





# GIAO THỨC MẠNG

Giảng viên HD: PGS.TS Nguyễn Đình Việt

Nhóm thực hiện:

Trịnh Việt Dũng

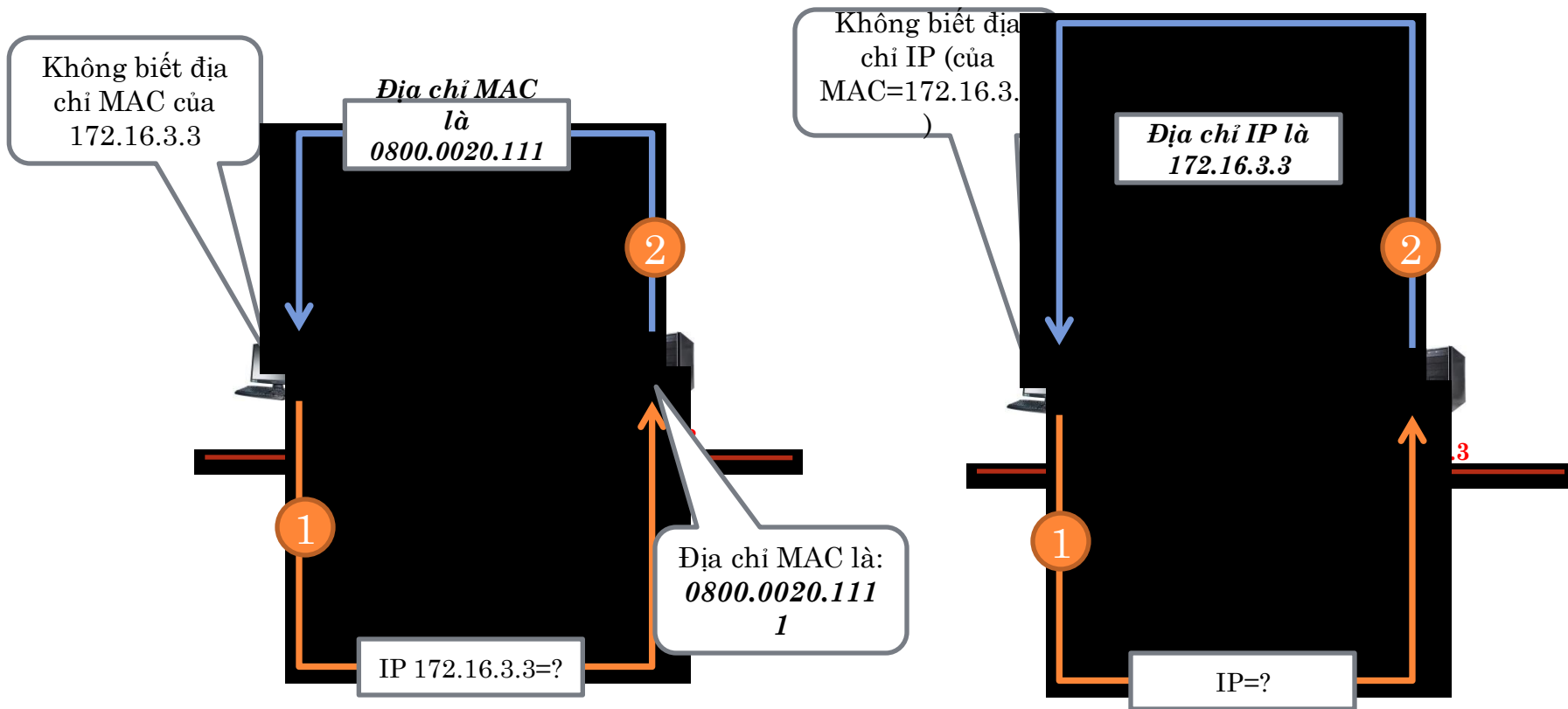
Trịnh Văn Thành

Hoàng Thị Vân

# NỘI DUNG TRÌNH BÀY




# 1. GIAO THỨC ARP & R-ARP



# 1.1. GIAO THỨC ARP

## 1.1.1. *Khái niệm*

-ARP là phương thức phân giải địa chỉ động giữa địa chỉ lớp network và địa chỉ lớp datalink. Quá trình thực hiện bằng cách: một thiết bị IP trong mạng gửi một gói tin local broadcast đến toàn mạng yêu cầu thiết bị khác gửi trả lại địa chỉ phần cứng (địa chỉ lớp datalink) hay còn gọi là Mac Address của mình.

- Ban đầu ARP chỉ được sử dụng trong mạng Ethernet để phân giải địa chỉ IP và địa chỉ MAC. Nhưng ngày nay ARP đã được ứng dụng rộng rãi và dùng trong các công nghệ khác dựa trên lớp hai.



# 1.1. GIAO THỨC ARP

## 1.1.2. Cấu trúc gói tin

Có hai dạng bản tin trong ARP :

- Một được gửi từ nguồn đến đích(Request)
- Một được gửi từ đích tới nguồn(Reply).

Request : Khởi tạo quá trình, gói tin được gửi từ thiết bị nguồn tới thiết bị đích

Reply : Là quá trình đáp trả gói tin ARP request, được gửi từ máy đích đến máy nguồn

Hardware type		Protocol type
HW add length	Protocol add length	Opcode
Sender Hardware Address		
Sender Protocol Address		
Target Hardware Address		
Target Protocol Address		



# 1.1. GIAO THỨC ARP

## 1.1.3. ARP cache

Vì ARP là một giao thức phân giải địa chỉ động. Quá trình gửi gói tin Request và Reply sẽ tiêu tốn băng thông mạng. Chính vì vậy càng hạn chế tối đa việc gửi gói tin Request và Reply sẽ càng góp phần làm tăng khả năng hoạt động của mạng. Từ đó sinh ra nhu cầu của ARP Caching (Có dạng giống như 1 bảng tương ứng giữa địa chỉ phần cứng và địa chỉ IP)

Có 2 loại ARP cache:

**Static** ARP Cache Entries

**Dynamic** ARP Cache Entries



# 1.1. GIAO THỨC ARP

## 1.1.3. ARP cache

### *a) Static ARP Cache Entries*

Đây là cách mà các thành phần tương ứng trong bảng ARP được đưa vào lần lượt bởi người quản trị. Công việc được tiến hành một cách thủ công

Ví dụ: Cấu hình ARP tĩnh cho máy tính:

+ **Windows XP/2003:** arp -s ip\_của\_computer mac\_của\_computer

Windows Vista/2008 : + netsh -c "interface ipv4

+ set neighbors tên\_card\_mạng ip\_của\_computer mac\_của\_computer

Ưu điểm: Đối workstation nên có static ARP entry đến router và file server nằm trong mạng. Điều này sẽ hạn chế việc gửi các gói tin để thực hiện quá trình phân giải địa chỉ

Nhược điểm: + Công việc thiết lập được tiến hành một cách thủ công.

+ Khi địa chỉ IP của các thiết bị trong mạng thay đổi thì sẽ dẫn đến việc phải thay đổi ARP cache



# 1.1. GIAO THỨC ARP

## 1.1.3. ARP cache

### *b) Dynamic ARP Cache Entries:*

Đây là quá trình mà các thành phần địa chỉ hardware/IP được đưa vào ARP cache một cách hoàn toàn tự động bằng phần mềm sau khi đã hoàn tất quá trình phân giải địa chỉ. Chúng được lưu trong cache trong một khoảng thời gian và sau đó sẽ được xóa đi (được lưu trữ khoảng 10-20 phút)

Ưu điểm: tất cả các quá trình diễn ra tự động và không cần đến sự tương tác của người quản trị



# 1.1. GIAO THỨC ARP

## 1.1.3. ARP cache

Dùng lệnh để xem bảng ARP cache: `arp -a`

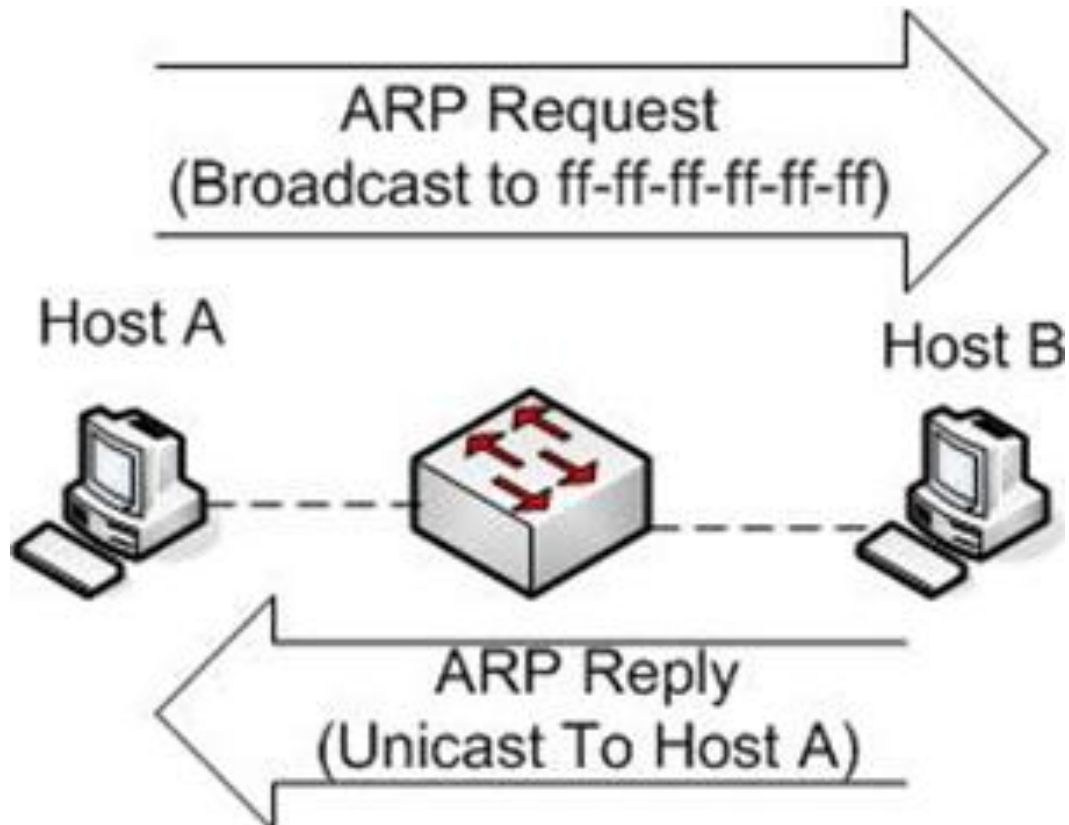
```
C:\Windows\system32\cmd.exe
C:\Users\admin>arp -a
Interface: 192.168.150.155 --- 0xb
Internet Address      Physical Address      Type
192.168.150.2         00-10-db-82-4d-52     dynamic
192.168.150.10        00-0e-7f-af-6d-b8     dynamic
192.168.150.24        00-0f-fe-25-74-40     dynamic
192.168.150.32        00-0b-cd-6e-b8-2c     dynamic
192.168.150.36        00-0f-fe-3a-aa-3f     dynamic
192.168.150.42        00-0f-fe-87-1e-98     dynamic
192.168.150.48        00-0e-7f-63-8d-d1     dynamic
192.168.150.54        00-16-35-ae-3b-a9     dynamic
192.168.150.58        00-16-35-ae-39-53     dynamic
192.168.150.60        00-21-63-68-e9-29     dynamic
192.168.150.62        00-0f-fe-9b-e8-38     dynamic
192.168.150.78        00-0f-fe-3a-a7-d7     dynamic
192.168.150.90        00-0e-7f-f2-f8-e8     dynamic
192.168.150.92        00-0f-fe-3a-a7-96     dynamic
192.168.150.98        00-0f-fe-85-8d-6b     dynamic
192.168.150.114       00-0e-7f-6c-81-25     dynamic
192.168.150.144       00-22-5f-12-67-a2     dynamic
192.168.150.156       00-0f-fe-d1-7e-1e     dynamic
192.168.150.157       00-0f-fe-d1-7e-1e     dynamic
192.168.150.159       00-06-1b-c2-e1-f3     dynamic
192.168.150.208       00-19-66-32-53-25     dynamic
192.168.150.219       00-00-aa-8c-be-07     dynamic
192.168.150.221       00-0e-7f-64-5f-d0     dynamic
192.168.150.255       ff-ff-ff-ff-ff-ff     static
224.0.0.22           01-00-5e-00-00-16     static
224.0.0.251          01-00-5e-00-00-fb     static
224.0.0.252          01-00-5e-00-00-fc     static
224.0.1.134          01-00-5e-00-01-86     static
239.255.255.250      01-00-5e-7f-ff-fa     static
```



# 1.1. GIAO THỨC ARP

## 1.1.4. Proxy ARP

TH1: Gửi nội mạng

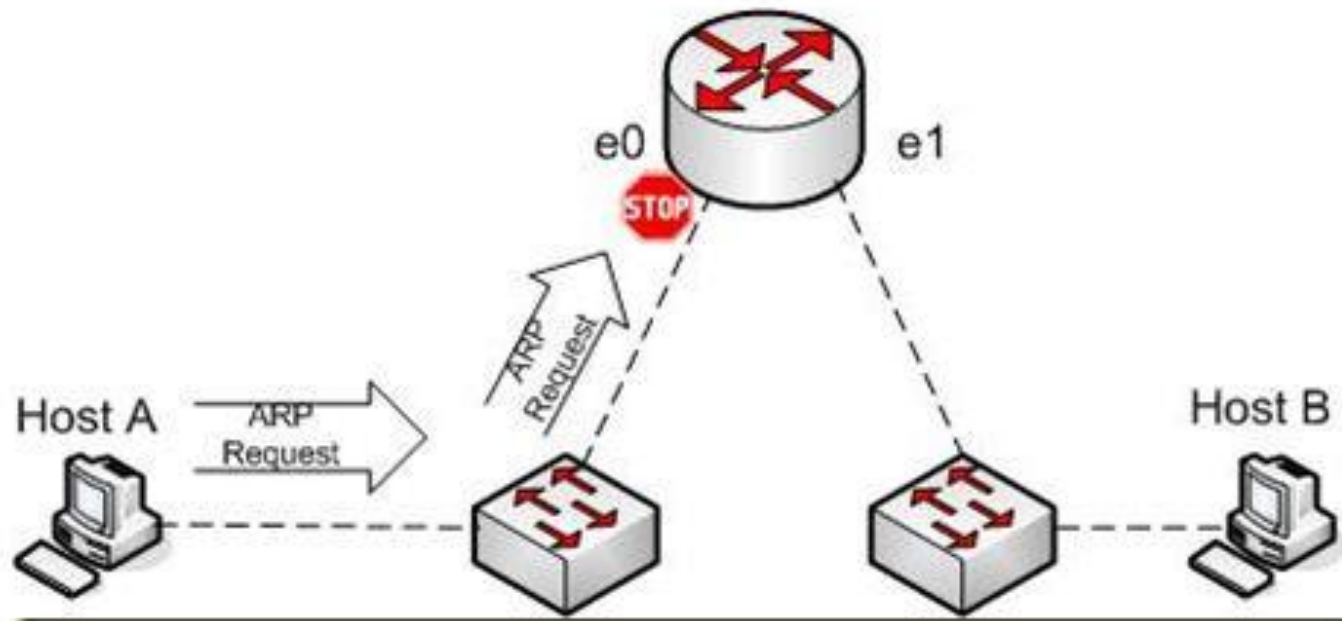


# 1.1. GIAO THỨC ARP

## 1.1.4. Proxy ARP

TH2: Gửi ngoại mạng (Thực trạng)

A không gửi được yêu cầu tới B vì thế A không nhận được trả lời từ B

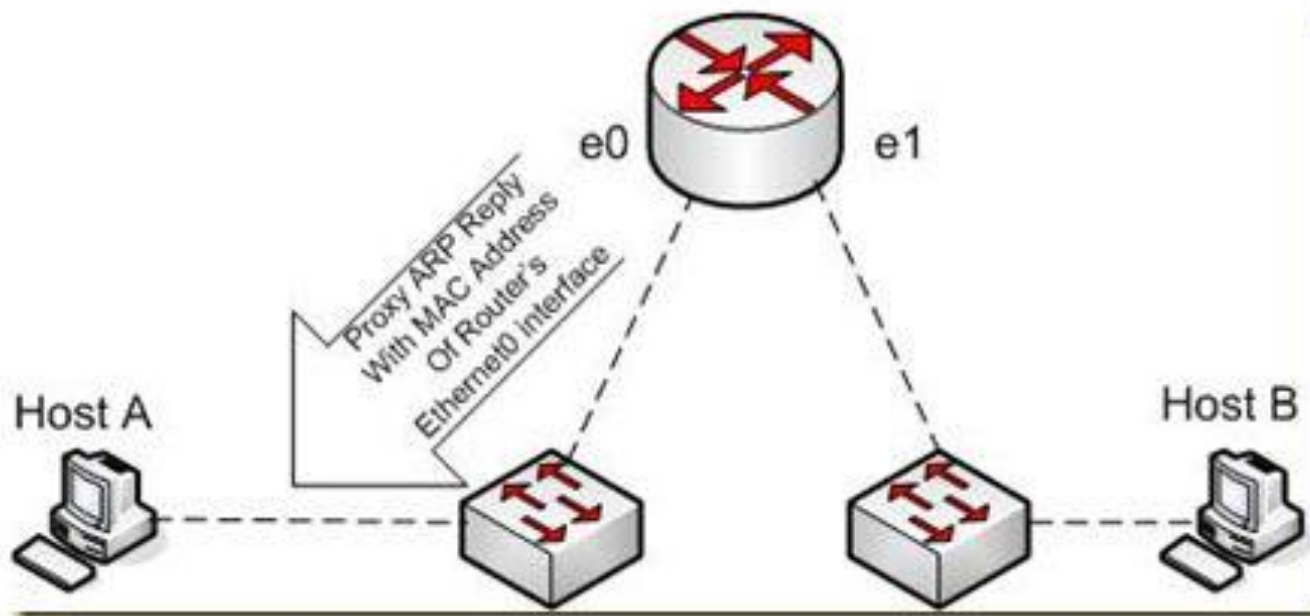


# 1.1. GIAO THỨC ARP

## 1.1.4. Proxy ARP

TH2: Gửi ngoại mạng (Giải pháp)

Router gửi cho A địa chỉ MAC của chính Router

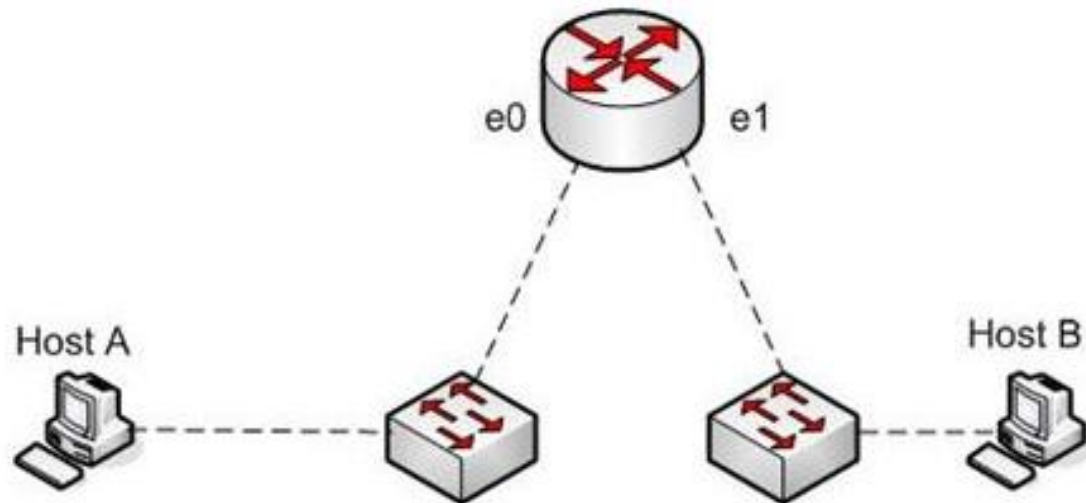


# 1.1. GIAO THỨC ARP

## 1.1.4. Proxy ARP

TH2: Gửi ngoại mạng

A gửi gói tin (request) cho Router sau đó Router gửi lại cho B và ngược lại



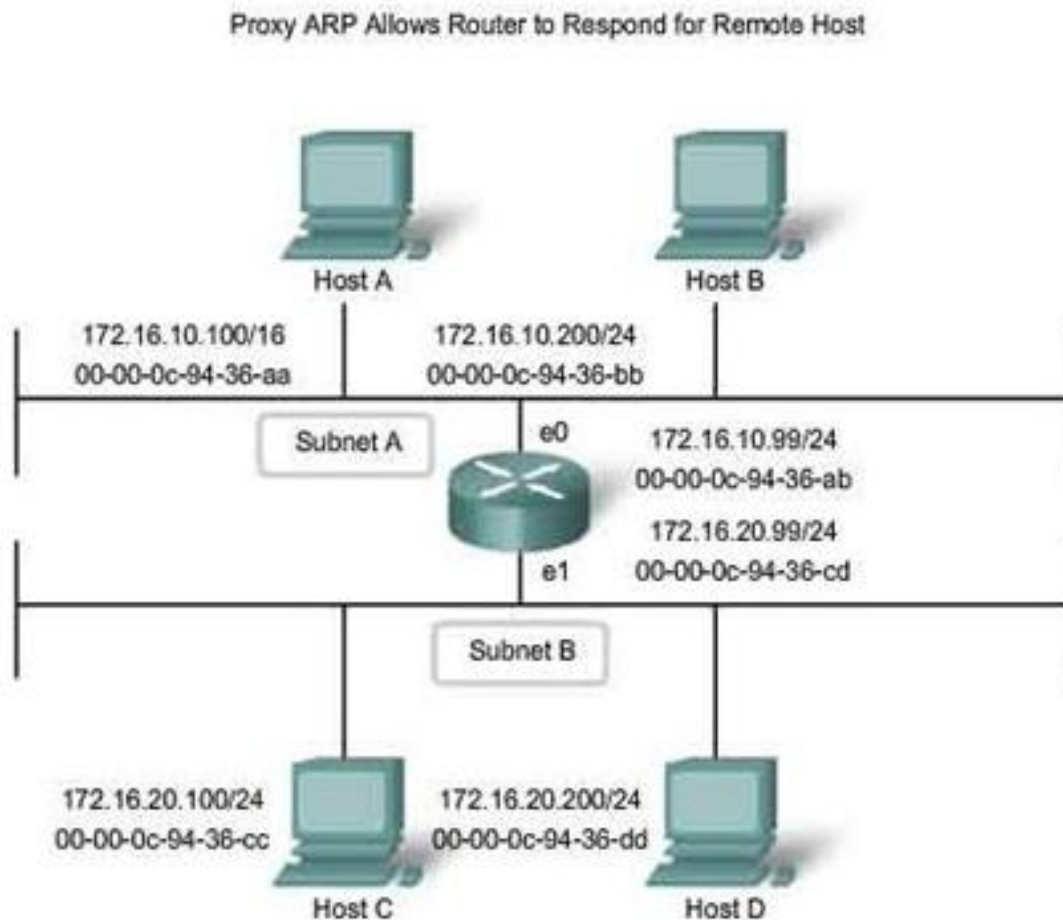
Transmitting To Host B:  
IP Destination Address Is Host B's,  
MAC Destination Address is the  
MAC address of the router's  
ethernet0 interface.



# 1.1. GIAO THỨC ARP

## 1.1.4. Proxy ARP

Ví dụ:



# 1.1. GIAO THỨC ARP

## 1.1.4. Proxy ARP

### Ưu điểm

Các router hoạt động nhưng các thiết bị không hề cảm nhận được sự hoạt động của nó. Các hoạt động gửi nhận giữa hai thiết bị thuộc hai LAN khác nhau vẫn diễn ra bình thường

### Nhược điểm

Thứ nhất, nó làm tăng độ phức tạp của mạng

Nếu nhiều hơn một router kết nối tới hai LAN cùng nằm trong một mạng IP, nhiều vấn đề có thể phát sinh.

Công nghệ này cũng tạo nên những mối nguy cơ tiềm ẩn về an ninh và bảo mật, khi các router được cấu hình proxy, tạo nguy cơ về giả mạo địa chỉ.

Giải pháp: Thiết kế lại topo mạng để chỉ một router kết nối tới hai LAN nằm trong một mạng IP.





# GIAO THỨC R-ARP

- Giao thức RARP (Reverse Address Resolution Protocol) là giao thức ngược của ARP, được sử dụng để tìm kiếm địa chỉ IP tương ứng với địa chỉ vật lý của thiết bị mạng.
- Địa chỉ IP của thiết bị được cấp bởi một máy chủ RARP trong mạng.
- Gói tin RARP có khuôn dạng hoàn toàn giống với gói tin ARP.



## 2. GIAO THỨC BOOTP

2.1. Giới thiệu

2.2. Cấu trúc gói tin BOOTP

2.3. Cách thức hoạt động BOOTP

2.4. BOOTP Relay Agents



# NỘI DUNG TRÌNH BÀY

## 1. GIAO THỨC ARP & R-ARP

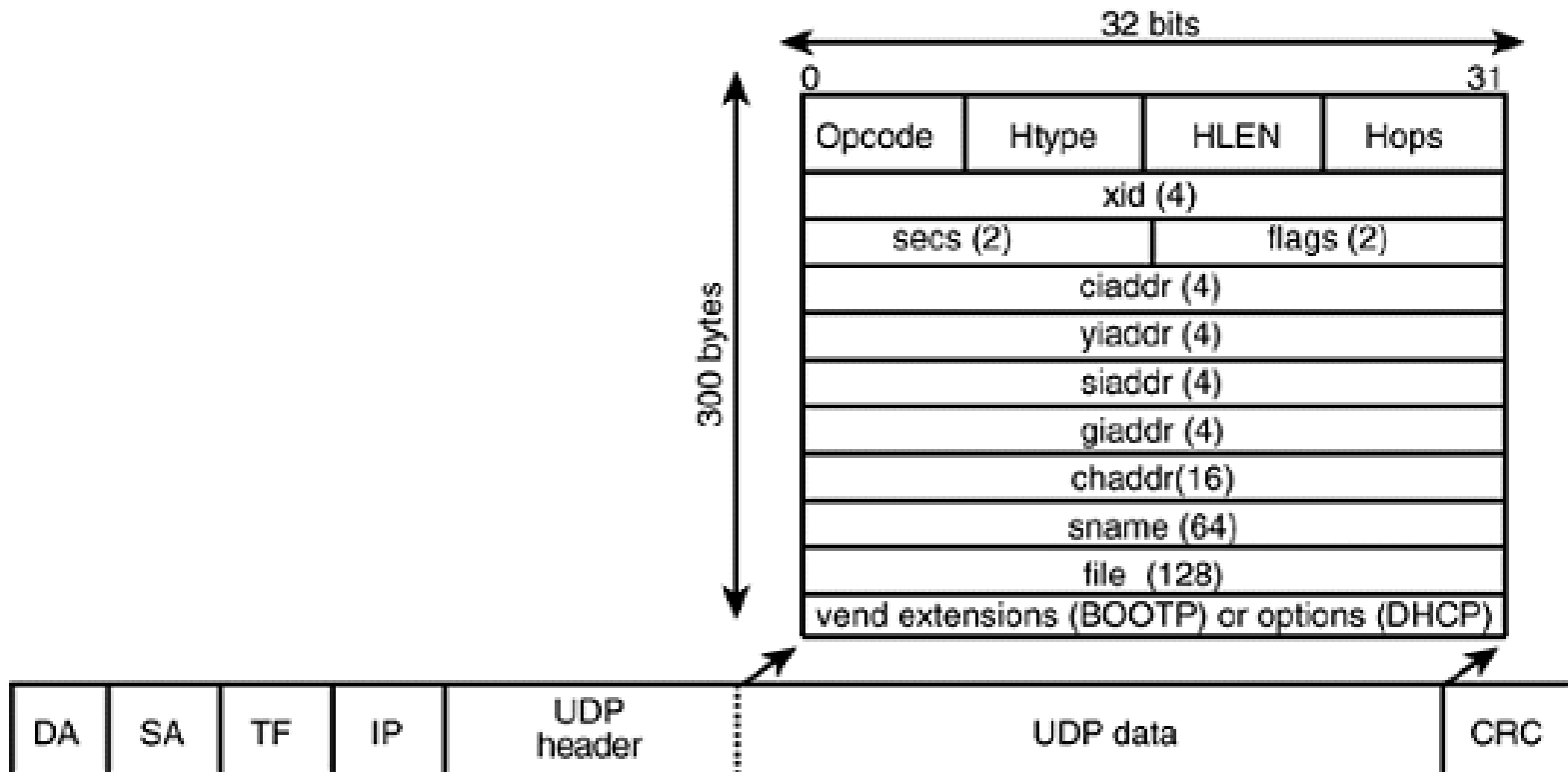


## 2.1. GIỚI THIỆU BOOTP

- BOOTP (Bootstrap Protocol): là giao thức mạng mà máy client sử dụng để lấy thông tin cần thiết từ máy server.
- Được sử dụng trong tiến trình đầu tiên của quá trình bootstrap.
- Sử dụng UDP làm giao thức vận chuyển.
- Khắc phục 2 hạn chế của RARP

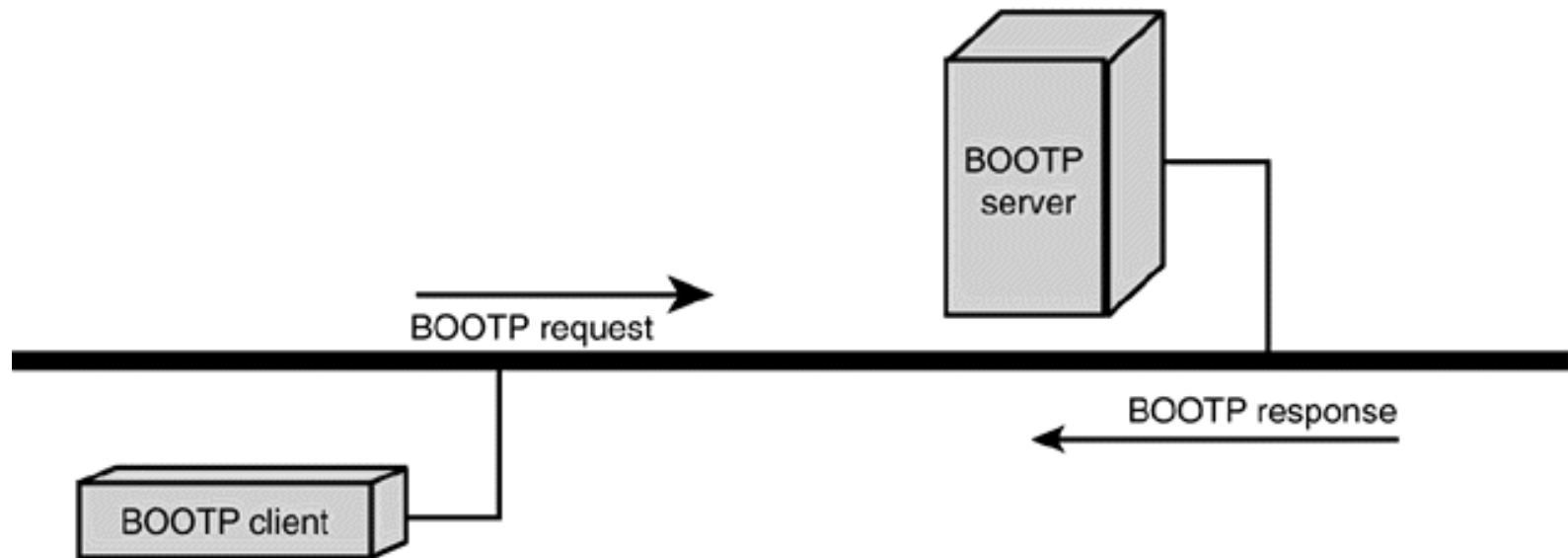


## 2.2. CẤU TRÚC GÓI TIN BOOTP

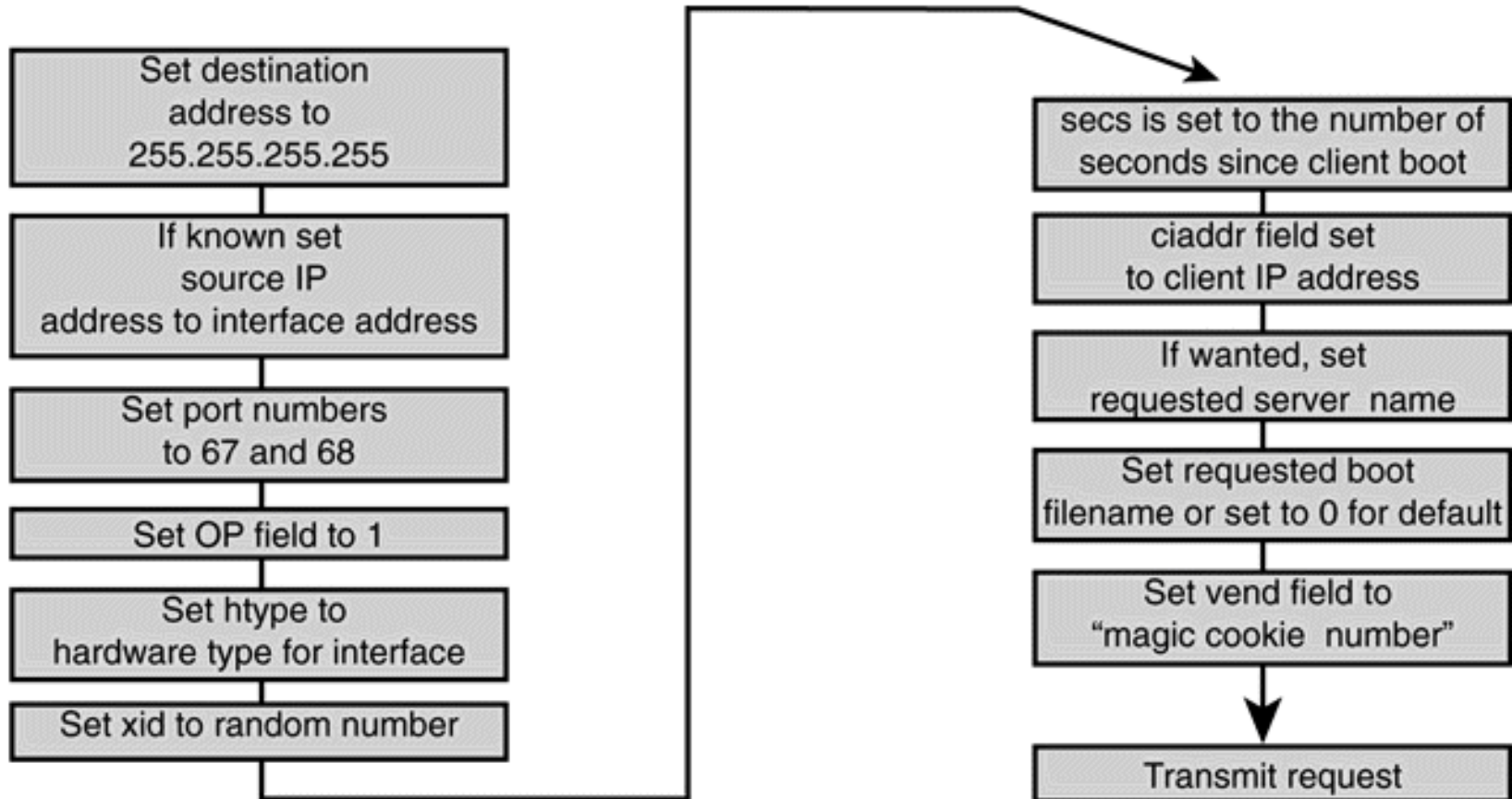


## 2.3. CÁCH THỨC HOẠT ĐỘNG BOOTP

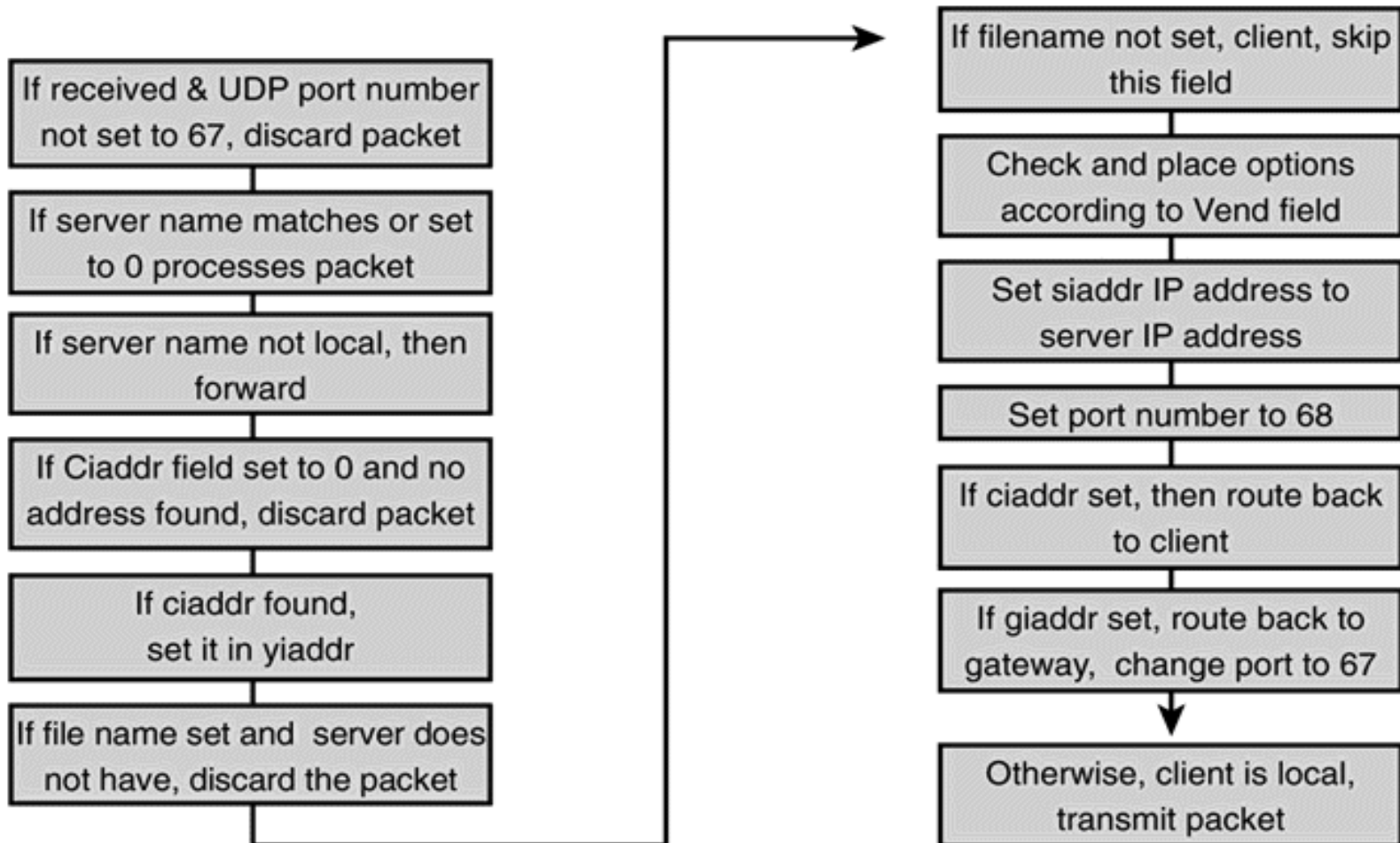
- Bao gồm 2 quá trình: BOOTREQUEST và BOOTRESPONSE.



## 2.3.1. BOOTREQUEST



## 2.3.2. BOOTRESPONSE





## 2.4. BOOTP RELAY AGENTS

- Sử dụng khi BOOTP client và server không cùng mạng (tách biệt bởi router).
- Giúp chuyển BOOTREQUEST và BOOTRESPONSE thông qua router.



