiều (three way handshake)



Cơ chế truyền dữ liệu trong TCP

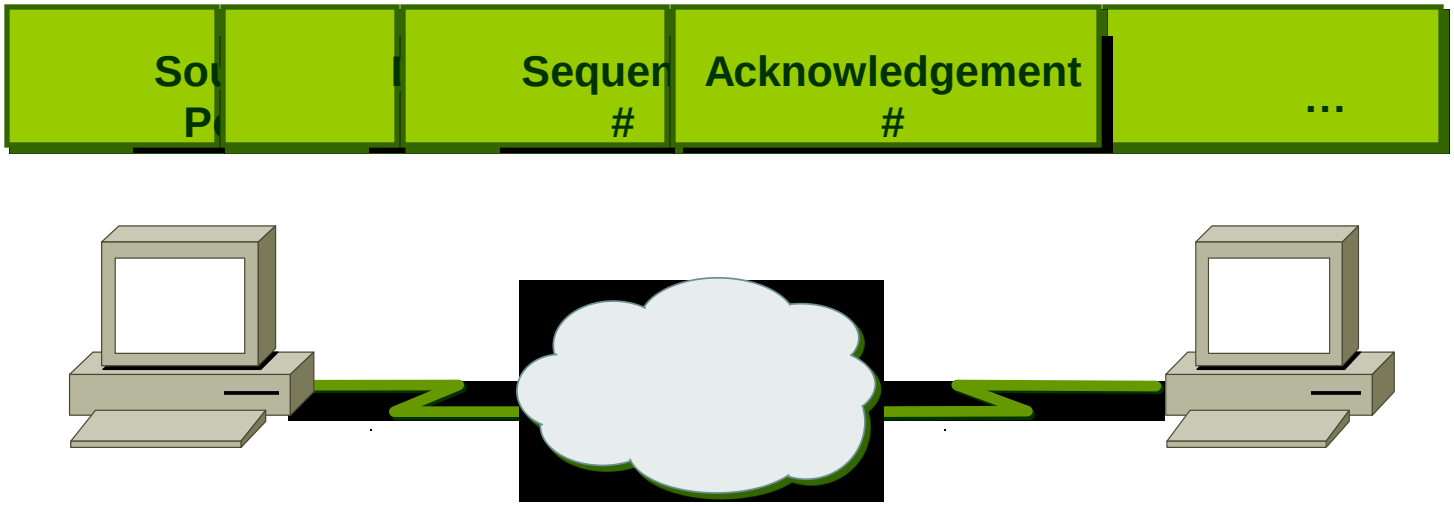


TCP: cơ chế truyền đơn giản



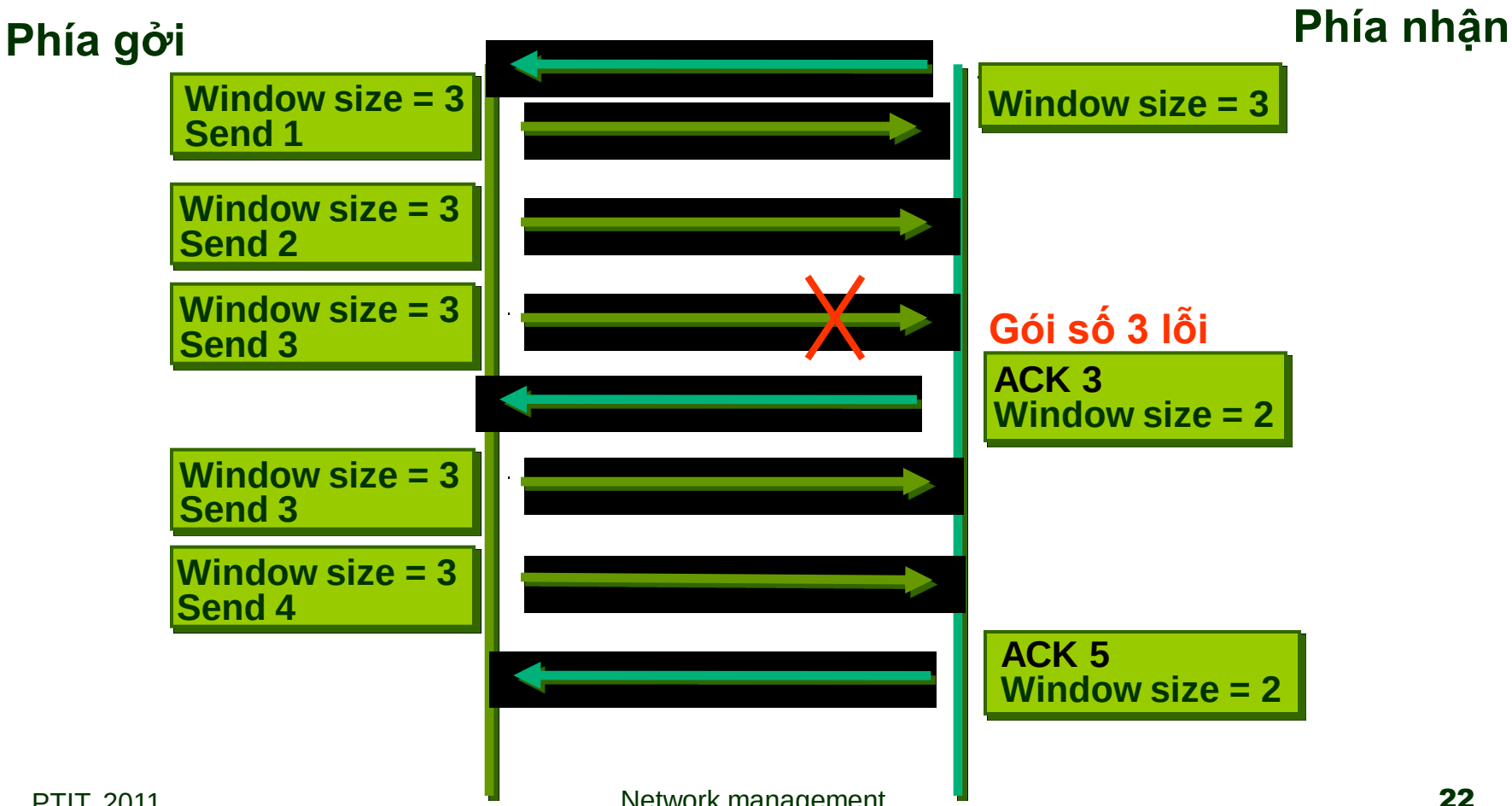
- Window size = 1

TCP: cơ chế truyền đơn giản



TCP: Cơ chế dịch cửa sổ

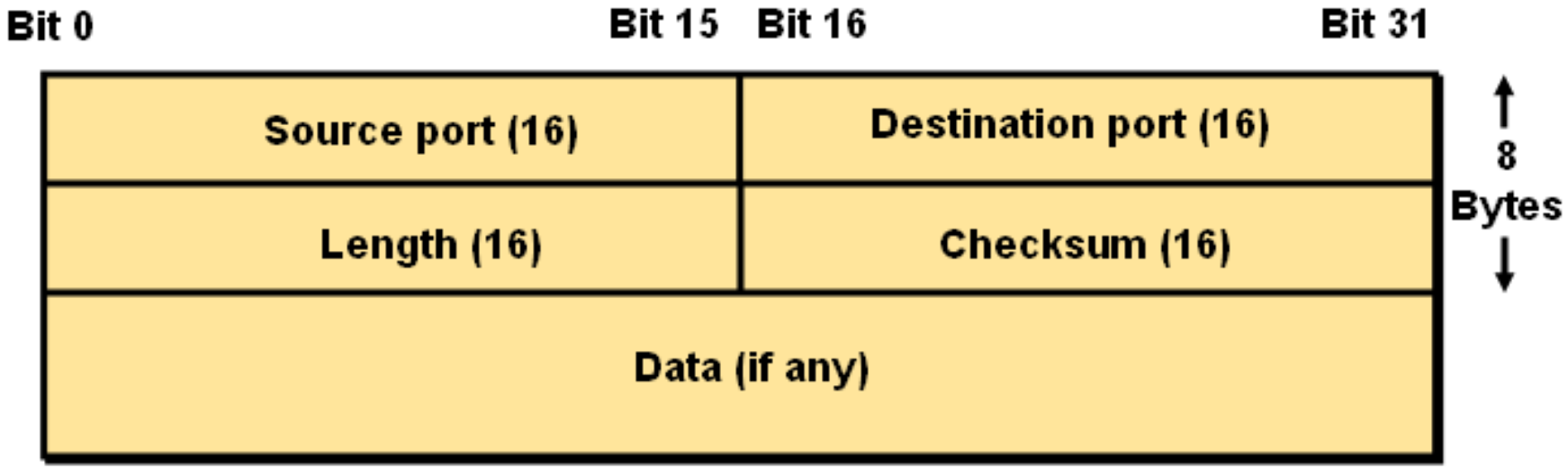
Giảm thời gian chờ, tăng hiệu suất truyền



Đặc điểm giao thức TCP

- Là giao thức có kết nối (connection-oriented), sử dụng thủ tục bắt tay 3 chiều để thiết lập kết nối.
