

# Nhắc lại một số kiến thức TCP/IP cơ bản



Ts. Phạm Huy Hoàng

Bộ môn Mạng & Truyền thông

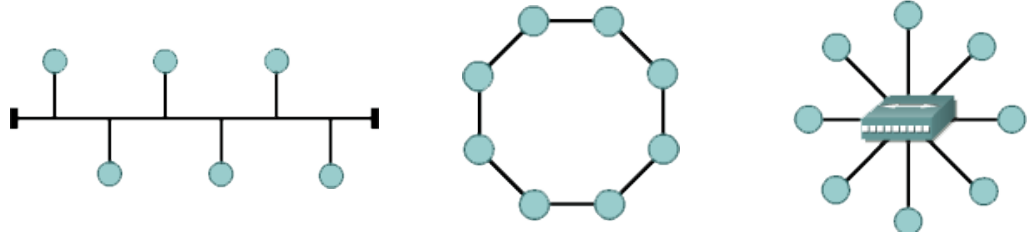
Khoa CNTT, ĐHBK Hà nội

Email: [hoangph@soict.hust.edu.vn](mailto:hoangph@soict.hust.edu.vn)

# Kiến trúc mạng máy tính

## ⌘ Hình trạng (Topology)

- ☑ Bus
- ☑ Ring
- ☑ Star
- ☑ Kết hợp các topology

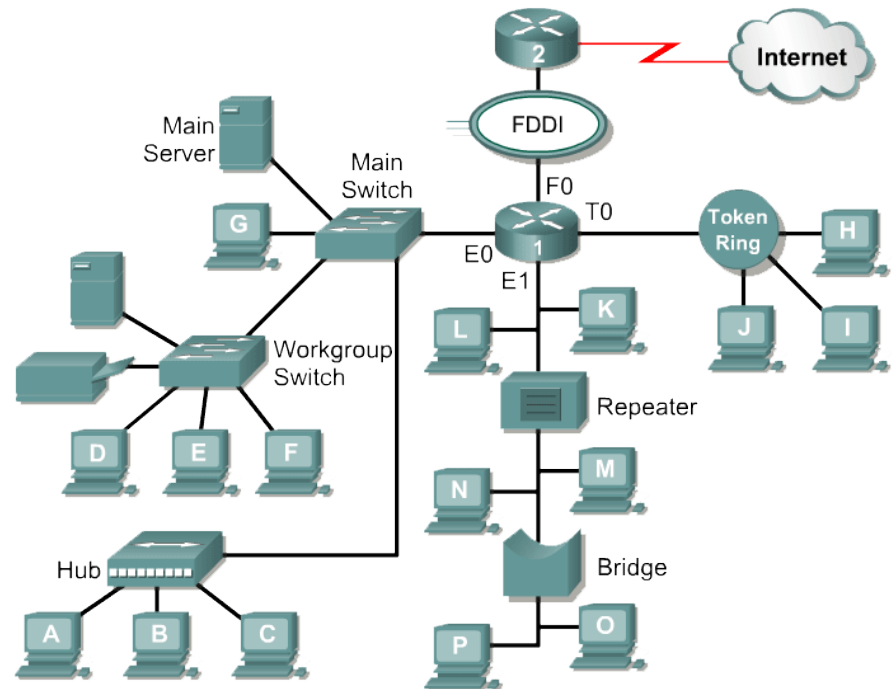


## ⌘ Phần cứng mạng (phần thiết bị)

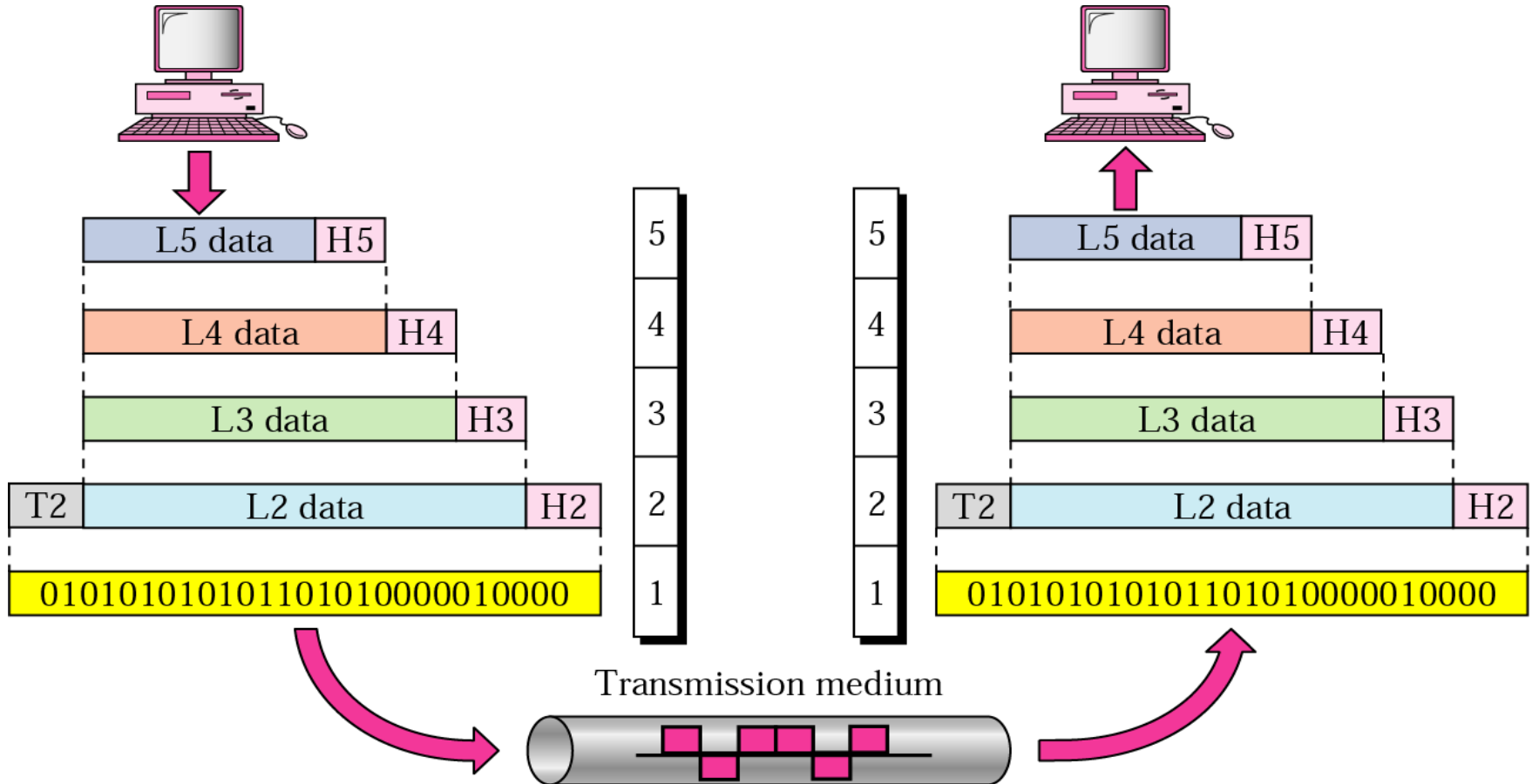
- ☑ Card mạng (NIC)
- ☑ Các thiết bị đường truyền
- ☑ Các thiết bị chuyển mạch