# NỘI DUNG TRÌNH BÀY

**1. GIAO THỨC ARP & R-ARP**

**2. GIAO THỨC BOOTP**



# 3. GIAO THỨC DHCP

- Giới thiệu
- Cách thức hoạt động DHCP
- Các bản tin DHCP
- Cấp phát địa chỉ IP
- Trả lại địa chỉ IP



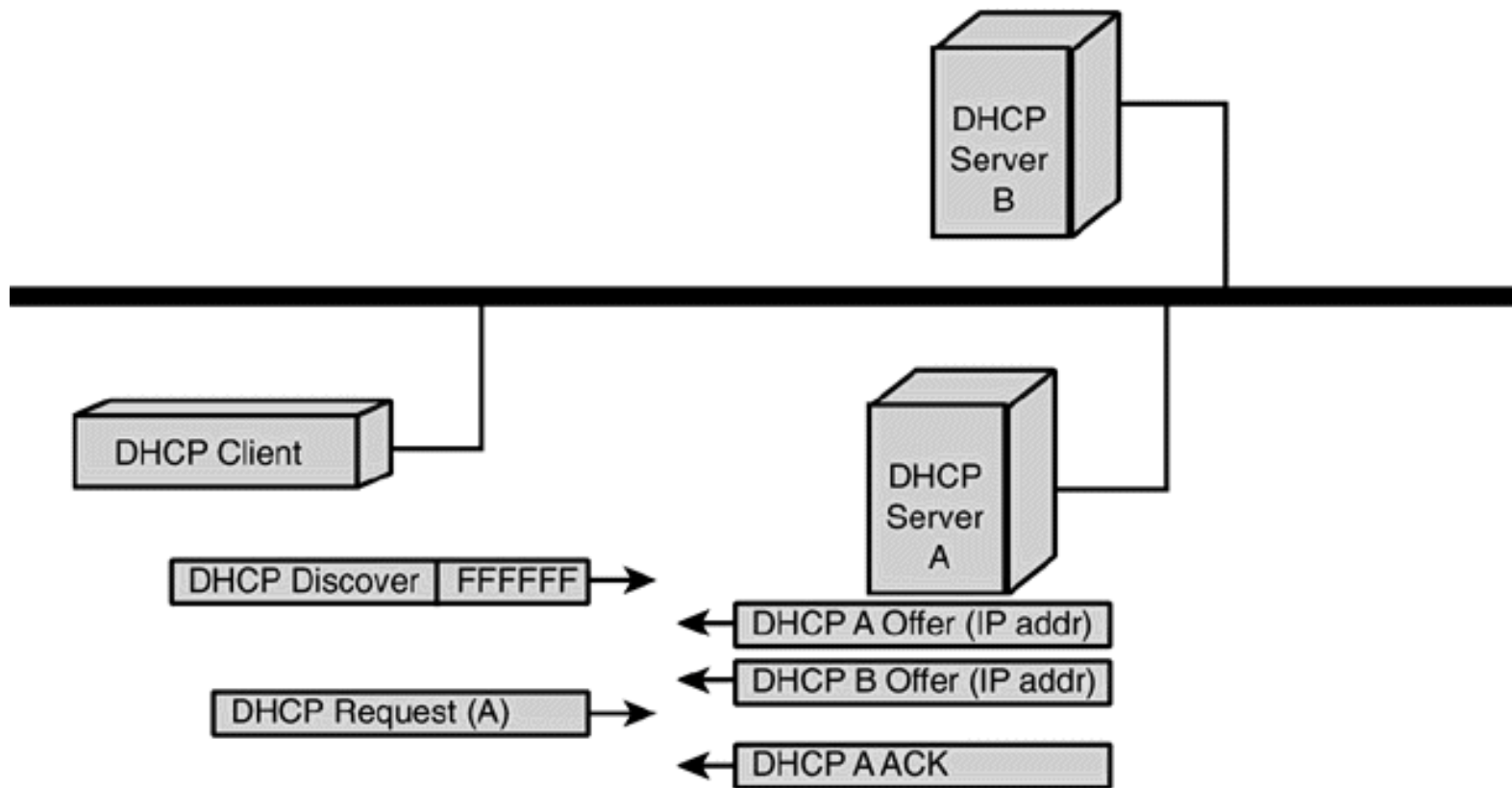
## 3.1. GIỚI THIỆU

- DHCP (Dynamic Host Configuration Protocol): cấu hình địa chỉ IP động cho máy DHCP client khi kết nối mạng.
- Cấu trúc gói tin của DHCP giống BOOTP.



## 3.2. CÁCH THỨC HOẠT ĐỘNG

- Hoạt động dựa trên mô hình client-server (quá trình tương tác giữa client và server).



## 3.3. CÁC BẢN TIN DHCP

- DHCPDISCOVER
- DDHCPREQUEST
- DHCPACK
- DHCPNAK
- DHCPRELEASE
- DHCPINFORM
- DHCPPOFFER



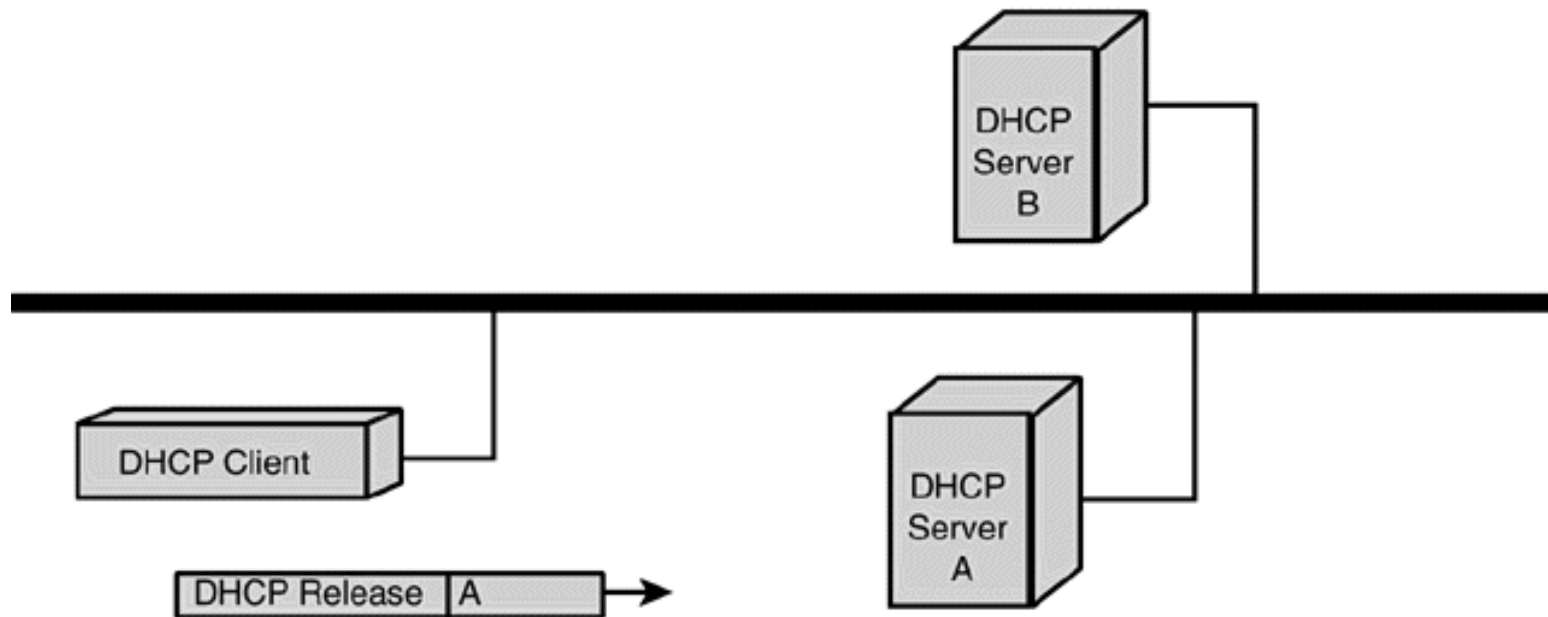
## 3.4. CẤP PHÁT ĐỊA CHỈ IP

- DHCP hỗ trợ 3 phương thức để cấp phát địa chỉ IP:
  - (1). Automatic: địa chỉ IP được gán tới máy client được sử dụng lâu dài.
  - (2). Dynamic: địa chỉ IP được gán tới máy client được sử dụng trong một khoảng thời gian xác định.
  - (3). Manual: gán địa chỉ IP cố định cho client .



## 3.5. TRẢ LẠI ĐỊA CHỈ IP

- Để dừng việc sử dụng địa chỉ IP mà client được cấp, nó sẽ gửi bản tin DHCPRELEASE cho DHCP server.





# NỘI DUNG TRÌNH BÀY

**1. GIAO THỨC ARP & R-ARP**

**2. GIAO THỨC BOOTP**

**3. GIAO THỨC DHCP**



## 4. GIAO THỨC ICMP

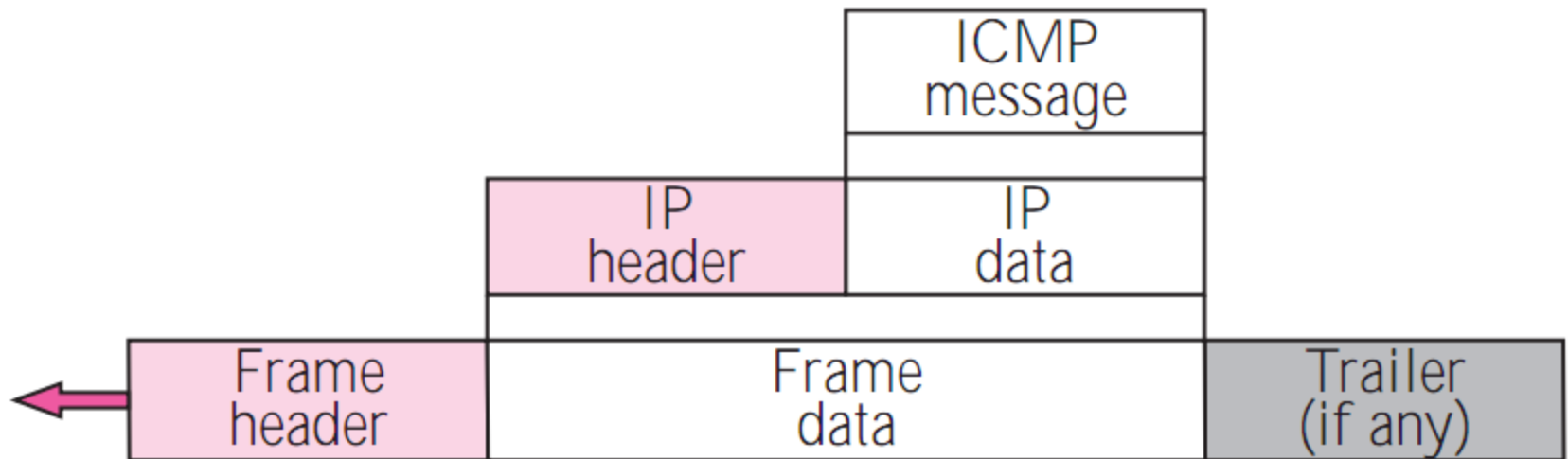
ICMP = Internet control message protocol

- Hoạt động trên layer 2 - Internetwork trong mô hình TCP/IP hoặc layer 3 - Network trong mô hình OSI.
- Cho phép kiểm tra và xác định lỗi bằng cách định nghĩa ra các loại bản tin có thể sử dụng để xác định xem mạng hiện tại có thể truyền được gói tin hay không.



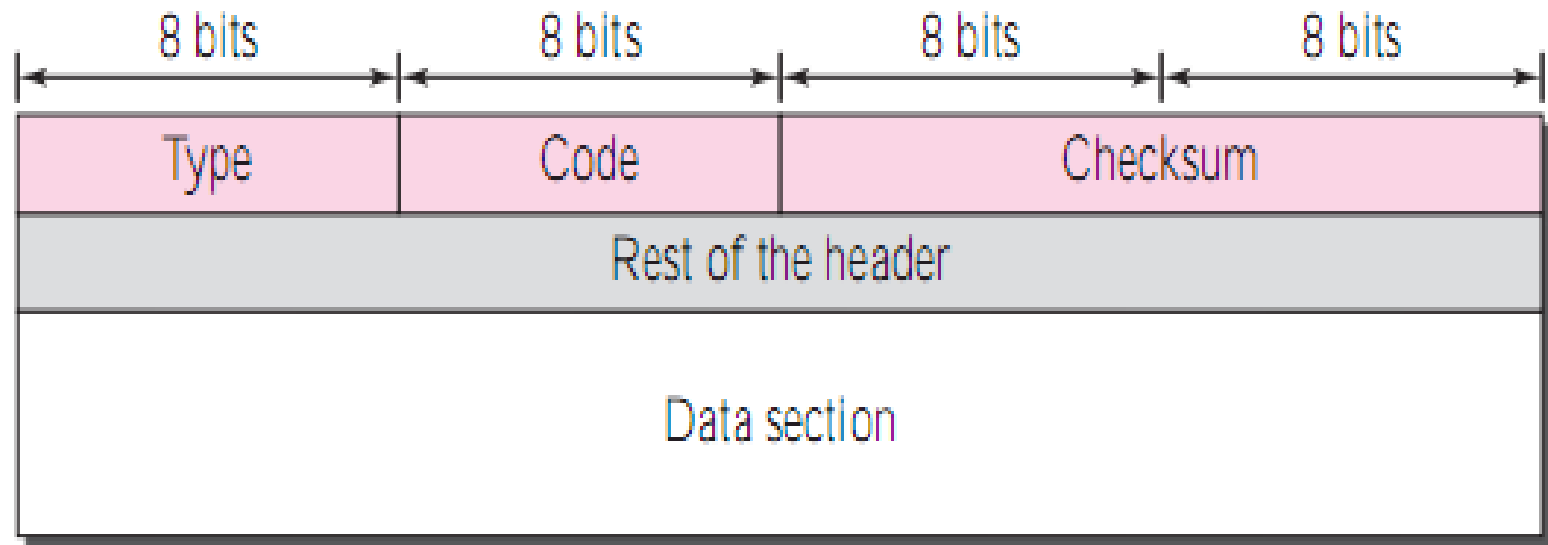
## 4. GIAO THỨC ICMP

- ICMP được đóng gói và truyền đi như thế nào trong mạng?



## 4. GIAO THỨC ICMP

- Cấu trúc thông điệp ICMP



## 4. GIAO THỨC ICMP

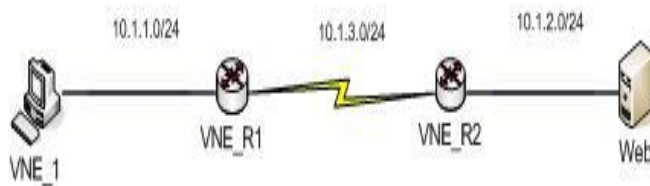
### ❖ Các thông điệp ICMP thường gặp :

Catelogy	Type	Description
Error- Reporting Messages	3	Destination unreachable
	4	Source Quench
	11	Time Exceeded
	12	Parameter Problem
	5	Redirection
Query messages	8 or 0	Echo Request or Reply
	12 or 14	Timestamp request or reply

# 4. GIAO THỨC ICMP

## ❖ Destination Unreachable

Type: 3	Code: 0 to 15	Checksum
Unused (All 0s)		
Part of the received IP datagram including IP header plus the first 8 bytes of datagram data		



Code	Description
0	Network Unreachable
1	Host Unreachable
2	Protocol Unreachable
3	Port Unreachable
...	....



## 4. GIAO THỨC ICMP

### ❖ Thông điệp Echo Request and Reply :

Type: 8 or 0	Code: 0	Checksum
Identifier		Sequence number
Optional data Sent by the request message; repeated by the reply message		



# NỘI DUNG TRÌNH BÀY

**1. GIAO THỨC ARP & R-ARP**

**2. GIAO THỨC BOOTP**

**3. GIAO THỨC DHCP**

**4. GIAO THỨC ICMP**





# 5. TÌM HIỂU VỀ WIRESHARK VÀ DEMO

## 5.1. Lịch sử Wireshark

Gerald Combs là người đầu tiên phát triển phần mềm này. Phiên bản đầu tiên được gọi là Ethereal được phát hành năm 1998. Tới năm 2006 ông và nhóm phát triển xây dựng một thương hiệu mới cho sản phẩm “Ethereal”, dự án tên là WireShark

## 5.2. Các giao thức được hỗ trợ bởi WireShark

WireShark vượt trội về khả năng hỗ trợ các giao thức (khoảng 850 loại), từ những loại phổ biến như TCP, IP đến những loại đặc biệt như là AppleTalk và Bit Torrent



# 5. TÌM HIỂU VỀ WIRESHARK VÀ DEMO

## 5.3. Điểm mạnh của wireshark

- Thân thiện với người dùng: Giao diện của Wireshark là một trong những giao diện phần mềm phân tích gói dễ dùng nhất. Wireshark là ứng dụng đồ họa với hệ thống menu rất rõ ràng và được bố trí dễ hiểu.
- Giá rẻ: Wireshark là một sản phẩm miễn phí GPL. Bạn có thể tải về và sử dụng Wireshark cho bất kỳ mục đích nào, kể cả với mục đích thương mại.
- Hỗ trợ: Cộng đồng của Wireshark là một trong những cộng đồng tốt và năng động nhất của các dự án mã nguồn mở.
- Hệ điều hành hỗ trợ Wireshark: Wireshark hỗ trợ hầu hết các loại hệ điều hành hiện nay.



**DEMO BẮT GÓI TIN CỦA CÁC  
GIAO THỨC ICMP, ARP, R-ARP, DHCP  
BẰNG PHẦN MỀM WIRESHARK**





# TỔNG QUAN GIAO THỨC ĐỊNH TUYẾN BGP TRÊN ROUTER CISCO

---

Sinh viên thực hiện : Lê Vũ Thắng

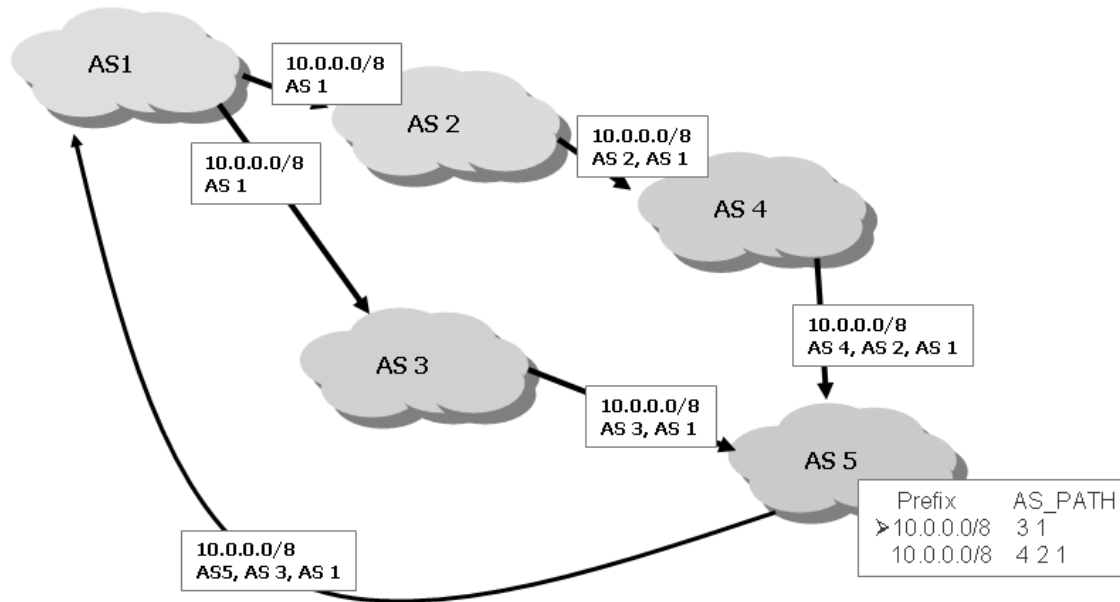
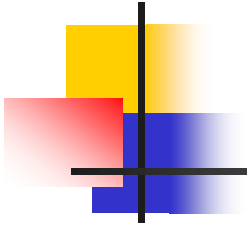
Giáo viên hướng dẫn: Th.s Trần Quang Đạt



# 1. Tổng quan về giao thức BGP

---

- BGP(Border Gateway Protocol) là giao thức định tuyến nòng cốt trên Internet.
- BGP hoạt động dựa trên việc cập nhật một bảng chứa các địa chỉ mạng và cho biết mối liên kết giữa các hệ tự trị
- BGP là giao thức path vector quá trình định tuyến của BGP dựa trên các thuộc tính (Attribute)





## 2. AS và các thuộc tính của BGP

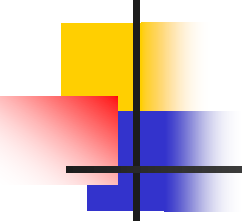
---

### 2.1 Hệ tự trị (Autonomous System)

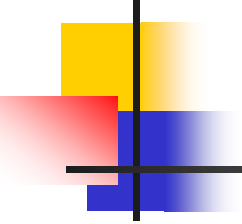
Là một tập hợp các mạng có cùng chính sách định tuyến và thường thuộc quyền quản lý, khai thác của một chủ thể.

### 2.2 Các thuộc tính của BGP

Các thuộc tính của BGP được chia làm 4 nhóm:

- 
- 
- Nhóm well-known mandatory: Nhóm này là các thuộc tính cần được biết bởi tất cả các router và quảng bá tới tất cả các router
  - Nhóm well-known discretionary: Nhóm này là các thuộc tính không bắt buộc các router cần biết tuy nhiên nếu biết các router sẽ có xử lý thích hợp



- 
- 
- Nhóm optional transitive : Các thuộc tính của router nếu thuộc nhóm này sẽ được chuyển tiếp nếu router không hiểu nó
  - Nhóm optional nontransitive: Các thuộc tính này sẽ không được chuyển tiếp nếu router không hiểu nó



---

## 2.2.1 Thuộc tính WEIGHT

- Đây là trọng số sử dụng để định tuyến. Nó có tác dụng cục bộ trên mỗi router. Router có trọng số weight lớn sẽ được ưu tiên hơn trong việc lựa chọn đường đi
- Thuộc tính này do cisco định nghĩa. Trên các router giá trị này mặc định là 32768

Figure 8-6. WEIGHT Attribute

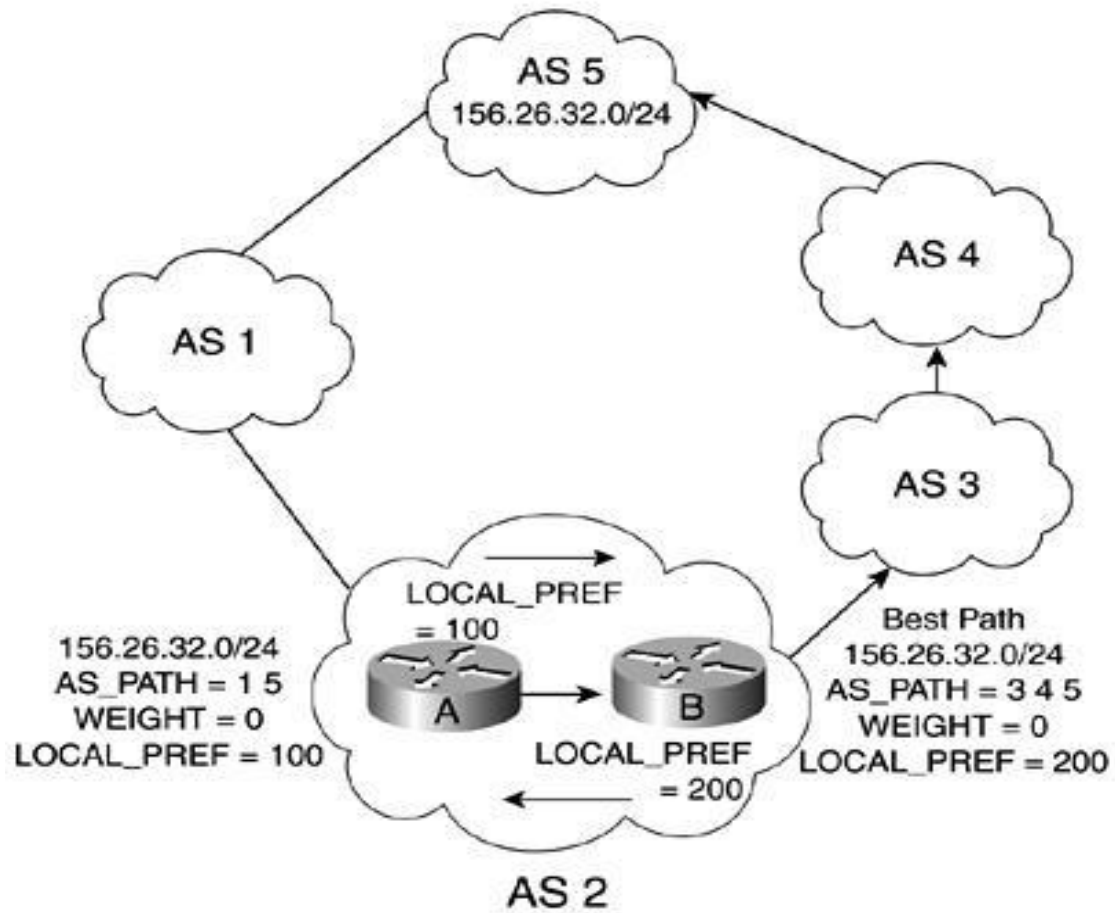
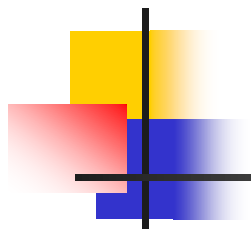




---

## 2.2.2 Thuộc tính Local Preference (LOCAL\_PREF)

- Thuộc tính này thuộc nhóm well-known discretionary.
- LOCAL\_PREF là một số 32 bit, được sử dụng trong một AS và có giá trị mặc định trên các router là 100
- Các giá trị lớn của LOCAL\_PREF được ưu tiên sử dụng để định tuyến

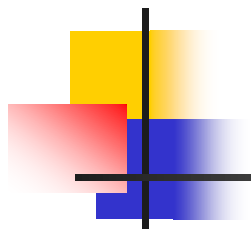




---

- 2.2.3 Thuộc tính AS\_PATH

- Thuộc tính này thuộc nhóm well-known mandatory và được sử dụng cho việc chọn tuyến và tương tự như việc đếm số hop trong rip
- AS\_PATH là một số 16 bit chia làm 2 loại: AS public và AS private
- AS\_PATH còn được sử dụng để chống loop trong BGP



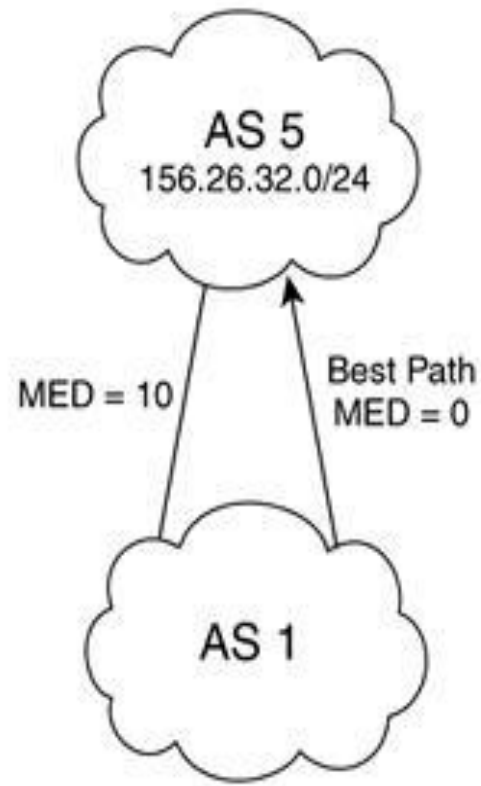
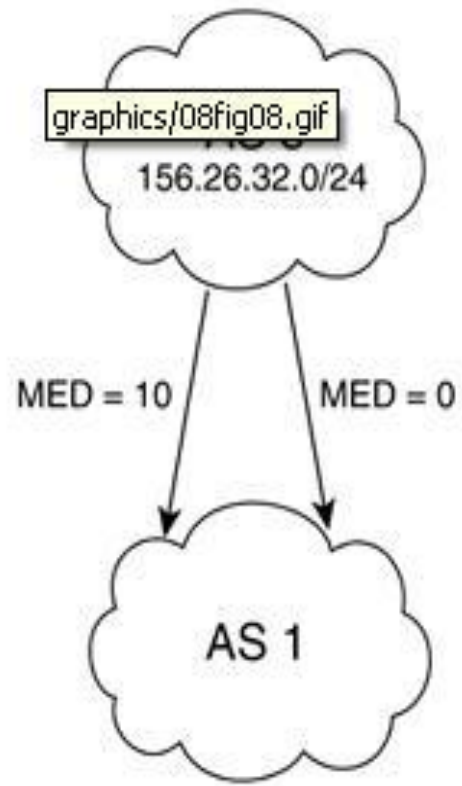
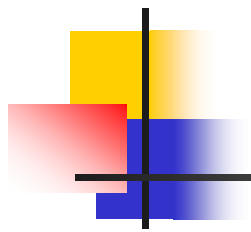


---

## ■ 2.2.4 Thuộc tính MED (Metric)

- Thuộc tính này thuộc nhóm optional nontransitive và được sử dụng để chọn tuyến giữa các AS khác nhau
- Khác với các thuộc tính trên MED ưu tiên các giá trị nhỏ hơn khi nó được sử dụng để định tuyến
- Độ ưu tiên định tuyến của thuộc tính này đứng sau các thuộc tính Weight, Local\_Pref, AS\_Path.







---

- 2.2.5 Thuộc tính ORIGIN

- Nằm trong nhóm well-known mandatory :  
Thuộc tính này cho phép xác định các tuyến được học như thế nào:

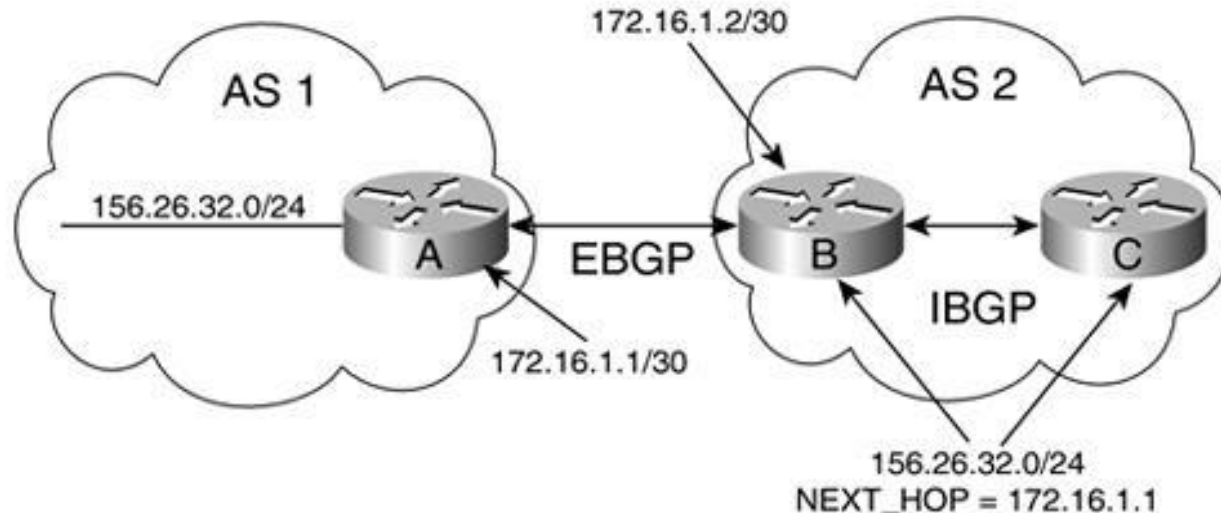
- + IGP: Từ câu lệnh network

- + EGP: Từ một EBGP hàng xóm

- + Incomplete : Tuyến được học thông qua redistribution

## ■ 2.2.6 Thuộc tính NEXT\_HOP

Là tiêu chí đầu tiên để kiểm tra điều kiện định tuyến. Nếu NEXT\_HOP không đạt được trên một router. Quá trình định tuyến sẽ chấm dứt





# 3. Thiết lập mối quan hệ giữa các router trong BGP

---

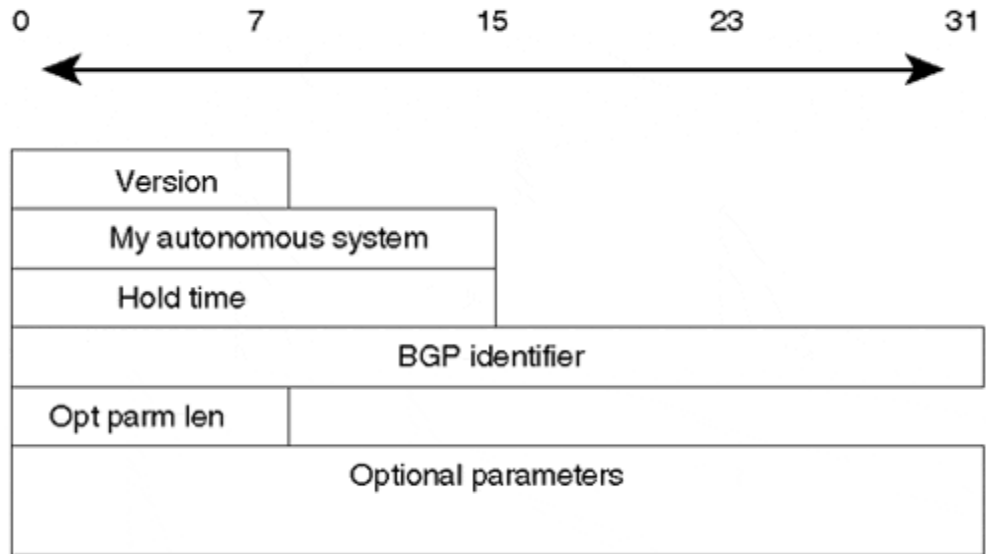
- Khi sử dụng BGP cho việc định tuyến cần chỉ rõ hàng xóm cho nó. Quá trình thiết lập mối quan hệ với hàng xóm và trao đổi thông tin định tuyến được thực hiện qua 6 giai đoạn:

- 
- 
- Idle State
  - Connect State
  - Active State
  - OpenSent State
  - OpenConfirm State
  - Entablished

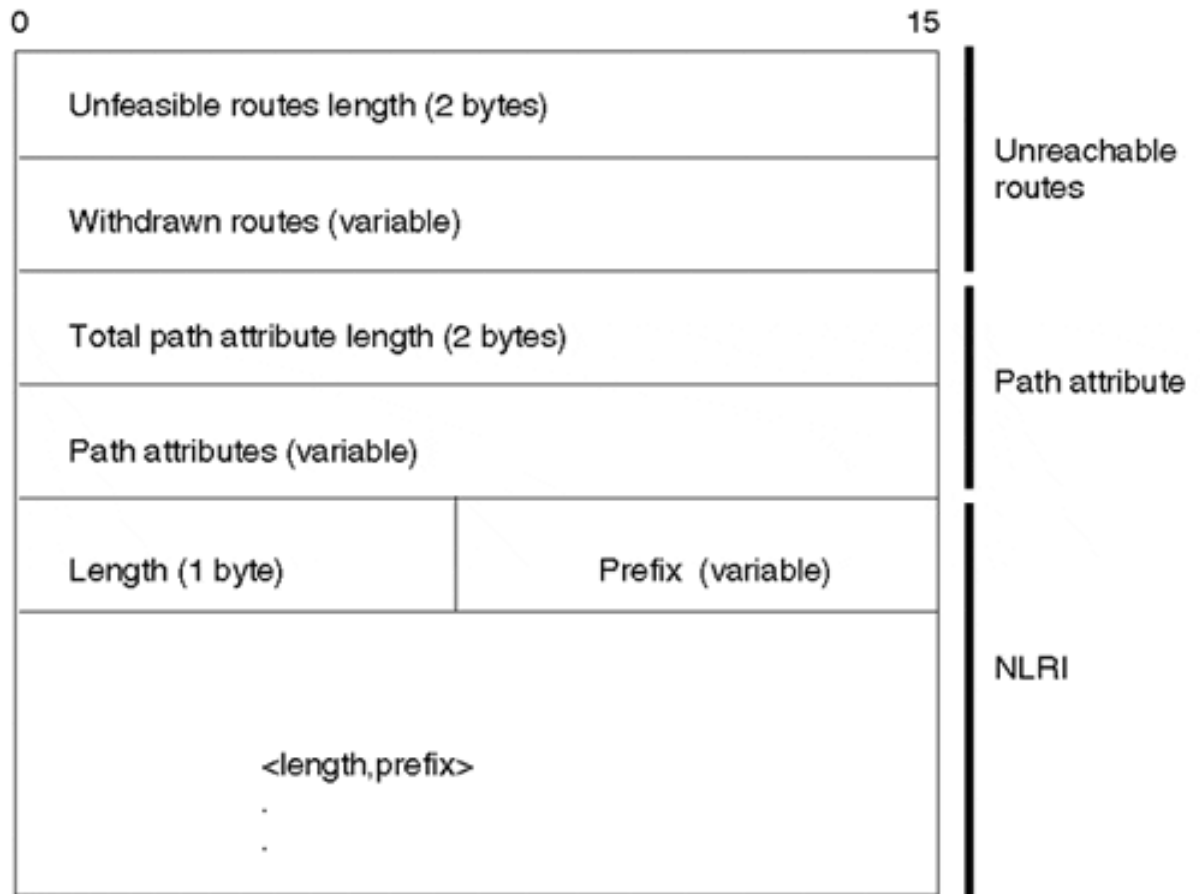
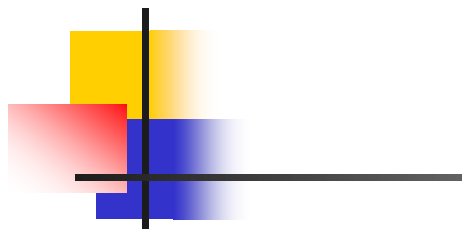


---

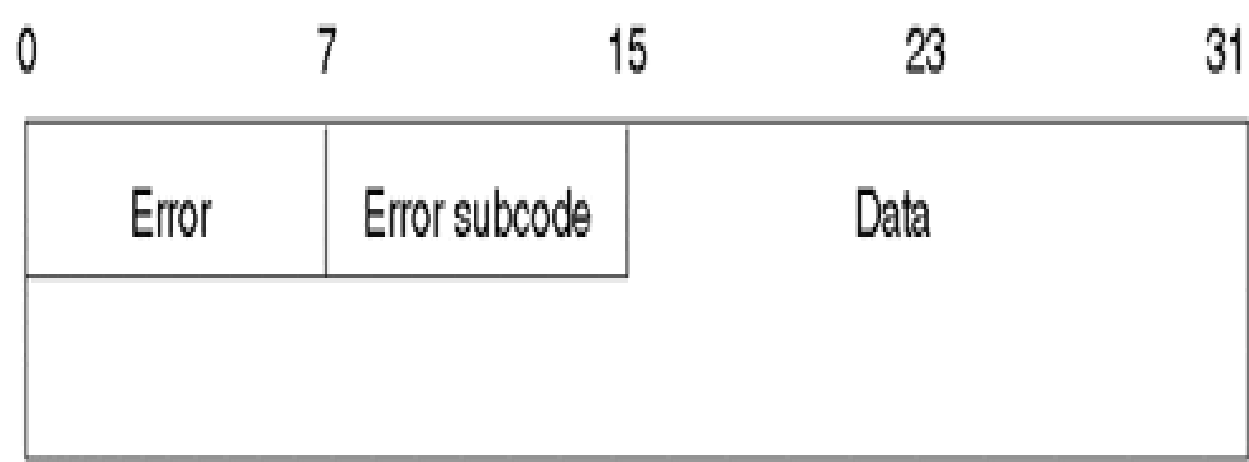
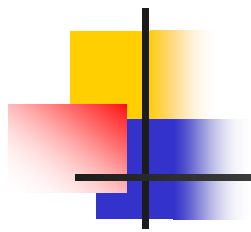
- Cấu trúc các thông báo trong BGP



- Thông báo OPEN



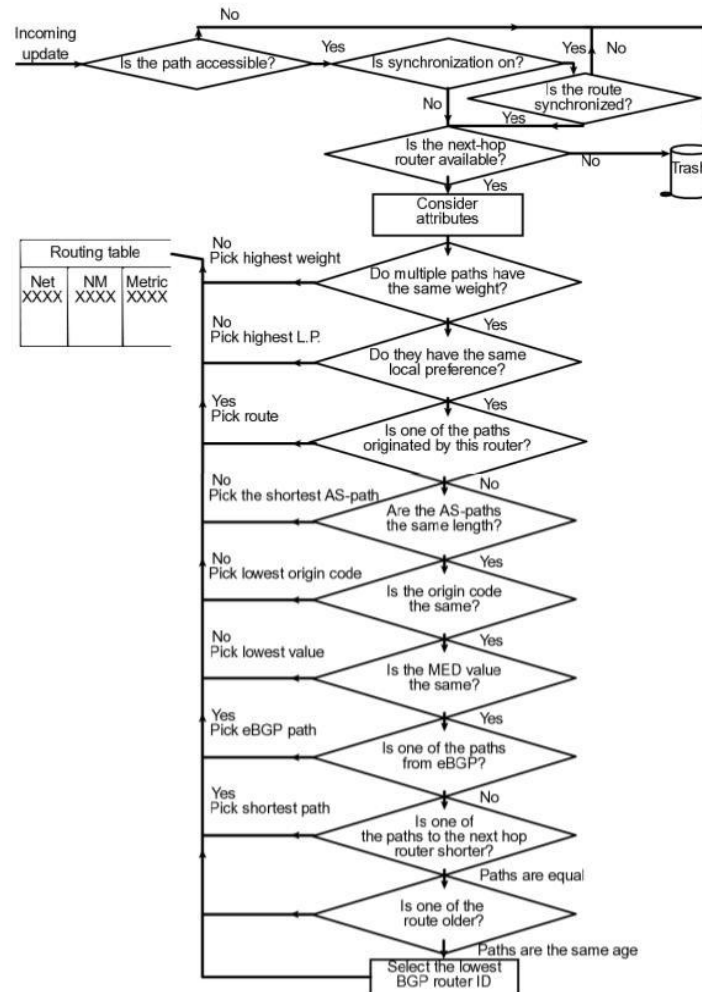
# Thông báo UPDATE

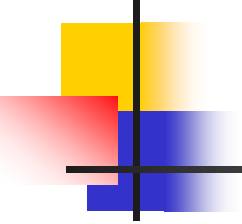


- Thông báo Notification



# 4. Quá trình định tuyến trong BGP

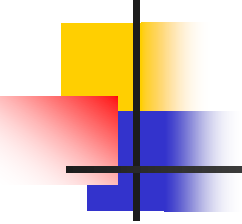


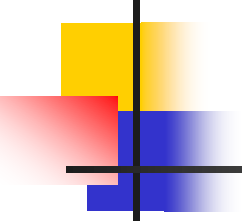


---

Quá trình định tuyến được thực hiện theo thứ tự các bước sau:

- Ưu tiên tuyến có thuộc tính WEIGHT cao nhất (Chỉ có tác dụng trên router)
- Ưu tiên tuyến có thuộc tính LOCAL\_PREF cao nhất (Thuộc tính này có tác dụng trong một AS). Những tuyến không có thuộc tính LOCAL\_PREF sẽ có giá trị mặc định là 100.
- Ưu tiên các tuyến gốc (có NEXT\_HOP là 0.0.0.0)

- 
- 
- - Ưu tiên tuyến có thuộc tính AS\_PATH thấp nhất
  - - Ưu tiên các tuyến có nguồn gốc thấp nhất theo thứ tự ưu tiên IGP, EGP và INCOMPLETE
  - - Ưu tiên tuyến có thuộc tính MED (metric) thấp nhất (thuộc tính này có giá trị giữa các AS khác nhau )

- 
- 
- - Ưu tiên các tuyến EBGP hơn so với IBGP
  - - Ưu tiên những tuyến có IGP metric thấp nhất đến BGP NEXT\_HOP
  - - Ưu tiên những tuyến từ BGP router có RouterID nhỏ nhất

## 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)**

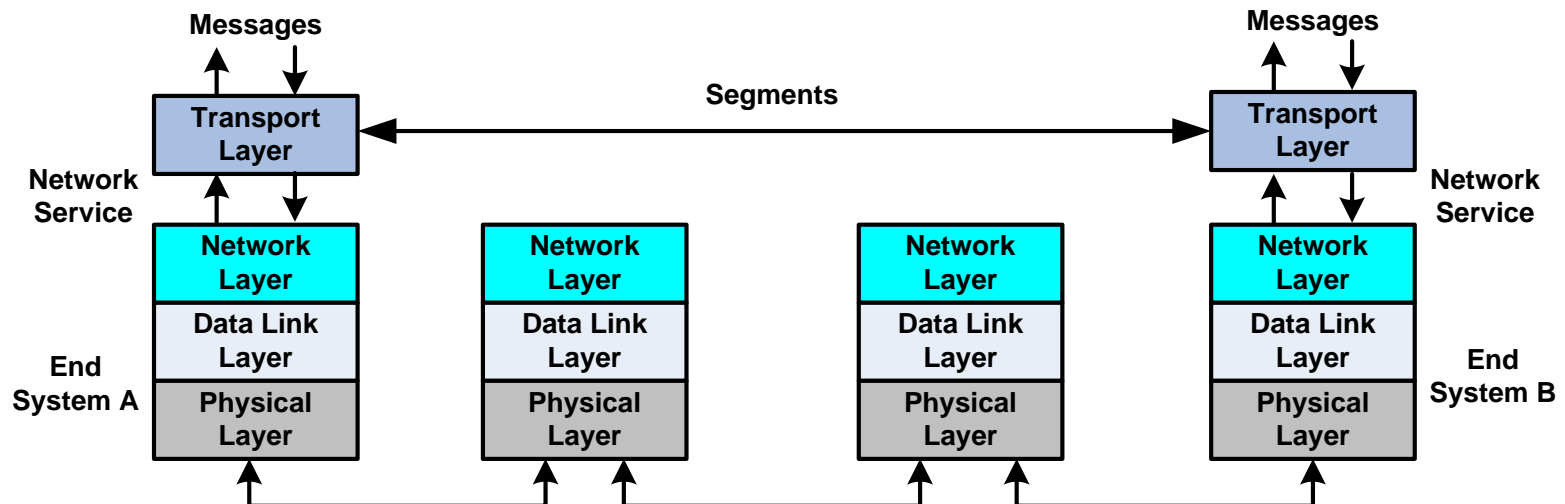
- ✓ **Vấn đề của Network layer**

- 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

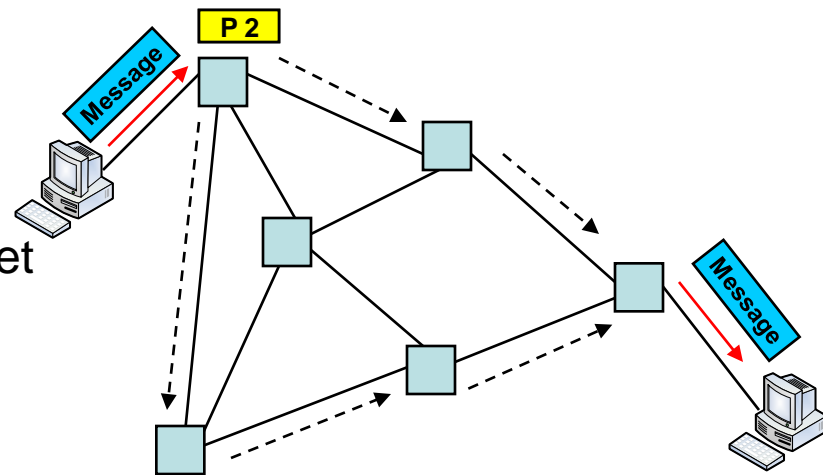
- Truyền thông tin qua các packet (gói tin)
- Khả năng có trễ ngẫu nhiên và mất packet
- Mỗi ứng dụng có yêu cầu truyền tin khác nhau

## ✓ 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



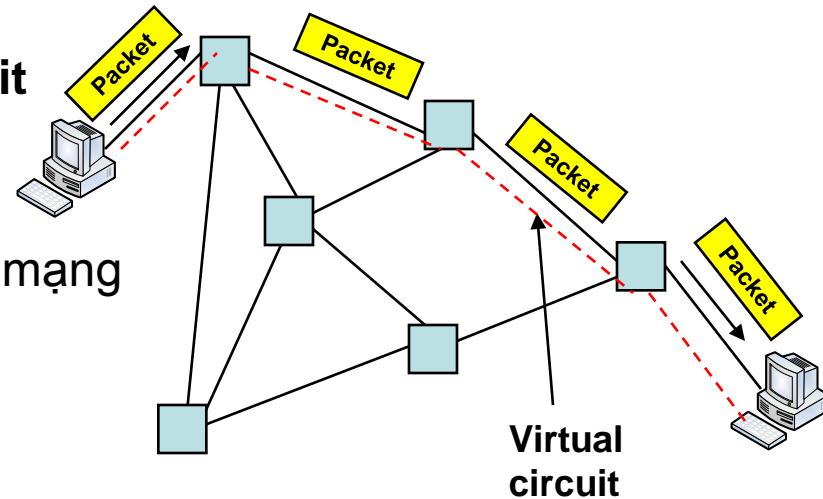


### ✓ Routing table trong mạng chuyển mạch gói

- 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

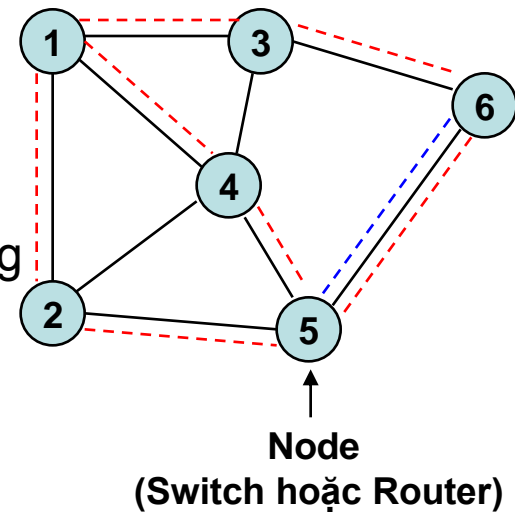
### ✓ Packet switching – Virtual circuit

- 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

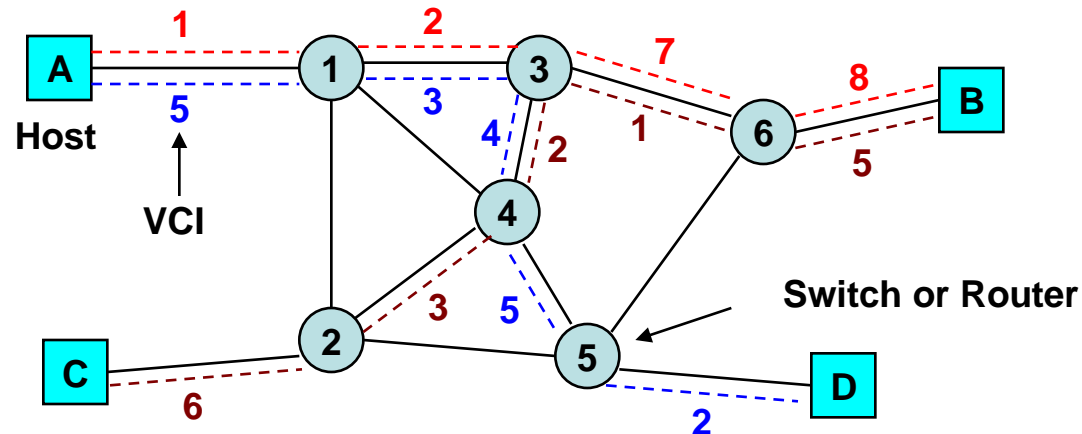


- ✓ **Thiết lập liên kết**
  - o Thông tin báo hiệu (signaling message) xác định liên kết và các bảng khởi tạo (setup table) trong các chuyển mạch
  - o Các liên kết được xác định nhờ **virtual circuit identifier (VCI)**
  - o Khi setup table được thiết lập, packet được truyền trên đường dẫn
- ✓ **Virtual 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...

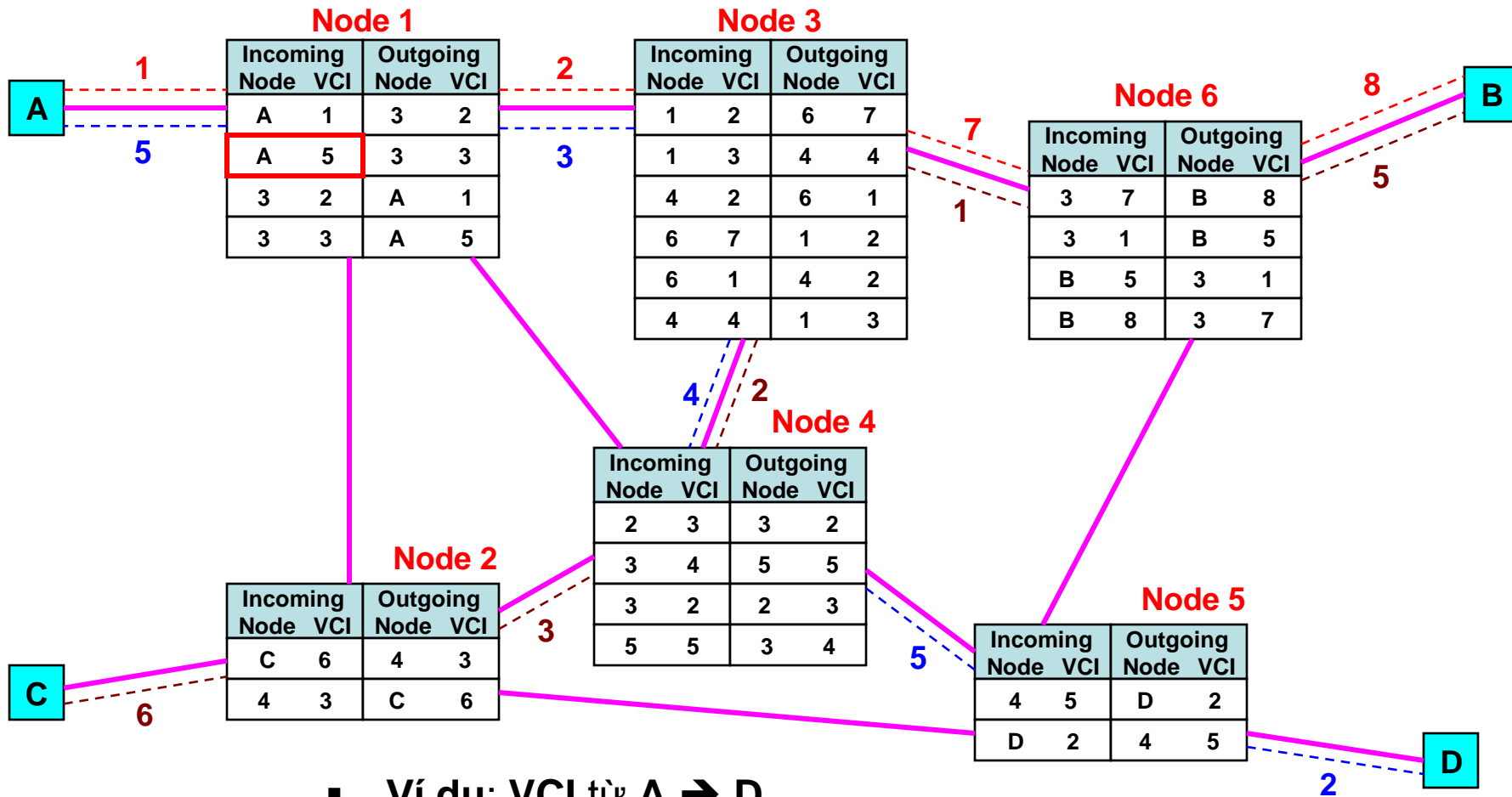
- ✓ **Định tuyến trong mạng chuyển mạch gói**
  - Có thể có 3 tuyến từ node 1 tới node 6: 1-3-6, 1-4-5-6, 1-2-5-6
  - Tuyến nào tối ưu nhất? : Min delay, min hop, max BW, min cost
  - Thuật toán định tuyến
    - Truyền nhanh và chính xác
    - Thích ứng với thay đổi của cấu hình mạng (link & node failure)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



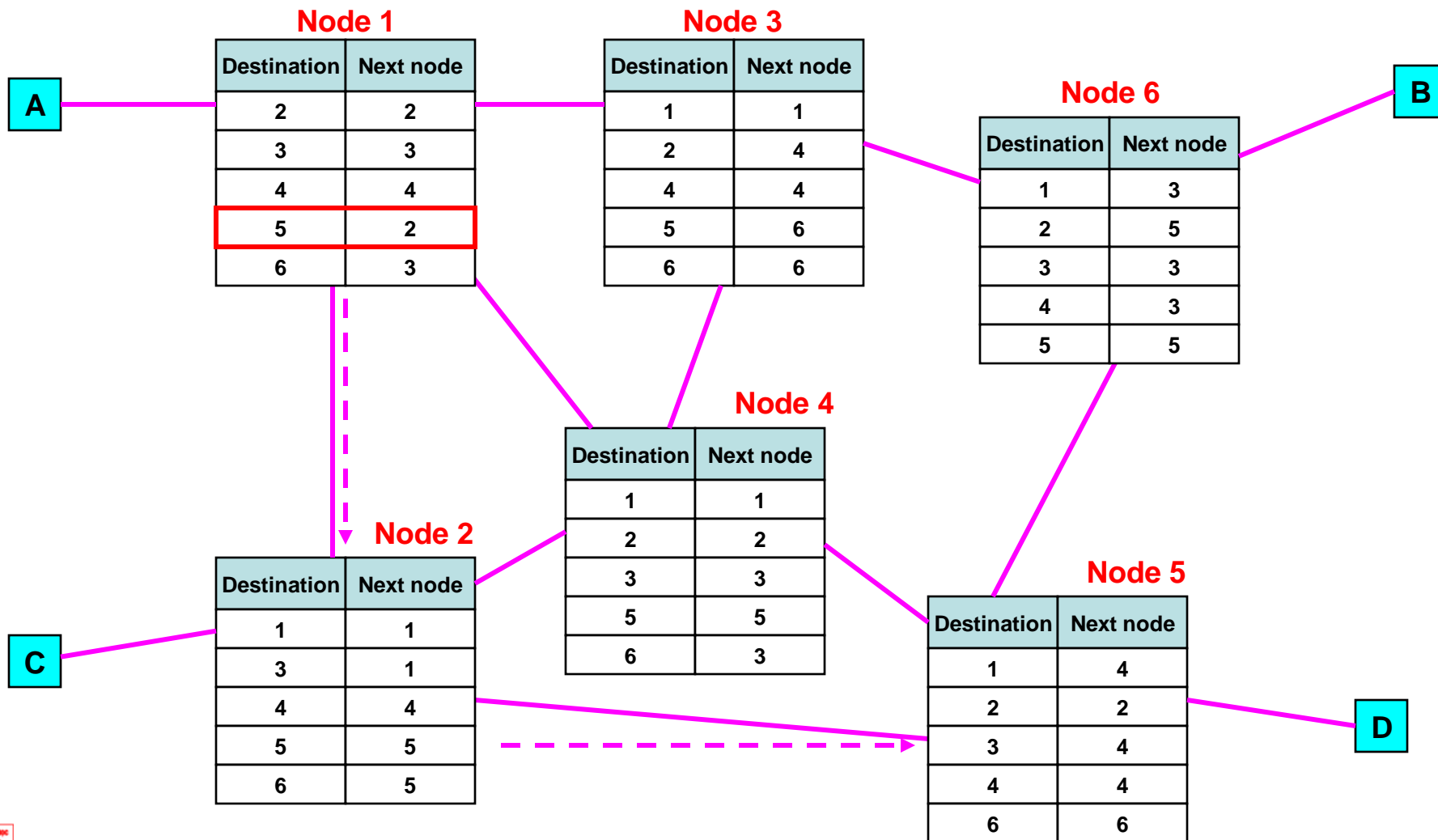
## o RT trong VC packet network



▪ 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

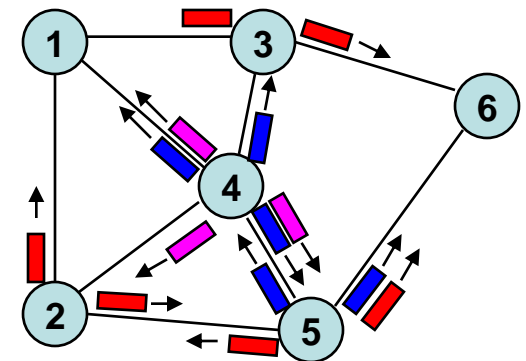
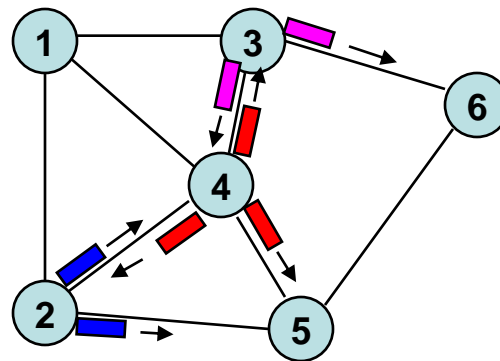
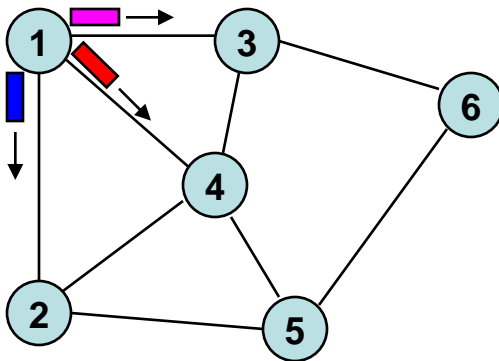


- Định tuyến (routing) trong mạng chuyển mạch gói

- ✓ Định tuyến đặc biệt: flooding và deflection

- o Flooding

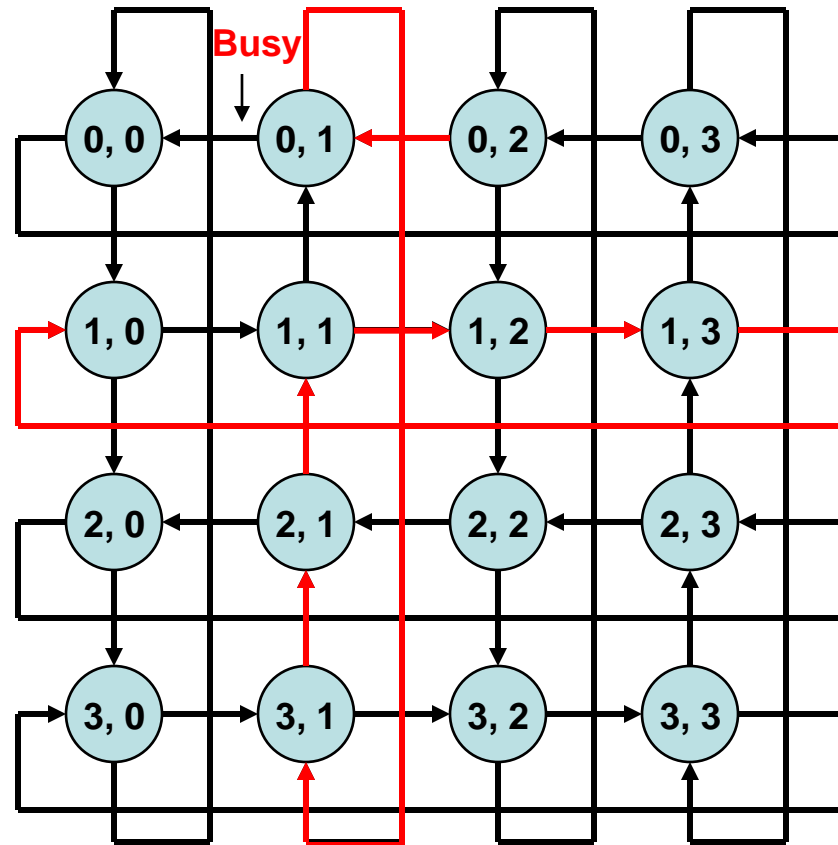
- Gửi gói tin tới tất cả các node trong mạng: Không cần bảng định tuyến, sử dụng kiểu quảng bá để gửi các packet tới các nút mạng
- Limited-flooding:
  - ❖ Time-to-live cho mỗi gói tin: giới hạn số chặng chuyển tiếp
  - ❖ Trạm nguồn điền số thứ tự cho mỗi packet



o 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

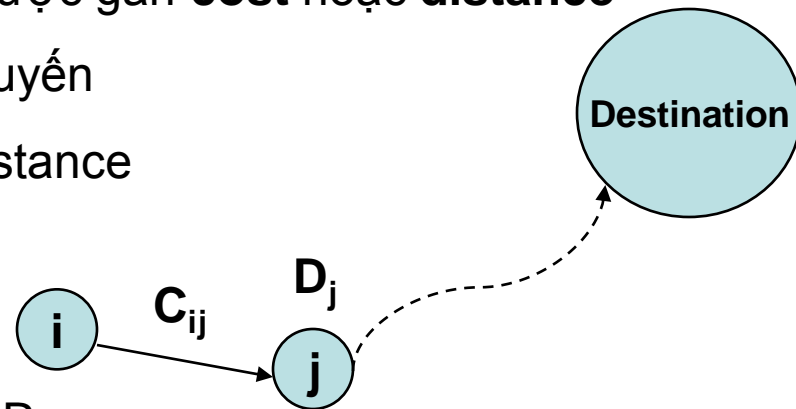
### ✓ Shortest path & routing

- 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**  $\rightarrow D_i = C_{ij} + D_j$

✓ 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:  $D_d = 0$
- Khoảng cách từ node **i** bất kỳ tới **d**:  $D_i = \infty, i \neq d$
- Node tiếp theo chưa được xác định:  $n_i = -1, i \neq d$

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**
  - ❖  $D_i(d) = \text{Min}_j\{C_{ij} + D_j\}, i \neq j$
  - ❖ Thay cặp giá trị cũ  $(n_i, D_i(d))$  bằng giá trị mới  $(n_i^*, D_j^*(d))$  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	$(-1, \infty)$	$(-1, \infty)$	$(-1, \infty)$	$(-1, \infty)$	$(-1, \infty)$

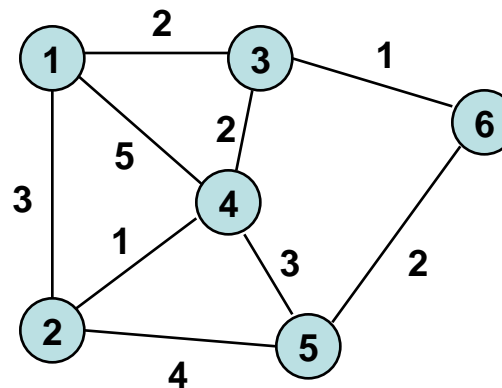
### Node 2 → Node 6

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

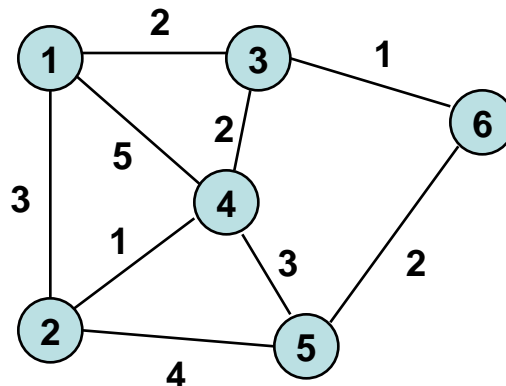
Đường nào ngắn nhất?

### $(n, D_i)$


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

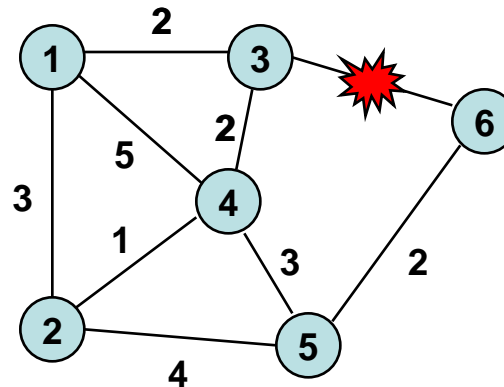


Iteration	Node 1	Node 2	Node 3	Node 4	Node 5
Initial	$(-1, \infty)$	$(-1, \infty)$	$(-1, \infty)$	$(-1, \infty)$	$(-1, \infty)$
1	$(-1, \infty)$	$(-1, \infty)$	$(6, 1)$	$(-1, \infty)$	$(6, 2)$
2	$(3, 3)$	$(5, 6)$	$(6, 1)$	$(3, 3)$	$(6, 2)$
3	$(3, 3)$	$(4, 4)$	$(6, 1)$	$(3, 3)$	$(6, 2)$
4	$(3, 3)$	$(4, 4)$	$(6, 1)$	$(3, 3)$	$(6, 2)$



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:

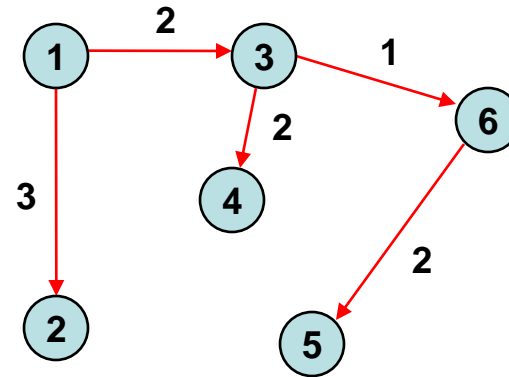
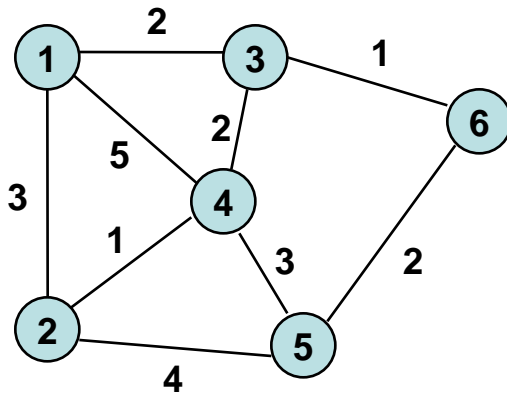
- **ID** của node liền kề:  $N_i =$  tập hợp của các node liền kề node  $i$
      - Khoảng cách tới node liền kề của nó  $\{C_{ij} \mid j \in N_i\}$

- ✓ **Dijkstra algorithm:** tìm **SP** theo thứ tự
  - **N:** tập hợp các node đã tìm thấy **SP**
  - **Initialization** (*Bắt đầu với node nguồn s*)
    - **$N = \{s\}$ ,  $D_s = 0$ :** Khoảng cách từ node **s** tới chính nó bằng 0
    - **$D_j = C_{sj}$ ,  $j \neq s$ :** Khoảng cách tới node liền kề kết nối trực tiếp
  - **Step A** (*Tìm node i gần nhất*)
    - Tìm node **i**  $\in N$  sao cho  **$D_i = \min D_j$**  với  **$j \in N$**
    - Cập nhật node **i** vào tập hợp **N**
    - Nếu **N** chứa tất cả các node, **STOP**
  - **Step B** (*cập nhật minimum cost*)
    - Với mỗi node **j**  $\in N$ , 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	$\infty$	$\infty$
1	{1, 3}	3	2	4	$\infty$	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  $\infty$  và gửi thông báo cập nhật sử dụng phương pháp *flooding*
- Tất cả các router sẽ tính toán và cập nhật **SP**

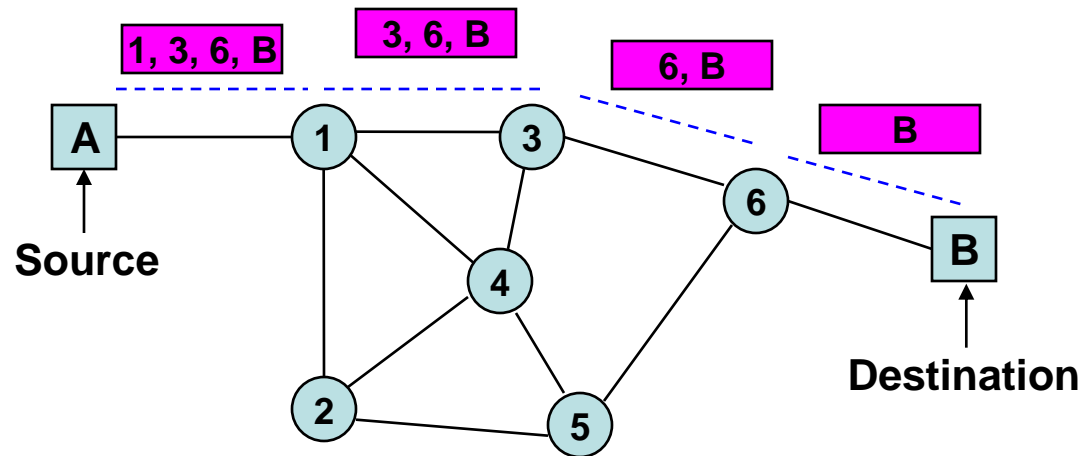
## o Vấn đề thông báo cập nhật link cost

- 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	

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 trạm 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 là 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

7

24



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

### Class B

14

16



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

### Class C

21

8



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

### Class D

28



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

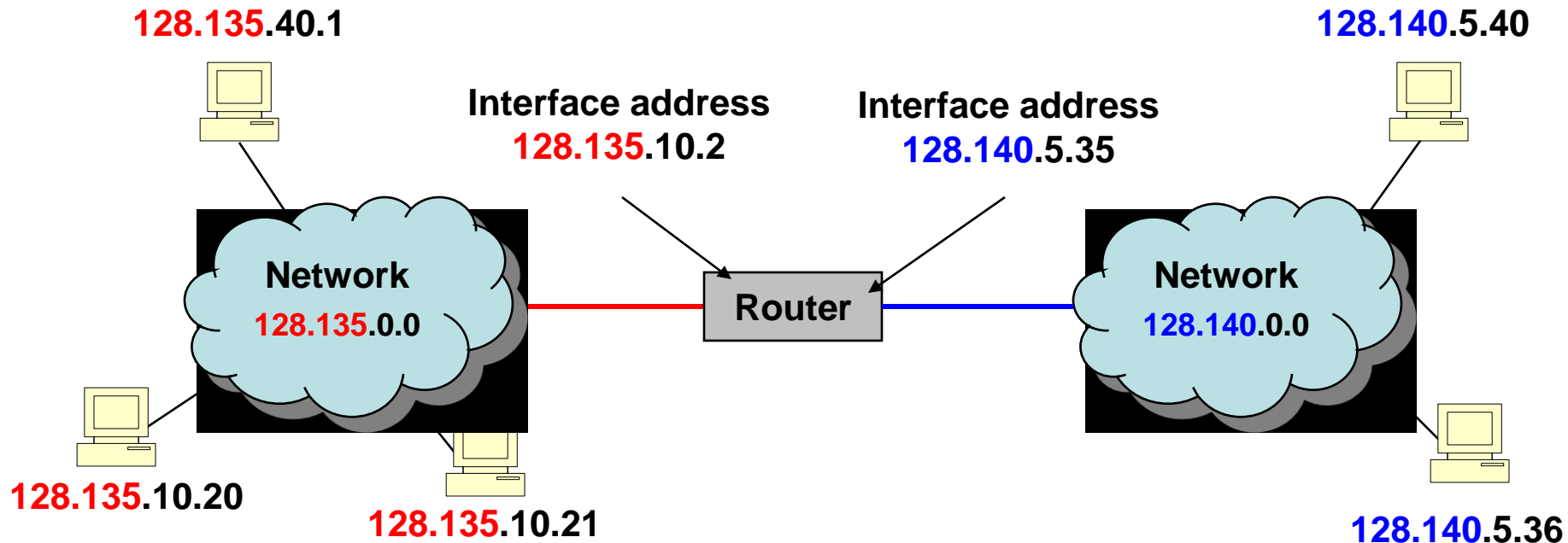
✓ Một số địa chỉ IP đặc biệt

0 0 ... 0 0	0 0 ... 0 0	This host (used in booting up)
0 0 ... 0 0	Host	A host in this network
1 1 ... 1 1	1 1 ... 1 1	Broadcast on a local network
NetID	1 1 ... 1 1	Broadcast on a distant network

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

- Router trong mạng chung từ chối packet với các địa chỉ IP này
- **Range 1: 10.0.0.0 – 10.255.255.255**
- **Range 2: 172.16.0.0 – 172.31. 255.255**
- **Range 3: 192.168.0.0 – 192.168.255.255 (Home LAN)**
- **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**)

Địa chỉ IP (lớp B)

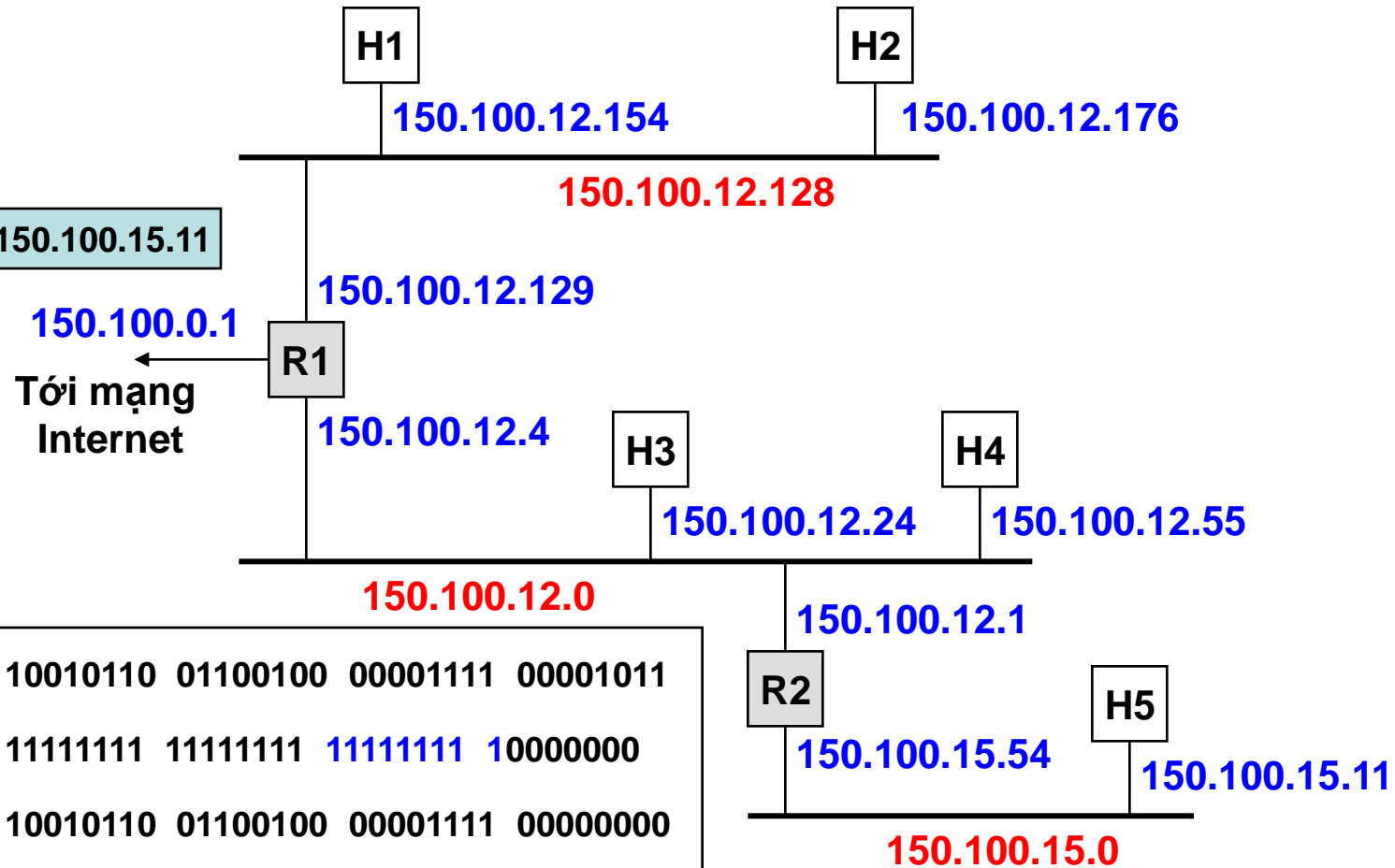


Địa chỉ **subnet**



- ✓ **Ví dụ:** 1 tổ chức có địa chỉ IP lớp B với netID: **150.100.0.0** (16 bit hostID)
  - 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
  - Áp dụng **subnet mask** cho địa chỉ IP để tìm mạng con tương ứng
  - Ví dụ: Tìm **subnet** cho địa chỉ IP **150.100.12.176**
  - Địa chỉ IP:       **10010110 01100100 00001100 10110000**
  - Mask:             **11111111 11111111 11111111 10000000**
  - AND:              **10010110 01100100 00001100 10000000**
  - Subnet:           **150.100.12.128**
  - Broadcast subnet: **150.100.12.255**
  - Các host kết nối vào subnet: **150.100.12.129 – 150.100.12.254**
  - Các router chỉ sử dụng địa 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**

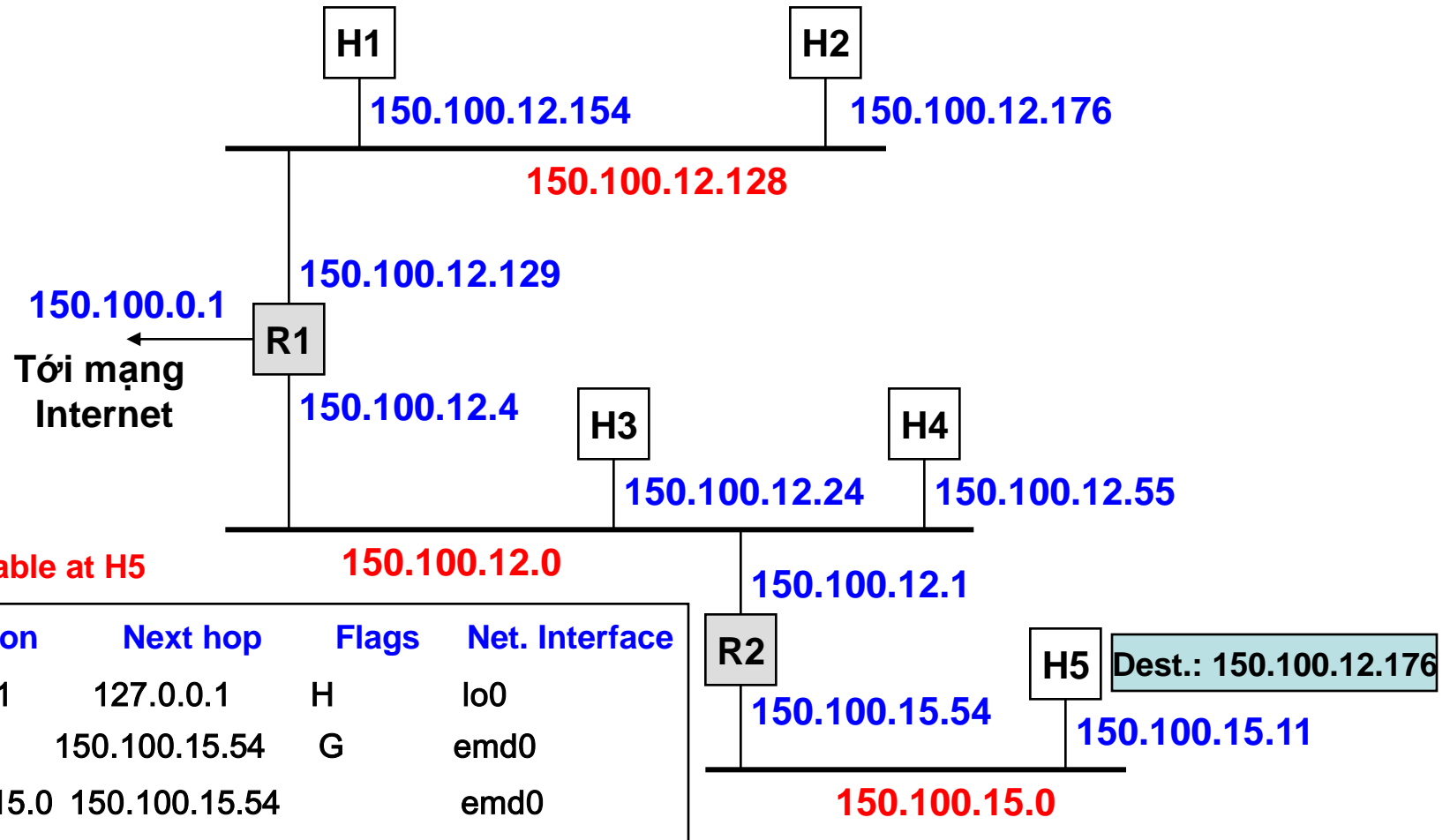
o **Router**: Kiểm tra địa chỉ IP của packet nhận được