- Truyền dữ liệu tin cậy (có sửa sai, sắp xếp gói theo thứ tự).
- Dùng port để nhận dạng dữ liệu của từng dịch vụ

Giao thức UDP



Cấu trúc gói dữ liệu rất đơn giản, không có các trường điều khiển như TCP

Đặc điểm của giao thức UDP

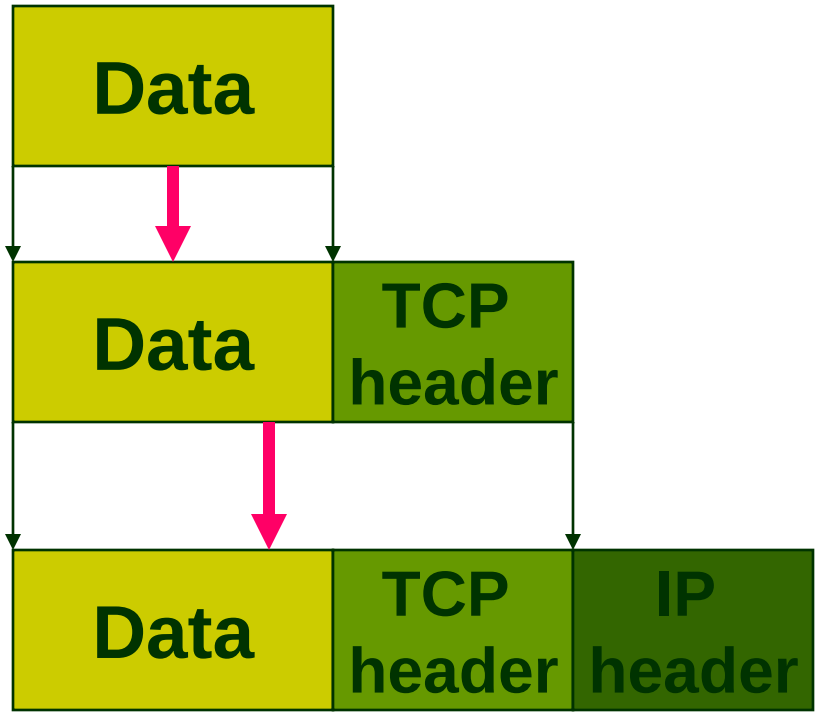
- Là giao thức không có kết nối
- Hoạt động đơn giản, không có các chức năng điều khiển.
- Truyền dữ liệu không tin cậy.
- Thích hợp với các dịch vụ có lượng dữ liệu nhỏ, tính đáp ứng nhanh.