## ⌘ Phần mềm mạng

- ☑ Hệ điều hành mạng
- ☑ Các ứng dụng trên mạng

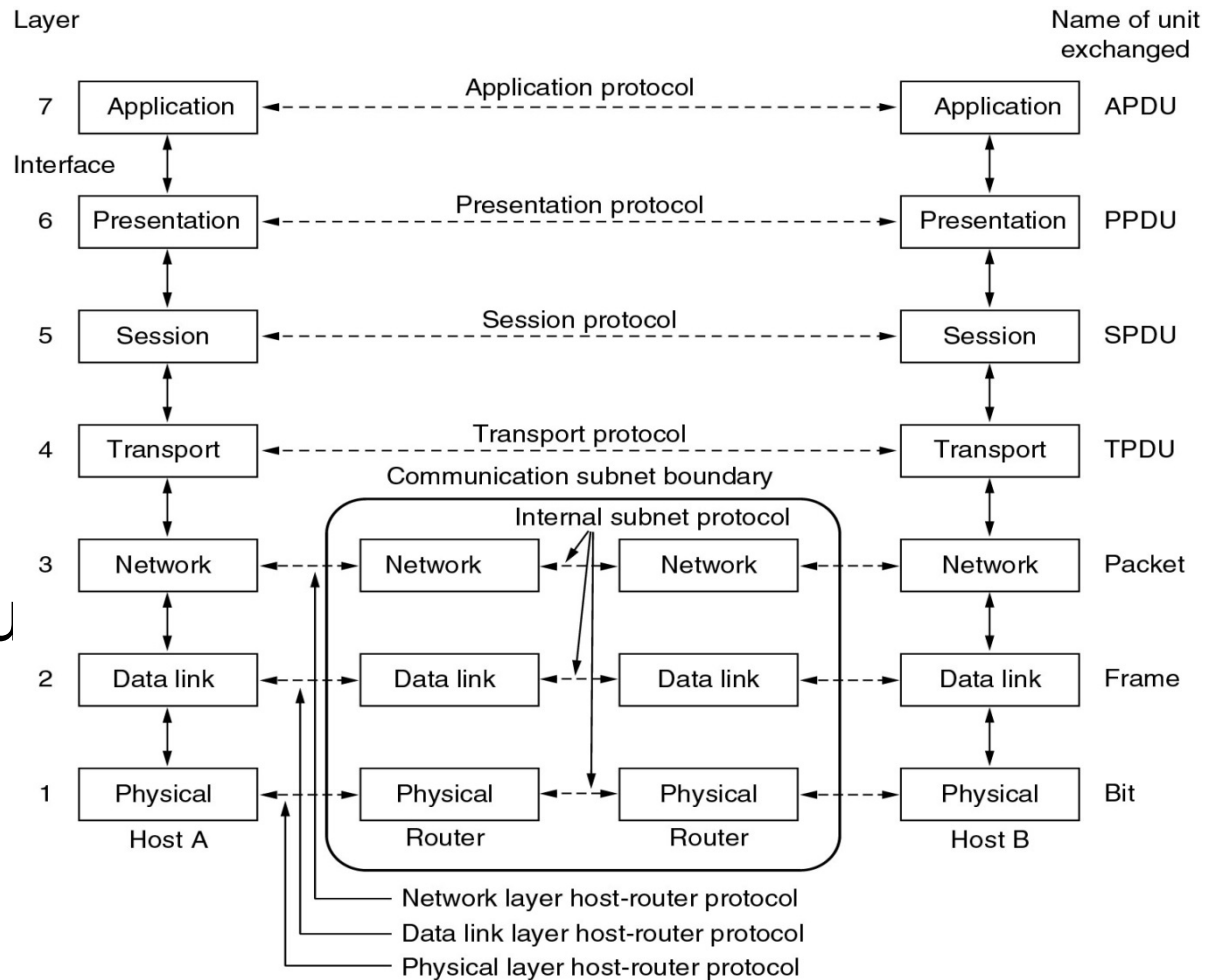


# Chuyển mạch gói & kiến trúc phân tầng



# Chuẩn OSI: Mô hình tham chiếu

- ⌘ Ứng dụng
- ⌘ Trình diễn
- ⌘ Phiên
- ⌘ Giao vận
- ⌘ Mạng
- ⌘ Liên kết dữ liệu
- ⌘ Vật lý

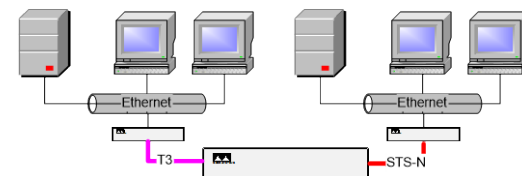
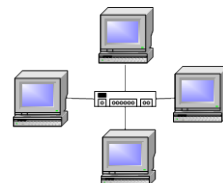


# Kết nối ở các tầng khác nhau

## ⌘ Hub (tầng 1)

☑ “trung tâm” của mạng hình sao

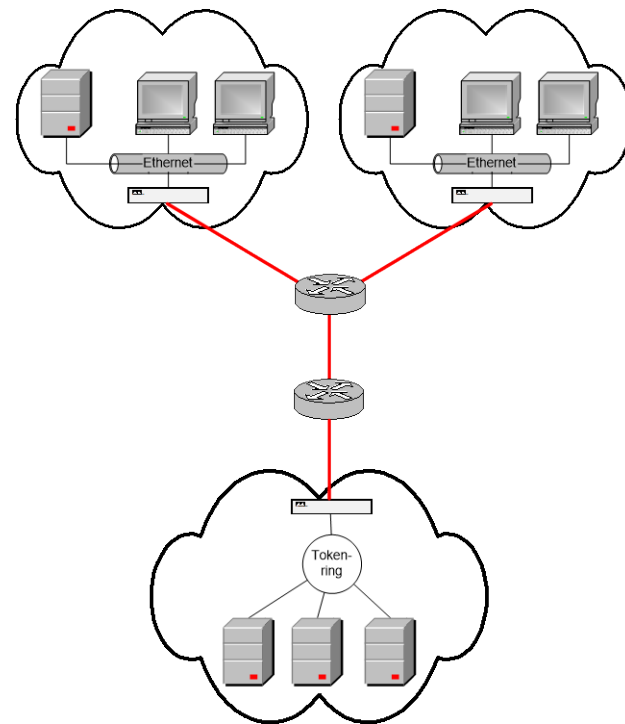
☑ Ethernet: multiport repeater/concentrator



## ⌘ Switch (tầng 2)

☑ packet-switched (dựa trên địa chỉ MAC)

☑ Giảm ñụng ñộ



## ⌘ Router (tầng 3)

☑ Default gateway

☑ Chuyển tiếp gói tin dựa trên địa chỉ IP

☑ Cô lập các broadcast domains

# Bên trong kiến trúc phân tầng

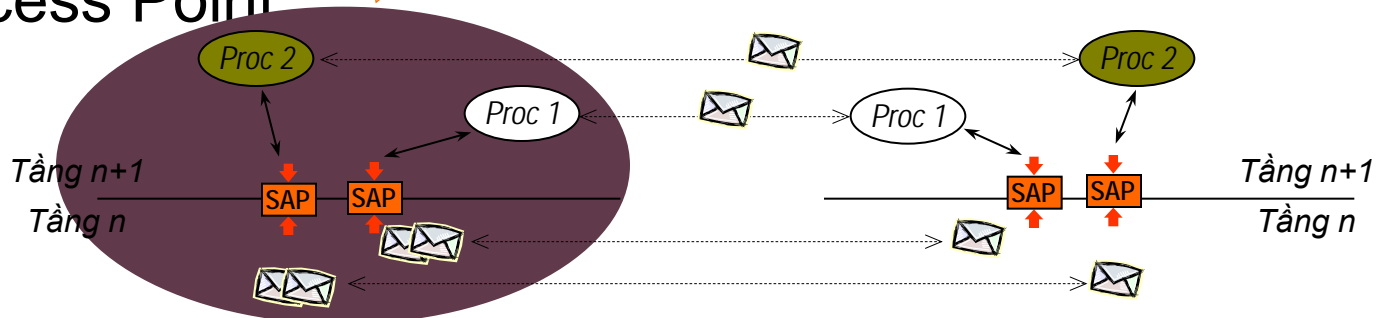
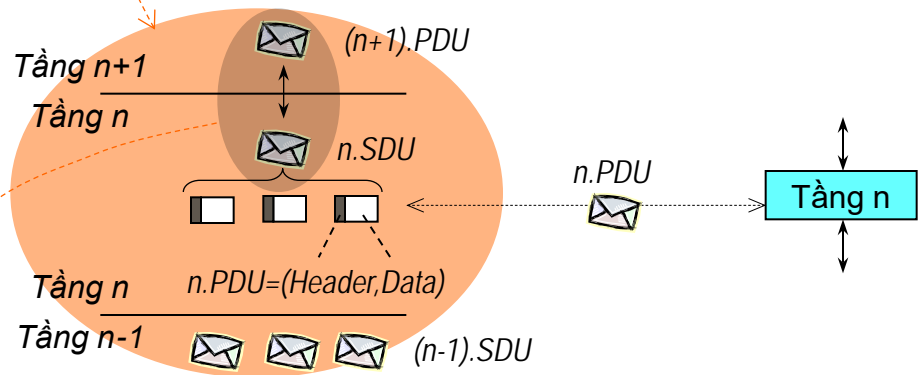
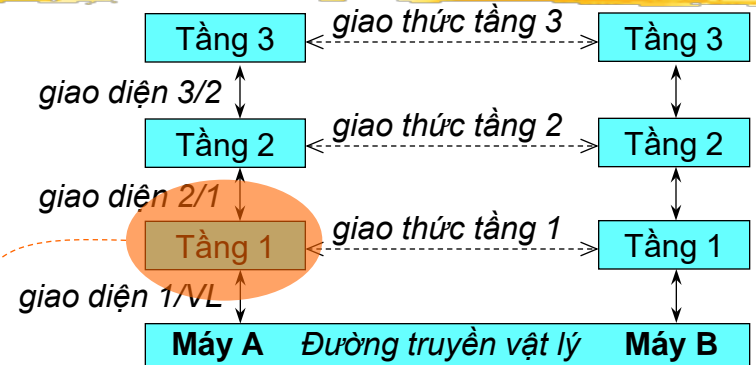
## ⌘ Kiến trúc phân tầng