- Nếu không biết IP đích, tham chiếu RT và xác định **next hop**

o **Routing table**

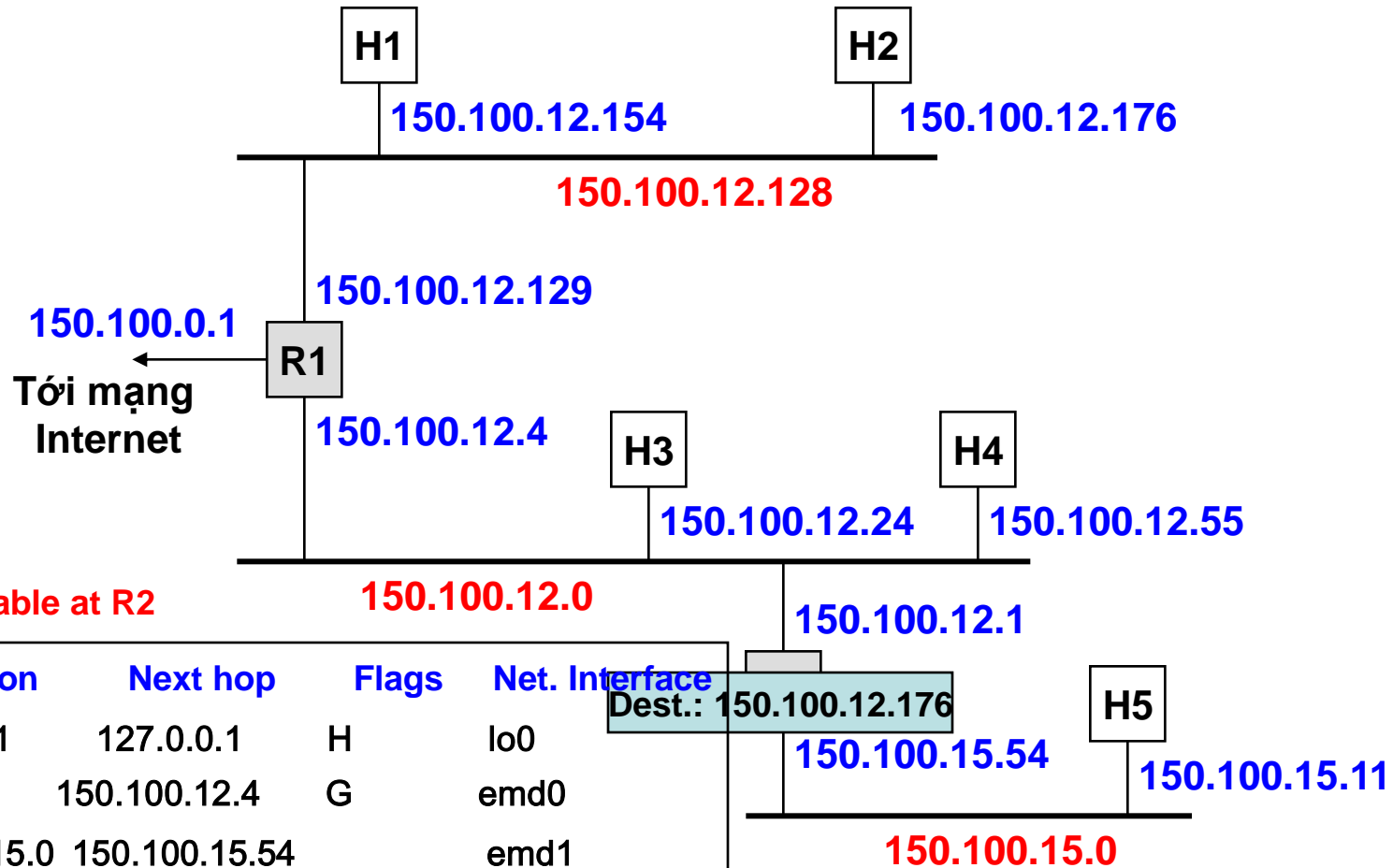
- Mỗi dòng trong RT chứa: **Dest. IP** , **next-hop router IP**, **subnet mask**, **phy. address**, **network interface**, **statistics**, **flag**
- Flag
  - ❖ **H = 1/0**: định tuyến tới host/network
  - ❖ **G = 1/0**: định tuyến tới router (gateway)/trực tiếp

o Ví dụ: **Host 5** → **Host 2**





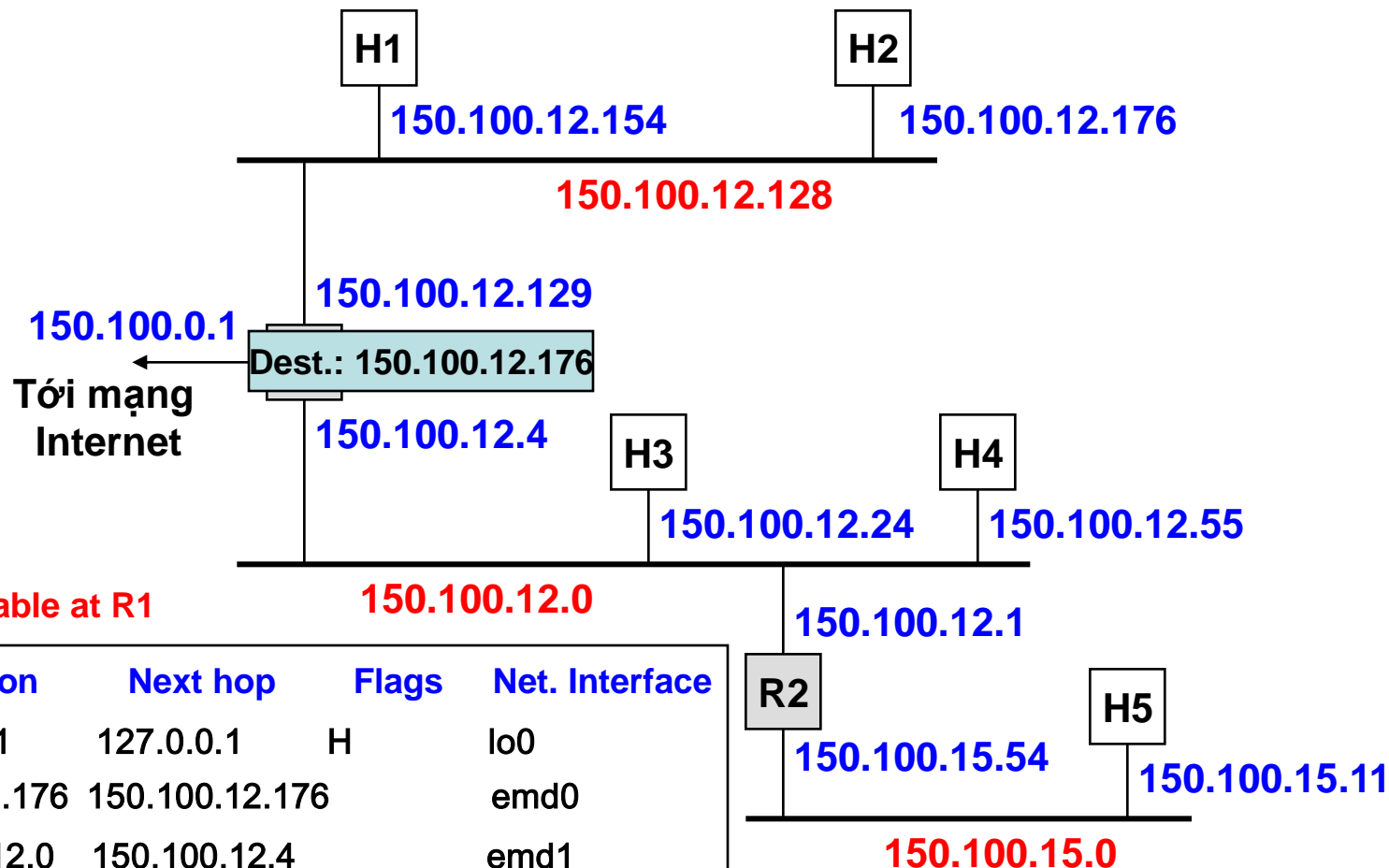
o Ví dụ: **Host 5** → **Host 2**



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**



✓ Vấn đề địa chỉ IP

o 1900: 2 vấn đề nảy sinh

- Hết các dải địa chỉ IP
- Bảng định tuyến IP phát triển công kênh

o Giải pháp tạm thời

- Subnetting
- Classless Interdomain Routing (**CIDR**)
- Network Address Translation (**NAT**)

o Giải pháp lâu dài: **IPv6**

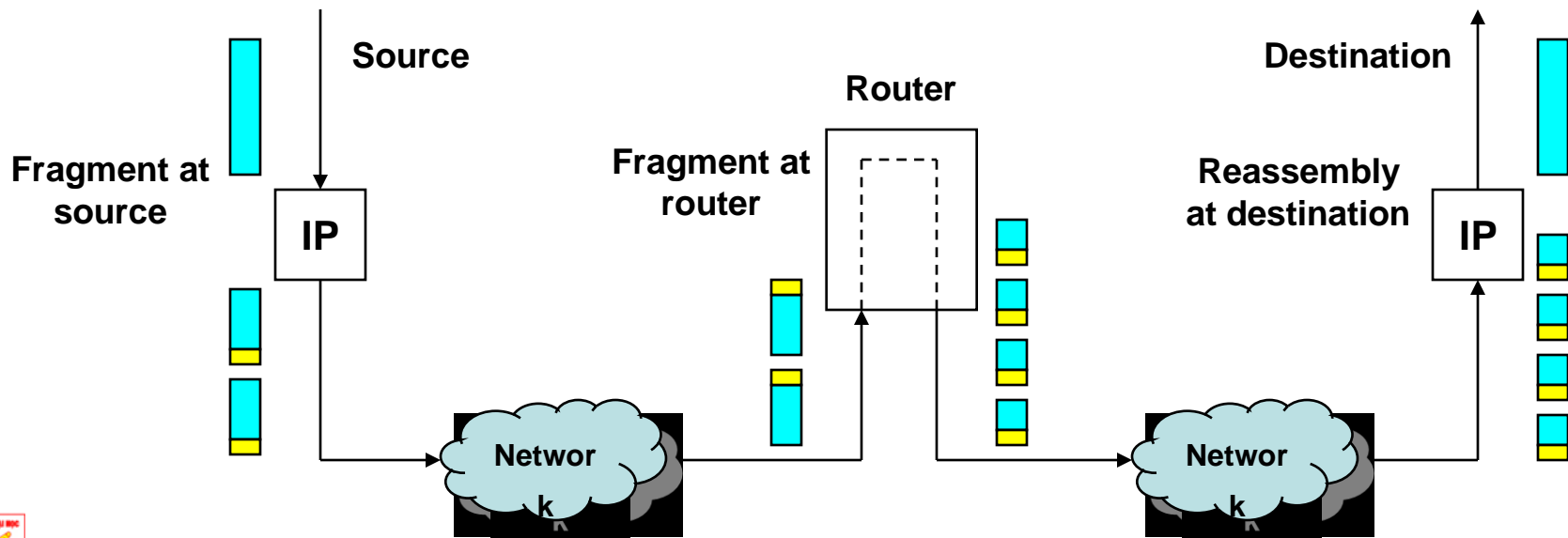
## ✓ CIDR và supernetting

- 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**)

0	4	8	16	19	24	31
Version	IHL	Type of Service	Total Length			
Identification			Flags	Fragment Offset		



- ✓ Ví dụ: Packet được truyền qua mạng với **Max. Transfer Unit (MTU)**  
**MTU = 576 byte, header = 20 byte, data = 1484 byte**
- o **Max. data length/fragment:  $576 - 20 = 556$  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

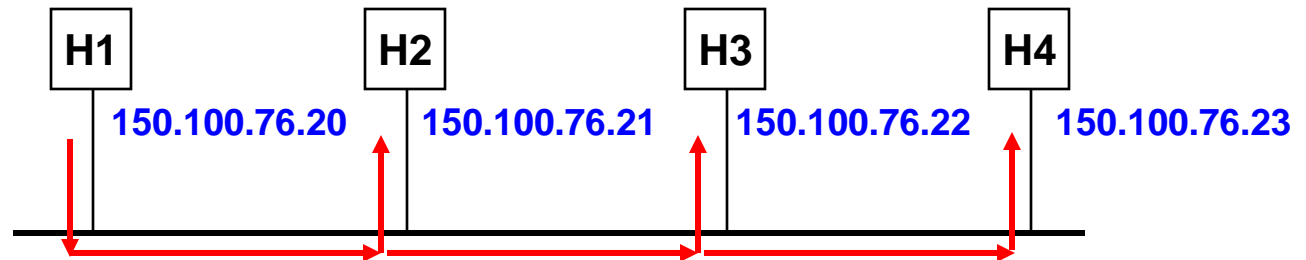
## ✓ Address Resolution Protocol (ARP)

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

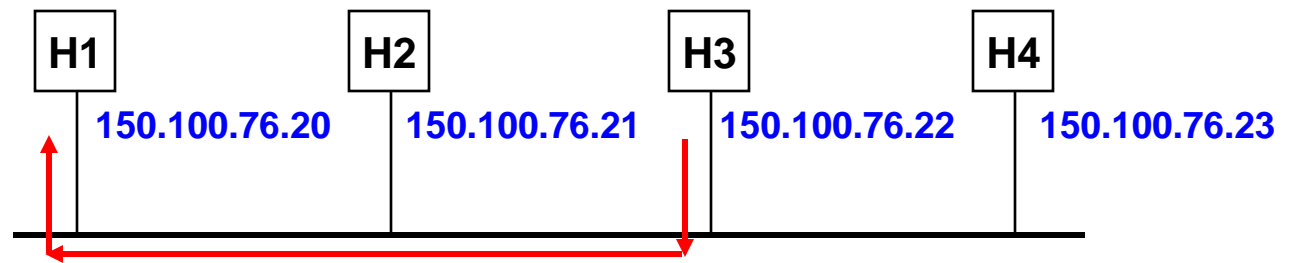
D: 150.100.76.22

MAC = ?



D: 150.100.76.20

MAC = 08:00:5A:3B:94



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

```

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

```

**ARP\_Notebook - 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	192.168.1.255	BROWSE	Host Announcement NOTEBOOK, wor

Frame 1 (254 bytes on wire, 254 bytes captured)

- Ethernet II, Src: 00:01:4a:c2:40:ab, Dst: ff:ff:ff:ff:ff:ff
  - Destination: ff:ff:ff:ff:ff:ff (Broadcast)
  - 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: 192.168.1.255 (192.168.1.255)
  - Version: 4
  - Header length: 20 bytes
  - Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
  - Total Length: 240
  - Identification: 0xcc35 (52277)
  - Flags: 0x00
  - Fragment offset: 0
  - Time to live: 128
  - Protocol: UDP (0x11)
  - Header checksum: 0xe974 (correct)
  - Source: 192.168.1.3 (192.168.1.3)
  - Destination: 192.168.1.255 (192.168.1.255)
- User Datagram Protocol, Src Port: netbios-dgm (138), Dst Port: netbios-dgm (138)
- NetBIOS Datagram Service
- SMB (Server Message Block Protocol)
- SMB Mailslot Protocol
- Microsoft windows Browser Protocol

```

0010  00 f0 cc 35 00 00 80 11 e9 74 c0 a8 01 03 c0 a8  ...5.... .t.....
0020  01 ff 00 8a 00 8a 00 dc 51 b8 11 02 80 b6 c0 a8  .....Q.....
0030  01 03 00 8a 00 c6 00 00 20 45 4f 45 50 46 45 45  .....EOEPFEE
0040  46 45 43 45 50 45 50 45 4c 43 41 43 41 43 41 43  FECEPEPE LCACACAC
0050  41 43 41 43 41 43 41 43 41 00 20 45 4c 44 47 43  ACACACAC A. ELDGC
0060  41 43 41 43 41 43 41 43 41 43 41 43 41 43  ACACACAC ACACACAC

```

Frame (frame), 254 bytes | P: 1 D: 1 M: 0



## ✓ 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 (210.245.0.21)  
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

```

0000  00 74 05 01 00 01 00 00 00 00 00 00 00 00 00 00  ..<.]... ..
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
  
```

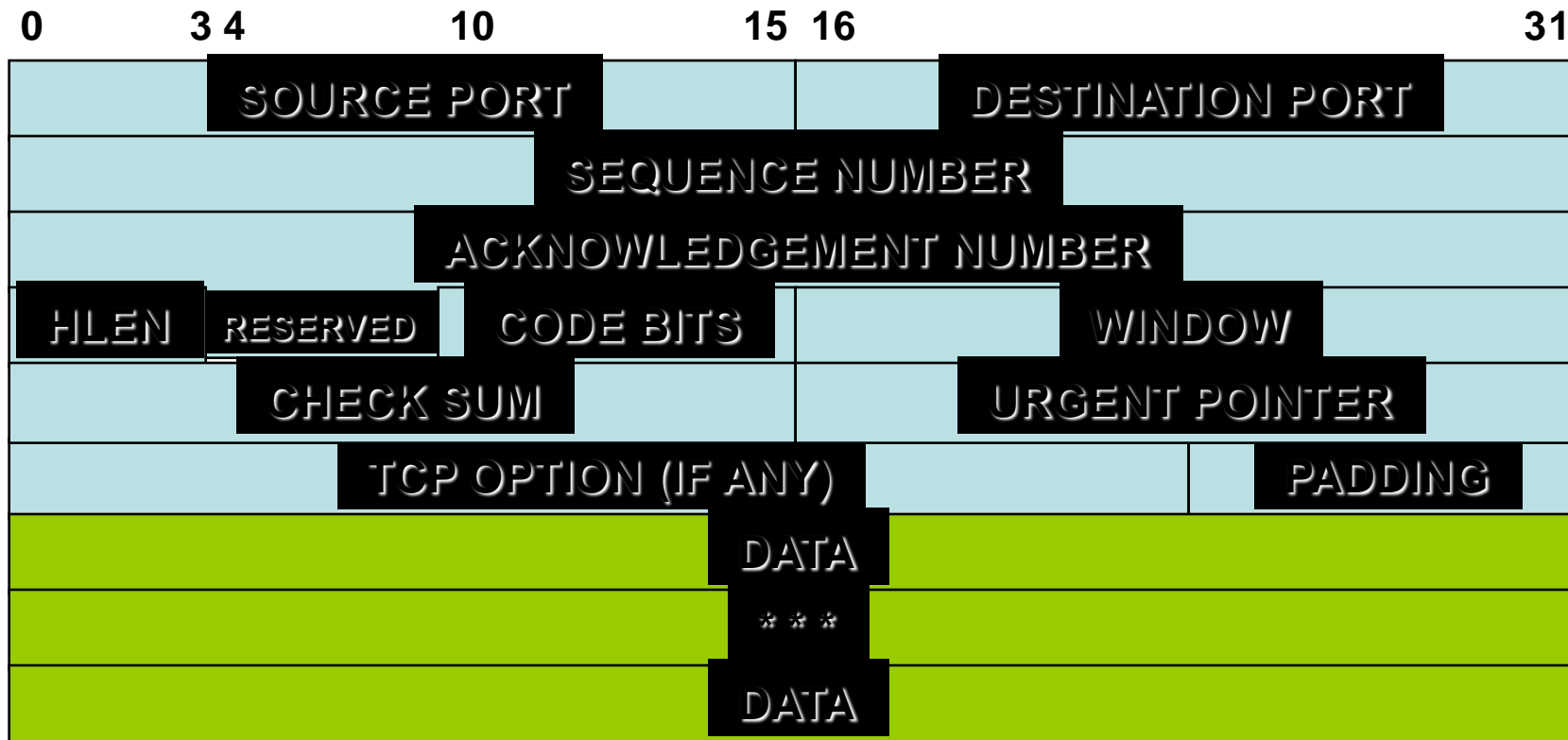
Internet Control Message Protocol | P: 10 D: 10 M: 0

- ✓ **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

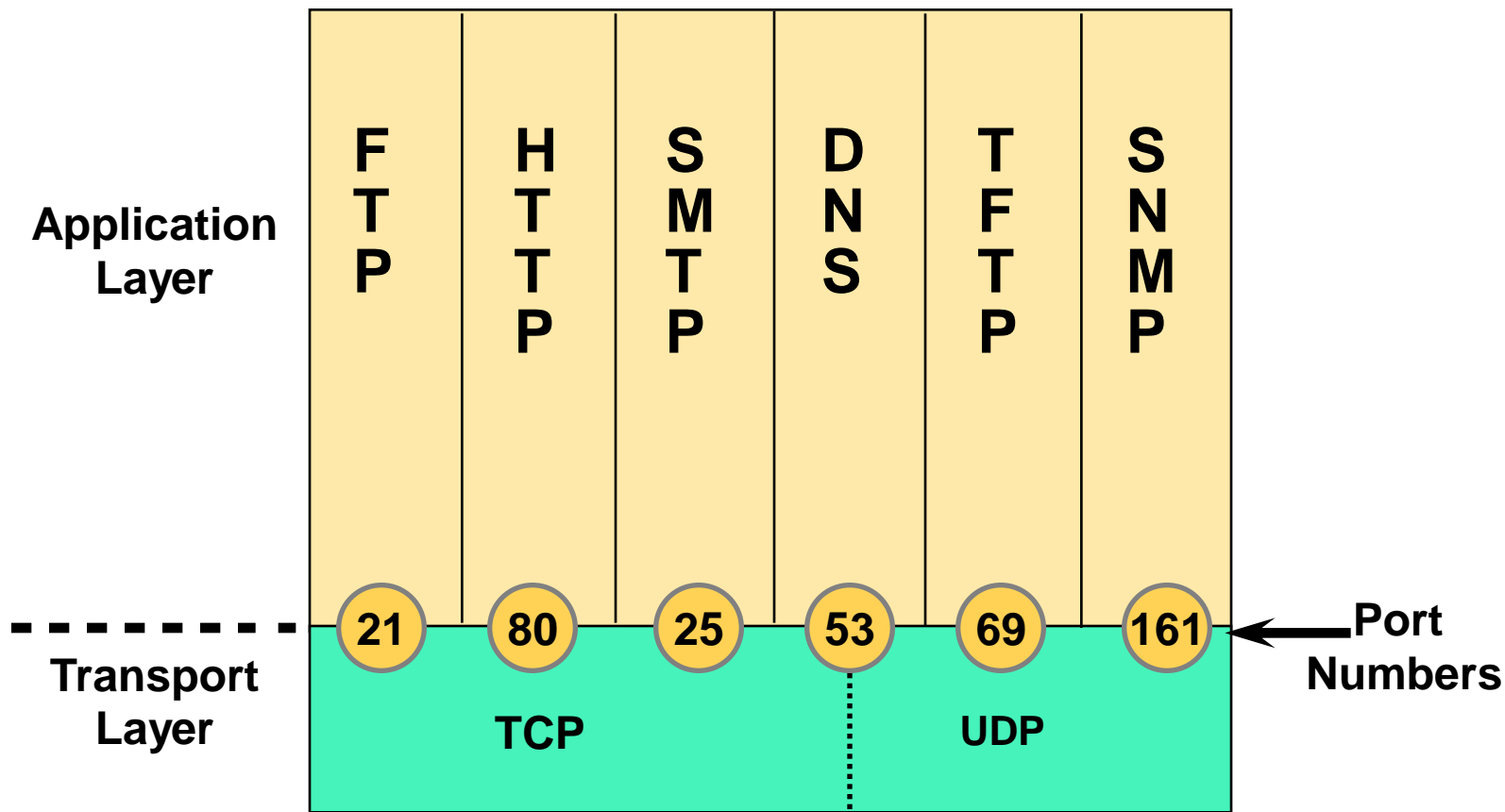
## o Transmission Control Protocol

- 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 Acknowledgment 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.

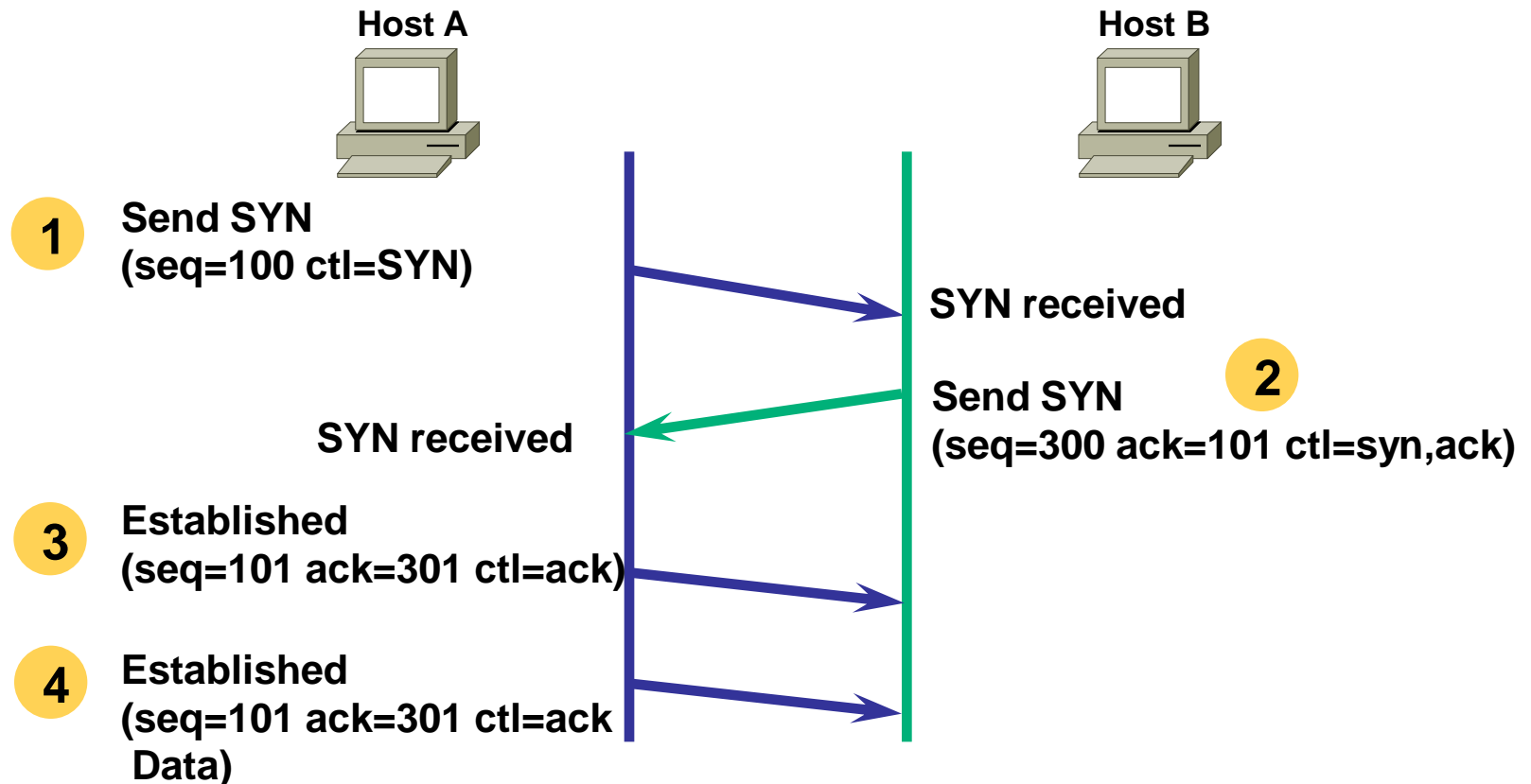
## Cấu trúc segment TCP



## Port number, phân kênh và dồn kênh

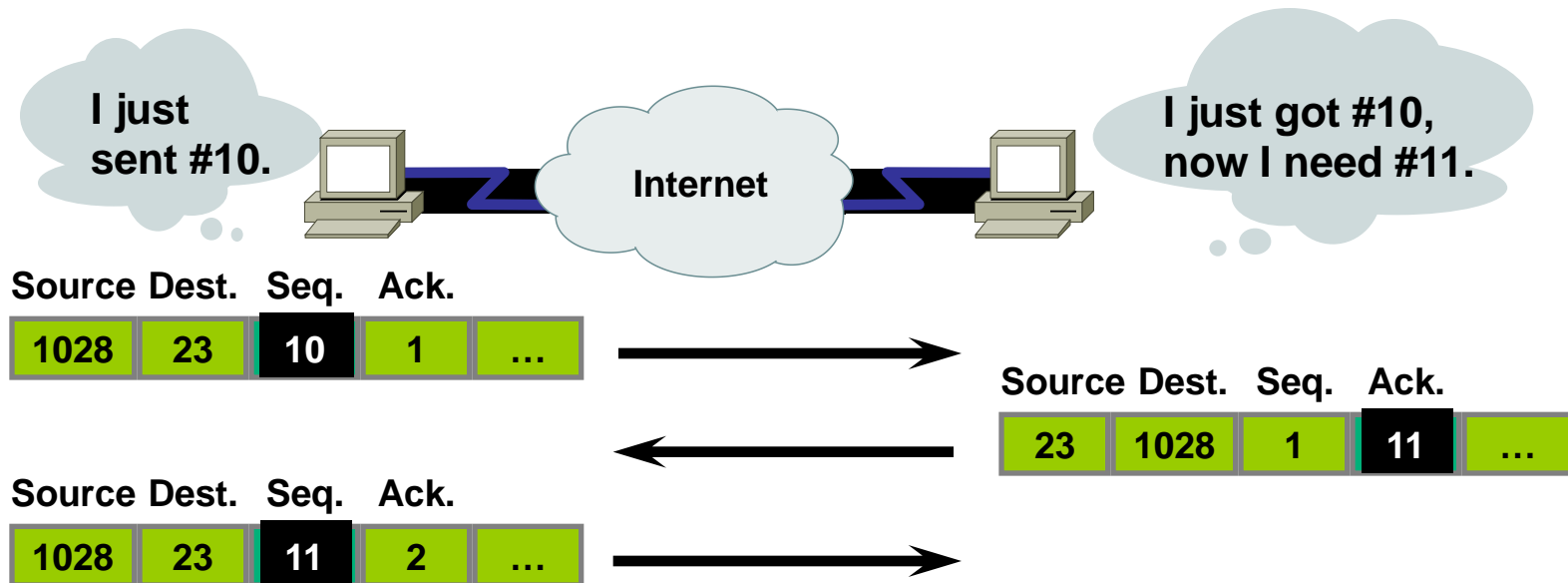


# TCP Handshake/Open Connection



# TCP Sequence and Acknowledgment Numbers

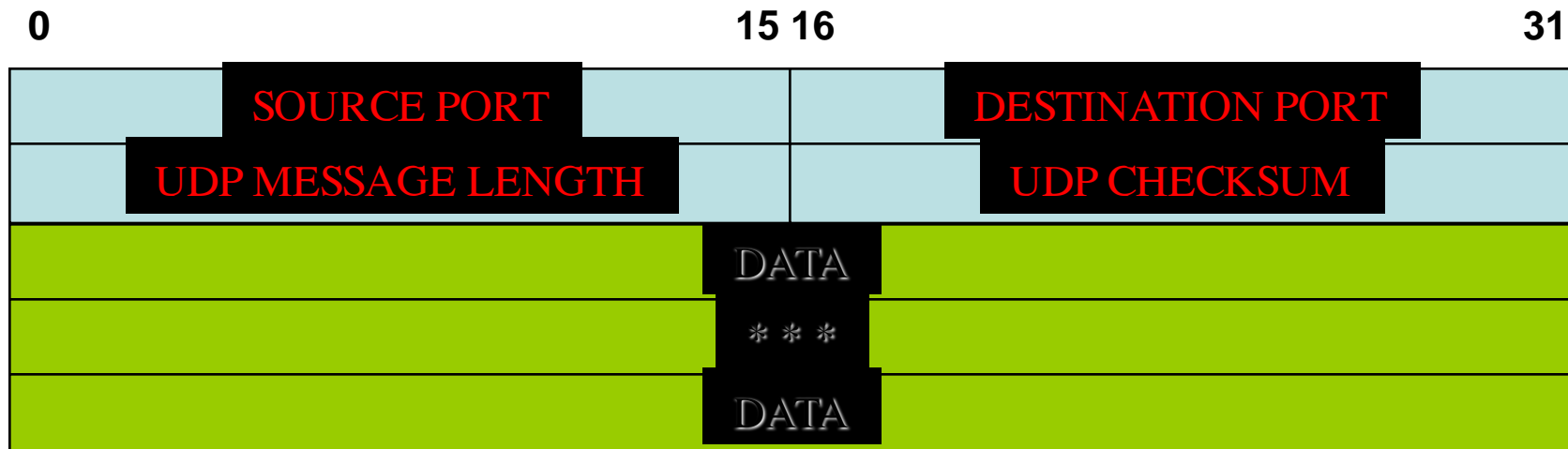
Source Port	Dest. Port	Sequence #	Acknowledgement #	...
-------------	------------	------------	-------------------	-----

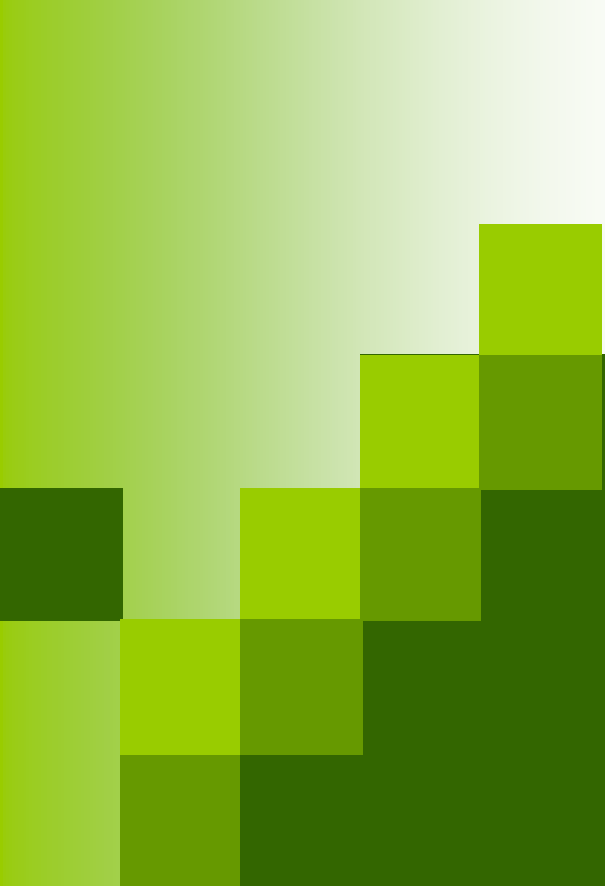




## o User data protocol

- UDP cho phép chương trình ứng dụng truy cập trực tiếp đến gói tin của dịch vụ chuyển giao giống như 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.