Đóng gói dữ liệu trong TCP/IP

Lớp ứng dụng

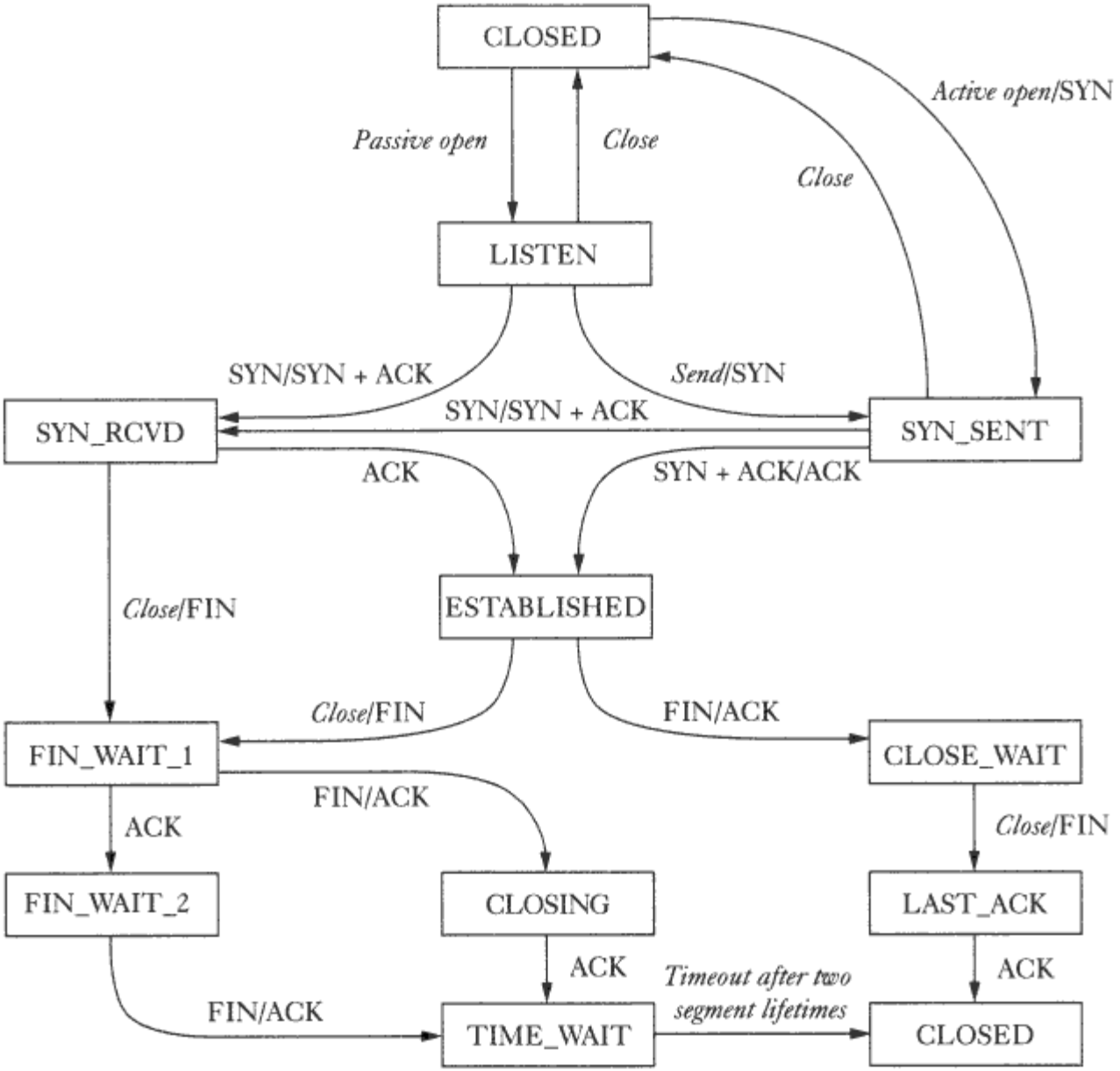
Giao thức TCP

Giao thức IP

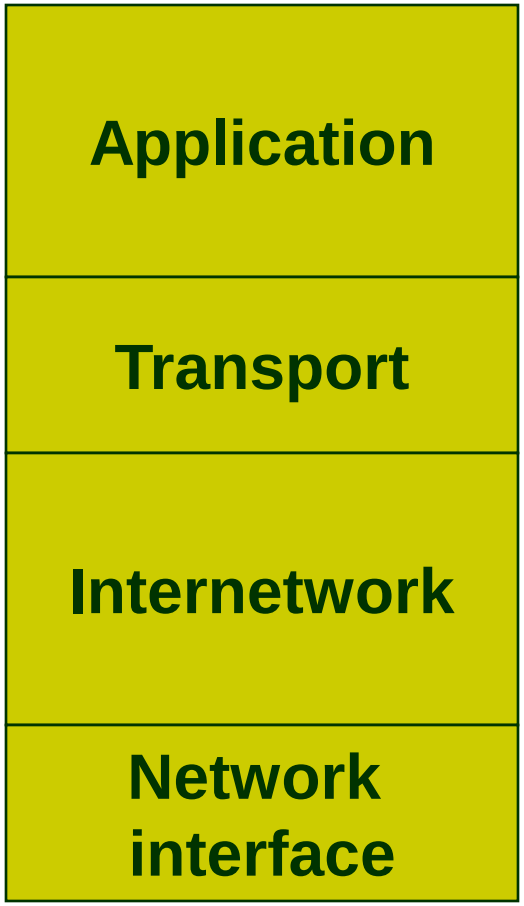


Sơ đồ trạng thái TCP

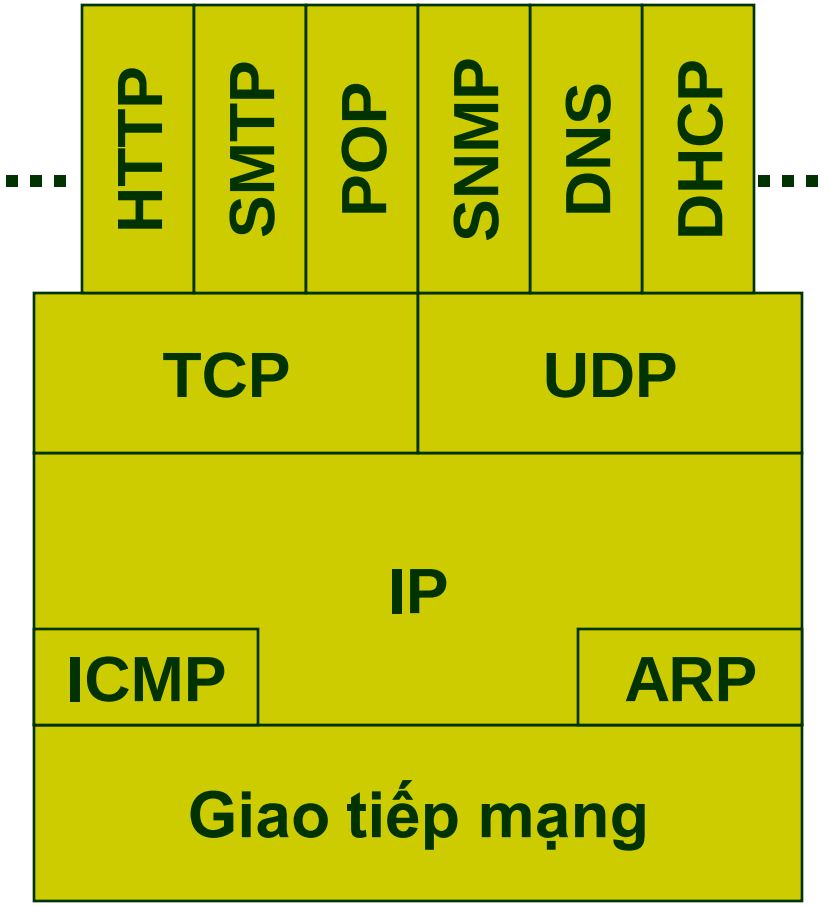
- Liệt kê các trạng thái (state) của giao thức TCP
- Các cơ chế chuyển đổi trạng thái của giao thức TCP.


CLOSED
Active open/SYN
Passive open
Close
Close
LISTEN
SYN/SYN + ACK
Send/SYN
SYN/SYN + ACK
SYN_RCVD
SYN_SENT
ACK
SYN + ACK/ACK
Close/FIN
ESTABLISHED
Close/FIN
FIN/ACK
FIN_WAIT_1
CLOSE_WAIT
FIN/ACK
ACK
Close/FIN
FIN_WAIT_2
CLOSING
LAST_ACK
ACK
Timeout after two segment lifetimes
ACK
FIN/ACK
TIME_WAIT
CLOSED

Tóm tắt bộ giao thức TCP/IP



Mô hình TCP/IP