- ☑ Số lượng & chức năng các tầng
- ☑ Giao tiếp giữa hai “agent” cùng tầng (trên 2 trạm khác nhau) → giao thức tầng N
- ☑ Giao tiếp giữa 2 tầng kề nhau

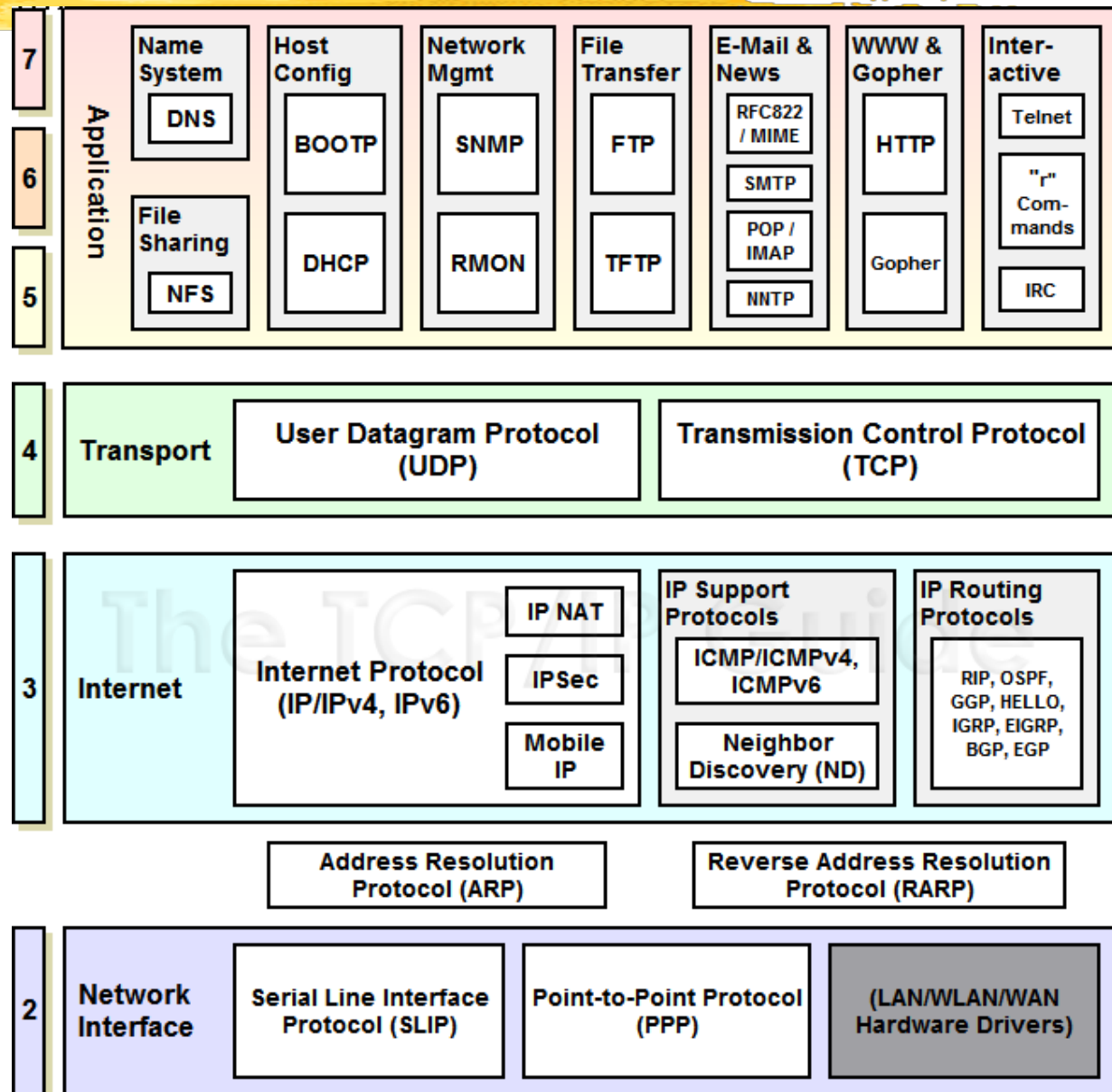
## ⌘ PDU (Package Data Unit)

## ⌘ SDU (Service Data Unit)

## ⌘ Service Access Point



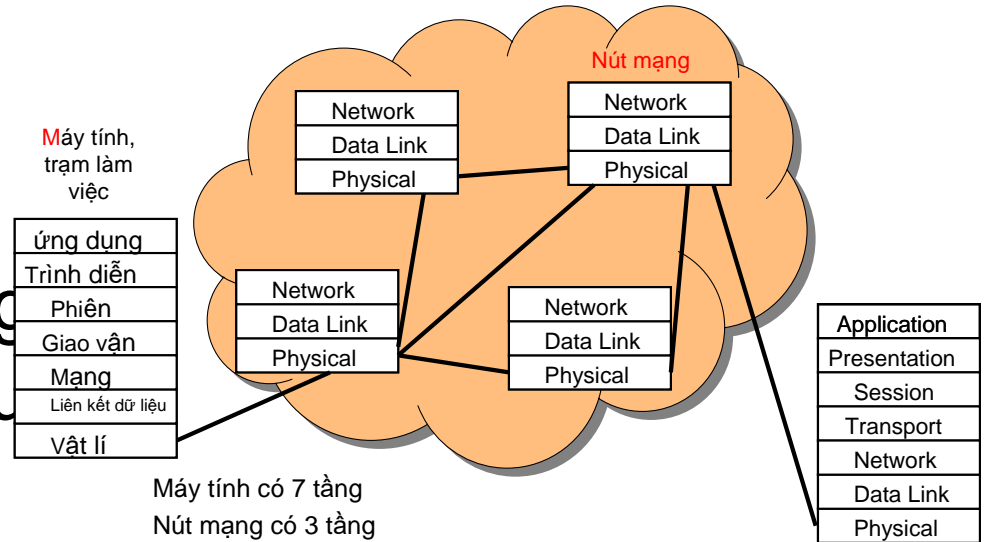
# TCP/IP & Internet: tham chiếu OSI



# Kết nối inter-net: giao thức IP

⌘ Kết nối các mạng không thuần nhất

⌘ Người sử dụng không nhận ra sự khác nhau giữa các mạng



⌘ Liên kết giữa các mạng thực hiện ở tầng mạng:

☑ Tìm đường

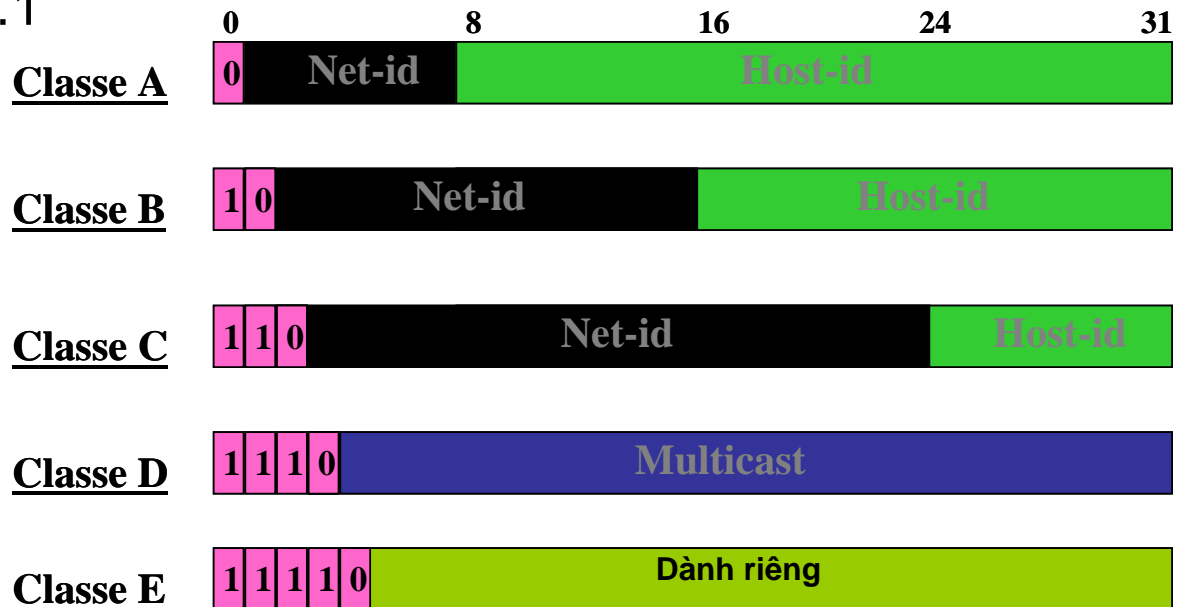
☑ Chuyển tiếp các gói tin giữa các mạng

⌘ Cơ chế chuyển tiếp (store and forward)

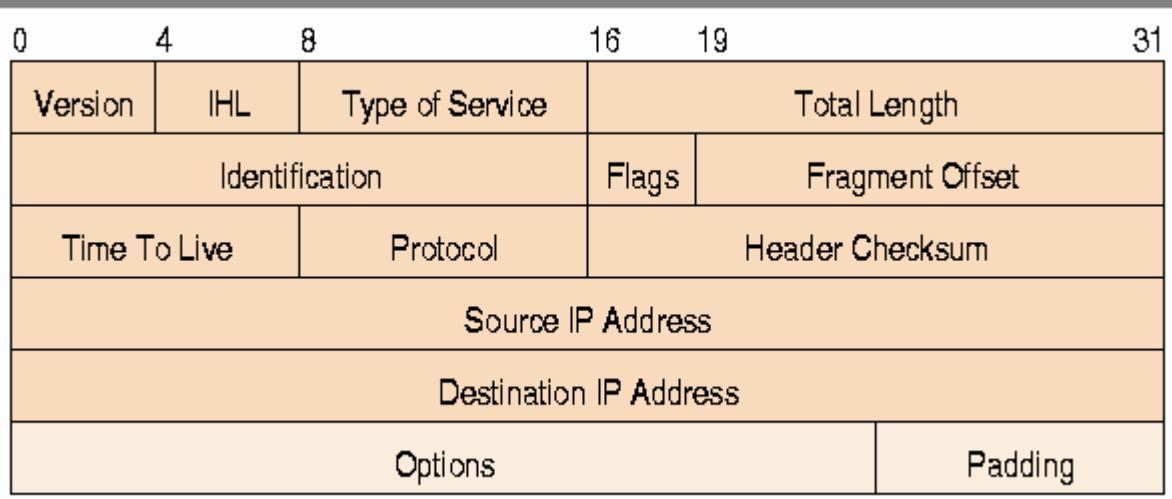


# Địa chỉ IP (phiên bản 4)

- ⌘ 1 @ = 32 bits @IP gồm hai phần (netid, hostid)
- ⌘ Hai phần này tạo ra 5 lớp địa chỉ
- ⌘ @ đặc biệt:
  - ⊞ 127.0.0.1
  - ⊞ hostid = 0...0/1...1



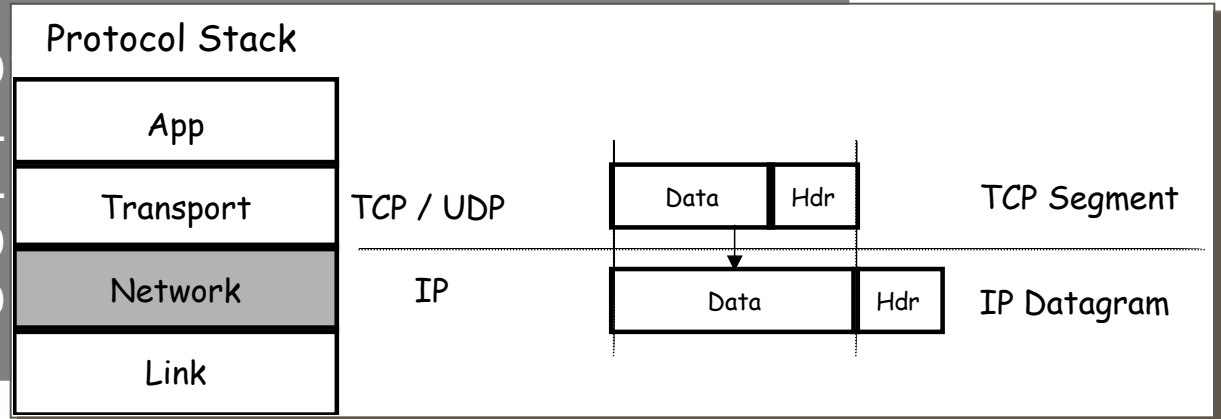
# Gói tin IP



Header

```

101010111010101010
110100101010100101
101000010111010101
001000000000101010
..... 10110110
    
```



# Subnet & Netmask

- ⌘ Dùng cho các mạng có kích thước không “khớp” với các class chuẩn (A, B, C)
- ⌘ Sử dụng một số bit trong netid làm hostid → tăng độ lớn mạng (~gộp nhiều mạng thành một mạng)
- ⌘ Sử dụng một số bit trong hostid làm netid → chia mạng thành nhiều mạng con
- ⌘ Netmask: cho phép thông báo với hệ điều hành các bit trong netid & các bit trong hostid
- ⌘ Cách thể hiện khác: a.b.c.d/n (n = số bit trong netid)  
VD: 192.168.10.153/24

# IP cho mạng riêng (Private IP)

- ⌘ Khái niệm mạng riêng
- ⌘ Nhu cầu kết nối Internet → phân biệt máy trong mạng & máy trên Internet
- ⌘ IP riêng:
  - ☑ Class A: 10.x.x.x (1 mạng)
  - ☑ Class B: 172.16.x.x – 172.31.x.x (16 mạng)
  - ☑ Class C: 192.168.0.x – 192.168.255.x (256 mạng)