# Tổng quan về giao thức TCP/IP

Mail:

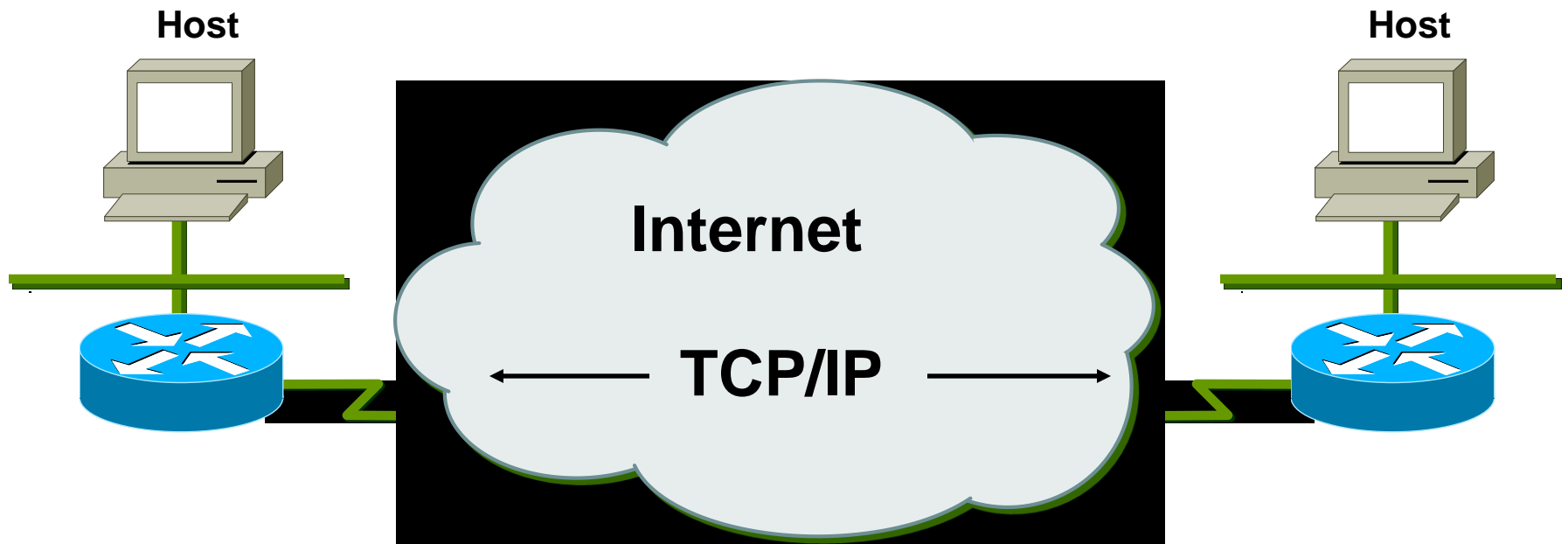
[lephuc@ptithcm.edu.vn](mailto:lephuc@ptithcm.edu.vn)

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

# Nội dung

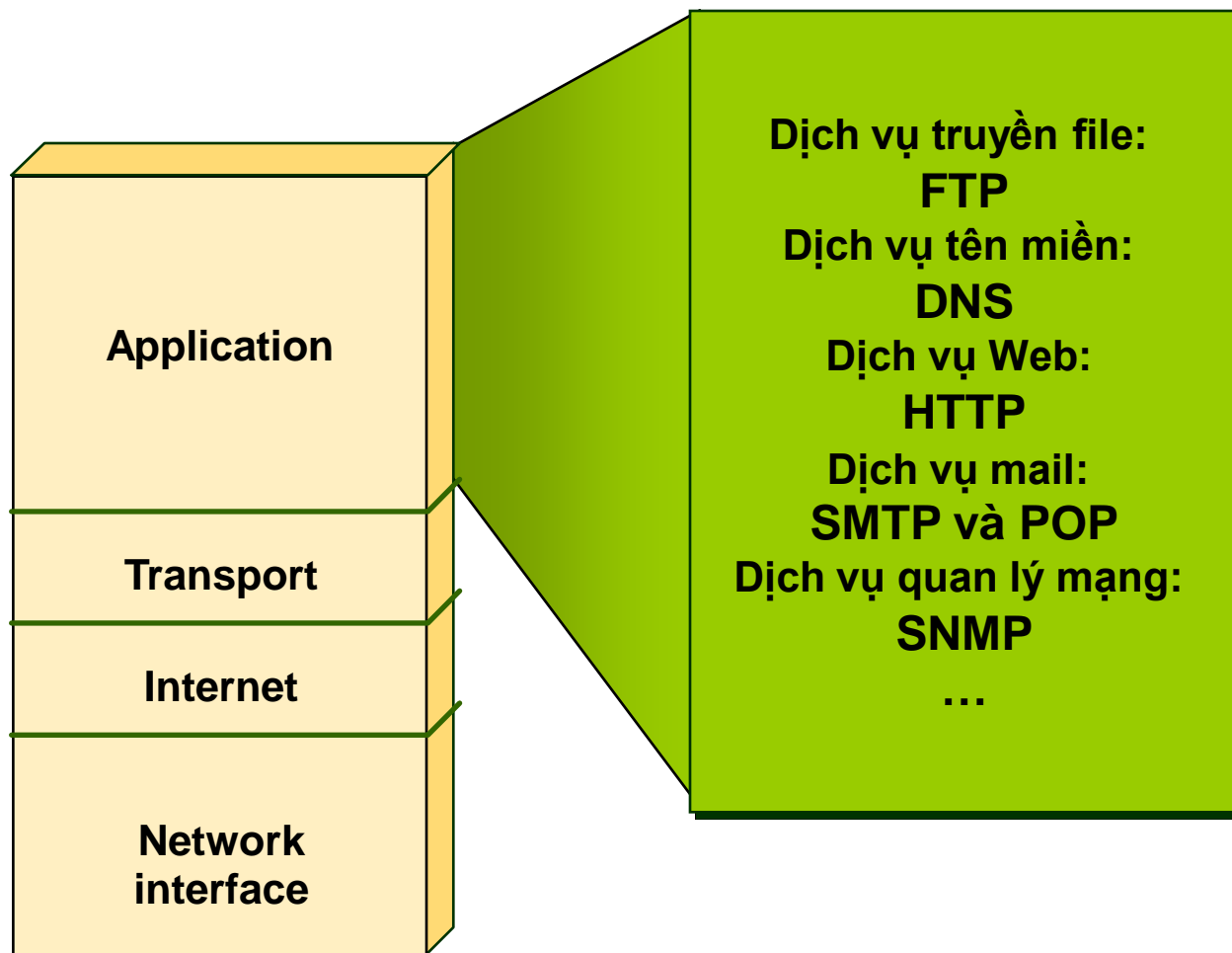
- Các thành phần của TCP/IP
- Giao thức IP
- Giao thức TCP
- Giao thức UDP
- Các giao thức phụ trợ (ARP, ICMP)
- Sơ đồ chuyển đổi trạng thái TCP

# TCP/IP trên mạng Internet

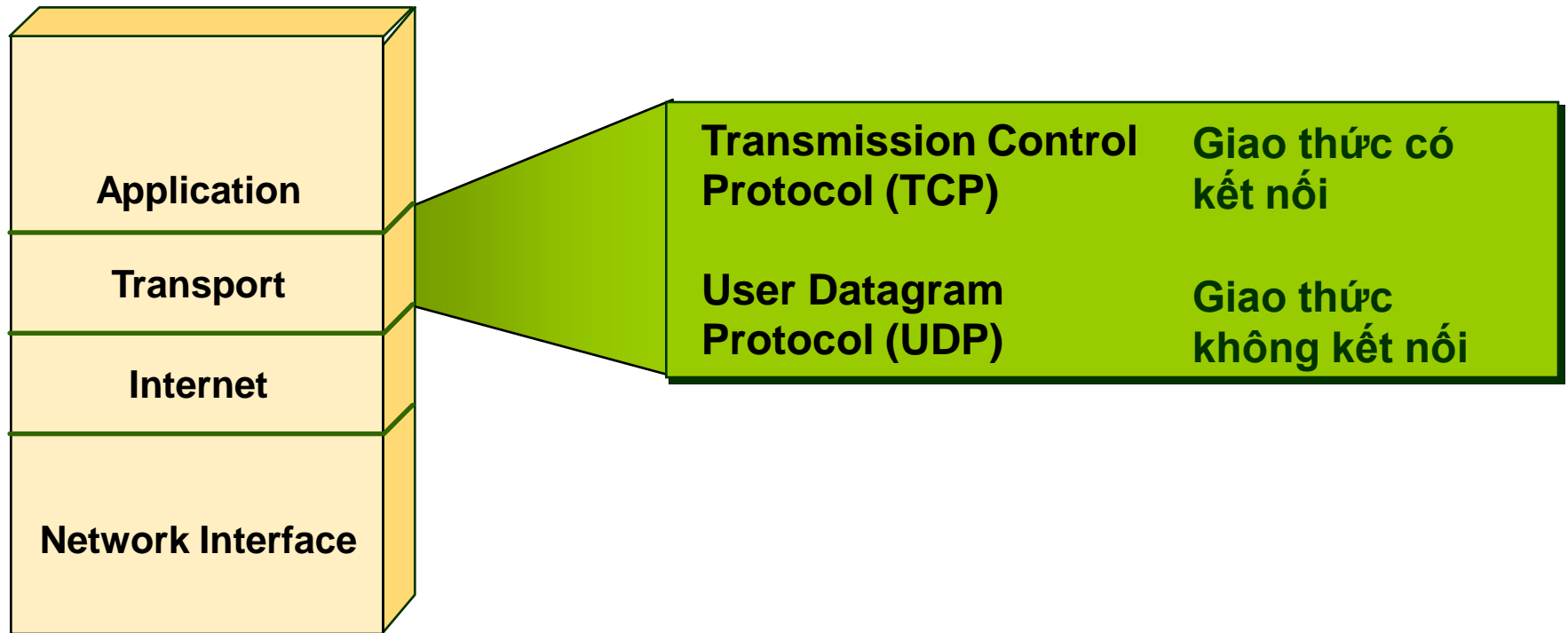


- Là giao thức truyền thống, bắt buộc đối với mạng Internet.

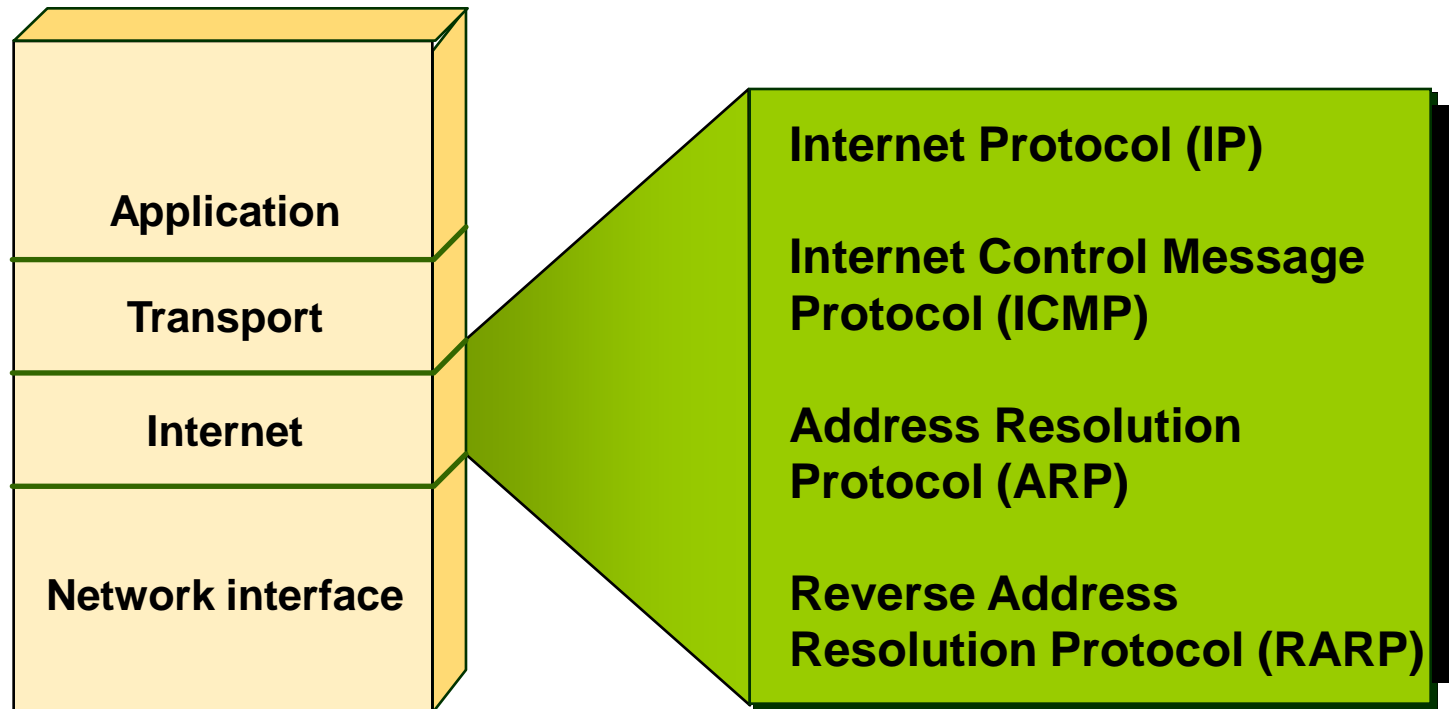
# Lớp ứng dụng của TCP/IP



# Lớp Vận chuyển trong TCP/IP



# Lớp Liên kết mạng trong TCP/IP

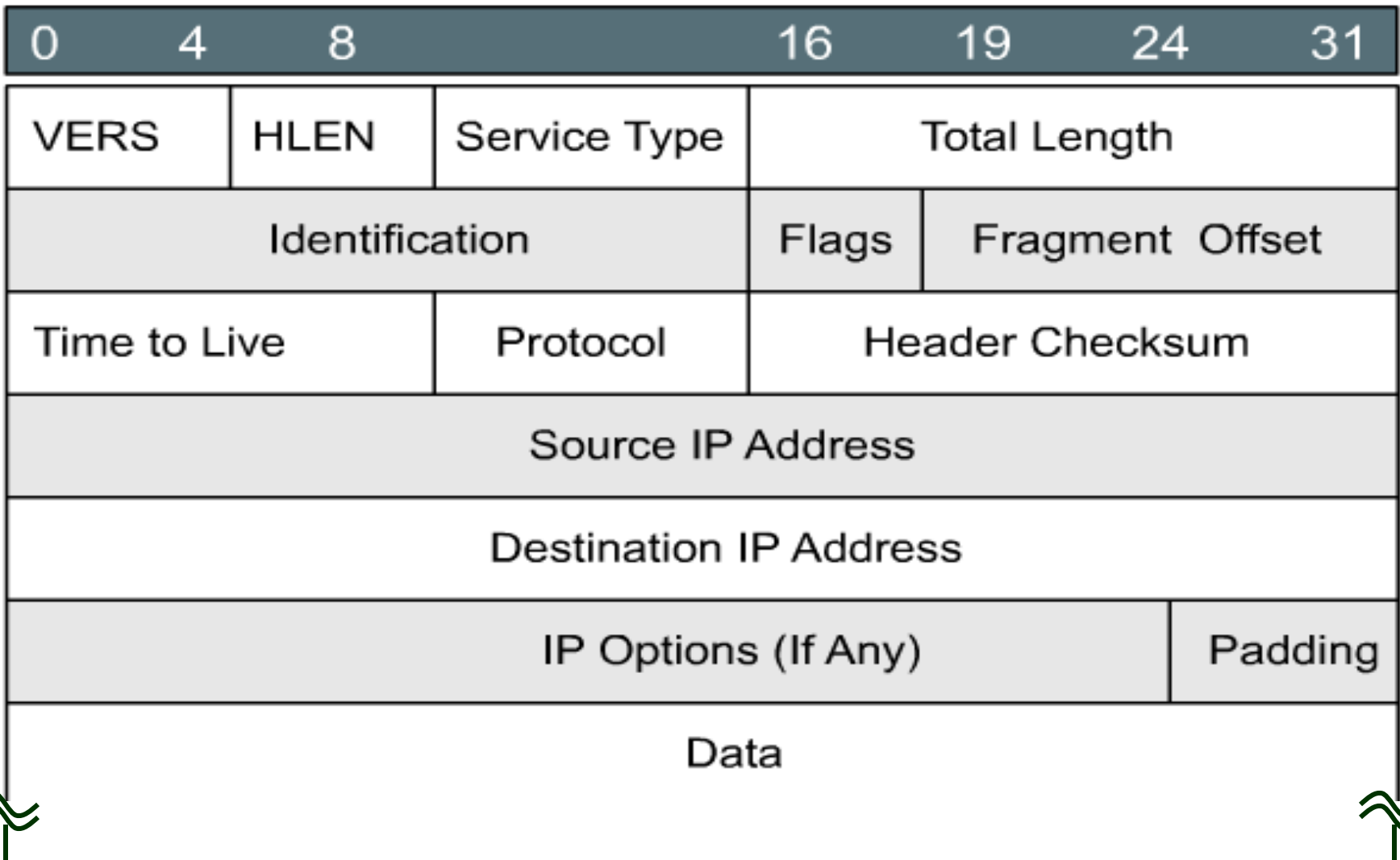


# Giao thức IP

- 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:

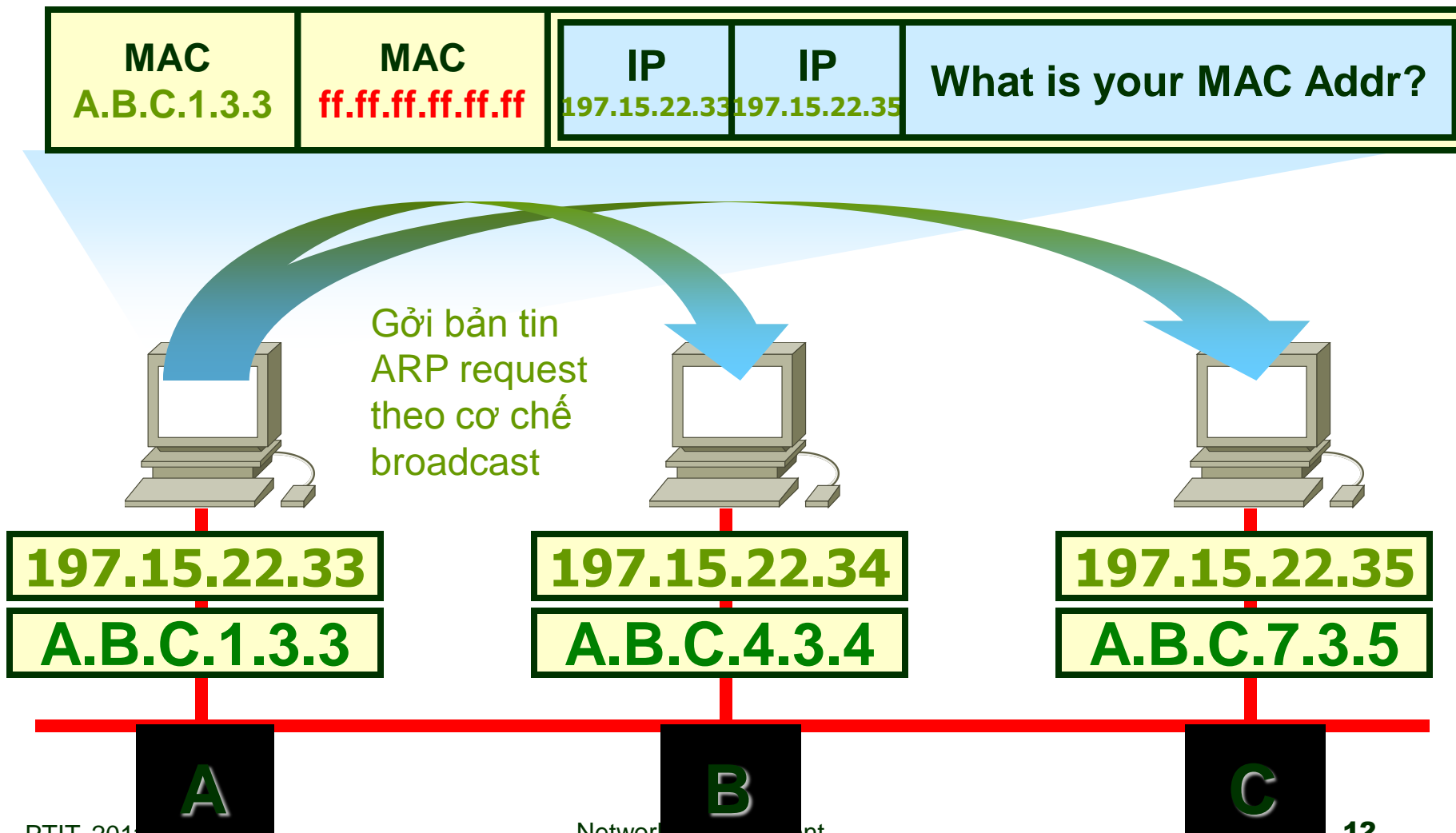
  0  1      <1 ms    <1 ms    <1 ms    192.168.10.254
  1  2       3 ms     2 ms     2 ms     203.162.100.33
  2  3       8 ms    11 ms    8 ms     203.162.143.73
  3  4      19 ms    19 ms    19 ms    localhost [123.30.120.41]
  4  5       9 ms    12 ms    11 ms    localhost [123.30.120.22]
  5  6      19 ms    19 ms     9 ms    www.tuoitre.com.vn [203.162.163.35]