Giao thức TCP/IP

Giao thức TCP/IP và Mạng Internet

Giảng viên hướng dẫn : Th.s Đỗ Quang Trung
Sinh viên thực hiện : LÊ THỊ THANH HIỀN
Lớp : S0809G
MSSV : HDSM 252601

Lời mở đầu

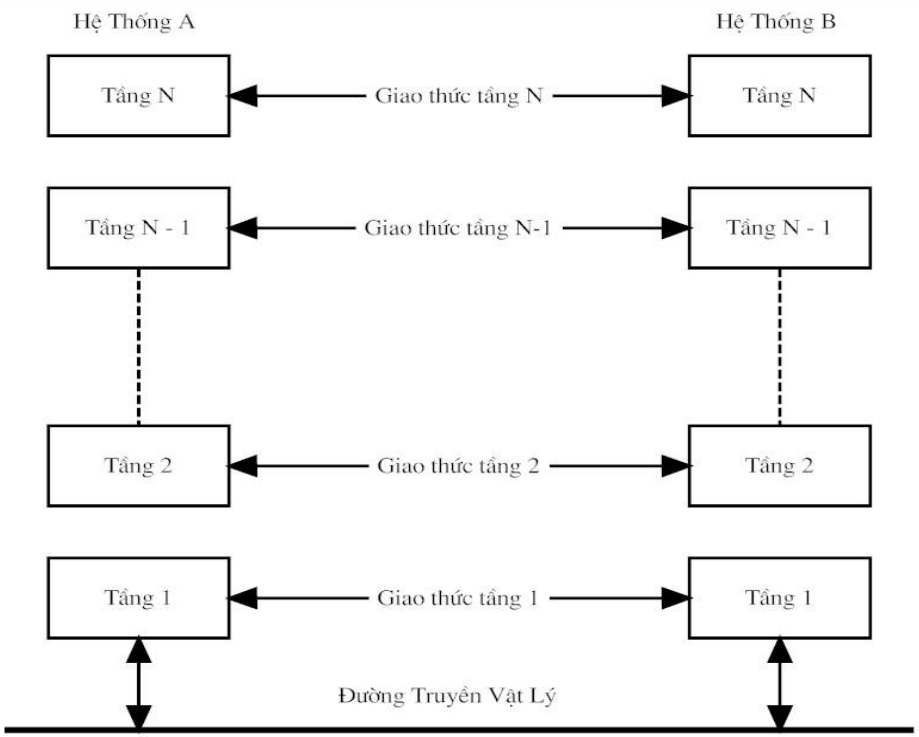
- Để hoàn thành đề án “Giao thức TCP/IP và Mạng Internet" tôi xin gửi lời cảm ơn trân thành tới Thầy giáo Đỗ Quang Trung đã tận tình hướng dẫn cũng như tạo điều kiện cho tôi hoàn thành tốt đề án này, tôi xin cảm ơn đến các bạn đã cùng thảo luận và giúp đỡ tôi trong suốt thời gian làm đồ án.
- Mục tiêu của đề án
  Tìm hiểu và thu thập kiến thức về” Giao thức TCP/IP và mạng Internet”. Đồ án góp phần cho người đọc có được kiến thức tổng quát và đầy đủ về TCP/IP.