# Địa chỉ MAC & giao thức ARP

⌘ Tầng 2 làm việc với các địa chỉ MAC

⌘ Tầng 3 làm việc với các địa chỉ IP  
→ chuyển đổi IP & MAC

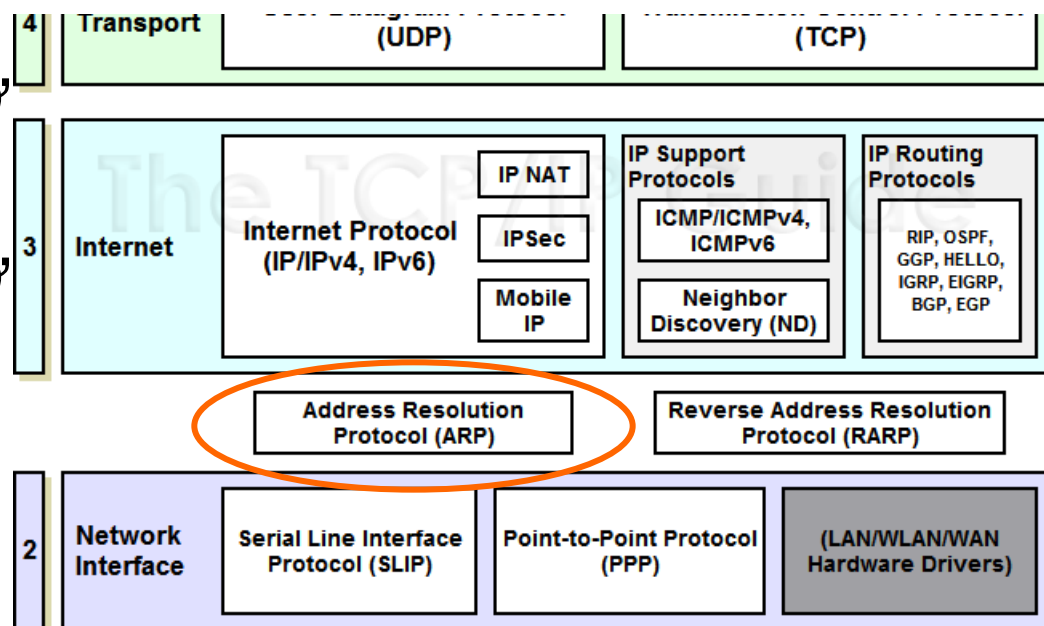
⌘ Giao thức ARP

☑ Máy nguồn quảng bá @IP trên mạng (vật lí)

☑ Máy có @IP trùng trả lời kèm @MAC

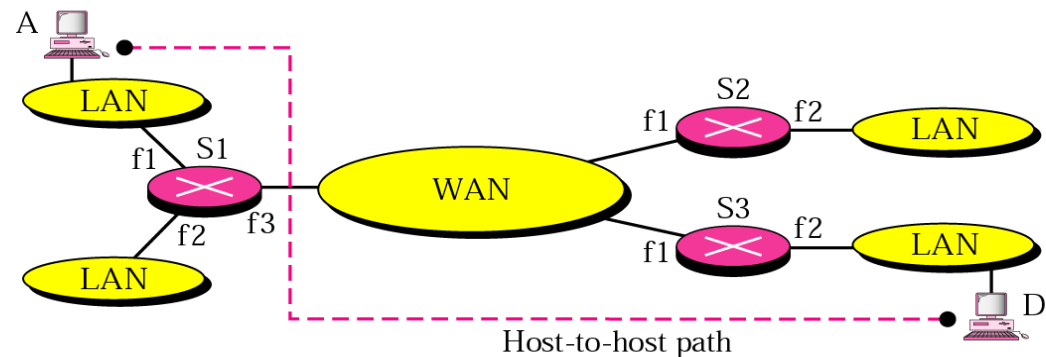
☑ Các máy khác không trả lời

☑ Mỗi máy giữ một bộ nhớ đệm MAC-IP



# Tìm đường - routing

- ⌘ A gửi dữ liệu cho D
- ⌘ A: PDU chuyển từ tầng ứng dụng xuống tầng IP
- ⌘ Theo qui tắc định trước (default gateway) → S1
- ⌘ S1 tìm đường → S3
- ⌘ S3 tìm đường → D



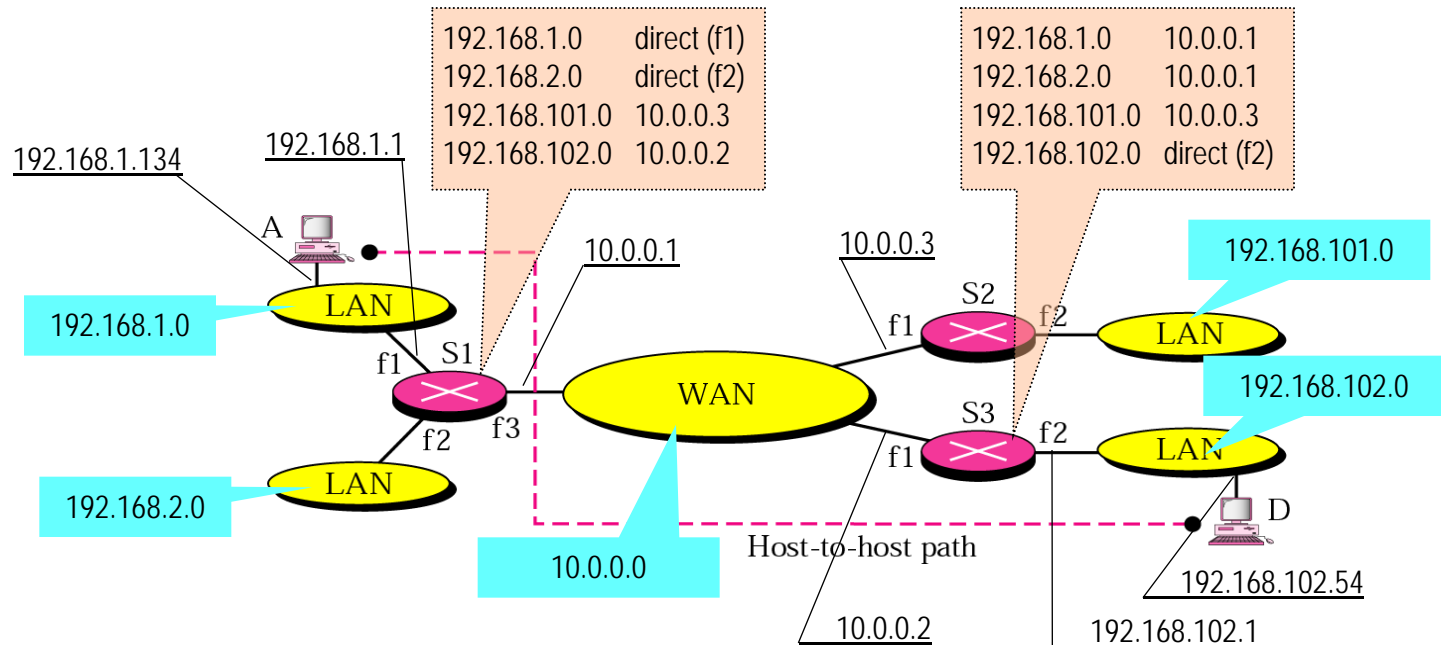
# Tìm đường - routing (2)

⌘ Bảng routing

⌘ Bảng routing được admin config (tìm đường tĩnh)

☑ Mạng đích - next hop (IP hoặc interface)

☑ Tìm đường ~ so sánh địa chỉ đích trong PDU với các địa chỉ mạng đích trong bảng routing



# Tìm đường - routing (3)

## ⌘ Tìm đường tĩnh

- ☑ Đơn giản
- ☑ Không phù hợp với các mạng cỡ lớn (hay có sự thay đổi kiến trúc - thêm mạng, bớt mạng, v.v...)
- ☑ Không cho phép tìm đường tối ưu theo các thông số thay đổi trong quá trình làm việc

## ⌘ Tìm đường động

- ☑ Dựa trên bảng routing
- ☑ Tự động cập nhật bảng routing bằng protocol đặc biệt giữa các router
- ☑ RIP & OSPF → distance vector & link state

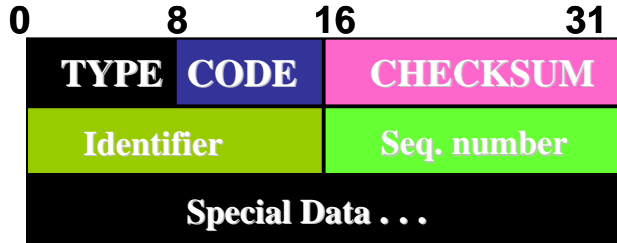
⌘ IP không phải là giao thức tìm đường !!! → routed protocol



# Giao thức ICMP

- ⌘ Kiểm soát quá trình truyền tin bằng IP: KS lỗi, KS tình trạng của mạng
- ⌘ Thông báo lỗi cho trạm gửi.
- ⌘ VD về lỗi kết nối:
  - ⊞ Máy đích không kết nối,
  - ⊞ TTL=0,
  - ⊞ Tắc nghẽn ở các trạm trung gian
- ⌘ Khi một cổng phát hiện lỗi của một gói tin, cổng hủy gói tin và gửi về cho nguồn một thông báo ICMP báo lỗi.
- ⌘ Thông báo ICMP được chuyển bằng một gói tin IP.
- ⌘ Lỗi của thông báo ICMP không sinh ra một thông báo ICMP khác.