Trace complete.
```

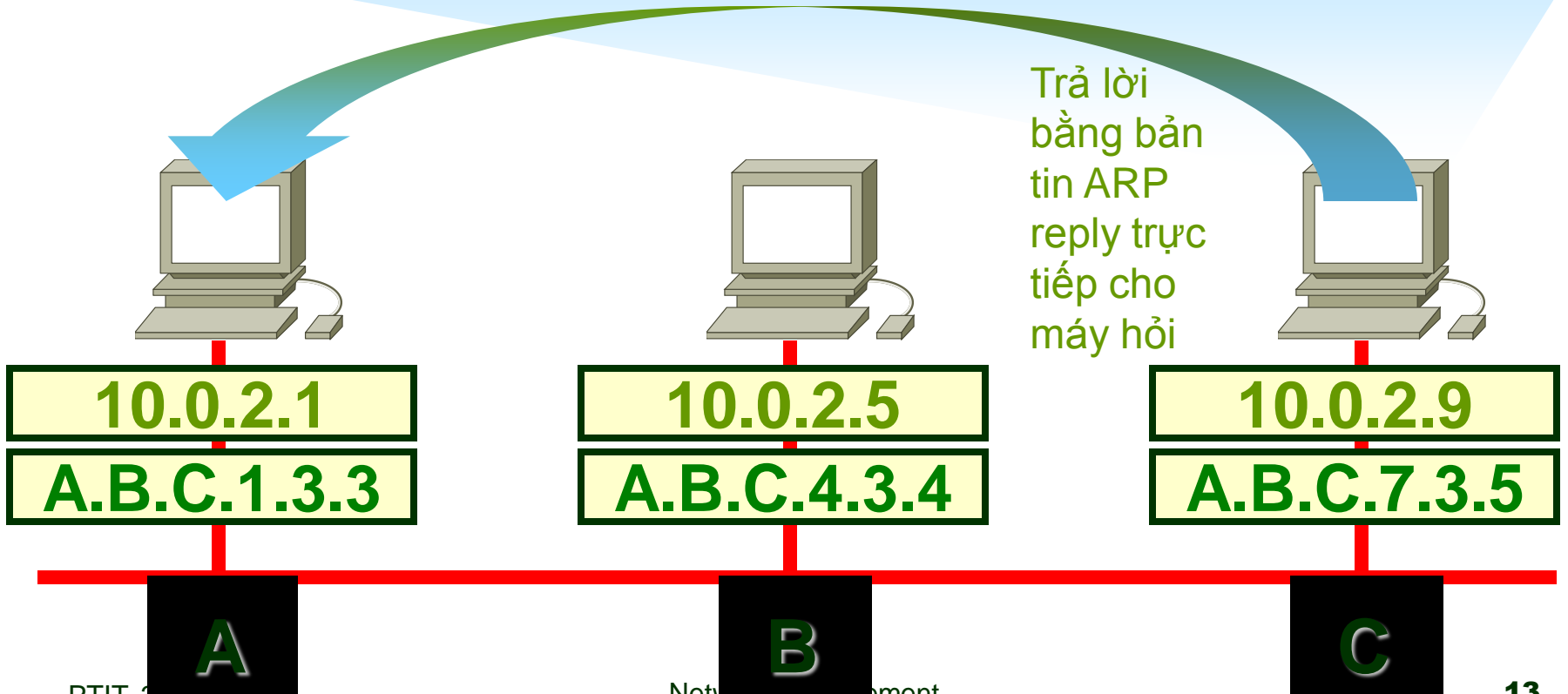
# Giao thức ARP

- 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 gọi kế tiếp.

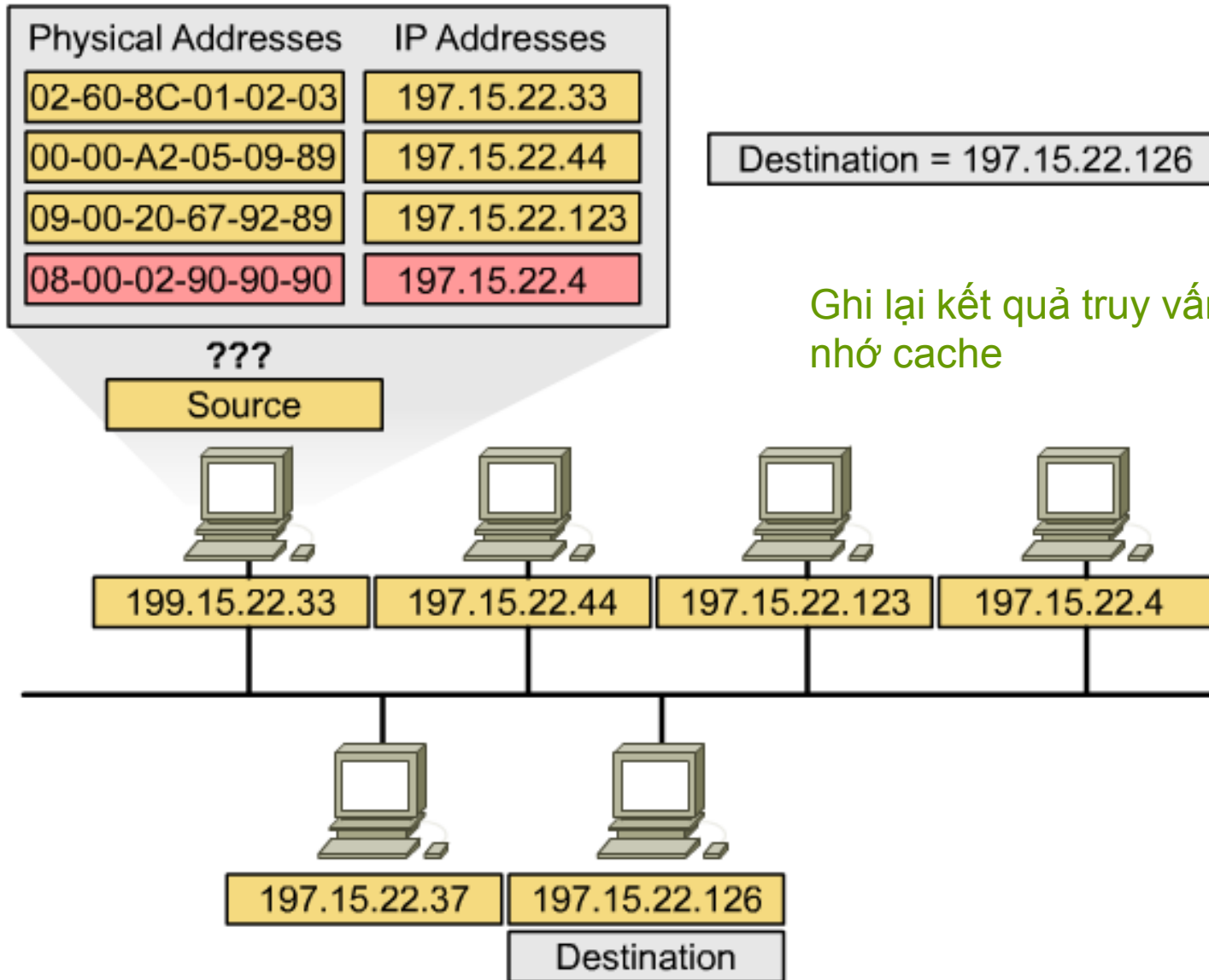
# Giao thức ARP: ARP request



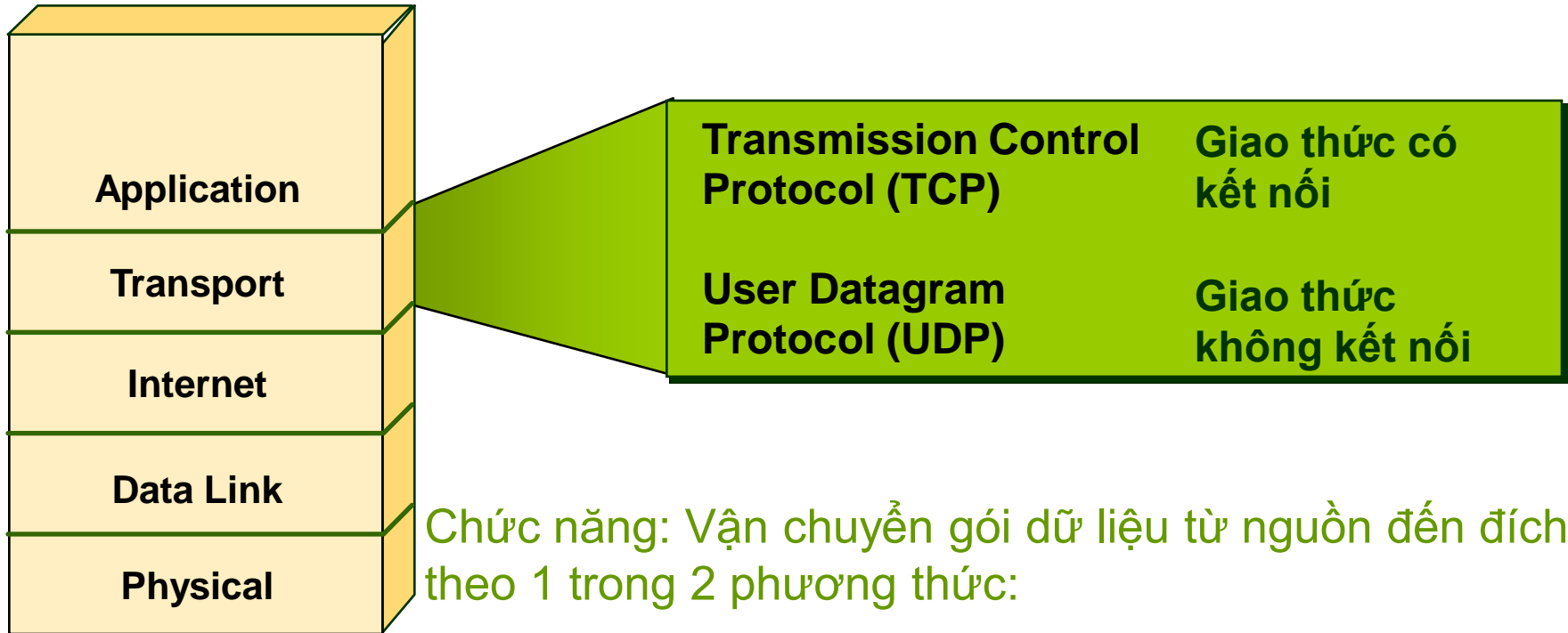
# Giao thức ARP: ARP reply



# Giao thức ARP: Caching



# Lớp Vận chuyển trong TCP/IP

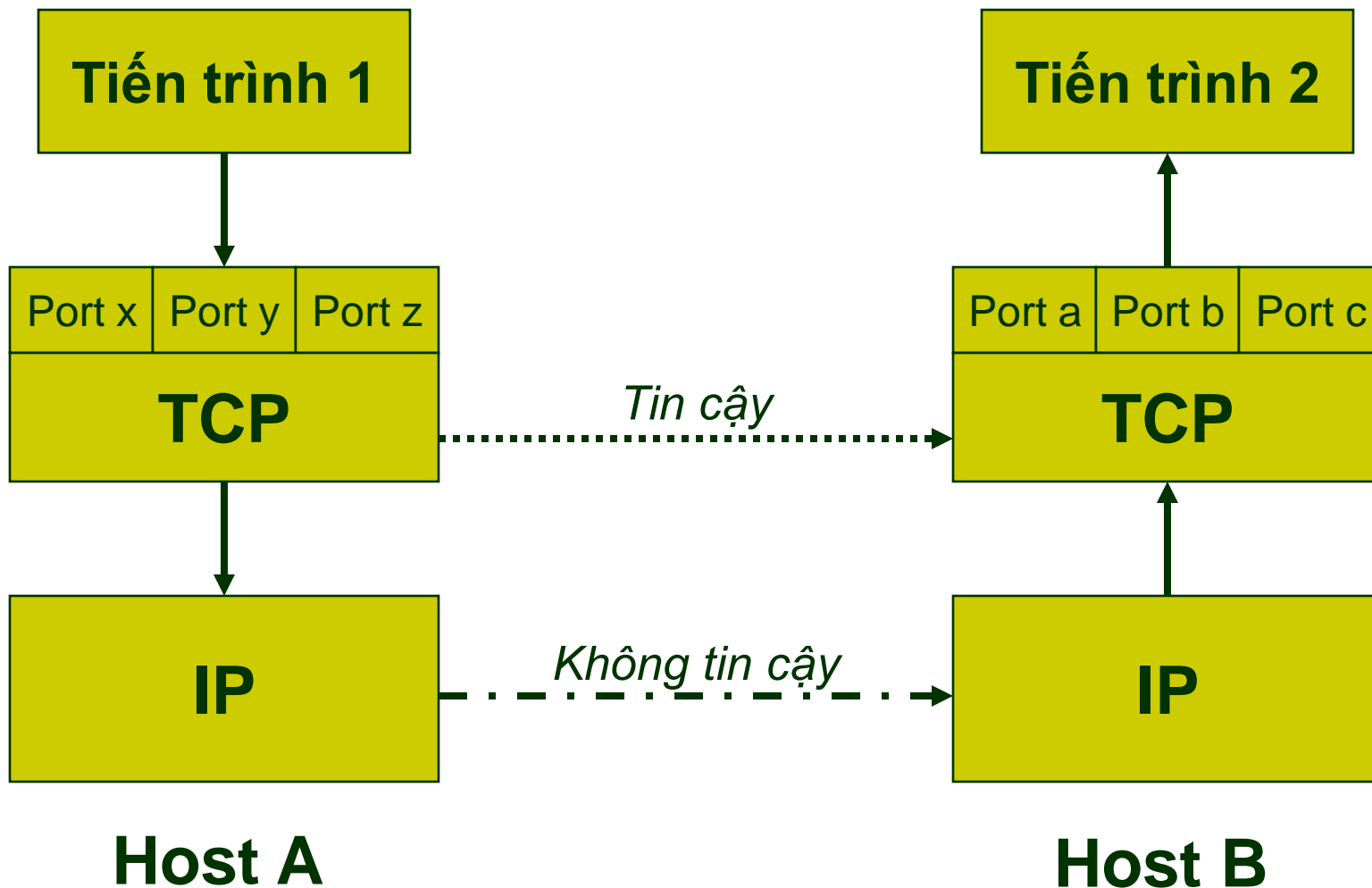


Chức năng: Vận chuyển gói dữ liệu từ nguồn đến đích theo 1 trong 2 phương thức:

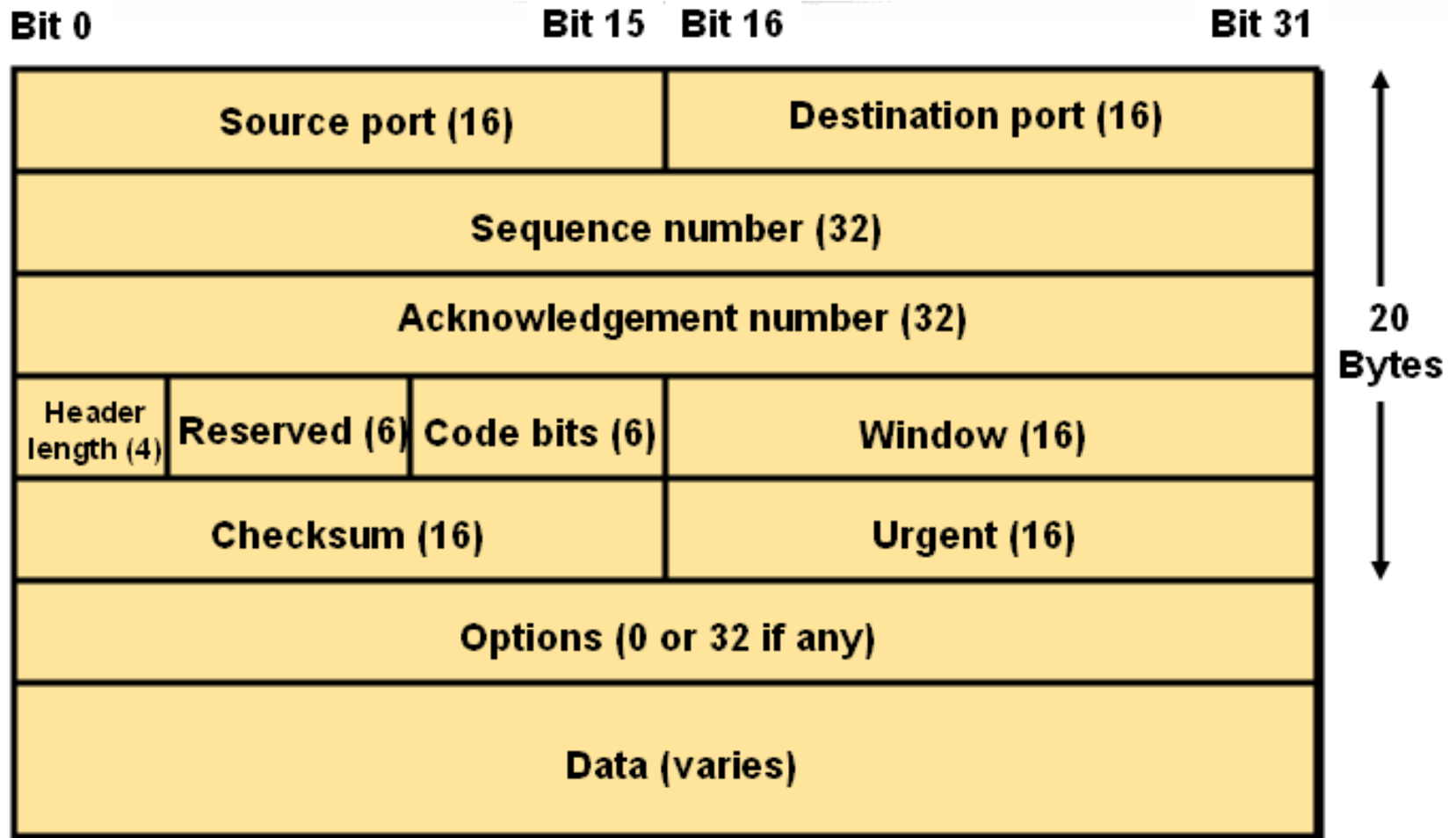
- **Tin cậy (TCP):** Có sửa lỗi và điều khiển
- **Không tin cậy (UDP):** Không sửa lỗi, không điều khiển



# Giao thức TCP

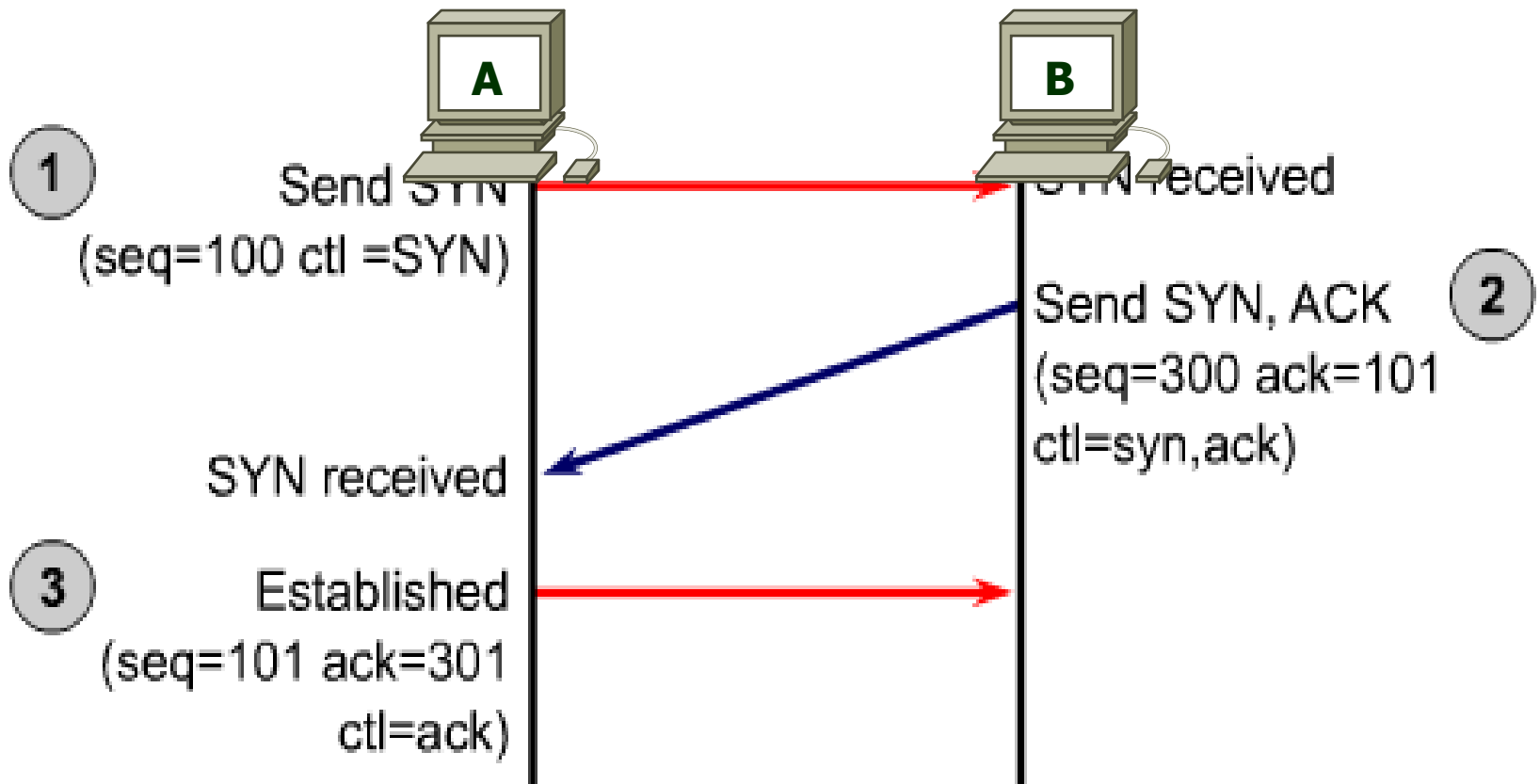


# Cấu trúc gói dữ liệu TCP

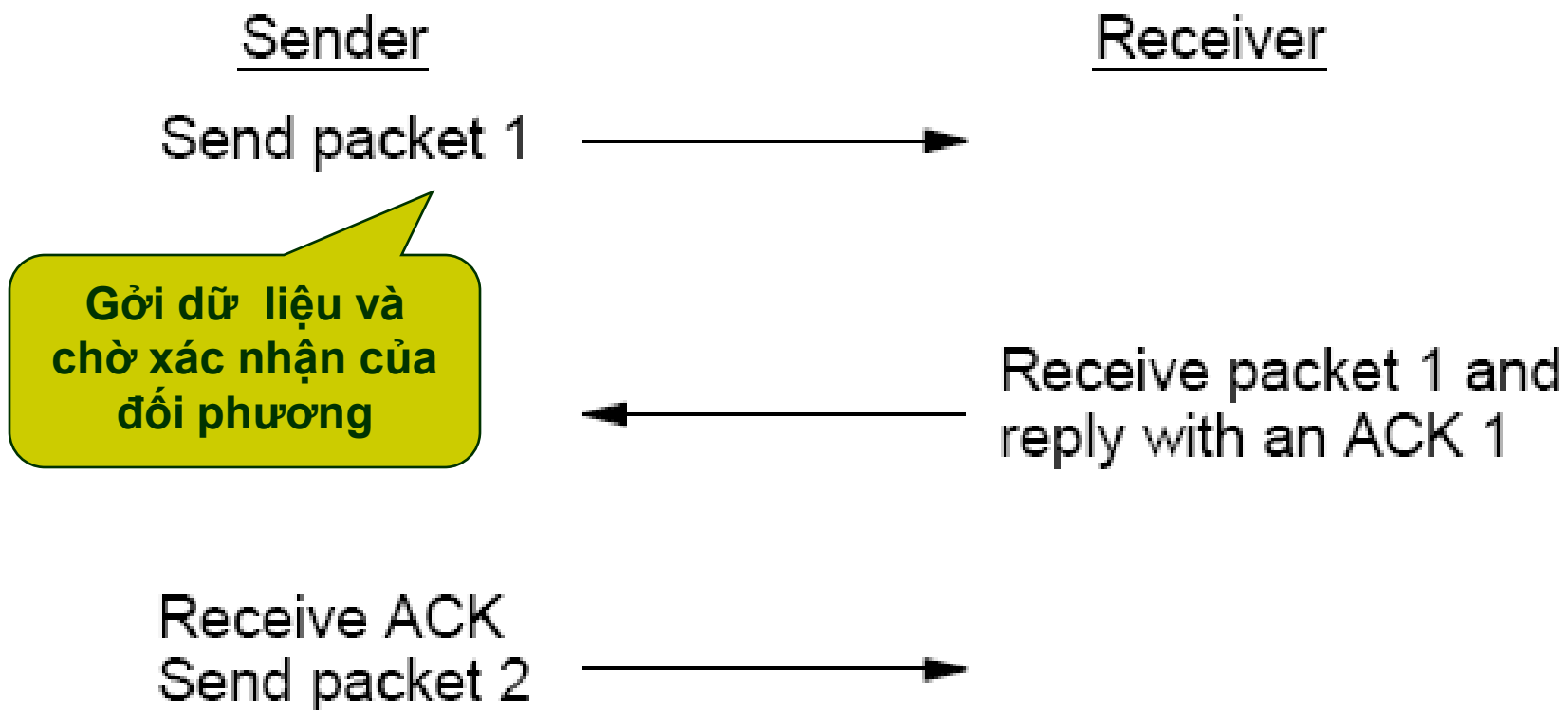


# Thủ tục thiết lập kết nối TCP

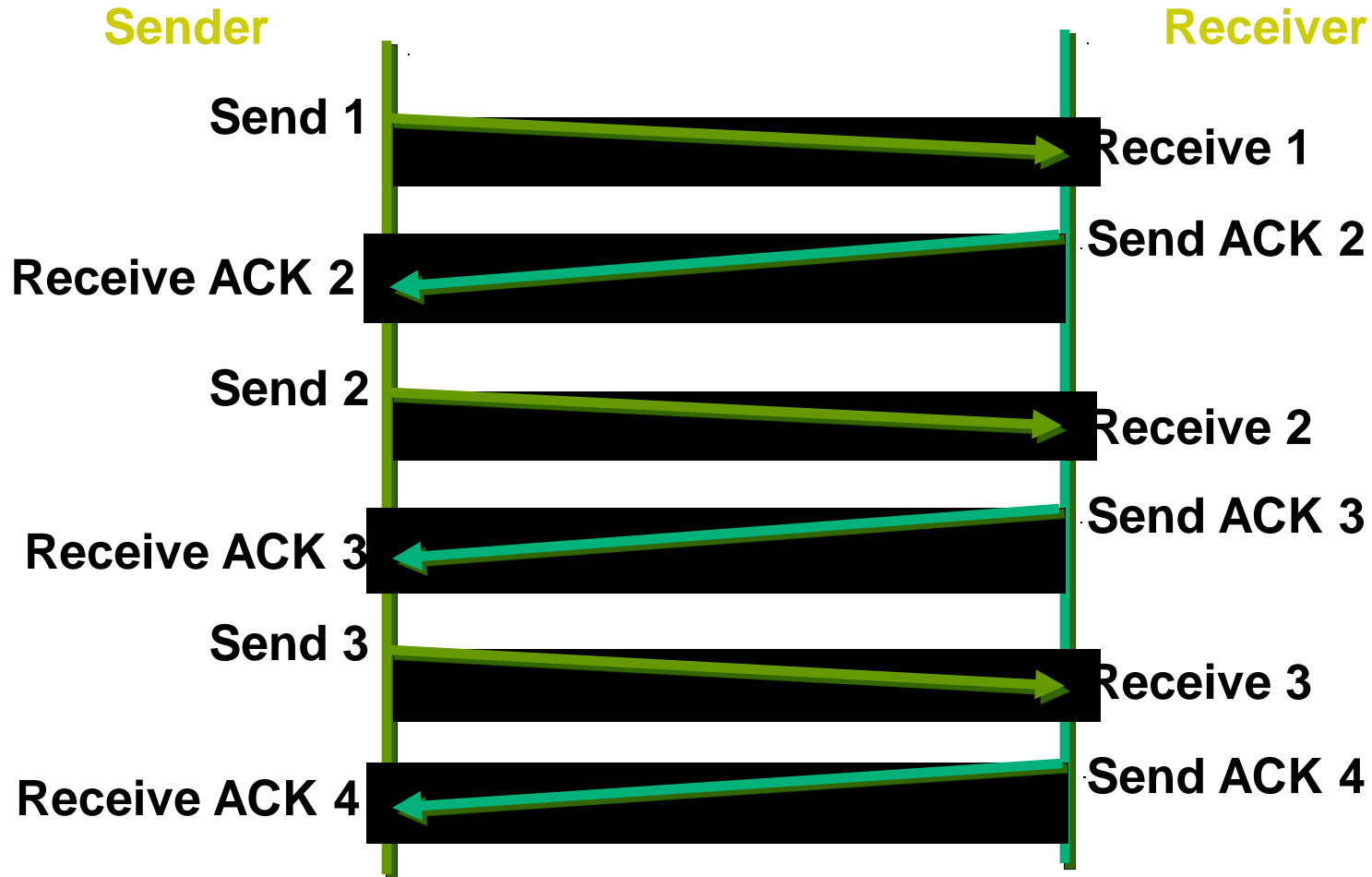
*Thủ tục bắt tay 3 chiều (three way handshake)*



# Cơ chế truyền dữ liệu trong TCP

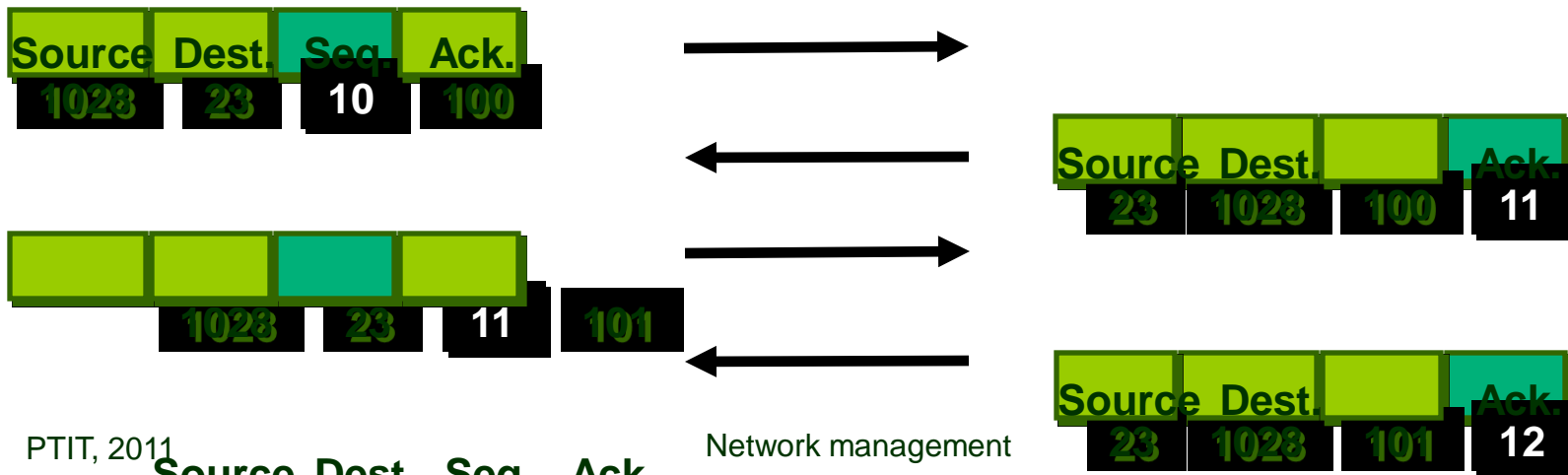
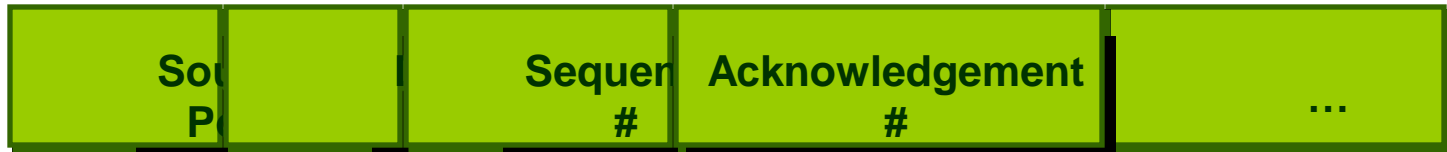


# TCP: cơ chế truyền đơn giản



■ Window size = 1

# TCP: cơ chế truyền đơn giản



# TCP: Cơ chế dịch cửa sổ

*Giảm thời gian chờ, tăng hiệu suất truyền*

Phía gửi

Window size = 3  
Send 1

Window size = 3  
Send 2

Window size = 3  
Send 3

Window size = 3  
Send 3

Window size = 3  
Send 4

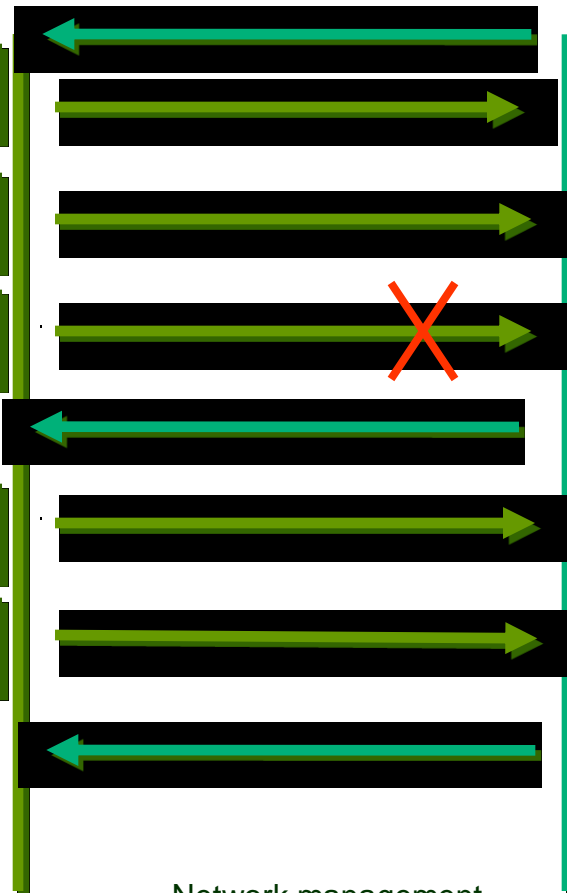
Phía nhận

Window size = 3

**Gói số 3 lỗi**

ACK 3  
Window size = 2

ACK 5  
Window size = 2

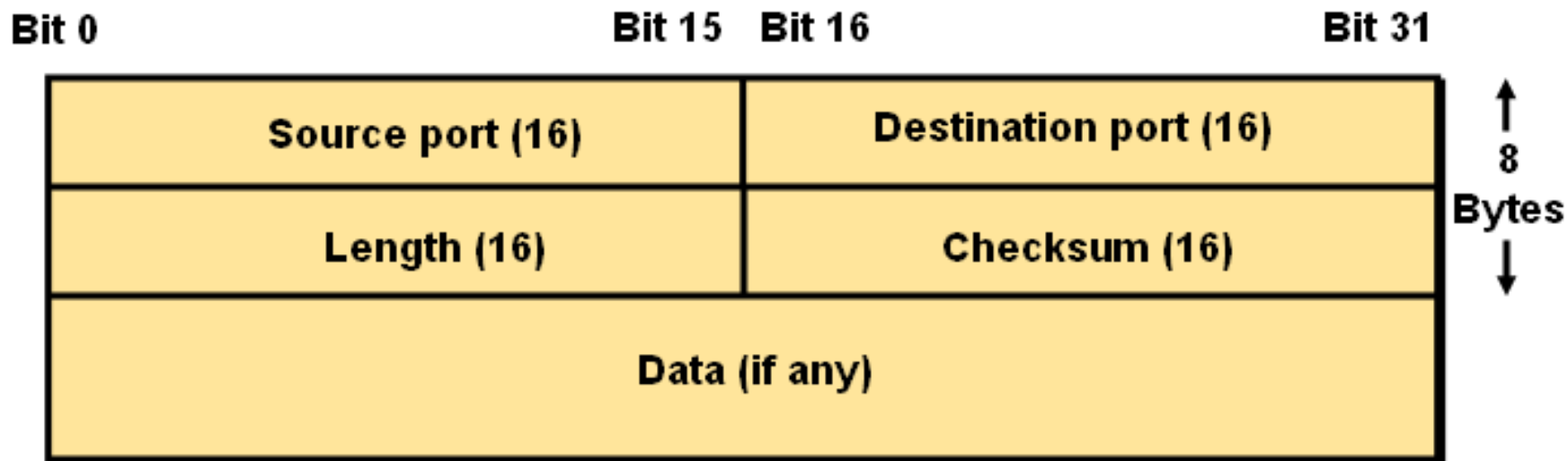


# Đặc điểm giao thức TCP

- 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

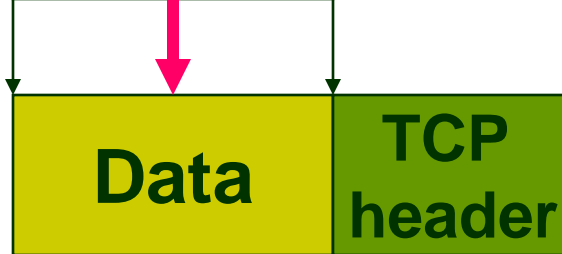
- Là giao thức không có kết nối
- Hoạt động đơn giản, không có các chức năng điều khiển.
- Truyền dữ liệu không tin cậy.
- Thích hợp với các dịch vụ có lượng dữ liệu nhỏ, tính đáp ứng nhanh.

# Đóng gói dữ liệu trong TCP/IP

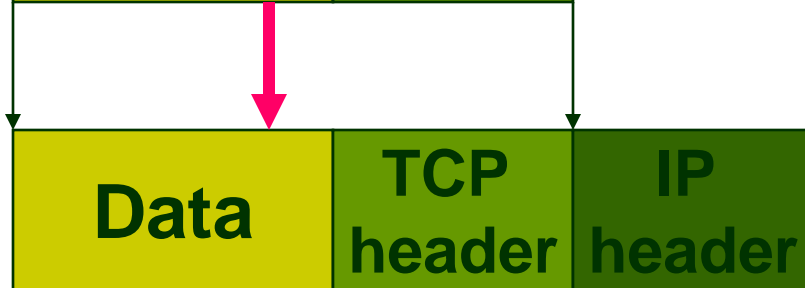
Lớp ứng dụng



Giao thức TCP

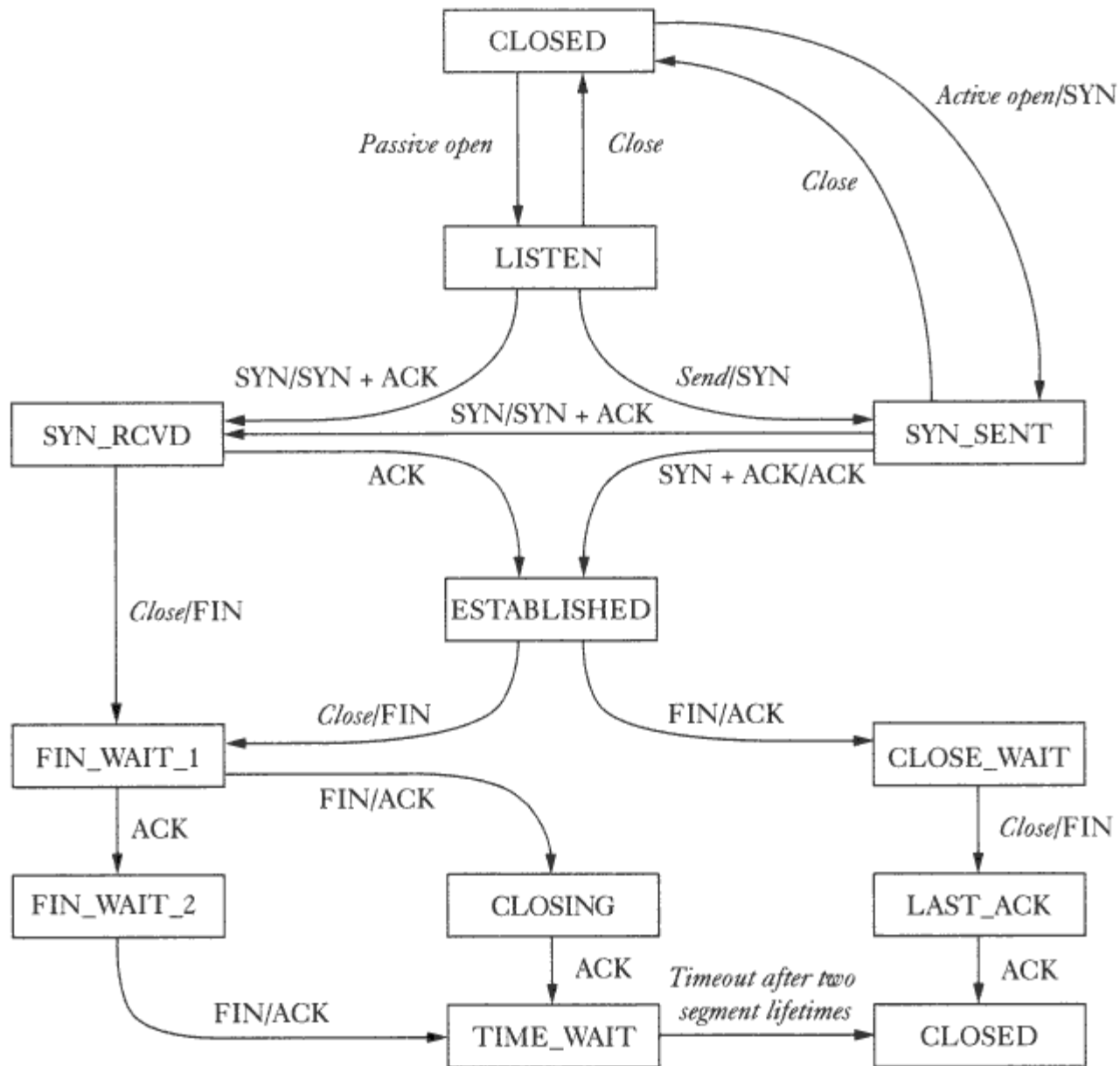


Giao thức IP

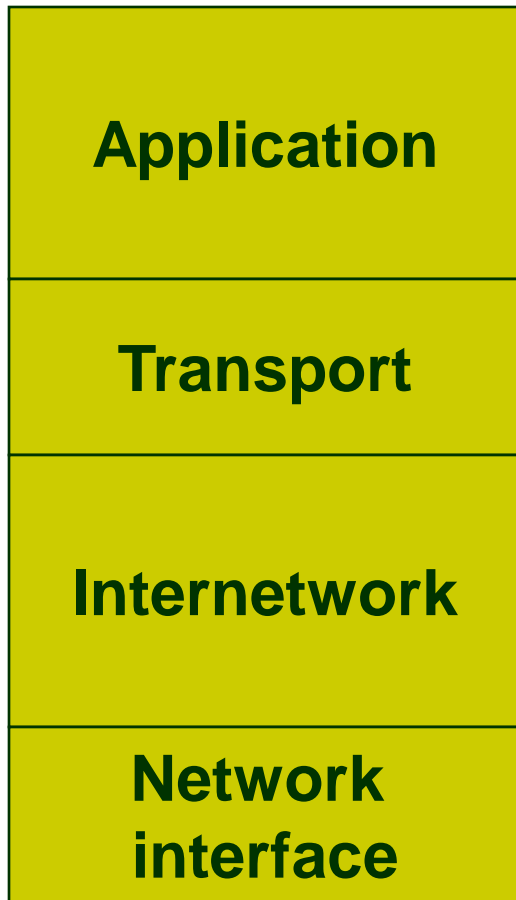


# Sơ đồ trạng thái TCP

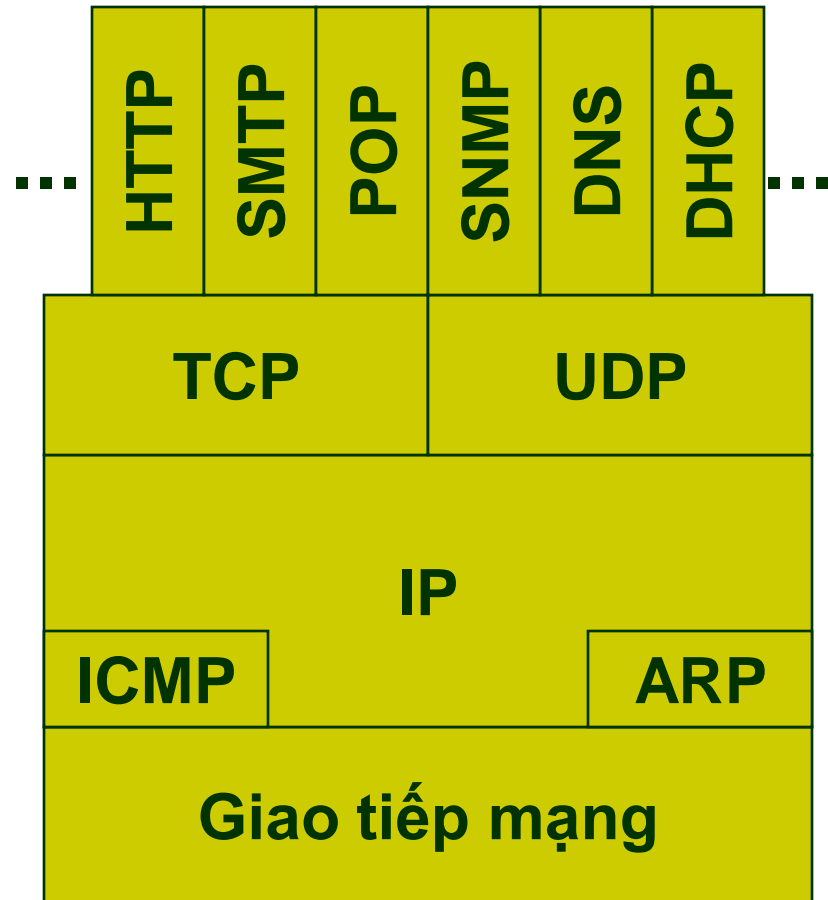
- Liệt kê các trạng thái (state) của giao thức TCP
- Các cơ chế chuyển đổi trạng thái của giao thức TCP.



# Tóm tắt bộ giao thức TCP/IP



**Mô hình TCP/IP**



**Giao thức TCP/IP**

# Các mô hình và giao thức mạng

# Mục tiêu bài học

- Nhận biết nhu cầu sử dụng giao thức mạng
- Mô hình OSI
- Chi tiết 7 lớp mô hình OSI

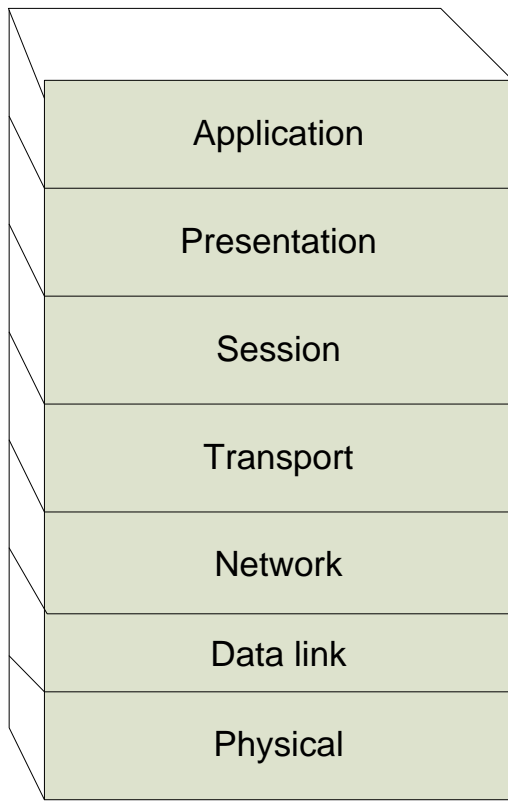


# Nhu cầu sử dụng giao thức

- Cú pháp (Syntax)
  - Giải quyết các nhu cầu về định dạng dữ liệu
  - Mức độ tin hiệu
- Ngữ nghĩa giao thức (Semantics)
  - Thông tin điều khiển
  - Kiểm soát lỗi
- Phân thời (timing)
  - Giải quyết vấn đề tương quan tốc độ
  - Tuần tự hóa thông tin

# Mô hình OSI

- OSI – Open System Interconnection



**Mô hình chuẩn 7 lớp mạng gồm các lớp:**

Ứng dụng

Trình diễn

Phiên (hội)

Vận chuyển

Mạng

Liên kết dữ liệu

Vật lý

# Mô hình OSI

- Mục đích thiết kế
  - Chia nhỏ logic mạng thành các phần nhỏ hơn
  - Cung cấp chuẩn giao tiếp
  - Duy trì đối xứng các chức năng
  - Cung cấp một chuẩn biệt ngữ cho nhà thiết kế

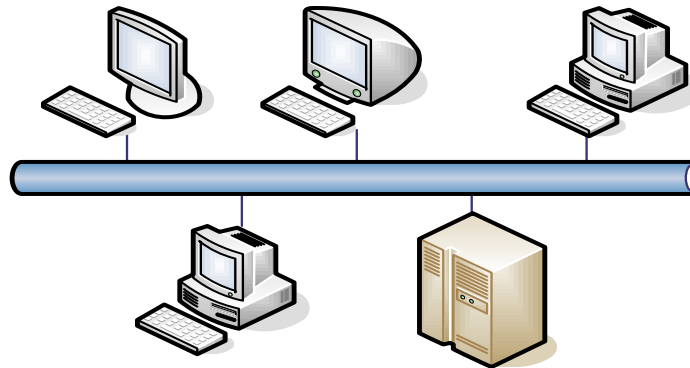
# Mô hình OSI

- **Lớp vật lý**

- Cung cấp cơ chế và các thủ tục cho các thực thể kết nối mạng
- Cung cấp đường kết nối vật lý
- Các kiểu kết nối có thể là: điểm-nối-điểm hoặc đa điểm kết nối
- Cấu trúc vật lý của thiết bị truyền dẫn gọi là topo mạng bao gồm các kiểu
  - Đường trục (BUS)
  - Vòng (RING)
  - Hình sao (STAR)
  - Hình mạng lưới (MESH)
  - Kiểu tế bào (CELLS)

# Topô mạng

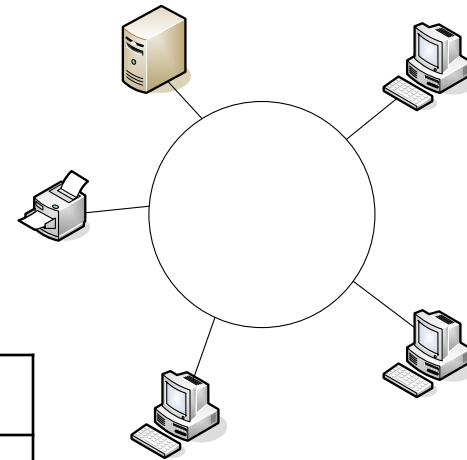
- Đường trục (BUS)



Lợi ích	Những lưu ý
Sử dụng các tiêu chuẩn đã được xây dựng sẵn	Khó trong việc tái cấu hình đặc biệt khi khoảng cách hoặc số lượng các kết nối đạt đến con số tối đa cho phép
Sử dụng ít vật liệu truyền dẫn hơn các topo mạng khác	Khó trong việc giải quyết những lỗi nếu xảy ra

# Topô mạng

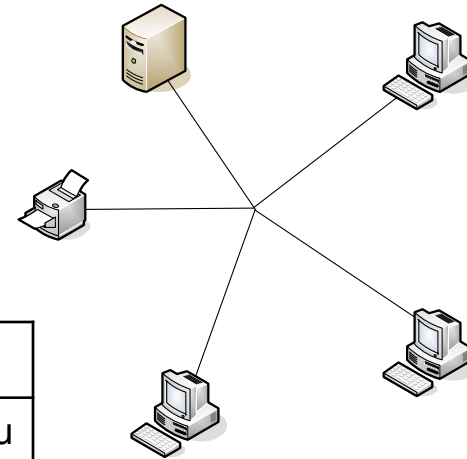
- Vòng (RING)



Lợi ích	Những lưu ý
Dễ dàng phát hiện nếu có lỗi của cáp truyền	Khó cài đặt và cấu hình hơn topo bus
Sử dụng các vòng lặp kép sẽ hỗ trợ tốt khả năng dung lỗi của hệ thống nếu cáp không bị đứt tại điểm lỗi.	Lỗi gián tiếp trên đường truyền hoặc vòng lặp đơn sẽ gây ra lỗi trên toàn mạng.

# Topô mạng

- Hình sao (STAR)

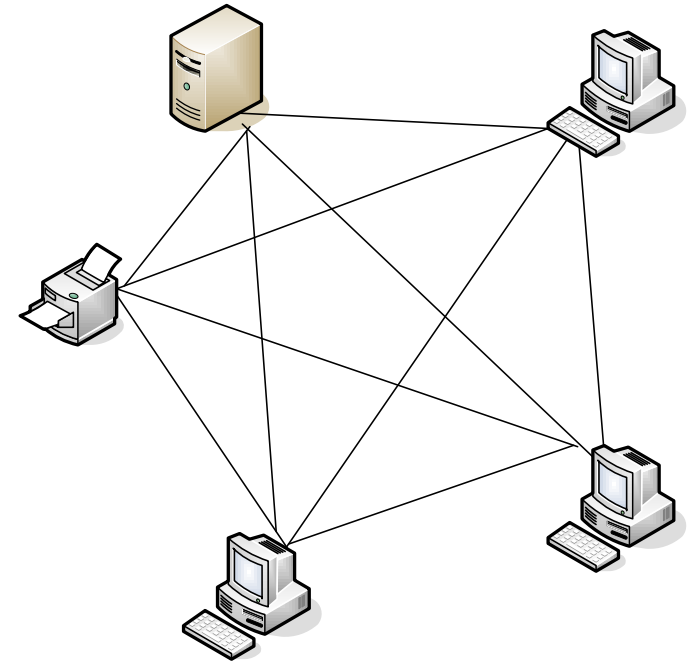


Lợi ích	Những lưu ý
Tương đối dễ cho việc cấu hình	Nhu cầu sử dụng nhiều cáp hơn các topo mạng khác
Dễ dàng xử lý các lỗi	Tương đối khó cài đặt
Lỗi đường truyền sẽ tự động bị cô lập mà không ảnh hưởng đến phân đoạn khác	

# Topô mạng

- Hình mạng lưới (MESH)

Lợi ích	Những lưu ý
Dễ dàng cô lập và giải quyết các lỗi	Rất khó cấu hình và cài đặt đặc biệt khi số lượng các trạm trong mạng là lớn
Có khả năng dung lỗi rất tốt bởi khả năng điều tiết thông qua việc định tuyến lại các đường truyền	

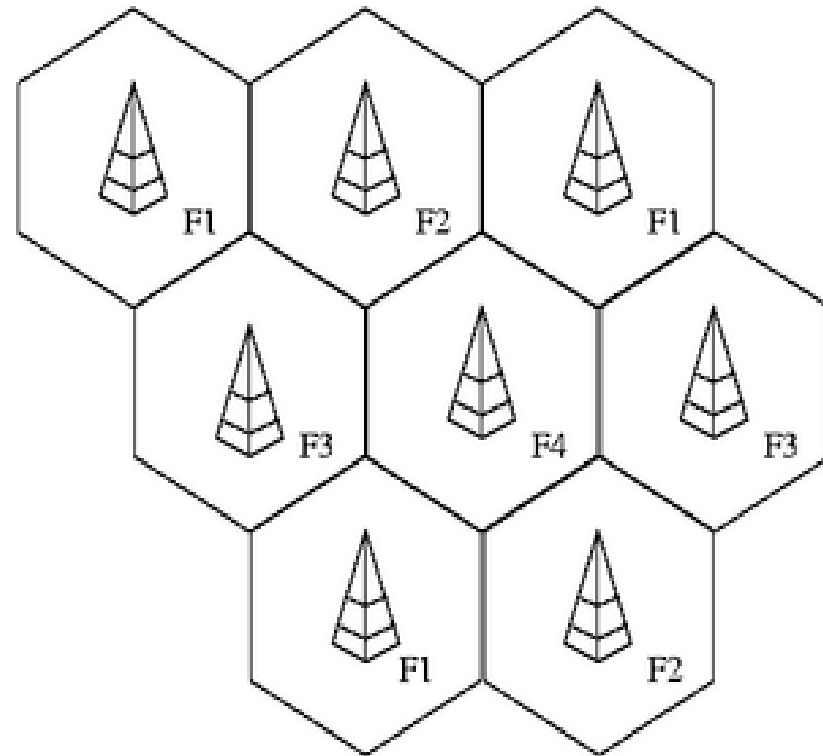




# Topô mạng

- Mạng tế bào (CELL)

Lợi ích	Những lưu ý
Tương đối dễ cài đặt	Các thiết bị sử dụng chung hub sẽ bị phụ thuộc rủi ro nếu hub xảy ra lỗi
Khả năng cô lập và sửa lỗi dễ dàng	



# Tín hiệu mạng

- Tín hiệu số
  - Thiết bị dùng thường đơn giản và rẻ tiền
  - Ít xảy ra xung đột
  - Hiện tượng suy hao nhiều so với tín hiệu tương tự (khi truyền trên cùng một chiều dài)
- Tín hiệu tương tự
  - Cho phép sử dụng các tín hiệu trộn để tăng hiệu suất băng thông đường truyền
  - Ít bị suy hao trên đường truyền
  - Thường xảy ra lỗi xung đột

# Mô hình OSI

- **Lớp liên kết dữ liệu (Data link)**

- Chức năng chính

- Cung cấp giao diện dịch vụ cho lớp trên
- Đóng khung dữ liệu
- Điều khiển luồng
- Kiểm soát lỗi
- Quản lí liên kết

- Điều khiển truy cập đường truyền (MAC – Media Acces Control)

- Điều khiển liên kết logic (LLC – Logical link control)

# Mô hình OSI

- **Lớp mạng (Network layer)**
  - Cấp địa chỉ
    - Cấp địa chỉ logic
    - Địa chỉ dịch vụ
  - Chuyển mạch
    - Chuyển mạch vòng
    - Chuyển mạch thông báo
    - Chuyển mạch gói
  - Phát hiện định tuyến
  - Lựa chọn định tuyến
  - Dịch vụ kết nối
    - Điều khiển luồng lớp mạng
    - Kiểm soát lỗi
  - Dịch vụ cổng kết nối

# Mô hình OSI

- **Lớp vận chuyển (Transport)**
  - Phân giải tên / địa chỉ
  - Phương pháp cấp địa chỉ
  - Phát triển phân đoạn mạng
  - Các dịch vụ kết nối

# Mô hình OSI

- **Lớp phiên (hội) – Session**

- Điều khiển giao tiếp

- Đơn công (Simplex)



- Song công (Duplex)



- Bán song công (Half-duplex)



- Quản trị phiên

- Khởi tạo kết nối

- Truyền dữ liệu

- Giải phóng kết nối

# Mô hình OSI

- Lớp trình diễn (Presentation layer)
  - Quản lý việc trình diễn và xử lý dữ liệu
  - Nén và mã hóa dữ liệu
- Lớp ứng dụng (Application layer)
  - Dịch vụ mạng
    - File, máy in, thư tín, CSDL và ứng dụng
  - Quảng bá dịch vụ
    - Tích cực
    - Thụ động
  - Sử dụng dịch vụ
    - Chặn các lời gọi từ hệ điều hành
    - Hoạt động từ xa
    - Sự cộng tác

# Nhóm các giao thức dùng trong mạng máy tính



# Mục tiêu bài học

- Mô tả bộ giao thức Netware
- Bộ giao thức IP
- Giao thức DNA
- Giao thức AppleTalk
- Bộ giao thức SNA
- Các giao thức SLIP, PPP và IEEE

# Bộ giao thức Netware

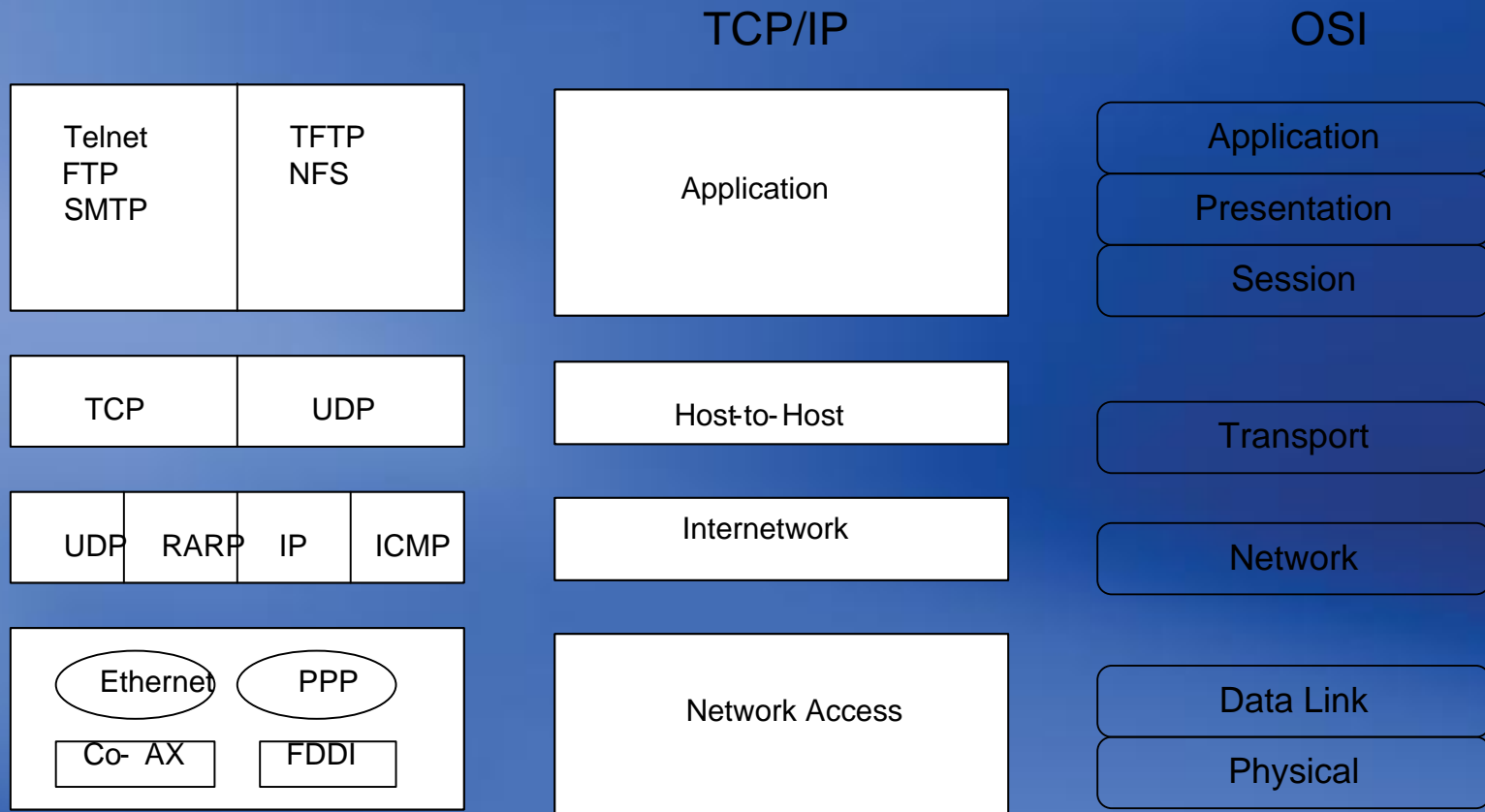
- MLID – Multiple Link Interface Driver
  - Kiểm soát vòng cạnh tranh
- LSL – Link support layer
  - Giao tiếp giữa MLID và giao thức phù hợp lớp trên
- IPX – Internetworking Package Exchange Protocol
  - Các dịch vụ mạng logic
  - Phi kết nối động
- RIP – Router Information Protocol
  - Tìm kiếm định tuyến
- NLSP – Netware Link Services Protocol
  - Tình trạng liên kết

# Bộ giao thức Netware

- SPX – Sequenced Packet Exchange Protocol
  - Xác nhận kết nối
  - Kết hợp và phân rã
  - Xếp thứ tự phân đoạn
  - Điều khiển lỗi
  - Điều khiển luồng kết nối điểm-điểm
- NCP – Netware Core Protocols
  - Xếp thứ tự phân đoạn
  - Điều khiển lỗi
  - Điều khiển luồng kết nối điểm-điểm
  - Mã ký tự
  - Cú pháp file
  - Bộ chuyển hệ điều hành
  - Cộng tác (phụ thuộc vào máy trạm Netware)
- SAP – Service Advertising Protocol
  - Truyền file
  - Tích cực

# Bộ giao thức TCP/IP

- Ánh xạ sang mô hình tham chiếu OSI



# Bộ giao thức TCP/IP

- Tóm tắt giao thức Internet
  - Giao thức điều khiển thư tín (ICMP)
  - Giao thức thông tin định tuyến (RIP)
  - Giao thức điều khiển truyền dẫn (TCP – Transmission Control Protocol)
  - Giao thức dữ liệu người dùng (UDP – User Datagram Protocol)
  - Giao thức phân giải địa chỉ (ARP - Address Resolution Protocol)

# Bộ giao thức TCP/IP

- Tóm tắt giao thức Internet (tiếp)
  - Giao thức truyền file (FTP – File transfer Protocol)
  - Giao thức truyền thư tín đơn giản (SMTP – Simple Mail Transfer Protocol)
  - Giả lập thiết bị đầu cuối từ xa (TELNET – Remote Terminal Emulation)
  - Hệ thống file mạng (NFS – Network File System)

# Kiến trúc mạng số

- (DNA – DIGITAL NETWORK ARCHITECTURE)
  - Tạo lập năm 1974 bởi DEC - Digital Equipment Corporation
  - Hiện tại sử dụng phiên bản 5
  - Chủ yếu sử dụng thiết bị sản phẩm DECNet

# Bộ giao thức AppleTalk

- Sản phẩm của hãng Apple bao gồm
  - TokenTalk
    - Dựa theo chuẩn mạng token-ring
  - EtherTalk
    - Dựa theo chuẩn mạng Ethernet
  - LocalTalk
    - Dựa trên CSMD/CD
  - RTMP: Giao thức duy trì bảng định tuyến
    - Thực thi chức năng tìm kiếm định tuyến
    - Tạo và duy trì bảng định tuyến trên AppleTalk



# Bộ giao thức AppleTalk

- ZIP (Zone information protocol):
  - Tổ chức logic các tên trên liên mạng
  - Sử dụng khi số lượng các máy cung cấp dịch vụ lớn
- ASP (AppleTalk Session Protocol):
  - Tối ưu hóa các chức năng dịch vụ mạng

# Kiến trúc mạng SNA

- Được phát triển bởi IBM năm 1974, còn được gọi là kiến trúc mạng ngang hàng nâng cao

## Các lớp của SNA

Các dịch vụ giao dịch  
Các dịch vụ trình diễn  
Điều khiển luồng dữ liệu  
Điều khiển truyền dẫn  
Điều khiển đường  
Điều khiển liên kết dữ liệu  
Điều khiển vật lý

# Kiến trúc mạng SNA

- Các giao thức chính:
  - Token-Ring
    - Được sử dụng như mô hình IEEE 802.5
  - SDLC – Synchronous Data Link Control
    - Được sử dụng một phần cứng cụ thể để giao tiếp với một đường thuê riêng hoặc một đường điện thoại
    - Hỗ trợ điểm-nối-điểm hoặc đa điểm kết nối
  - CICS – Customer Information Control System
  - SNADS – SNA Distribution Services
  - DIA – Document Interchange Architecture

# Sự tiến triển của SLIP và PPP

- SLIP (Serial Line Internet Protocol)
  - Giao thức đầu tiên hỗ trợ IP qua kết nối điện thoại
  - Đơn giản, không hỗ trợ đa giao thức trên cùng một đường truyền
  - SLIP không phải là một chuẩn tuyệt đối
- PPP – Point-to-Point Protocol
  - Cấp địa chỉ động
  - Hỗ trợ nhiều giao thức trên cùng một đường truyền
  - Mật khẩu truy cập
  - Kiểm soát lỗi

# Các giao thức IEEE 802.X

- **IEEE 802.2**
- **IEEE 802.3**
- **IEEE 802.4**
- **IEEE 802.5**
- **IEEE 802.6**
- **IEEE 802.9**
- **IEEE 802.11**
- **IEEE 802.12**

# FDDI, ISDN và ATM

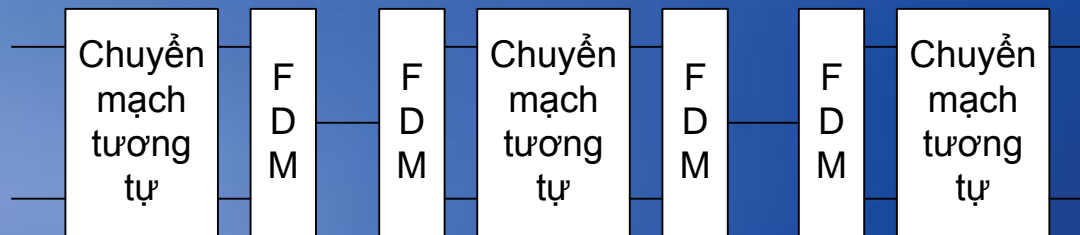
- FDDI – Fiber Distributed Data Interface
  - Chuẩn chỉ rõ nguyên tắc chuyển việc chiếm đường truyền theo chuẩn vòng trên đường cáp quang tốc độ 100Mbps
  - Sử dụng phương pháp truy cập đường truyền “chuyển thẻ” (token-Passing)
  - Hỗ trợ tốc độ truyền dữ liệu cao và khoảng cách lớn
  - Sử dụng “trạm kết nối kép” để phân tải

# FDDI, ISDN và ATM

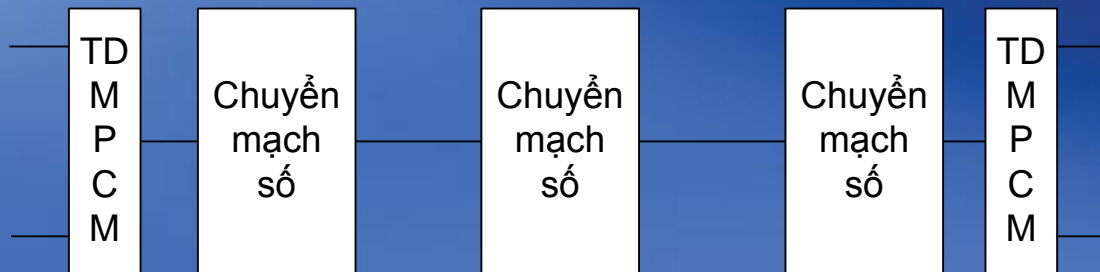
- Giao thức X25
  - Nhóm giao thức mạng diện rộng
  - Đặc tả kỹ thuật cho việc truyền dữ liệu giữa máy tính kết nối vật lý với mạng chuyển mạch gói
  - Hỗ trợ cả các vòng chuyển mạch ảo và các vòng mạch cố định

# FDDI, ISDN và ATM

- ISDN (Integrated Services Digital Network)
  - ISDN dựa vào Mạng tích hợp số (IDN – Intergrated Digital Network)



A. Không tích hợp



B. Tích hợp



# FDDI, ISDN và ATM

- Các dịch vụ cung cấp bởi ISDN
  - Telephone
  - Kết nối mạng
  - Facsimile
  - Teletext
  - Video Text

# FDDI, ISDN và ATM

- ATM – Asynchronous Transfer Mode (Chế độ truyền không đồng bộ)
  - Chạy trên lớp vật lý
  - Hỗ trợ các giao thức lớp trên
  - Xác định bốn lớp của dịch vụ dựa trên hoạt động giao thức lớp trên:
    - Tốc độ bất biến khi truyền các bit
    - Cấp phát các khoảng trễ nhất định khi truyền
    - Dữ liệu định hướng kết nối
    - Dữ liệu phi kết nối



**Giao thức chuyển mạch gói X25**

## **X.25 - Giao thức mạng chuyển mạch gói**

- 1. Mở đầu
- 2. X.25 (84) cấp 1 - cấp vật lý
- 3. X.25 (84) cấp 2 - cấp tuyến số liệu
  - ✓ 3.1. Thẻ thức khung của LAPB
  - ✓ 3.2. Các kiểu khung LAPB
  - ✓ 3.3. Các trường (vùng) N(R) và N(S)
  - ✓ 3.4. Bit P
  - ✓ 3.5. Thao tác cấp tuyến số liệu
  - ✓ 3.6. Các tham số hệ thống
- 4. Cấp X.25 thứ 2 - Một số gợi ý thực tế
- 5. Cấp X.25 (84) cấp 3 - cấp mạng (lớp mạng)
  - ✓ 5.1. Khuôn mẫu gói cấp mạng
  - ✓ 5.2. Các kiểu gói cấp mạng
  - ✓ 5.3. Các địa chỉ dây cấp mạng
  - ✓ 5.4. Trường mã nhận dạng khuôn mẫu
  - ✓ 5.5. Cung đoạn tái khởi động
  - ✓ 5.6. Thiết lập các cuộc gọi thử
  - ✓ 5.7. Cung đoạn chuyển giao tin
  - ✓ 5.8. Trường mã dịch vụ
- 6. Cấp X.25 - 3 - Một số hướng dẫn thực tế
- 7. X.75
  - ✓ 7.1. Cấp vật lý của X.75
  - ✓ 7.2. Cấp tuyến của X.75

### 7.3. Cấp mạng của X.75

#### ❖ Tài liệu tham khảo:

1. Nguyễn Hồng Sơn, Cơ sở kỹ thuật chuyển mạch và tổng đài.

2. Dương Văn Thành, Cơ sở kỹ thuật chuyển mạch, Học viện công nghệ bưu chính viễn thông, 2000.
3. Nguyễn Văn Điềm, Giáo trình chuyển mạch kỹ thuật số.
4. Nguyễn Duy Nhật Viễn, Công nghệ chuyển mạch.
5. Hoàng Trọng Minh, Hệ thống chuyển mạch-Switching Systems.
6. Mai Văn Quý, Nguyễn Hữu Kiên, Nguyễn Văn Giáo: Kỹ thuật chuyển mạch.

## ***X.25. Giao thức mạng chuyển mạch gói***

### **1 Mở đầu**

Giao thức này là giao thức CCITT X.25 (84), nó là giao thức quan trọng nhất trong các giao thức chuyển mạch gói. Chữ số 84 sau X.25 thể hiện tài liệu khuyến nghị X.25 xuất bản năm 1984. CCITT xuất bản 4 năm một lần. Điều đó không có nghĩa là giao thức này thay đổi nhiều tới 4 năm một lần. X.25 (80) xuất bản vào năm 1980 là cơ sở của tất cả các khuyến nghị X.25 đã lỗi thời. Điều này chủ yếu là do các mạng quốc gia (ví dụ luồng chuyển mạch gói BT) đã tiêu chuẩn hoá theo X.25 (80).

X.25 (84) có một số đặc tính mới không trình bày ở X.25 (80). Các đặc tính mới này đã được đưa vào mọi trường hợp để trợ giúp cho dịch vụ mạng ghép nối định hướng cho các hệ thống mở (càng về sau càng nhiều). Hầu hết những điều bổ sung mới ở X.25 (84), nó giải thích vì sao giao thức này lại được chọn.

IOS còn có kiểu X.25 (84) cấp 3. Nó được IOS công bố là ISO/DIS 8202 và BSI công bố là DD117. Kiểu giao thức ISO này trên cơ bản giống kiểu CCITT, nhưng khác là nó cho phép thao tác từ DTE tới DTE ở mức gói còn CCITT chỉ quan tâm tới thao tác giữa DTE và DCE.

Cuối cùng chúng ta sẽ xem xét qua X.75. Đây là một dạng của X.25 sử dụng cho các mạng liên kết X.25 và bao gồm một số các tính năng đặc biệt và các thể thức gói khác.

### **2. X.25 (84) cấp 1 - Cấp vật lý**

Cấp vật lý của giao thức này xác định các vấn đề như báo hiệu điện và kiểu các bộ đầu chuyển được sử dụng. Cho phép hai kiểu giao tiếp chính. Đó là X.21 và X.21 bis. Khuyến nghị này cũng cho phép giao tiếp nối tiếp V khi cần.

### 3. X.25 (84) Cấp 2 - Cấp tuyến số liệu

Nói ngắn gọn là : cấp 2 cung cấp một đường thông tin điều khiển dòng, không có lỗi giữa hai đầu cuối của một tuyến liên lạc. Nó tạo điều kiện cho các cấp cao hơn làm việc mà không quản ngại về việc số liệu bị sai lạc và cho cấp dưới để điều khiển luồng. Giao thức cấp tuyến sử dụng một số khái niệm từ giao thức HDLC (giao thức điều khiển tuyến số liệu cấp cao).

Có hai kiểu giao thức X.25 cấp 2: LAP và LAPB. LAP có nghĩa là: thể thức xâm nhập tuyến (Link access procedure). Còn LAPB có nghĩa là thể thức xâm nhập tuyến có cân bằng (Link access procedure balanced). LAPB hoàn thiện hơn LAP một ít và là kiểu mà hầu hết mọi người sử dụng.

Chi tiết hơn ta có hai kiểu giao thức LAPB. Dạng chung nhất của LAPB là kiểu SLP - thể thức tuyến đơn (Single Link Procedure), tức là giao thức giữa DTE và DCE chỉ dùng một tuyến thông tin. Một kiểu mới của LAPB cũng được giới thiệu ở X.25 (84) là kiểu MLP thể thức đa tuyến. Nó cho phép sử dụng đa tuyến liên lạc giữa DTE và DCE. Nếu một trong các tuyến có sự cố thì các tuyến khác được tuyến dụng mà không bị mất số liệu. Điều này cho phép phân tải giữa các tuyến ghép và tự động khắc phục lỗi cho một hoặc nhiều tuyến.

#### 3.1. Thể thức khung của LAPB

Đơn vị tin ở giao thức LAPB là "khung". Hình 1 trình bày cấu trúc của các khung LAPB. Trường F chứa 1 byte cờ. Khi các khung chưa được phát đi, các byte cờ liên tục được chuyển đi (byte mẫu nhị phân 01111110).

Trường "A" chứa địa chỉ gói tin. Vùng này có thể chứa hoặc 00000011 (địa chỉ A) hoặc 00000001 (địa chỉ B). Việc sử dụng địa chỉ A và B sẽ được mô tả sau này. Các trường C là trường điều khiển khung. Nó được sử dụng để xác định khung chứa những gì. Chú ý rằng ở hình 1.a và 1.b trường điều khiển luôn dài 8 bits, trong khi đó ở hình 1.c và 1.d, trường điều khiển này có thể dài đến 8 đến 16 bits. Đó là do có sự thay đổi thêm của giao thức mà hiện chưa được nhắc tới. Kiểu LAPB chuẩn này cho phép kích thước cửa sổ tối đa (xem chương 2 dành để giải thích các cửa sổ giao thức) của 7 số liên tiếp từ 0 tới 7. Một vùng 3 bit cần cho công việc này, nó ghép khớp trong trường điều khiển. Có thể xảy ra trường hợp kích thước cửa sổ lớn hơn sẽ hay hơn. Để có điều đó kiểu LAPB mở rộng được xác định, nó có thể trợ giúp các kích thước cửa sổ tới 127. Khi đó cần phải có trường 7 bits. Khi trường điều khiển có độ dài thay đổi thì nhiều điều khoản của X.25 không trợ giúp được cho phương thức làm việc mở rộng này.

Ở trường hợp hình 1.a và 1.b chỉ có một trường "I" được dùng để chuyển tin của giao thức cấp cao hơn các gói X.25 cấp 3.

Trường FCS chứa dãy kiểm tra khung. Nó được sử dụng để bộ thu khung kiểm tra để đảm bảo nó đã thu mà không có lỗi. Thiết bị phát khung đưa thêm vào FCS, trị số của nó được tính toán theo nội dung khung.

## Mai Xuân Hoàn

Cuối cùng có một trường "F" khác. Cờ này xác định điểm cuối của khung. Hoàn toàn có khả năng một khung khác tiếp theo ngay sau cờ này, vì vậy chỉ có một cờ giữa các khung. Có một vấn đề nảy sinh từ cấu trúc khung này. Giả sử nội dung của khung giữa các trường cờ có kiểu bit 01111110, là kiểu bit cờ. Vì cờ đánh dấu điểm cuối của khung, vì vậy có thể khung không thu được chính xác.

Để khắc phục vấn đề này, số liệu được phát đi theo cách riêng. Nếu nội dung của khung chứa 5 hoặc hơn 5 bits 1 ở một dãy thì máy phát sẽ bổ sung vào một bit 0 sau 5 bit 1. Điều này đảm bảo không bao giờ xảy ra 6 bit 1 liên tiếp ở giữa của một khung. Máy thu nhận biết được điều máy phát đã thực hiện, nếu nó thấy bit 0 theo sau 5 bit 1 thì nó biết rằng bit 0 này cần bị loại bỏ đi vì nó đã được máy phát đưa thêm vào. Kỹ thuật này được coi như kỹ thuật chèn bit.

Thứ tự bit phải	12345678	12345678	12345678	16 tới 1	12345678
	Cờ	Địa chỉ	Điều khiển	FCS	Cờ
	F 01111110	A 8 bits	C 16 bits	FCS 16 bits	F 01111110

Thứ tự bit phải	12345678	12345678	12345678		16 tới 1	12345678
	Cờ	Địa chỉ	Điều khiển	Thông tin	FCS	Cờ
	F 01111110	A 8 bits	C 16 bits	INFO N bits	FCS 16 bits	F 01111110

Thứ tự bit phải	12345678	12345678	1 tới *)	16 tới 1	12345678
	Cờ	Địa chỉ	Điều khiển	FCS	Cờ
	F 01111110	A 8 bits	C *) bits	FCS 16 bits	F 01111110

Thứ tự bit phải	12345678	12345678	1 đến *)	16 tới 1	12345678
	Cờ	Địa chỉ	Điều khiển	FCS	Cờ

## Mai Xuân Hoàn

	F 01111110	A 8 bits	C *) bits	FCS 16 bits	F 01111110
--	---------------	-------------	--------------	----------------	---------------

\*) 16 đối với thể thức khung chứa địa chỉ dãy liên tiếp, 8 cho thể thức khung không chứa địa chỉ dãy liên tiếp.

Hình1. Các thể thức khung

### 3.2. Các kiểu khung LAPB

Giao thức LAPB xác định một kiểu khung chính thống được dùng để chuyển tin theo giao thức LAPB và chuyển tin theo giao thức cấp cao hơn.

Kiểu khung này được xác định ở trường điều khiển.

Bảng 1 trình bày các loại trường điều khiển hợp thức ở LAPB. Tùy theo phương thức LAPB đã đưa ra có hai dạng khác nhau của các kiểu khung. Các chức năng khung vẫn giữ nguyên, chỉ các chức năng được đưa ra mới được mô tả ở bảng này.

Thể thức	Lệnh	Đáp ứng	Mã hoá			
			0	N(S)	P	N(R)
Chuyển tin	I (Tin)		0	N(S)	P	N(R)
Giám sát	RR (sẵn sàng thu)	RR (sẵn sàng thu)	1.....0.....0.....0		P/F	N/R
	RNR (chưa sẵn sàng thu)	RNR (chưa sẵn sàng thu)	1.....0.....1.....0		P/F	N/R
	REJ (không chấp nhận)	REJ (không chấp nhận)	1.....0.....0.....1		P/F	N/R
Không đánh số	SABM (thiết lập phương thức cân bằng không đồng bộ)	...	1.....1.....1.....1		P	1...0...0
	DISC Cắt tuyến nối (giải toà)	...	1.....1.....0.....0		P	0...1...0
		DM (phương thức không đầu nối)	1.....1.....1.....1		F	0...0...0
		UA (xác nhận không đánh số)	1.....1.....0.....0		F	1...1...0
	FRMR (không chấp nhận khung)		1.....1.....1.....0		F	0...0...1

**Bảng 1 Thể thức trường điều khiển**



Chủ yếu có hai kiểu khung: Khung lệnh và khung đáp ứng. Khung đáp ứng được phát để xác nhận công việc thu một lệnh. Ví dụ như các khung *I* là các khung lệnh. Sau khi thu được một khung *I* hay nhiều khung *I*, một đáp ứng cần được chuyển đi để xác nhận rằng, khung hoặc các khung đã thu được chính xác. Chú ý rằng, các khung *S* có thể là các khung lệnh hoặc khung đáp ứng (trả lời). Chúng được sử dụng làm vai trò gì tùy theo điều kiện cụ thể.

Các lệnh và các đáp ứng được phân biệt nhờ giá trị ở trường *A* của khung. Lưu là trường này có thể chứa địa chỉ của *A* hoặc địa chỉ *B*. Đáp ứng được phát cho một lệnh thu được luôn có cùng trường *A* vì nó là của lệnh này. Nếu DCE phát lệnh thì dùng địa chỉ *A*. Nếu DTE phát lệnh thì dùng địa chỉ *B*.

Thực ra ở cấp tuyến số liệu thì đây là sự khác nhau chủ yếu giữa *DTE* và *DCE*.