Giao thức TCP/IP và Mạng Internet

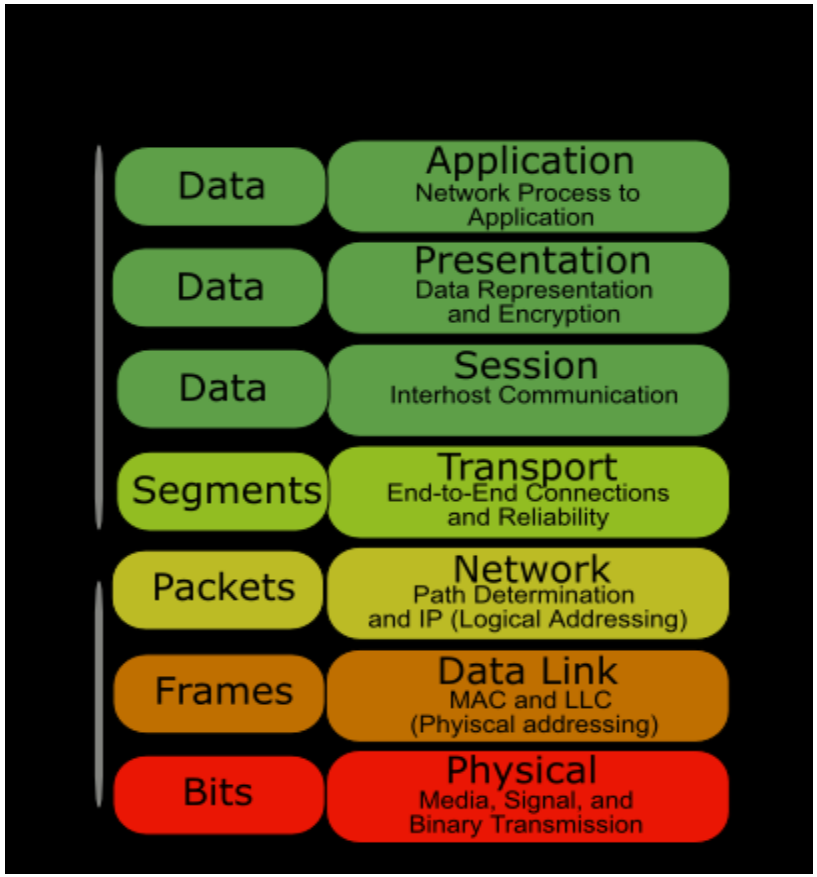
- Tổng quan hệ thống mạng TCP/IP.
- Bộ giao thức TCP/IP.
- Định tuyến.
- Mạng Internet.

Tổng quan hệ thống mạng TCP/IP.

- Kiến trúc phân tầng của mạng.

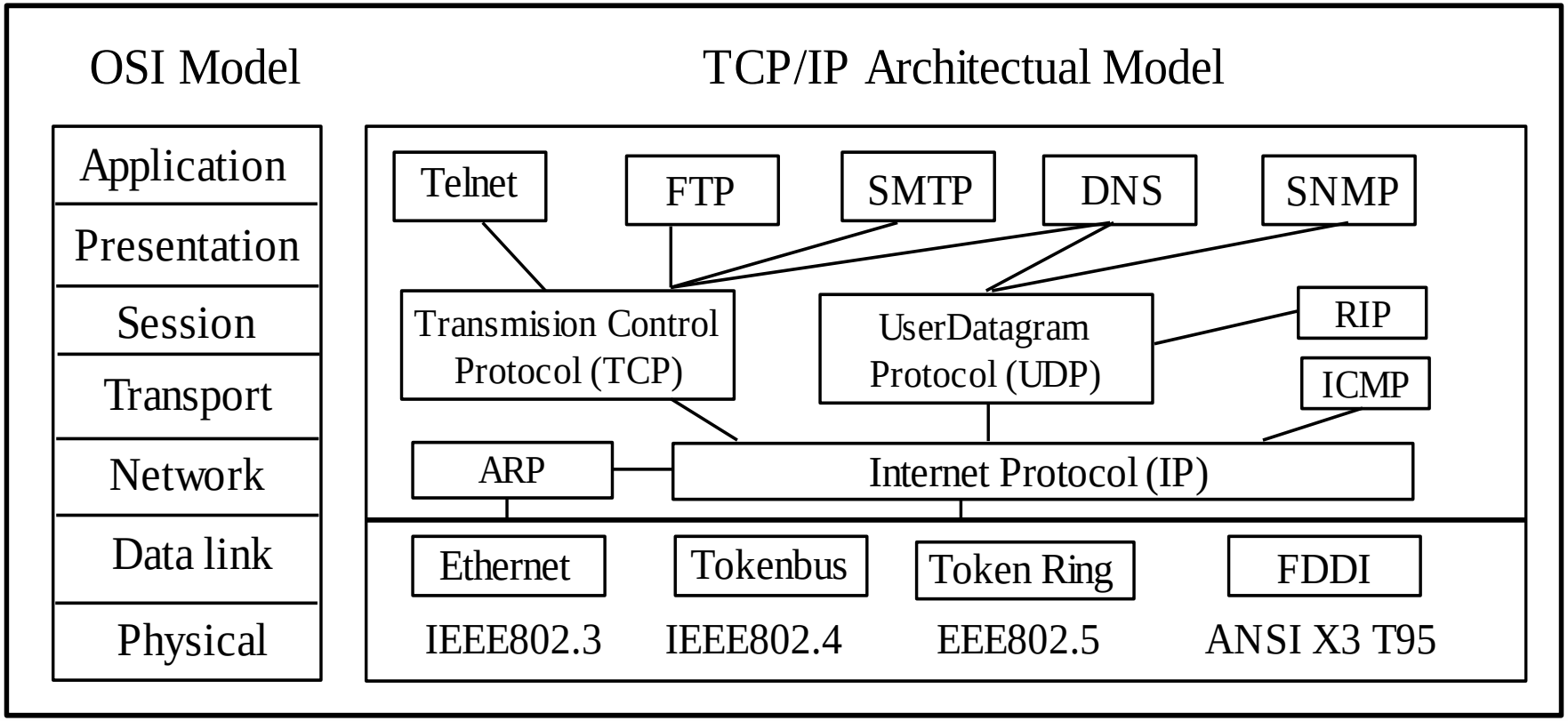


- Mô hình OSI.