# Gói tin ICMP



<u>TYPE</u>	<u>Message ICMP</u>
0	Echo Reply
3	Destination Unreachable
4	Source Quench
5	Redirect (change a route)
8	Echo Request
11	Time Exceeded (TTL)
12	Parameter Problem with a

<u>TYPE</u>	<u>Message ICMP</u>
13	Timestamp Request
14	Timestamp Reply
15	Information Request (obsolete)
16	Information Reply (obsolete)
17	Address Mask Reques
18	Address Mask Reply

# Tầng giao vận

⌘ Cung cấp giao diện cho các chương trình ứng dụng (API)

- ☑ trao đổi thông tin sử dụng mạng các máy tính
- ☑ Trong suốt
- ☑ Chất lượng dịch vụ

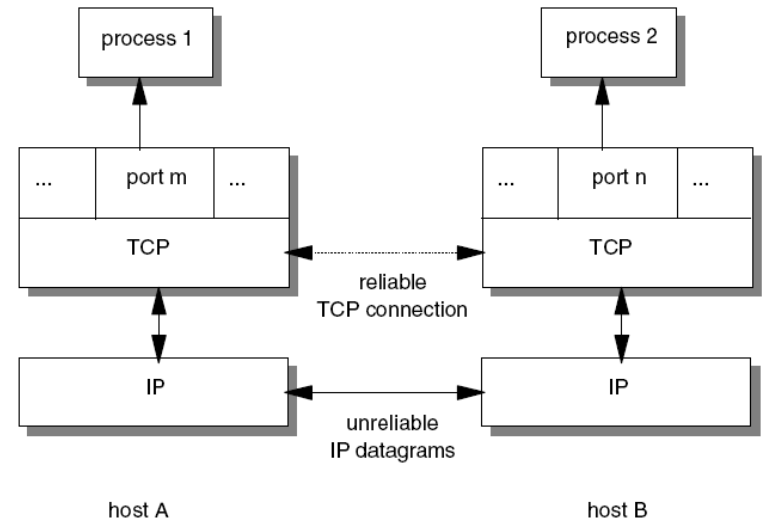
⌘ Là các module phần mềm nằm trên máy nguồn và máy đích

(không nằm trên các máy trung gian như ở tầng mạng)

- ☑ Dồn tách kênh (nhiều chương trình ứng dụng sử dụng cùng 1 kết nối mạng)
- ☑ Kiểm soát lỗi
- ☑ Phục hồi

⌘ Giao thức không tin cậy, không hướng kết nối: UDP

⌘ Giao thức tin cậy hướng kết nối: TCP



# Giao thức TCP

## ⌘ Truyền tin tin cậy

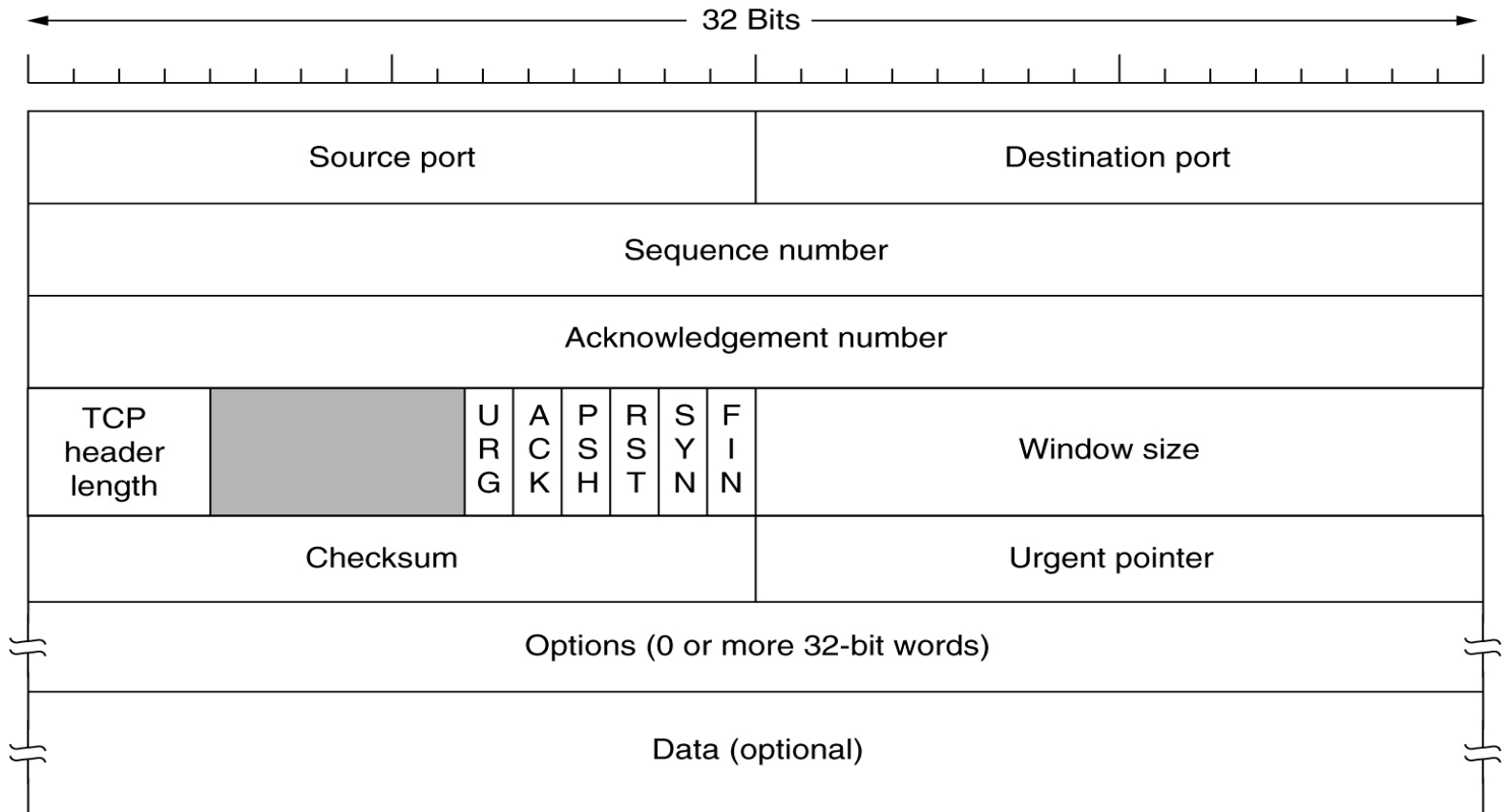
- ☑ Chia thông tin thành các phân đoạn thông tin
- ☑ Đóng gói trong các gói tin IP và chuyển đến đích bằng IP
- ☑ Kiểm soát lỗi, báo nhận, truyền lại nếu cần

## ⌘ Dịch vụ hướng kết nối, đảm bảo truyền tin tin cậy, đảm bảo thứ tự của các phân đoạn

## ⌘ Quy trình

- ☑ Thiết lập kết nối (Thỏa thuận các tham số truyền dữ liệu)
- ☑ Trao đổi dữ liệu (trao đổi các phân đoạn đánh số + báo nhận)
- ☑ Giải phóng kết nối (giải phóng kết nối đảm bảo không mất dữ liệu)