Bây giờ đến lượt mô tả các kiểu khung khác nhau. Khung "*I*" là "khung tin". Nó được dùng để chuyển tin cho giao thức cấp cao hơn.

Các khung *S* gọi là các khung giám sát. Có 3 kiểu khung *S*: *RR* (máy thu sẵn sàng làm việc), *RNS* (máy thu chưa sẵn sàng làm việc) và *REJ* (khung phát lại). Các khung này liên quan tới công việc điều khiển luồng cho khung *I* và khắc phục lỗi tuyến thông tin do hỏng khung.

Các khung "*U*" gọi là các khung không được đánh số. Chúng được gọi như vậy vì chúng không chứa địa chỉ dãy. Các khung này được dùng để khởi xướng, chọn tuyến (*SABM*, *SABME*, *DISC*, *DM*, và *UA*) và báo cáo những sự vi phạm giao thức (*FRMR*).

Lệnh *SABM* (Set Asynchronous Balanced Mode) thiết lập phương thức cân bằng không đồng bộ và *SABME* (Set Asynchronous Balanced Mode Extended) thiết lập phương thức cân bằng không đồng bộ mở rộng) dùng để thiết lập tuyến vào trạng thái chuyển tin (tức là tới trạng thái cao). Sự khác nhau duy nhất giữa hai lệnh này là: *SABM* đòi hỏi phương thức làm việc thông thường (kích cỡ cửa sổ tối đa là 7) còn *SABME* đòi hỏi phương thức làm việc mở rộng (kích cỡ cửa sổ tối đa 127).

Khung lệnh *DISC* (giải toả) dùng để đưa tuyến về trạng thái thấp (dưới) và như vậy ở chừng mực nào đó nó ngược với các lệnh *SABM* và *SABME*.

Đáp ứng *DM* (phương thức giải toả) dùng để trả lời cho *SABM* hoặc *SABME* đã thu được nếu máy phát *DM* không muốn đưa tuyến vào trạng thái chuyển tin.

Đáp ứng *UA* (xác nhận không đánh số) dùng để khẳng định lệnh *DISC* hoặc *SABM* đã thu được.

Đáp ứng *FRMR* (không chấp nhận khung) dùng để chỉ thị lệnh sau cùng hoặc đáp ứng sau cùng không hợp lệ về mặt nào đó. *FRMR* mang thông tin mô tả lý do.

### 3.3 Các trường (vùng) *N (R)* và *N (S)*

Cụm *N(R)* do bộ phát khung số liệu sử dụng để báo cho máy thu số thứ tự của khung tin tiếp theo mà máy thu đang đợi. Các khung *RR* và *RNR* dùng cụm này để khẳng định công việc thu các khung tin có thứ tự tới *N(R)*. Khung *REJ* dùng để yêu cầu phát lại các khung tin có số thứ tự bắt đầu từ *N(R)*. Cụm *N(S)* dùng để chỉ thị số thứ tự của một khung tin.

### 3.4 Bit *P*

Bit *P* (hoặc bit đầu/cuối) được sử dụng chung để chỉ thị một khung đã được phát lại.

## Mai Xuân Hoàn

Khi sử dụng trong một lệnh thì bit này gọi là bit đầu, còn khi sử dụng trong một đáp ứng thì nó gọi là bit cuối. Khi một đáp ứng được tạo ra cho một lệnh thì bit cuối phải bằng bit đầu của lệnh.

Tổng quát, lúc đầu phát một lệnh, bit đầu là không (xoá). Khi lệnh đã được phát đi, cần có một đáp ứng. Nếu không thu được đáp ứng trong một khoảng thời gian xác định thì lệnh sẽ được phát lại, lần này bit đầu là lập.

Khoảng thời gian quy định, trong đó phải thu được một đáp ứng gọi là T1. Đó là một trong các tham số để cấu hình các tuyến đặc biệt. Mục các tham số hệ thống sau này sẽ đề cập nhiều hơn về vấn đề này.

Các bit của khối tin

- Trường điều khiển khung không được chấp nhận là cụm mã điều khiển của khung thu, đã gây ra sự từ chối khung.

- V(S) là biến số trạng thái phát hiện thời ở DCE hoặc DTE báo cáo trạng thái từ chối (bit 10=bit thứ tự thấp).

- CIR thiết lập một chỉ thị khung đã bị từ chối là một đáp ứng, còn RIS thiết lập 0 chỉ thị khung đã bị từ chối là một lệnh.

- V(R) là biến số trạng thái thu hiện thời ở DCE hoặc DTE báo cáo trạng thái từ chối (bit 14=bit thứ tự thấp).

- W ở trạng thái 1 chỉ thị trường điều khiển đã thu được và đã quay về các bit từ 1 tới 8 không được xác định hoặc không được thực hiện.

- X ở trạng thái 1 chỉ thị trường điều khiển đã thu được và đã quay về các bit từ 1 tới 8 bị coi là không hợp lệ do khung chứa trường tin không cho phép ở khung này hoặc khung này là một khung giám sát hay một khung không được đánh số có độ dài không chuẩn xác. Bit W cần ở trạng thái 1 phối hợp với bit này.

- Y ở trạng thái 1 chỉ thị trường tin đã thu được vượt quá dung lượng thiết lập cực đại.

- Z ở trạng thái 1 chỉ thị trường điều khiển đã thu được và đã quay về các bit từ 1 tới 8 chứa N(R) không hợp lệ.

12345678	9	10 11 12	13	14 15 16	17	18	19	20	21	22	23	24
Trường điều khiển khung không chấp thuận	0	V(S)	C/R	V(R)	W	X	y	Z	0	0	0	0

**Hình 2 Trường tin của FRMR**

### 3.5. Thao tác cấp tuyến số liệu

Có hai cung đoạn thao tác chính: cung đoạn lập tuyến và cung đoạn chuyển tin. Theo đặc điểm hình thái của giao thức thì hai cung đoạn này được chia nhỏ thành một số lớn các trạng thái. Vì vậy giao thức này được xác định theo bảng trạng thái. Điều này có nghĩa là nếu biến cố này xảy ra theo trạng thái này thì làm như vậy và chuyển sang trạng thái mới đó. Thực ra các bảng trạng thái chỉ cần cho người thực hiện giao thức, vì vậy chúng ta không quan tâm tới các bảng trạng thái ở đây.

Hai cung đoạn của tuyến số liệu sẽ được mô tả dưới đây. Thao tác đối với DTE cũng giống như đối với DCE. Vì vậy thuật ngữ DXE được dùng cho cả DTE và DCE.

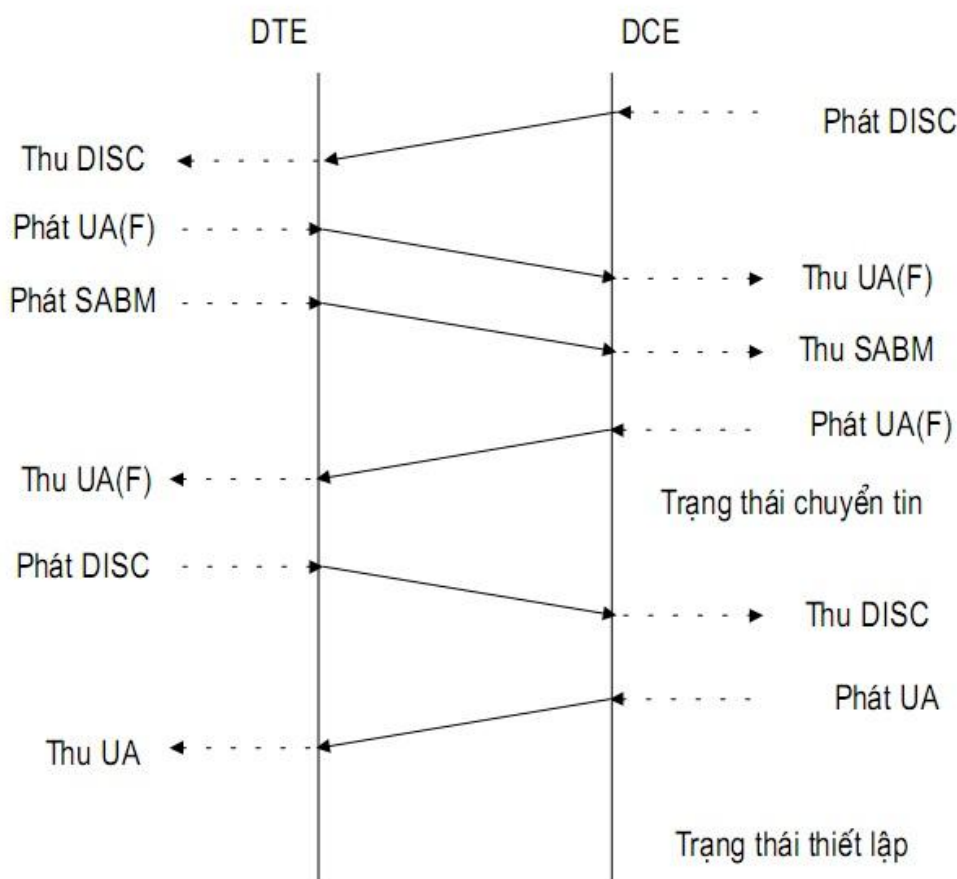
#### • *Cung đoạn lập tuyến*

Khi một DXE mới được khởi động, đó là cung đoạn lập tuyến. ở trạng thái này phổ biến là để phát DISC theo chu kỳ. Điều này chủ yếu để nói "tôi đang vào cuộc". Nếu không được trả lời trong khoảng  $T_1$  thì DISC được phát lần nữa nhưng có thiết lập bit P. Nó được viết là DISC (P). Hình 3 mô tả trạng thái này.

Nếu một DXE thu một DISC hoặc DISC(P) và muốn khởi động tuyến, nó trả lời bằng một UA hoặc UA(F) (tức là một UA có lập một bit cuối). DXE thu UA hoặc UA(F) này sẽ chờ một khoảng thời gian là  $T_3$ . Nếu một SABM hoặc SABME thu được trong khoảng thời gian này thì đáp ứng UA được phát đi và tuyến số liệu chuyển sang cung đoạn chuyển tin. Nếu một SABM(P) hoặc SABME(P) thu được thì một UA(F) được phát đi và tuyến chuyển sang cung đoạn chuyển tin. Lưu ý rằng nếu sự chậm trễ hơn xảy ra thì điều này có nghĩa là một SABM hoặc SABME đã bị mất vì sự thiết lập bit đầu chỉ thị rằng khung đã được phát đi.

#### • *Cung đoạn chuyển tin*

Hình 4 mô tả quá trình trình thiết lập một tuyến để đưa tuyến vào cung đoạn chuyển tin, tiếp theo là đưa tuyến quay về cung đoạn lập tuyến. ở cung đoạn chuyển tin I, các khung RR, RNR và REJ được dùng để điều khiển công việc chuyển giao số liệu giao thức cấp cao hơn qua tuyến. Nếu thu được một khung I chuẩn xác và DXE có thể tiếp nhận nữa thì nó trả lời cho khung I này bằng một khung đáp ứng RR. Nếu DXE không thể tiếp nhận nữa thì nó trả lời bằng một đáp ứng RNR, báo cho DXE kia rằng hiện nó bận và không thể tiếp nhận thêm số liệu ở thời điểm này. Đáp ứng REJ dùng để yêu cầu phát lại một hoặc nhiều khung I đã bị DXE nghi là mất (có thể bị loại bỏ do một lỗi FCS sinh ra trong khi thu).



**Hình 4. Thiết lập tuyến sau khi giải toả**

Các khung RR, RNR và REJ được dùng để trả lời khung I là các đáp ứng. Dạng lệnh của các khung RR, RNR và REJ dùng để hỏi DXE kia về trạng thái hiện tại của nó hoặc báo cho nó nếu trạng thái của DXE này đã thay đổi. Khi được sử dụng là lệnh thì các khung RR, RNR, và REJ luôn có sự thiết lập bit đầu. Vì vậy các đáp ứng tạo ra ở bên thu luôn được gắn bit cuối.

Để xem xét nó làm việc ra sao, giả thiết rằng một DXE đã trả lời cho một khung tin bằng một đáp ứng RNR do nó không thể tiếp nhận số liệu nữa. Khi lại có thể thu số liệu, nó có thể phát một lệnh RR(P) cho DXE kia, thông báo cho nó về trạng thái mới. Sau đó DXE thu có thể trả lời bằng một đáp ứng RR(F), RNR(F) hoặc REJ(F), (tùy thuộc vào trạng thái của nó) và lại tiếp tục phát các khung I. Điều này thể hiện ở hình 5. Cả DTE lẫn DCE có thể chuyển tuyến sang trạng thái thiết lập nhờ phát đi một lệnh DISC vào bất cứ lúc nào. Nếu một DXE đòi hỏi phục hồi tuyến thì nó phát đi lệnh SABM hoặc SABME. Cũng thế, điều này có thể xảy ra ở bất cứ lúc nào.

Phía thu phát một UA để trả lời và tuyến lại trở về cung đoạn chuyển tin.

• *Trạng thái từ chối khung*

Trạng thái từ chối khung được đưa vào khi thu một khung không hợp lệ. Điều đó có nghĩa là một khung đã không được thu nhận cùng với trường địa chỉ A hoặc B ở trường A và

không có lỗi FCS, nhưng nội dung của khung vẫn không chuẩn xác hoặc không tương ứng đối với trạng thái của phía máy thu. Hiện nhiên đây là trạng thái tương đối trầm trọng, có thể biểu hiện sự vi phạm giao thức và cần phải tái lập tuyến. Mặc dù tuyến có thể được tái lập ngay nhờ phát đi lệnh SABM hoặc SABME, nhưng cũng không thể báo cho DXE kia vì sao tuyến lại phải tái khởi động. Vì vậy khi một DXE thu một khung không hợp lệ thì nó phát một đáp ứng FRMF để báo cho DXE kia biết cái gì bị sai. Chủ yếu đây là một sự luận tội: "bạn đã phát cho tôi một khung bị sai và vì sao vậy".

Đáp ứng FRMF là một bit đặc biệt bởi vì nó là đáp ứng duy nhất có thể phát đi để trả lời một đáp ứng - tốt, có phải không? Ngay ở trạng thái từ chối khung, tuyến có thể được tái khởi động bằng một lệnh SABM hoặc SABME.

### 3.6. Các tham số hệ thống

Các tham số hệ thống là các tham số cấu hình, nó xác định các khía cạnh nào đó của sự thao tác cấp tuyến số liệu.

Đại lượng  $T_1$  là khoảng thời gian mà máy phát khung lệnh chờ một đáp ứng trước khi lại phát đi một lệnh có gắn bit đầu. Đôi khi gọi đây là khoảng tái thử.  $T_1$  phải lớn hơn thời gian dùng để phát một khung có độ dài cực đại và nhận một đáp ứng cho khung này, nó có thể là một khung cực đại. Nó tùy thuộc vào tốc độ phát các bit theo tuyến thông tin và khoảng trễ xử lý ở máy thu.

Còn có khoảng định thời nữa, đó là  $T_2$ , nó được xác định như là thời gian cực đại cần dùng trước khi máy thu thu một khung và phát đi một khung xác nhận việc thu khung này. Nó luôn ngắn hơn  $T_1$ . Điều này thực tế thích hợp để phát đi một khung xác nhận việc thu một khung càng sớm càng tốt.

Khoảng định thời gian  $T_3$  xác định một DXE phải chờ bao lâu đối với lệnh thiết lập tuyến trước khi bắt đầu phát đi các DISC ở cung đoạn lập tuyến. Giá trị này là  $T_1 \times N_2$ .

$N_2$  là số lần cực đại để một khung lệnh được phát lại trước khi tuyến được tái khởi động.

Thực chất nếu  $T_1$  đã hết  $N_2$  lần thì máy phát từ bỏ và tái khởi động tuyến bởi SABM hoặc SABME. PSS dùng giá trị 20 đối với  $N_2$ . Các mạng khác nhau có thể quy định các giá trị  $N_2$  khác nhau nhưng chúng hoàn toàn giống PSS.

$N_1$  là số bit cực đại có thể có trong một khung I. Nó bao gồm các cụm IA, C, I, và FCS. Ví dụ nếu kích cỡ cụm I cực đại cho một tuyến là 128 bytes thì  $N_1$  sẽ là 1064. Tham số hệ thống  $k$  là số lượng cực đại của các khung I được đánh số tuần tự mà một DXE có thể phát đi nhưng không được xác nhận ở bất cứ lần nào (tức là kích cỡ cửa sổ). PSS xác định giá trị  $k$  là 7 vì nó không hỗ trợ cho phương thức làm việc mở rộng.

## 4. Cấp X.25 thứ 2 - Một số gợi ý thực tế

Điều quan trọng nhất xảy ra khi thử nối hai thiết bị với nhau là tìm một sợi cáp để làm việc và cung cấp đồng hồ cho cấp tuyến số liệu (tức là các vấn đề cấp vật lý). Sau đó là tới cấp tuyến số liệu. Một DTE chỉ trao đổi với một DCE; các DTE khác sẽ không trao đổi với các DTE và các DCE cũng không trao đổi với các DCE. Vì vậy điều đầu tiên cần làm để đảm

bảo một thiết bị là một DCE, thiết bị khác là một DTE. Ngay sau khi điều đó đã được thực hiện thì cáp tuyến số liệu hoàn thành mà không còn sự rắc rối nào nữa.

Tham số cấu hình quan trọng nhất là giá trị định thời  $T_1$ . Nó cần được thiết lập chuẩn xác cho tốc độ mà tuyến được cấp đồng hồ nhịp. Đừng quên rằng  $T_1$  cần được thiết lập ở cả DCE và DTE.

Điều gì xảy ra khi  $T_1$  thiết lập không chuẩn xác? Nếu  $T_1$  không đủ lớn, có thể không đủ thời gian để phát đi một khung dài và xác nhận nó trước khi  $T_1$  hết hạn. Điều đó dẫn tới một loạt sự phát lại khung không cần thiết và làm cho việc truyền số liệu kém hiệu quả.

Nếu  $T_1$  quá lớn thì sự chậm trễ nghiêm trọng có thể xảy ra khi cần phải phát lại khung. Cả hai trường hợp đều không xảy ra nguy hiểm nhưng thực tế để tạo lập tuyến có hiệu suất tốt thì đòi hỏi phải thiết lập  $T_1$  chuẩn xác.

Tham số cấu hình quan trọng khác cho cáp tuyến là  $N_2$ . Đó là số lần phát lại một khung trước khi tuyến số liệu được tái lập. Giá trị 20 là hợp lý.  $N_2$  không phụ thuộc vào tốc độ của tuyến.

$T_1$  và  $N_2$  có quan hệ sao cho  $T_1 \times N_2$  là thời gian cực đại cần để phục hồi tuyến số liệu từ một sự cố nghiêm trọng. Nếu giá trị này quá lớn thì thời gian này có thể mất nhiều phút. Cho dù điều này không thường xuyên xảy ra.

Nếu tuyến thông tin giữa *DTE* và *DCE* có chất lượng tốt thì lỗi *FCS* hầu như không xảy ra. Nếu tuyến bị nhiễu loạn sẽ xảy ra nhiều lỗi *FCS*. Các khung bị lỗi có thể được đưa qua công việc kiểm tra *FCS*, *FCS* không làm gì khác, nó chỉ từ chối hầu hết các khung bị lỗi. Đây là một trạng thái nghiêm trọng vì sự huỷ hoại của số liệu cấp mạng có thể xảy ra mà không được phát hiện bởi các giao thức cấp mạng cao hơn.

Cho dù việc này không trầm trọng thì lỗi *FCS* thường xuyên sẽ làm cho tuyến số liệu kém hiệu quả vì phải phát lại khung nhiều lần như một hậu quả.

### **5. Cấp X.25 (84) cấp 3 - cấp mạng (lớp mạng)**

Cấp X.25 thứ 2 tạo ra phương thức để chuyển tin giao thức cấp cao hơn (trong các khung tin) giữa hai đầu cuối của một tuyến thông tin đảm bảo chuẩn xác, điều khiển lưu lượng chuyển số liệu. Cấp X.25 cấp 3 tạo cho số liệu được phát đi trong các khung tin. Đơn vị số liệu ở cấp mạng là gói.

Giao thức cấp mạng trên cơ bản xác định thao tác gọi ảo qua giao thức cấp tuyến. Mỗi cuộc gọi ảo được lớp mạng tạo ra cho các giao thức cấp cao hơn là một tuyến có điều khiển theo luồng giữa *DXE* nội hạt và một *DXE* xa qua mạng.

X.25 cấp 3 thực tế được định nghĩa là một giao thức giữa một *DTE* và một *DCE* đầu nối trực tiếp qua một tuyến thông tin. *DTE* có thể như là một *PAD* còn *DCE* có thể là một thiết bị chuyển mạch gói X.25. Có thể có nhiều kiểu mạng khác nhau đang được sử dụng để cung cấp tuyến nối giữa hai *DXE*. Thế nhưng điều quan trọng là giao tiếp cấp mạng giữa *DTE* và *DCE* phải giữ giống nhau dù cho các mạng bao gồm cả các tuyến truyền giữa *DXE* nội hạt và *DXE* xa.

#### **5.1. Khuôn mẫu gói cấp mạng**

Mỗi một gói cấp mạng có cùng khuôn mẫu đầu đề 3 bytes mô tả ở hình 8. Cụm nhận dạng khuôn mẫu chung (*GFI*) là khối 4 bit được dùng để chỉ thị khuôn mẫu chung cho phần còn lại của đầu đề. Công việc mã hoá của cụm *GFI* sẽ được mô tả trong khi mô tả các kiểu gói.



**Hình 8. Khuôn mẫu gói cấp mạng**

Cụm thứ hai của bytes đầu này của gói là địa chỉ nhóm kênh logic (*LCGN*). Nó kéo sang cả bytes thứ hai tạo thành địa chỉ kênh logic (*LCN*) 12 bit, nó dùng để nhận dạng cho từng cuộc gọi ảo riêng biệt. Byte thứ ba là cụm nhận dạng kiểu gói (*PTI*), nó định ra chức năng của gói.

### 5.2. Các kiểu gói cấp mạng

Bảng 2 trình bày các kiểu gói cấp (lớp) mạng. Lưu ý rằng cùng một gói có thể gọi tên khác nhau tùy thuộc vào *DTE* hay *DTE* phát nó đi. ở cả hai trường hợp, mã hoá cụm *PTI* là giống nhau vì khi chuyển tới đó thì các gói đều giống nhau.

Khác với cấp tuyến số liệu, ở đây *DCE* được phép làm một số việc mà *DTE* không thể làm, vì vậy sự phân biệt này là quan trọng.

Lưu ý rằng các gói thiết lập và xoá cuộc gọi chỉ có hiệu lực đối với các cuộc gọi ảo có chuyển mạch (*SVCS*), trong khi đó các gói khác có hiệu lực đối với cả *SVCs* và cả các mạch ảo cố định (*PVCs*). Các *PVC* giống các *SVC*, chỉ khác là ngay khi từ cấp tuyến số liệu đi lên, các *PVC* đi thẳng vào cung đoạn chuyển tin, bỏ qua cung đoạn thiết lập gọi. Các *PVC* ít khi được sử dụng nên chúng ta sẽ không thảo luận về chúng nữa.

#### • Các gói thiết lập và xoá cuộc gọi

Gói gọi vào và yêu cầu gọi dùng để yêu cầu thiết lập một cuộc gọi ảo giữa *DXE* phát gói này và *DXE* thu gói này. Gói chỉ cuộc gọi được đấu nối hay cuộc gọi được tiếp nhận được dùng để trả lời cho gói yêu cầu gọi hoặc gói chỉ cuộc gọi vào để chỉ thị rằng cuộc thử nối được tiếp nhận và bây giờ cuộc gọi được tiến hành.

Gói yêu cầu giải toả và biểu thị giải toả được dùng hoặc để kết thúc một tuyến nối đang làm việc hoặc để từ chối một yêu cầu thiết lập gọi (tức là để trả lời cho gói yêu cầu gọi hoặc gọi vào).

Gói xác nhận giải toả dùng để xác nhận rằng đã thu được gói chỉ thị giải toả trước đó hoặc gói yêu cầu giải toả.

## Mai Xuân Hoàn

	8	7	6	5	4	3	2	1
1	Nhận dạng thể thức chung				Địa chỉ nhóm kênh logic			
2	Địa chỉ kênh logic							
3	Nhận dạng kiểu gói							
	0	0	0	0	1	0	1	1
4	Độ dài địa chỉ DTE chủ gọi				Độ dài địa chỉ DTE bị gọi			
5	(Các) địa chỉ DTE							
					0	0	0	0
	Chiều dài mã dịch vụ							
	Các dịch vụ							
	Số liệu thuê bao gọi							

Gói gọi vào, yêu cầu gọi

	8	7	6	5	4	3	2	1
1	Nhận dạng thể thức chung				Địa chỉ nhóm kênh logic			
2	Địa chỉ kênh logic							
3	Nhận dạng kiểu gói							
	0	0	0	0	1	0	1	1
4	Độ dài địa chỉ DTE chủ gọi				Độ dài địa chỉ DTE bị gọi			
5	(Các) địa chỉ DTE							
					0	0	0	0
	Chiều dài mã dịch vụ							
	Các dịch vụ							
	Số liệu thuê bị gọi							

Gói chỉ cuộc gọi được đầu nối được tiếp nhận

	8	7	6	5	4	3	2	1
1	Nhận dạng thể thức chung				Địa chỉ nhóm kênh logic			
2	Địa chỉ kênh logic							
3	Nhận dạng kiểu gói							
	0	0	0	1	0	0/1	1	0/1
4	Nguyên nhân giải tỏa							
5	Mã phán đoán lỗi							
6	Chiều dài địa chỉ DTE chủ gọi				Chiều dài địa chỉ DTE bị gọi			
7	(Các) địa chỉ DTE							
					0	0	0	0
	Chiều dài dịch vụ							
	Các dịch vụ							
	Số liệu thuê bao xóa							

Gói chỉ thị xóa, gói yêu cầu xóa/gói xác nhận xóa



• Các gói số liệu và ngắt:

		Các bits							
		8	7	6	5	4	3	2	1
1	Nhận dạng thể thức chung	Q D 0 1				Địa chỉ nhóm kênh logic			
2	Địa chỉ kênh logic								
3	P(R)		M		P(S)		0		
		Số liệu thuê bao							

*A, module 8.*

		Các bits							
		8	7	6	5	4	3	2	1
1	Nhận dạng thể thức chung	Q D 1 0				Địa chỉ nhóm kênh logic			
2	Địa chỉ kênh logic								
3	P(S)							0	
4	P(R)							M	
		Số liệu thuê bao							

*B, module 128.*

D: bit xác nhận phân phát; Q: bit định tiêu chuẩn; M: bit tăng số liệu

Gói số liệu

Gói số liệu được dùng để chuyển số liệu cho giao thức cấp cao hơn giữa hai *DXE* đầu nối với nhau bởi cuộc gọi ảo. Gói ngắt được dùng để chuyển một phần nhỏ số liệu (tối đa 32 bytes) giữa hai *DXE* với độ ưu tiên rất cao. Gói ngắt có khả năng nhảy qua các gói số liệu và không phụ thuộc vào sự điều khiển lưu lượng cấp mạng.

Gói xác nhận ngắt được dùng để xác định việc thu một gói ngắt. Chỉ có thể có một gói ngắt không được xác nhận ở bất kỳ lần nào.

		Các bits							
		8	7	6	5	4	3	2	1
1	Nhận dạng thể thức chung					Địa chỉ nhóm kênh logic			
2	Địa chỉ kênh logic								
3	Nhận dạng kiểu gói								
4	0	0	1	0	0	0/1	1	1	
		Số liệu thuê ngắt							

Gói ngắt/gói xác nhận ngắt

• Các gói điều khiển luồng và tái lập

Các gói RR và RNR được dùng để xác nhận việc thu các gói số liệu. Sử dụng gói RR khi máy thu có thể thu thêm các gói số liệu. Gói RNR được sử dụng khi máy thu tạm thời bị bận và không thể thu thêm số liệu.

Gói *REJ* có thể được *DTE* sử dụng để yêu cầu chuyển các gói số liệu. Dịch vụ *REJ* không cần thiết hỗ trợ ở tất cả các *DCE* vì thực tế nó không cần cho thao tác chuẩn xác của nghi thức. Sử dụng gói *REJ* có ngụ ý là một gói số liệu đã thu được chuẩn xác bởi cấp tuyến

## Mai Xuân Hoàn

số liệu đã bị DTE làm mất vì lý do nào đó, có thể do nó bị đẩy ra khỏi vùng nhớ đệm dành cho gói tin thu được.

Gói chỉ thị tái lập/yêu cầu tái lập dùng để chuyển cuộc gọi ảo về trạng thái trước của nó khi cuộc gọi được thiết lập lúc ban đầu. Toàn bộ các việc chưa giải quyết xong của số liệu bị vớt bỏ, các địa chỉ dãy được lập không và các trạng thái điều khiển luồng bị xoá. Gói này thường được sử dụng khi lỗi giao thức được phát hiện hoặc điều gì đó để xoá số liệu bị "mắc kẹt" ở một cuộc gọi mà không cần phải xoá cuộc gọi hiện thời.

	8	7	6	5	4	3	2	1
1	Nhận dạng thể thức chung				Địa chỉ nhóm kênh logic			
	0	0	0	1				
2	Địa chỉ kênh logic							
3	P(R)			Nhận dạng kiểu gói				
				0/0/0	0/0/1	0/1/0	0/0/0	1/1/1

*A, module 8.*

	8	7	6	5	4	3	2	1
1	Nhận dạng thể thức chung				Địa chỉ nhóm kênh logic			
	0	0	1	0				
2	Địa chỉ kênh logic							
3	Nhận dạng kiểu gói							
	0	0	0	0	0/0/1	0/1/0	0/0/0	1/1/1
4	P(R)							D

Gói RR/RNR/REJ.

Gói xác nhận tái lập được dùng để xác nhận việc thu của gói chỉ thị tái lập/yêu cầu tái lập và nhờ vậy thể thức tái lập được thực hiện.

	8	7	6	5	4	3	2	1
1	Nhận dạng thể thức chung				Địa chỉ nhóm kênh logic			
2	Địa chỉ kênh logic							
3	Nhận dạng kiểu gói							
	0	0	0	1	1	0/1	1	1
4	Lý do tái lập							
5	Mã đoán lỗi							

Gói chỉ thị tái lập, yêu cầu tái lập/xác nhận tái lập

• *Các gói tái khởi động:*

Gói chỉ thị tái khởi động/yêu cầu tái khởi động được dùng để xoá đi tất cả các cuộc gọi ảo đang xúc tiến và chuyển tải toàn bộ cấp mạng về trạng thái khởi đầu của nó. Gói này là gói đầu tiên được cấp mạng phát đi khi cấp tuyến số liệu chuyển sang cung đoạn chuyển tin.

## Mai Xuân Hoàn

	8	7	6	5	4	3	2	1
1	Nhận dạng thể thức chung				0	0	0	0
2	0	0	0	0	0	0	0	0
3	Nhận dạng kiểu gói							
4	1	1	1	1	1	0	1	1
5	Nguyên nhân tái khởi động							
	Mã phán đoán lỗi							

A, Gói chỉ thị tái khởi động, gói yêu cầu tái khởi động

	8	7	6	5	4	3	2	1
1	Nhận dạng thể thức chung				0	0	0	0
2	0	0	0	0	0	0	0	0
3	Nhận dạng kiểu gói							
	1	1	1	1	1	1	1	1

B, Gói xác nhận tái khởi động

- Các gói tái khởi động

Gói xác nhận tái khởi động được dùng để xác nhận công việc thu một gói chỉ thị tái khởi động/yêu cầu tái khởi động và để chỉ thị rằng cấp mạng hiện đang hoạt động

Kiểu gói		Byte 3
Từ DCE tới DTE	Từ DTE tới DCE	Các bit 8.....7.....6.....5.....4.....3.....2.....1
<b>Thiết bị và giải toả cuộc gọi</b>		
Gọi vào	Yêu cầu gọi	0.....0.....0.....0.....1.....0.....1.....1
Đầu nối cuộc gọi	Tiếp nhận cuộc gọi	0.....0.....0.....0.....1.....1.....1.....1
Chỉ thị giải toả	Yêu cầu giải toả	0.....0.....0.....1.....0.....0.....1.....1
Xác nhận giải toả DCE	Xác nhận giải toả của DTE	0.....0.....0.....1.....0.....1.....1.....1
<b>Số liệu và ngắt</b>		