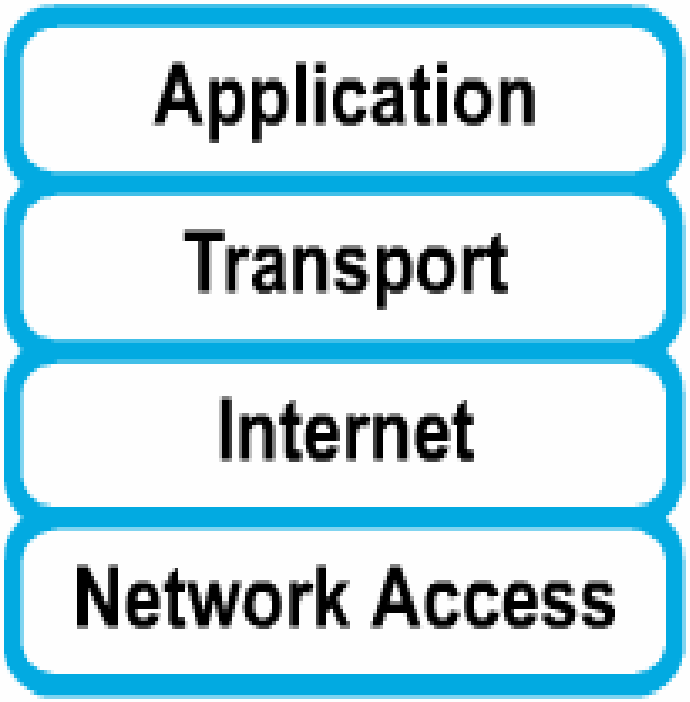
Tổng quan hệ thống mạng TCP/IP

Giao thức TCP/IP và Mô hình OSI



Bộ giao thức TCP/IP.

TCP/IP là mô hình áp dụng cho mạng Internet



- **The Application Layer (Lớp ứng dụng):** quản lý các giao thức, như hỗ trợ việc trình bày, mã hóa, và quản lý cuộc gọi. Lớp này cũng hỗ trợ nhiều ứng dụng .

- **The Network Access Layer (Link)**: xác định việc truy xuất đến các thiết bị, chuyển đổi dữ liệu thành các bit rồi truyền.

TCP/IP Layers

Transport Layer

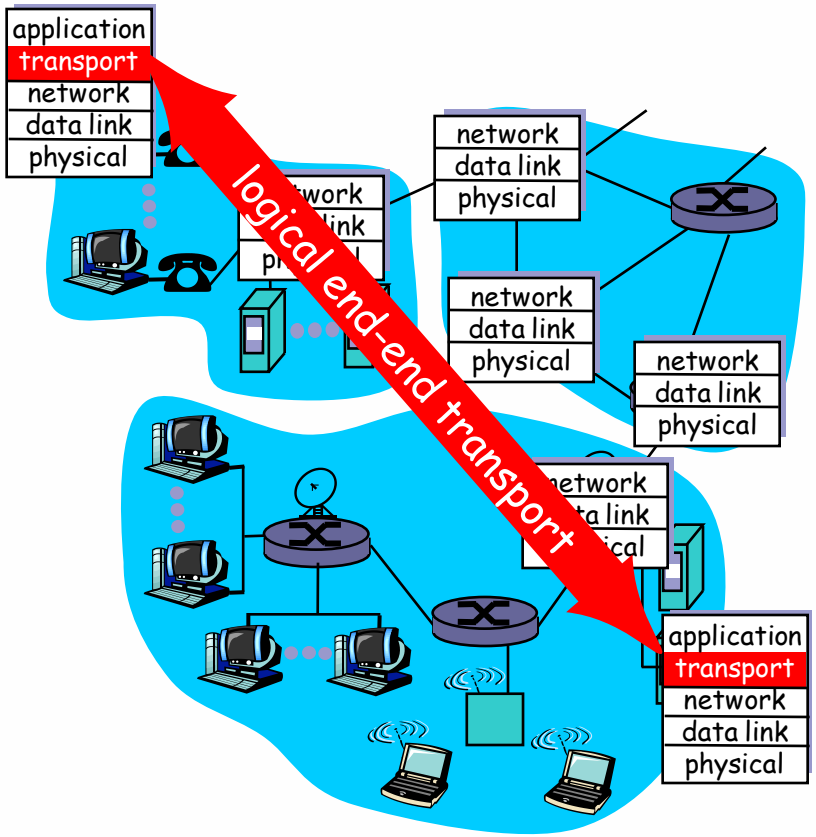
- Đảm nhiệm việc vận chuyển dữ liệu từ nguồn tới đích thông qua 2 giao thức:

- TCP (Transmission Control Protocol)
  - có liên kết.
  - điều khiển luồng.
  - điều khiển chống nghẽn mạng.

  → Giao thức tin cậy.

- UDP (User Datagram Protocol)
  - Không kết nối.
  - Không có kiểm soát luồng và kiểm soát nghẽn mạng.

  → Giao thức không tin cậy.