# Gói tin TCP

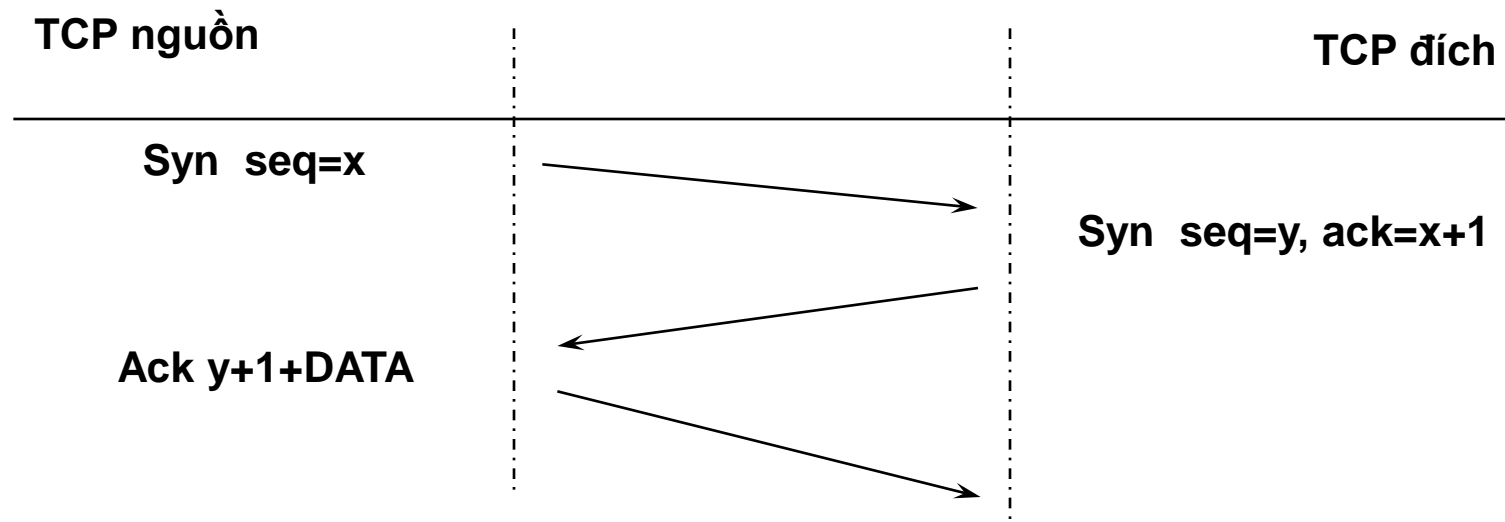


# Kết nối TCP

- ⌘ Một kết nối ảo được thiết lập trước khi truyền tin
- ⌘ Một kết nối = liên kết giữa hai tiến trình = bộ hai điểm đầu kết nối
- ⌘ Một điểm đầu = bộ (@ IP, cổng) (khái niệm cổng giống UDP, 16 bits) → Socket
- ⌘ Thiết lập kết nối:
  - ☑ Một tiến trình (chủ - server) mở một kết nối thụ động, đăng ký nghe trên 1 cổng (nhận dữ liệu trên một cổng),
  - ☑ Một tiến trình khác (khách - client) chủ động mở kết nối vào cổng đã mở bằng cách gửi yêu cầu kết nối vào cổng đó.

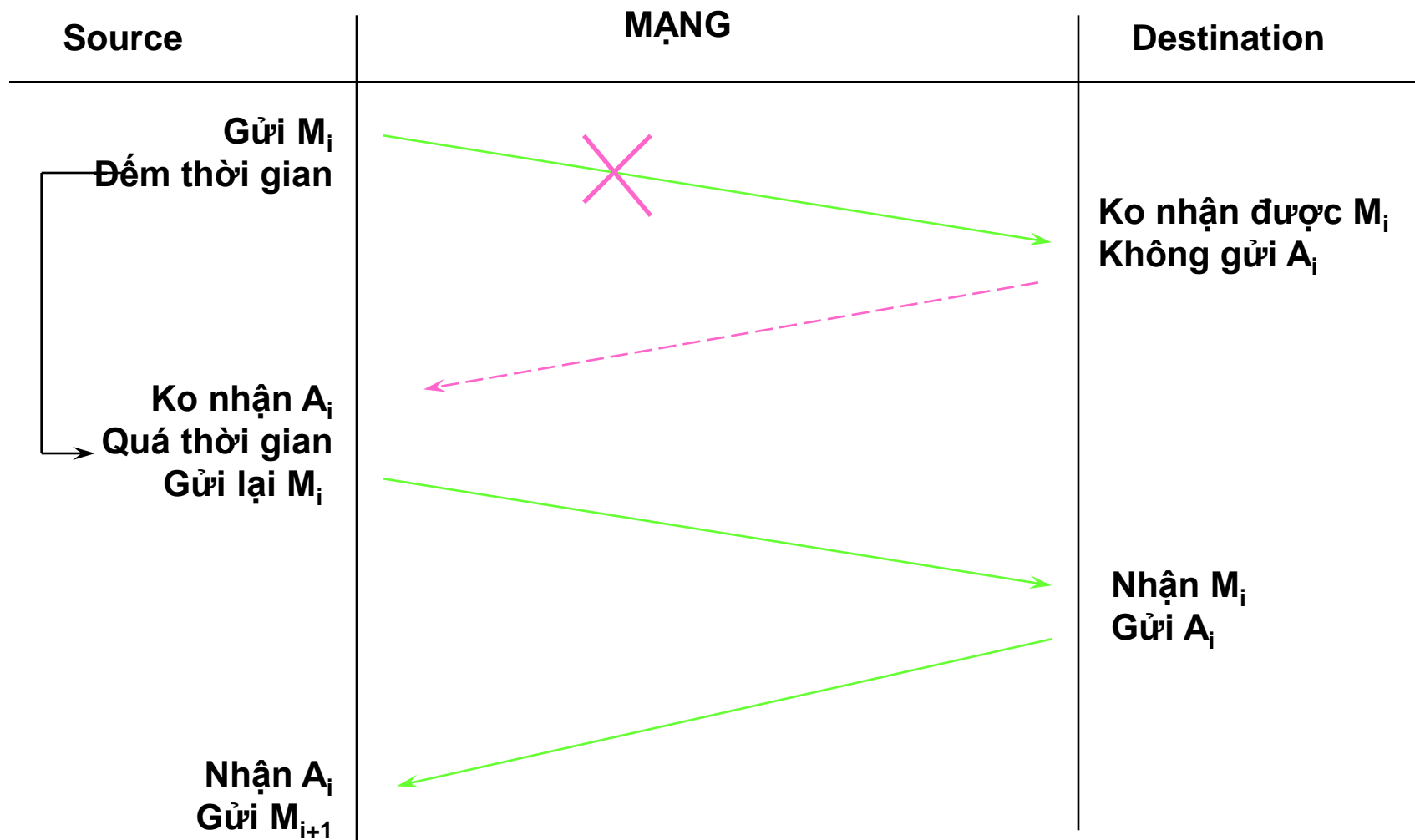
# Kết nối TCP - Thiết lập liên kết

- ⌘ Một kết nối TCP được thiết lập qua 3 bước đảm bảo đồng bộ giữa hai đầu kết nối:



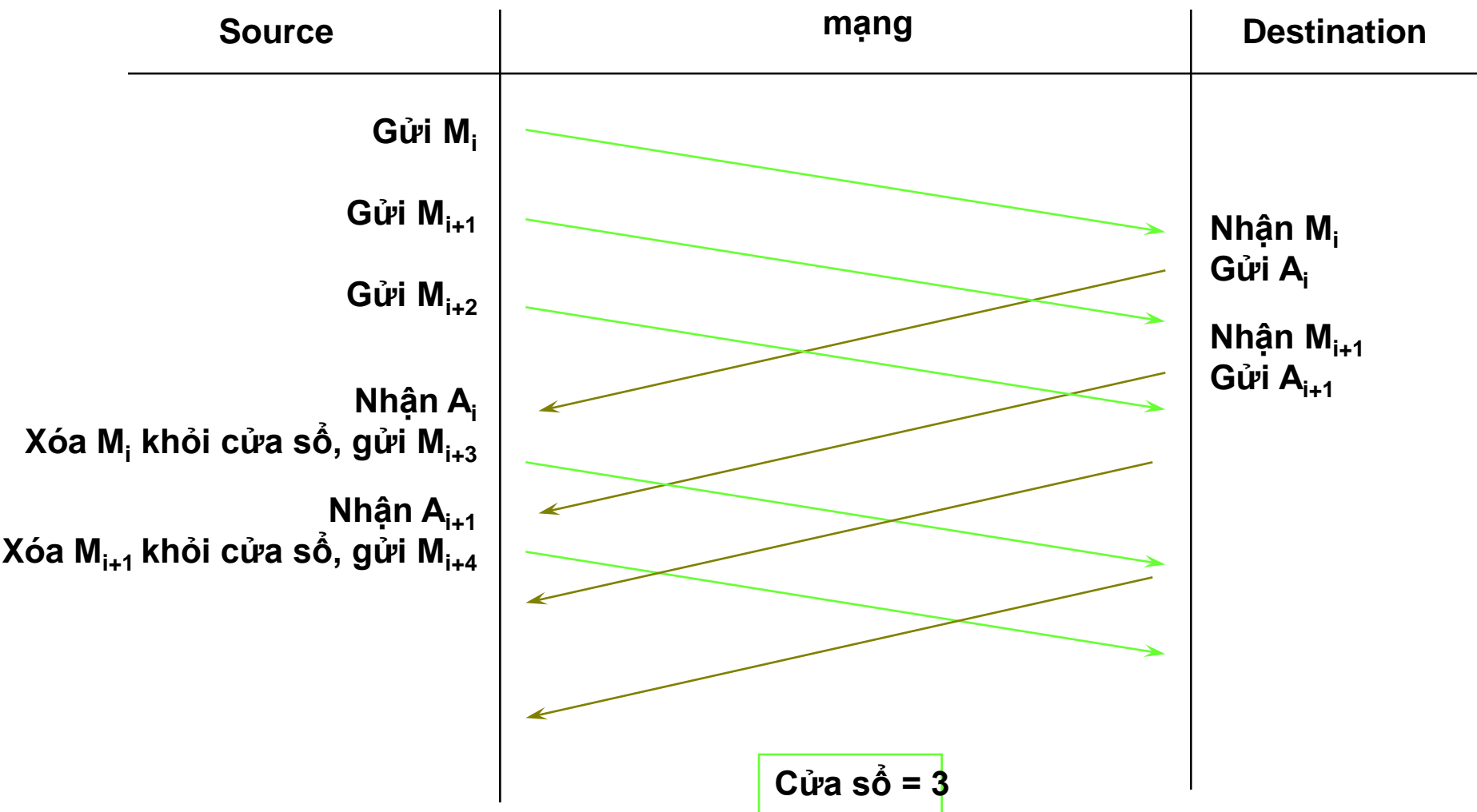
- ⌘ Sơ đồ dùng cho trường hợp cả hai đầu cùng thiết lập liên kết. TCP bỏ qua tất cả các yêu cầu thiết lập liên kết khi đã có liên kết được thiết lập.

# Kết nối TCP - Stop & Wait





# Kết nối TCP - Sliding Window



# Giải phóng kết nối TCP

TCP nguồn

TCP đích

**FIN seq=x**  
Ngừng gửi dữ liệu  
Tiếp tục nhận dữ liệu

**ack=x+1**  
**+ FIN-> tăng ứng dụng**  
Ngừng gửi, nhận dữ liệu

**Ứng dụng -> close**  
**FIN seq=y ack=x+1**

**Ack y+1**  
Ngừng nhận dữ liệu

# Cổng TCP thông dụng

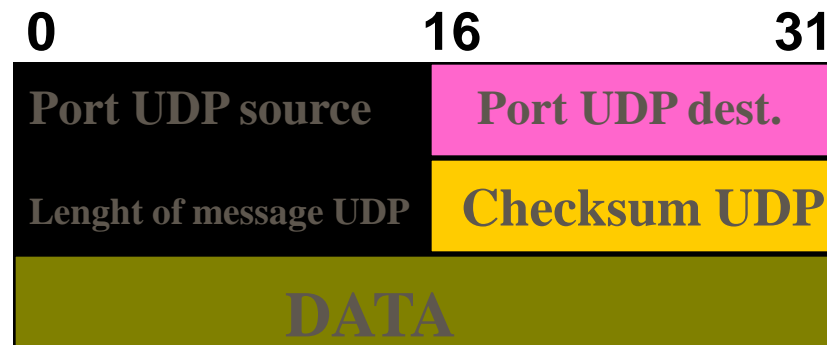
<u>No_port</u>	<u>Keyword</u>	<u>Description</u>
20	FTP-DATA	File Transfer [Default Data]
21	FTP	File Transfer [Control]
23	TELNET	Telnet
25	SMTP	Simple Mail Transfer
37	TIME	Time
42	NAMESERVER	Host Name Server
43	NICNAME	Who Is
53	DOMAIN	Domain Name Server
79	FINGER	Finger
80	HTTP	WWW
110	POP3	Post Office Protocol - Version 3
111	SUNRPC	SUN Remote Procedure Call

# Giao thức UDP

- ⌘ Giao thức thông báo (bức dữ liệu) của NSD : giao thức truyền tin hướng không kết nối, không tin cậy:
  - ☑ Truyền các thông báo của NSD, CT thiết lập các kết nối
  - ☑ Kết quả truyền tin và thứ tự của các thông báo không được đảm bảo.
  - ☑ Ví dụ: trao đổi thông tin bằng thư
- ⌘ Định danh của dịch vụ: cổng (ports)
  - ☑ @IP định danh máy tính. Trên 1 máy có nhiều CT cùng chạy. Các CT muốn trao đổi thông tin lẫn nhau
  - ☑ Định danh các tiến trình bằng một biến trừu tượng, không phụ thuộc máy :
    - ☑ Tiến trình khởi động, kết thúc động trên các máy, ,
    - ☑ Tiến trình có thể thay thế lẫn nhau,
    - ☑ Dịch phải được định danh theo dịch vụ, ko phải theo CT
    - ☑ Một tiến trình có thể cung cấp nhiều dịch vụ.
  - ☑ Port ~ Service Access Point giữa tầng ứng dụng và tầng giao vận

# Gói tin UDP

- ⌘ Port → số nguyên 0 đến 65535
- ⌘ Trao đổi thông báo tiến hành giữa một cổng nguồn và một cổng đích
- ⌘ Các tiến trình có thể truy cập một cổng trên một máy
- ⌘ Truy cập vào cổng thường là đồng bộ, các thao tác trên cổng được đánh dấu và ghi lại.

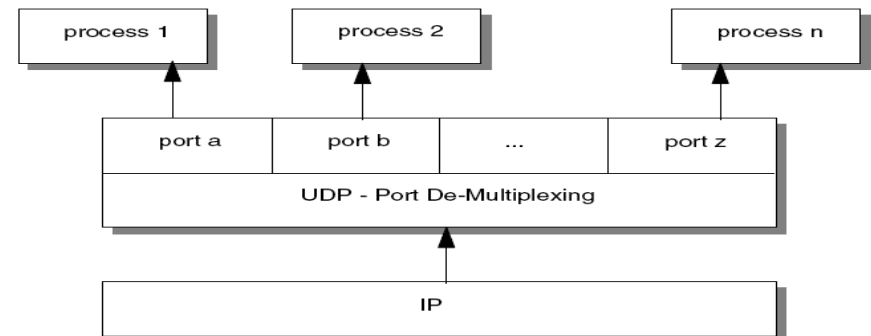


# Các cổng UDP thông dụng

⌘ *well-known port assignments:*

No port	Name	Description
7	ECHO	Echo
11	USERS	Active Users
13	DAYTIME	Daytime
37	TIME	Time
42	NAMESERVER	Host Name Server
53	DOMAIN	Domain Name Server
67	BOOTPS	Boot protocol server
68	BOOTPC	Boot protocol client
69	TFTP	Trivial File Transfer protocol
123	NTP	Network Time Protocol
161	SNMP	Simple Network Management Protocol.

⌘ Các cổng khác được gán động cho các tiến trình



# Internet Socket

- ✂ Bộ đôi (địa chỉ IP, số hiệu cổng TCP/UDP): xác định duy nhất một ứng dụng đang chạy trên Internet
- ✂ Kết hợp 2 bộ đôi (truyền & nhận): một dòng dữ liệu truyền trên Internet
- ✂ Ứng dụng tập trung vào xử lý bài toán nghiệp vụ riêng, không phải quan tâm đến vấn đề truyền các gói tin trên mạng
- ✂ Thư viện hàm Socket API cho phép ứng dụng giao tiếp với Protocol Stack thông qua Socket để gửi/nhận dữ liệu