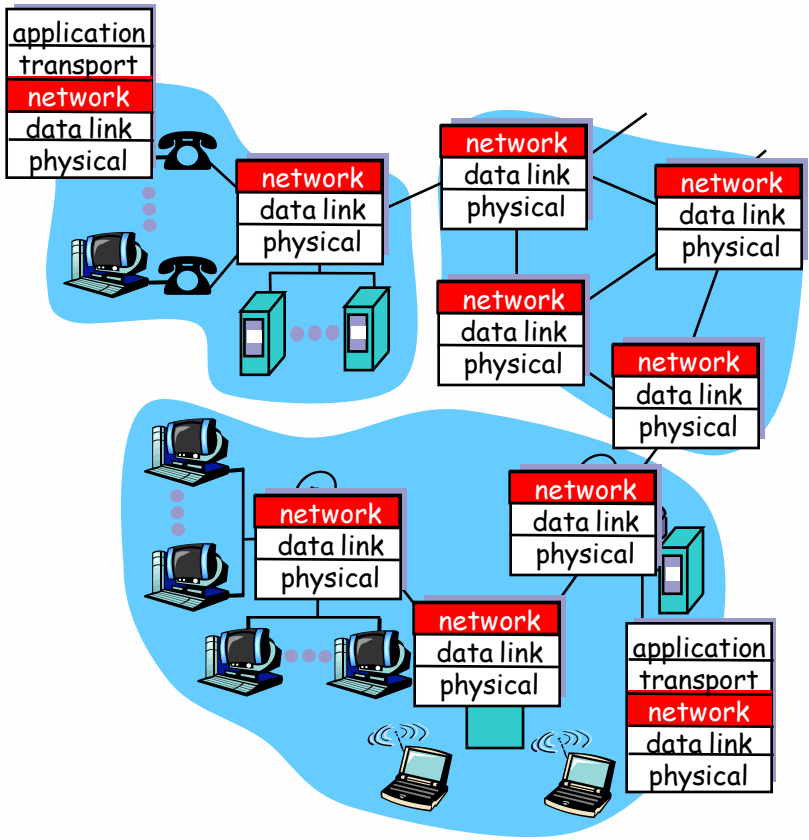
TCP/IP Layers

Chức năng của tầng mạng

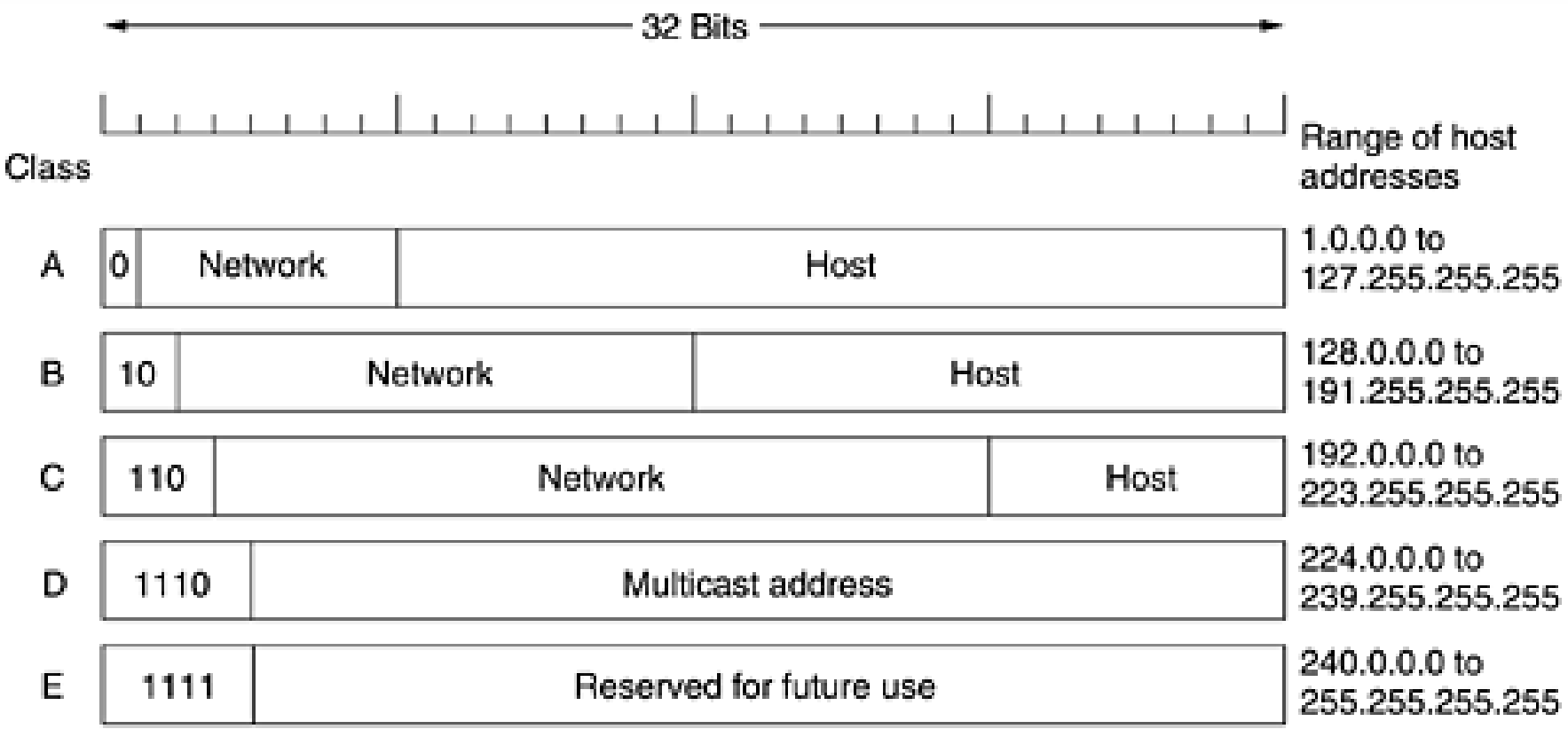
- Đảm nhiệm việc lựa chọn đường đi tốt nhất cho các gói tin (chọn đường, chuyển mạch, thiết lập liên kết) thông qua giao thức IP.

Chức năng giao thức IP :

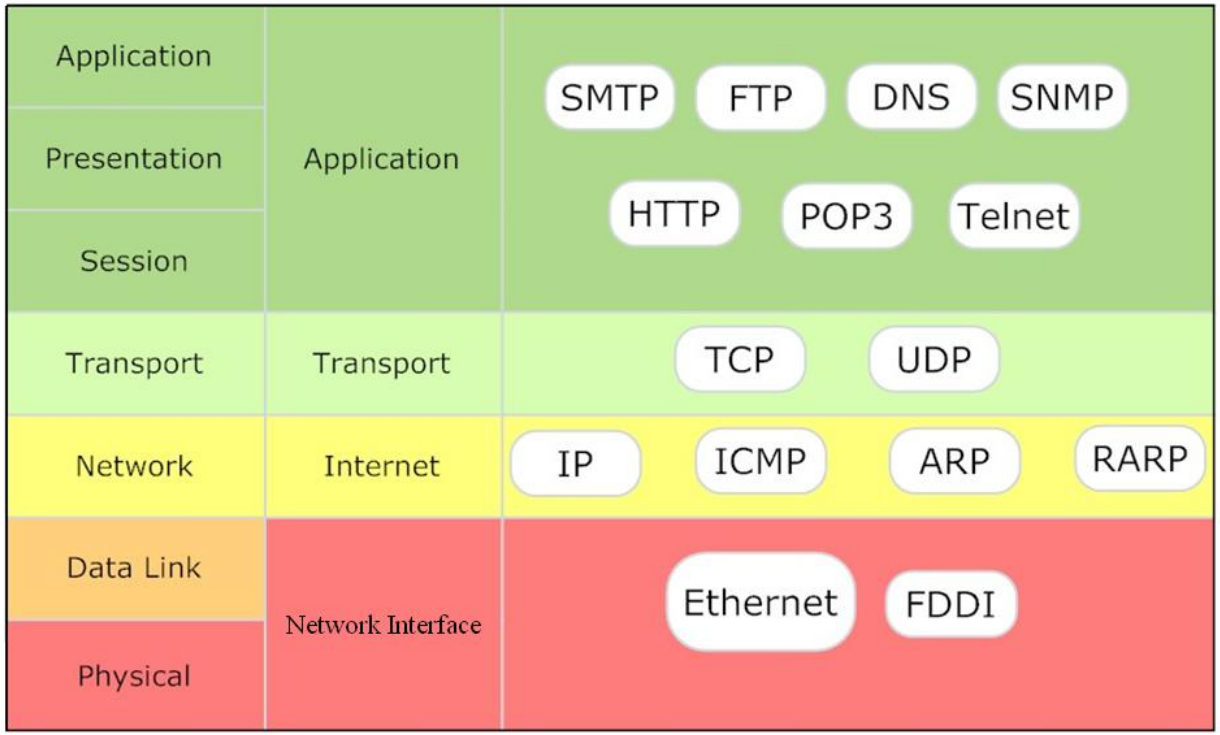
- Xác định lược đồ địa chỉ Internet.
- Di chuyển dữ liệu giữa tầng giao vận và tầng liên kết.
- Dẫn đường cho các đơn vị dữ liệu tới các trạm ở xa.
- Thực hiện việc cắt và hợp dữ liệu.



IP addressing

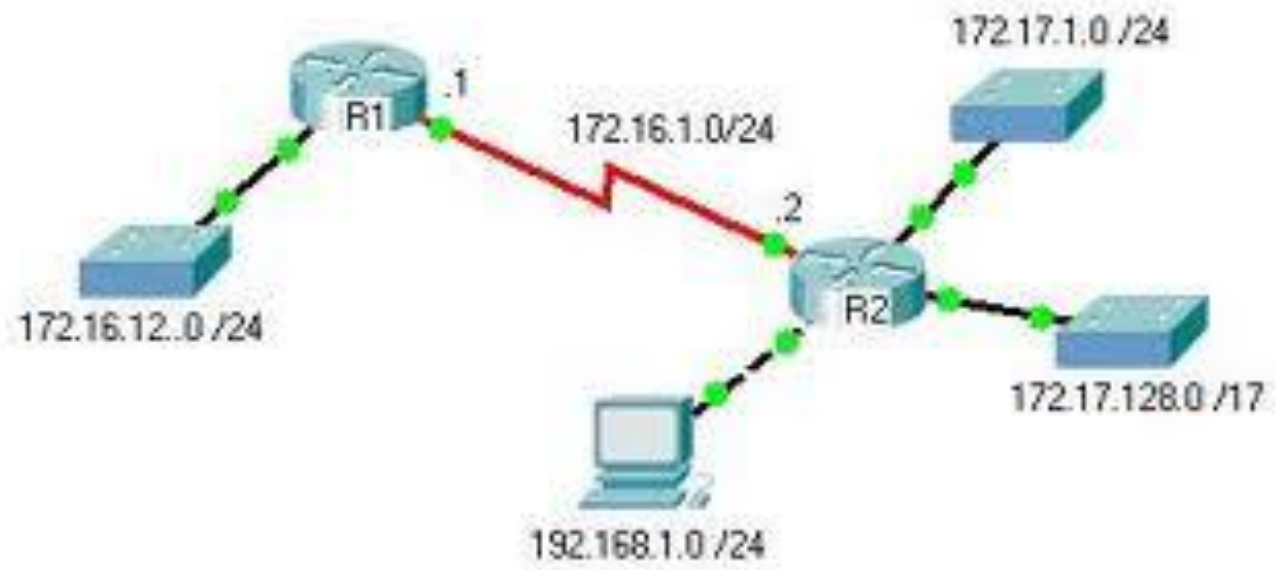


So sánh mô hình OSI và mô hình TCP/IP



Định tuyến

- Định tuyến là quá trình chọn một con đường để truyền một đợn vị dữ liệu từ trạm nguồn đến đích.



Định tuyến

- **Phân loại :**

- Định tuyến tập trung.
- Định tuyến phân tán.
- Định tuyến tĩnh.
- Định tuyến động.

Một số giao thức định tuyến động như RIP, EIGRP, OSPF, IGRP…

Mục tiêu:

- Tối ưu hiệu năng mạng.
- Tối thiểu giá thành mạng.
- Tối ưu tham số mạng như băng thông, độ trễ, độ tin cậy, chất lượng gói tin…

Mạng Internet

Sơ lược lịch sủ phát triển :

- Năm 1969, mạng ARPnet ra đời.
- Năm 1974, TCP/IP → Giao thức mạng ARPnet.
- Năm 1982, TCP/IP là giao thức chuẩn của DOD.
- Năm 1983, Internet chính thức ra đời.
- Năm 1989, Tim Berners Lee triển khai thành công dịch vụ WWW.
- 12-1997, Việt Nam chính thức tham gia vào mạng Internet.

Mạng Internet

Các dịch vụ cơ bản :

- Dịch vụ trang tin toàn cầu www.
- Dịch vụ thư điện tử E-mail.
- Dịch vụ truyền tệp tin FTP.
- Dịch vụ truy cập từ xa Telnet.

Kết Luận

- Đồ án được hoàn thành giúp người đọc có được những kiến thức tổng quát và đầy đủ về "Giao thức TCP/IP và mạng Internet".

Hướng phát triển:

- Phát triển giao thức IP thay thế IPv.4 bằng IPv.6, không gian địa chỉ IP từ 32 bits tăng lên 128 bits.
- Tối ưu hóa các giao thức định tuyến để gói tin đi từ nguồn tới đích, trước tiên nó phải tuân theo tập hợp các qui tắc, chọn đường đi để giảm thiểu độ trễ cũng như băng thông mạng.

Thank You!!!