Số liệu DCE	Số liệu DTE	x.....x.....x.....x.....x.....x.....x.....0
Ngắt của DCE	Ngắt của DTE	0.....0.....0.....0.....0.....0.....1.....1
Xác nhận ngắt của DCE	Xác nhận ngắt của DTE	0.....0.....1.....0.....0.....1.....1.....1
<b>Điều khiển luồng và tái lập</b>		
DCE RR (module 8)	DTE RR (module 8)	x.....x.....x.....0.....0.....0.....0.....1
DCE RR (module 128)	DTE RR (module 128)	0.....0.....0.....0.....0.....0.....0.....1
DCE RNR (module 8)	DTE RNR (module 8)	x.....x.....x.....0.....0.....1.....0.....1
DCE RNR (module 128)	DTE RNR (module 128)	0.....0.....0.....0.....0.....1.....0.....1
	DTE REJ (module 8)	x.....x.....x.....0.....1.....0.....0.....1
	DTE REJ (module 128)	0.....0.....0.....0.....1.....0.....0.....1
Chỉ thị tái lập	Yêu cầu tái lập	0.....0.....0.....1.....1.....0.....1.....1
Xác nhận tái lập DCE	Xác nhận tái lập DTE	0.....0.....0.....1.....1.....1.....1.....1
<b>Tái khởi động</b>		
Chỉ thị tái khởi động	Yêu cầu tái khởi động	1.....1.....1.....1.....1.....0.....1.....1
Xác nhận tái khởi động DCE	Xác nhận tái khởi động	1.....1.....1.....1.....1.....1.....1.....1

## Mai Xuân Hoàn

	DTE	
Phán đoán		
Phán đoán		1.....1.....1.....0.....0.....0.....1
Đăng ký		
Xác nhận đăng ký	Yêu cầu đăng ký	1.....1.....1.....0.....0.....1.....1 1.....1.....1.....1.....0.....1.....1.....1

**Bảng 2. Các trị số của cụm mã PTI**

• *Các gói phán đoán lỗi và đăng ký dịch vụ*

Gói phán đoán lỗi do DCE phát cho DTE khi DCE thu một gói tin bị lỗi trầm trọng. Ví dụ : Khi thu được một gói có trường GFI không chuẩn xác, DCE có thể phát một gói phán đoán lỗi cho DTE, gói này chứa mã phán đoán, lỗi thích hợp: Không phải toàn bộ các DCE đều tạo ra các gói phán đoán lỗi.

Gói yêu cầu đăng ký dịch vụ có thể được DTE phát cho DCE để yêu cầu được sử dụng hay không sử dụng một số dịch vụ nào đó trong khoảng thời gian nào đó. Các dịch vụ liên quan sẽ được mô tả sau này.

Gói xác nhận đăng ký do DCE phát cho DTE để trả lời cho một gói yêu cầu đăng ký dịch vụ từ DTE.

	8	7	6	5	4	3	2	1
1	Nhận dạng thể thức chung				0	0	0	0
2	0	0	0	0	0	0	0	0
3	Nhận dạng kiểu gói							
4	1	1	1	1	0	0/1	1	1
	Chiều dài địa chỉ DTE				Chiều dài địa chỉ DCE			
	Địa chỉ DCE và DTE							
					0	0	0	0
0	Chiều dài đăng ký							
	Đăng ký							

Gói yêu cầu đăng ký/gói xác nhận đăng ký

### 5.3. Các địa chỉ dây cáp mạng

Cũng như ở cấp tuyến số liệu, các kiểu gói xác định đều mang theo chúng các địa chỉ dây. Các địa chỉ dây này (chỉ số thứ tự) được dùng để đảm bảo cho các gói số liệu được chuyển đi không bị mất và theo một thứ tự chuẩn xác. Có hai địa chỉ dây được tải đi, đó là địa chỉ dây P(S) và địa chỉ dây P(R).

Địa chỉ dây P(S) chỉ được mang theo các góc số liệu và dùng để nhận dạng từ gói số liệu riêng.

Địa chỉ dây P(R) được mang theo ở gói số liệu, gói RR, RNR, và REJ. Vùng mã P(R) ở các gói này chuyển địa chỉ dây của gói số liệu tiếp theo mà máy phát sẽ chuyển cho máy thu.

Giống như ở cấp tuyến số liệu, có hệ thống đánh số dãy thông dụng, nó sử dụng cụm 3 bit và có địa chỉ dãy từ 0 tới 7, và một hệ thống đánh số dãy mở rộng nó sử dụng cụm mã 7 bit và có địa chỉ dãy từ 0 tới 127.

### 5.4. Trường mã nhận dạng khuôn mẫu.

Chúng ta đã đi qua các kiểu gói cấp mạng, bây giờ chúng ta chuyển sang công việc mã hoá cụm mã *GFI*. Bảng 3 trình bày các giá trị mã cụm *GFI* có thể nhận Bit "Q" chỉ xuất hiện ở các gói số liệu và được dùng để phân biệt gói số liệu theo hai loại khác nhau: Các gói số liệu thông thường và các gói số liệu "định phẩm chất". Các gói số liệu định phẩm chất thường được sử dụng để cho phép chuyển thông tin điều khiển giao thức cấp cao hơn mà không ảnh hưởng tới số liệu giao thức cấp cao hơn mà chúng được phát đi ở các gói số liệu thông thường. Một ví dụ về giao thức này là X.29.

Bit D là bit xác định chuyển giao. Bit này có thể xuất hiện ở các gói thiết lập gọi nhưng thực tế chức năng của nó chỉ liên quan đến việc chuyển giao các gói số liệu. Bit 5 và 6 của cụm mã *GFI* được sử dụng để chỉ thị hệ thống đánh số dãy nào được sử dụng. Hệ thống đánh số dãy mở rộng là một trong các kiểu tự chọn, gọi là kiểu đặt trước. Tức là hệ thống đánh số được dùng cần phải được quyết định khi tuyến X.25 được thiết lập. Toàn bộ các cuộc gọi ảo trên tuyến cần phải sử dụng hệ thống đánh số này đã đặt trước nó. Nếu dịch vụ đăng ký có hiệu lực thì nó có thể chuyển đổi hệ thống đánh số hiện thời theo những điều kiện nhất định. Phần lớn các trường hợp sử dụng hệ thống đánh số thông thường vì chỉ cần rất ít điều bổ sung hỗ trợ cho hệ thống đánh số mở rộng.

		Byte 1			
		Các bit			
		8	7	6	5
Các gói thiết lập gọi	Hệ thống địa chỉ dãy module 8	0	0	0	1
	Hệ thống địa chỉ dãy module 128	0	0	1	0
Góc giải toả, điều khiển luồng, ngắt, tái lập, tái khởi động đăng ký và phán đoán	Hệ thống địa chỉ dãy module 8	0	0	0	1
	Hệ thống địa chỉ dãy module 128	0	0	1	0
Các gói số liệu	Hệ thống địa chỉ dãy module 8	0	0	0	1
	Hệ thống địa chỉ dãy module 128	0	0	1	0
Mở rộng nhận dạng khuôn mẫu thông thường		0	0	1	1
Dùng cho các ứng dụng khác * Không xác định		*	*	0	0

**Bảng 3. Trị số của cụm mã GFI**

### 5.5 Cung đoạn tái khởi động

Ngay khi cấp tuyến số liệu chuyển sang trạng thái chuyển tin, cấp mạng cần được tái khởi động. Một yêu cầu tái khởi động/chỉ thị tái khởi động *DCE* hoặc *DTE* phát đi, sau đó nó được xác nhận bởi gói xác nhận tái khởi động. Vì nó là chung cho gói chỉ thị tái khởi động/gói yêu cầu tái khởi động mà cần phát đi ngay khi cấp tuyến làm việc, nên thường xảy ra "va vấp". Nếu một *DXE* đang chờ sự xác nhận tái khởi động mà nó thu được chỉ thị tái khởi động / yêu cầu tái khởi động thì "va vấp" tái khởi động đã bị xảy ra và gói thu được coi như là sự xác nhận tái khởi động.

Hình 6.9 mô tả thể thức của cụm mã chỉ thị tái khởi động / yêu cầu tái khởi động và các gói xác nhận tái khởi động. Lưu ý là cụm mã *LCGN* và *LCN* được mã hoá toàn bộ số 0. Địa chỉ đầy đủ của kênh logic 000000000000 có thể không dùng được bởi một cuộc gọi ảo ở tất cả các điều khiển bổ sung và cần loại ra ở bất cứ nơi nào có thể được.

Bảng 4: mô tả giá trị có thể có của cụm mã nguyên nhân tái khởi động trong gói chỉ thị tái khởi động.

	Các bit
	8.....7.....6.....5.....4.....3.....2.....1
Lỗi thủ tục tại chỗ	0.....0.....0.....0.....0.....0.....0.....0
ứ mạng	0.....0.....0.....0.....0.....0.....0.....1
Mạng làm việc	0.....0.....0.....0.....0.....0.....1.....1
Xác nhận đăng ký/ xoá	0.....1.....1.....1.....1.....1.....1.....1

Bảng 6.4. Mã hoá cho cụm mã chỉ nguyên nhân tái khởi động của các gói chỉ thị tái khởi động

		Các bit	
		8.....7.....6.....5.....4.....3.....2.....1	
Các byte	1	Nhận dạng khuôn mẫu chung	.....0.....0.....0.....0
	2	0.....0.....0.....0.....0.....0.....0.....0	
	3	Nhận dạng kiểu gói 1.....1.....1.....1.....1.....0.....1.....1	
	4	Nguyên nhân tái khởi động	
	5	Mã phán đoán lỗi	

a) Gói chỉ thị tái khởi động / gói yêu cầu khởi động

Các byte		Các bit 8.....7.....6.....5.....4.....3.....2.....1	
	1	Nhận dạng khuôn mẫu chung	.....0.....0.....0.....0
	2	0.....0.....0.....0.....0.....0.....0.....0	
	3	Nhận dạng kiểu gói 1.....1.....1.....1.....1.....1.....1.....1	

Hình 9

b) Gói xác nhận tái khởi động

Khi được dùng ở gói yêu cầu tái khởi động, bit 8 cần được thiết lập ở giá trị 1 thay cho giá trị 0. Vì vậy đây là sự hoàn thiện đáng kể so với X.25(80), nó nói lên rằng cụm mã này ở gói yêu cầu tái khởi động cần phải có giá trị mã toàn bộ không. Điều này thường có nghĩa là khi sử dụng các tuyến ghép X.25 thì lý do để tái khởi động đã mất đi. Cụm mã đã phán đoán lỗi có thể được dùng để cung cấp thêm thông tin về lý do khởi động lại.

Chỉ có gói khác có liên quan tới cung đoạn này là gói phán đoán lỗi. Hình 10 mô tả thể thức của gói này. Cụm giải nghĩa phán đoán bao gồm 3 byte đầu của cụm mã đầu đề trong gói chỉ thị nguyên nhân phải phát gói phán đoán lỗi. Nếu chỉ có dưới 3 byte tức là chỉ có những gì đã thu được.

Các byte		Các bit 8.....7.....6.....5.....4.....3.....2.....1	
	1	Nhận dạng thể thức chung	.....0.....0.....0.....0
	2	0.....0.....0.....0.....0.....0.....0.....0	
	3	Nhận dạng kiểu gói 1.....1.....1.....1.....0.....0.....0.....1	
	4	Mã phán đoán lỗi	
	5	Giải nghĩa phán đoán lỗi	

Hình 10

5.6. Thiết lập các cuộc gọi thực

Gói gọi vào/gói yêu cầu gọi yêu cầu một cuộc gọi thực có chuyển mạch cần được thiết lập giữa hai DXE được nối vào mạng. Hình 11 mô tả thể thức của gói này.

Lưu ý là bit 7 của GFI có nhãn là "D". Nếu bit này được thiết lập tức là thuê bao chủ gọi yêu cầu sử dụng dịch vụ bit D (xem mục cung đoạn chuyển giao tin ở chương này về vấn đề đó). Nếu nó bị xoá tức là chủ gọi không có ý định sử dụng nó.

*DXE* phát gói gọi vào/yêu cầu gọi phải chọn một địa chỉ nhóm kênh logic (*LCGN*) và địa chỉ kênh logic (*LCN*) để nhận dạng duy nhất một cuộc gọi riêng. Hiển nhiên nó không thể là địa chỉ đã dùng bởi cuộc gọi đang tiến hành. Dịch vụ *PSS* chia số lượng tổng thể kênh thành một nhóm. Các nhóm khác nhau được nhận dạng bởi trường mã *LCGN* này. Các nhóm này là:

*LCGN*, Kiểu, Tên, Sử dụng

Nó dùng để *DCE* định vị *LCNs* từ địa chỉ kênh thấp nhất trở lên, còn *DTE* lại định vị *LCNs* từ kênh cao nhất xuống.

Thông thường thì cả *DCE* và *DTE* đều sử dụng *SVC* cả hai cánh. Ví dụ nếu chỉ cần 16 kênh logic thôi thì *LCGN* 4 có thể được sử dụng cùng với các *LCN* từ 0 tới 15. *DCE* sẽ bắt đầu phân bổ các kênh từ 0 trở lên, còn *DTE* sẽ bắt đầu phân bổ các địa chỉ kênh từ 15 trở xuống.

Các bit	
8.....7.....6.....5.....4.....3.....2.....1	
Nhận dạng thể thức chung	Địa chỉ nhóm kênh logic
Địa chỉ kênh logic	
Nhận dạng kiểu gói 0.....0.....0.....0.....1.....0.....1.....1	
(Các) địa chỉ <i>DTE</i> 0.....0.....0.....0.....	
Chiều dài dịch vụ	
Các dịch vụ	
Số liệu thuê bao gọi	

**Hình 11. Gói gọi vào/gói yêu cầu gọi**

Có hai cụm mã gọi là địa chỉ *DTE* bị gọi và địa chỉ *DTE* chủ gọi. Mỗi địa chỉ có thể tới 15 chữ số, mỗi chữ số có thể từ 0 tới 9. Thường chỉ tối đa 14 chữ số được sử dụng 14 chữ số này được chia thành một địa chỉ 12 chữ số cùng với 2 chữ số cuối cùng là địa chỉ dưới cấp. Mạng X.25 thường tạo tuyến trên cơ sở địa chỉ 12 chữ số đầu. Còn 2 chữ số sau dùng cho địa chỉ một thực thể riêng trong *DXE* có địa chỉ đó. Như vậy địa chỉ này có thể là địa chỉ *DXE* chủ gọi và địa chỉ *DXE* bị gọi. Lý do vì sao *DTE* được sử dụng là đặc điểm của *DTE* cũng như các thiết bị được đấu nối vào mạng có *DCE* là giao tiếp cho mạng. Vì vậy bản thân của các



*DCE* không bao giờ là bị gọi. Tuy nhiên trong thực tế thì cả các *DTE* lẫn các *DCE* vẫn có thể là bị gọi.

Các địa chỉ này được thiết lập ở dạng gói sử dụng thể thức số thập phân được mã hoá nhị phân (*BCD*). Điều này có nghĩa là 2 chữ số trong mỗi byte: 4 bit cao (8-5) cho chữ số thứ nhất, 4 bit thấp (4-1) cho chữ số thứ hai. Địa chỉ đầu tiên là địa chỉ *DTE* bị gọi. Nó là địa chỉ của thiết bị mà thuê bao chủ gọi muốn thiết lập cuộc gọi ảo tới đó. Địa chỉ thứ 2 là địa chỉ thuê bao chủ gọi (*DTE*). Nó là địa chỉ của thiết bị bắt nguồn của gói tin. Đặc điểm này cho phép *DTE* không chứa địa chỉ *DTE* chủ gọi mặc dù *DCE* luôn luôn cần phải bao gồm cả nó. Như thế công việc thực hiện *X.25* có thể là *DCE* hoặc *DTE*, địa chỉ *DTE* chủ gọi luôn bao gồm hầu hết.

Một vấn đề nhỏ xảy ra khi tổng số các chữ số trong cụm địa chỉ *DTE* chủ gọi và bị gọi là lẻ vì chỉ một nửa byte cuối được dùng. Quy định này chọn số 0 đưa thêm vào để lấp vào nửa byte cuối cùng.

Tiếp theo các địa chỉ *DTE* là cụm mã dịch vụ mở đầu bởi byte chỉ chiều dài dịch vụ. Chiều dài này là tổng số byte của cụm mã dịch vụ được phép (chiều dài này có thể thay đổi được). Chiều dài tối đa của trường dịch vụ là 100 byte. Trường này dùng để mã hoá các thông tin đặc biệt về cuộc gọi và các yêu cầu dịch vụ trong khoảng thời gian tiến hành cuộc gọi. Các dịch vụ này sẽ được mô tả sau này.

Trường cuối cùng của gói là trường số liệu thuê bao của cuộc gọi (*CUD*). Trường này chứa các số liệu tùy chọn, nó được truyền dẫn không biến đổi giữa *DCX* chủ gọi và bị gọi. ở các gói gọi vào thông thường/các gói yêu cầu gọi trường *CUD* dài 16 byte là tối đa. Nếu dịch vụ "chọn nhanh" được sử dụng thì trường này có thể dài tới 128 bytes. Dịch vụ chọn nhanh là một trong các dịch vụ có thể có trong trường dịch vụ của gói.

Trường *CUD* có thể được dùng cho nhiều mục đích khác nhau tùy theo ứng dụng. Một ứng dụng chung là để cung cấp khả năng tạo lập địa chỉ mở rộng khi một phần của tuyến gọi bao gồm các mạng không dùng *X.25*.

Nếu cuộc gọi được phía thu gói gọi vào/gói yêu cầu gọi chấp nhận thì một gói chỉ thị cuộc gọi được đầu nối/cuộc gọi được tiếp nhận được phát trả lời. Hình 12 mô tả thể thức của gói chỉ thị cuộc gọi được đầu nối/cuộc gọi được tiếp nhận. Lưu ý là có hai dạng gói đầu nối gọi/tiếp nhận gọi. Thể thức thông thường có thể chứa cực đại là 3 bytes và không có trường số liệu thuê bao bị gọi. Thể thức mở rộng có ít nhất 5 bytes (nếu không có địa chỉ và các dịch vụ thì chiều dài dịch vụ và địa chỉ vẫn phải có) và có thể chứa trường số liệu thuê bao bị gọi nếu gói gọi vào/gói yêu cầu gọi đã quy định sử dụng dịch vụ chọn nhanh.

Các byte		Các bit	
		8.....7.....6.....5.....4.....3.....2.....1	
	1	Nhận dạng thể thức chung	Địa chỉ nhóm kênh logic

	2	Địa chỉ kênh logic	
	3	Nhận dạng kiểu gói 0.....0.....0.....0.....1.....0.....1.....1	
	4	Độ dài địa chỉ DTE chủ gọi	Độ dài địa chỉ DTE bị gọi
	5	(Các) địa chỉ DTE 0.....0.....0.....0.....	
		Chiều dài dịch vụ	
		Các dịch vụ	
		Số liệu thuê bao bị gọi	

**Hình 12 Gói chỉ cuộc gọi được đầu nối/được tiếp nhận**

Nhắc lại là trường *GFI* có bit 7 được gắn nhãn D. Nếu chủ gọi có yêu cầu dịch vụ này và *DXE* bị gọi có khả năng cung cấp thì cần phải thiết lập bit này trong gói. Nếu không thì bit này phải xoá đi.

Đáp ứng nhắc nhở cho gói gọi vào/gói yêu cầu gọi là gói chỉ thị xoá/gói yêu cầu xoá. Hình 13 mô tả thể thức của gói này. Cũng có hai kiểu biến thể như đối với gói đầu nối cuộc gọi và gói tiếp nhận cuộc gọi. Các quy tắc sử dụng địa chỉ, dịch vụ và xoá số liệu thuê bao dẫn ra từ các quy tắc của gói đầu nối cuộc gọi và gói tiếp nhận cuộc gọi.

Các byte		Các bit	
		8.....7.....6.....5.....4.....3.....2.....1	
	1	Nhận dạng thể thức chung	Địa chỉ nhóm kênh logic
	2	Địa chỉ kênh logic	
	3	Nhận dạng kiểu gói 0.....0.....0.....1.....0.....0.....1.....0	
	4	Nguyên nhân giả toả	
	5	Mã phán đoán lỗi	
	6	Chiều dài địa chỉ DTE chủ gọi	Chiều dài địa chỉ DTE bị gọi
		(Các) địa chỉ DTE 0.....0.....0.....0.....	
		Chiều dài dịch vụ	

		Các dịch vụ
		Số liệu thuê bao xoá

**Hình 13. Gói chỉ thị xoá/gói yêu cầu xoá**

Trường lý do giải toả (xoá) được dùng để chỉ thị lý do vì sao cuộc gọi bị xoá. Bảng 6.5 mô tả các trị số có thể dùng cho trường nguyên nhân xoá (mong muốn của các tác giả là không bị phạm lỗi). Lưu ý *DTE* có thể tạo lập trường này nếu bit 8 được thiết lập. Cũng như đối với trường lý do tái khởi động, đây là một trong các tính năng mới, quan trọng của X.25(84). ở X.25 (80) *DTE* mã hoá trường này toàn số không.

Gói xác nhận xoá dùng để xác nhận việc thu gói chỉ thị xoá/gói yêu cầu xoá. Thể thức của nó mô tả ở hình 14. ở đây lại có hai biến thể của gói này. Lúc này thể thức gói thông thường luôn được sử dụng. Gói thể thức mở rộng chỉ có thể do một *DCE* phát cho một *DTE* và nó chỉ được dùng liên quan tới dịch vụ. "thông tin tính cước". khác ở các dịch vụ tùy chọn.

	Các bit 8.....7.....6.....5.....4.....3.....2.....1
DTE khởi xướng gọi	0.....0.....0.....0.....0.....0.....0.....0
DTE khởi xướng gọi	x.....x.....x.....x.....x.....x.....x.....x
Địa chỉ bản	0.....0.....0.....0.....0.....0.....0.....0
Không được khai thác	0.....0.....0.....0.....0.....1.....0.....0.....1
Lỗi thủ tục hướng ra	0.....0.....0.....0.....1.....0.....0.....0.....1
Bị gọi không chấp nhận trả cước	0.....0.....0.....0.....1.....1.....0.....0.....1
Đích gọi không tương thích	0.....0.....0.....1.....0.....0.....0.....0.....1
Không chấp nhận chọn nhanh	0.....0.....1.....0.....0.....0.....0.....0.....1
Không lỗi	0.....0.....1.....1.....1.....1.....0.....0.....1
Yêu cầu dịch vụ không hợp lệ	0.....0.....0.....0.....0.....0.....0.....1.....1
Chặn tiếp cận	0.....0.....0.....0.....0.....1.....0.....1.....1
Lỗi thủ tục tại chỗ	0.....0.....0.....0.....1.....0.....0.....1.....1
ứ tải mạng	0.....0.....0.....0.....0.....0.....1.....0.....1
Không có khả năng tiếp cận	0.....0.....0.....0.....0.....1.....1.....0.....1
RPOA	0.....0.....0.....0.....1.....0.....1.....0.....1

**Bảng 5 Bảng giá trị của trường nguyên nhân xoá**

Các byte		Các bit	
		8.....7.....6.....5.....4.....3.....2.....1	
	1	Nhận dạng thể thức chung	Địa chỉ nhóm kênh logic
	2	Địa chỉ kênh logic	
	3	Nhận dạng kiểu gói 0.....0.....0.....1.....0.....1.....1.....1	
	4	Chiều dài địa chỉ DTE chủ gọi	Chiều dài địa chỉ DTE bị gọi
	5	(Các) địa chỉ DTE	
		Chiều dài dịch vụ	
		Các dịch vụ	

Hình 14. Gói xác nhận xoá

Các byte		Các bit			
		8.....7.....6.....5.....4.....3.....2.....1			
	1	Nhận dạng thể thức chung Q.....D.....1.....1	Địa chỉ nhóm kênh logic		
	2				
	3	P(R)	M	(P)S	0
		Số liệu thuê bao			

Các byte		Các bit	
		8.....7.....6.....5.....4.....3.....2.....1	
	1	Nhận dạng thể thức chung Q.....D.....1.....0.....	Địa chỉ nhóm kênh logic
	2	Địa chỉ kênh logic	

	3	P(S)	0
	4	P(R)	M
		Số liệu thuê bao	

(Khi đã mở rộng module tới 128)

*D*: Bit xác nhận phân phát;

*Q*: Bit định tiêu chuẩn

*M*: Bit tăng thêm số liệu

**Hình 16. Gói số liệu**

### 5.7 Cung đoạn chuyển giao tin

Một cuộc gọi ảo sẽ mất nhiều thời gian nhất ở cung đoạn chuyển số liệu. Các gói số liệu được trao đổi giữa các *DXE* ở mỗi đầu của tuyến gọi.

Hình 16 mô tả thể thức của gói số liệu. Lưu ý ở đây lại có hai thể thức có thể xảy ra. Nó được tạo ra do ta đánh số dãy mở rộng và thông thường.

Bit *M* là bit chỉ thị thêm số liệu. Nó dùng để tạo ra khả năng chuyển các bản tin số liệu đi. Các bản tin này dài hơn một gói cấp mạng đơn. Bảng tin này được chia thành các gói cấp mạng và gói cuối cùng sẽ tạo lập bit *M*. Gói cuối mà không có bit *M* thể hiện đây là cuối của dãy.

Khi một *DXE* thu một gói số liệu nó phải phát trở lại cho *DXE* đầu kia tin xác nhận cuộc gọi. Có thể có hai gói, gói *RR* hoặc *RNR*. Hình 17 mô tả thể thức của các gói này. Gói *RNR* được phát đi khi *DXE* không có khả năng thu thêm các gói tin nữa, thường do nó đã hết tiềm năng. Gói *RR* được phát đi khi *DXE* có khả năng thu tiếp tục các gói số liệu. Như trước đây đã nhắc tới, một *DTE* có thể phát đi một gói *REJ* cho *DCE* để yêu cầu phát lại các gói số liệu. Thể thức của nó ở hình 18.

Sự hiện diện của bit *D* được dùng để đảm bảo rằng *DXE* đầu kia đã thực sự thu được gói số liệu. Vì sao điều này lại cần thiết? Vấn đề là ở chỗ mặc dù *DXE* phát gói số liệu này có thể nhận được gói trả lời xác nhận là *RR* hay *RNR*, nó không biết chính xác gói này từ đâu tới. Nếu bit *D* được thiết lập thì yêu cầu sử dụng dịch vụ "xác nhận phân phát", khi đó lớp mạng không phát đi sự xác nhận cho tới khi *DXE* đối phương đã thu được gói số liệu và xác nhận điều đó.

Các byte		Các bit
		8.....7.....6.....5.....4.....3.....2.....1

	1	Nhận dạng thể thức chung 0.....0.....0.....1	Địa chỉ nhóm kênh logic
	2	Địa chỉ kênh logic	
	3	P(R)	Nhận dạng kiểu gói 0.....0.....0.....0.....1

(Module 8)

Các byte		Các bit	
		8.....7.....6.....5.....4.....3.....2.....1	
	1	Nhận dạng thể thức chung 0.....0.....1.....0....	Địa chỉ nhóm kênh logic
	2	Địa chỉ kênh logic	
	3	Nhận dạng kiểu gói 0.....0.....0.....0.....0.....0.....0.....1	
	4	P(R)	D

a) Gói RR

Các byte		Các bit	
		8.....7.....6.....5.....4.....3.....2.....1	
	1	Nhận dạng thể thức chung 0.....0.....0.....1	Địa chỉ nhóm kênh logic
	2	Địa chỉ kênh logic	
	3	P(R)	Nhận dạng kiểu gói 0.....0.....1.....0.....1

Các byte		Các bit
----------	--	---------

		8.....7.....6.....5.....4.....3.....2.....1	
	1	Nhận dạng thể thức chung 0.....0.....1.....0....	Địa chỉ nhóm kênh logic
	2	Địa chỉ kênh logic	
	3	Nhận dạng kiểu gói 0.....0.....0.....0.....0.....1.....0.....1	
	4	P(R)	D

b) Gói RNR (Khi module mở rộng 128)

**Hình 17. Gói RR(a) và gói RNR(b)**

Các byte		Các bit	
		8.....7.....6.....5.....4.....3.....2.....1	
	1	Nhận dạng thể thức chung 0.....0.....0.....1	Địa chỉ nhóm kênh logic
	2	Địa chỉ kênh logic	
	3	P(R)	Nhận dạng kiểu gói 0.....1.....0.....0.....1

a) Module 8

Các byte		Các bit	
		8.....7.....6.....5.....4.....3.....2.....1	
	1	Nhận dạng thể thức chung 0.....0.....0.....1...	Địa chỉ nhóm kênh logic
	2	Địa chỉ kênh logic	
	3	Nhận dạng kiểu gói 0.....0.....0.....0.....1.....0.....0.....1	
		P(R)	D

b) Module mở rộng 128

Các gói cuối liên quan tới cung đoạn chuyển tin là các gói chỉ thị tái lập/yêu cầu tái lập và xác định tái lập. Hình 20 mô tả thể thức của gói tin này. Bảng 6 là bảng mã hoá của trường lý do tái lập. DTE có thể thiết lập trường này khi bit 8 được thiết lập. Mã phán đoán lỗi sẽ cung cấp thêm thông tin về lý do tái lập. Thể thức của các gói yêu cầu đăng ký và xác nhận đăng ký mô tả ở hình 21.

Các byte		Các bit	
		8.....7.....6.....5.....4.....3.....2.....1	
	1	Nhận dạng thể thức chung	Địa chỉ nhóm kênh logic
	2	Địa chỉ kênh logic	
	3	Nhận dạng kiểu gói 0.....0.....1.....0.....0.....0.....1.....1	
	4	Số liệu thuê ngắt	

Hình 18. Gói REJ

a) Gói ngắt

Các byte		Các bit	
		8.....7.....6.....5.....4.....3.....2.....1	
	1	Nhận dạng thể thức chung	Địa chỉ nhóm kênh logic
	2	Địa chỉ kênh logic	
	3	Nhận dạng kiểu gói 0.....0.....1.....0.....0.....1.....1.....1	

Hình 18. Gói REJ

b) Gói xác nhận ngắt

	<i>Các bit</i>
	8.....7.....6.....5.....4.....3.....2.....1
DTE khởi xướng	0.....0.....0.....0.....0.....0.....0.....0
DTE khởi xướng	1.....x.....x.....x.....x.....x.....x.....x
Không làm việc tốt	0.....0.....0.....0.....0.....0.....0.....1
Lỗi thể thức phía xa	0.....0.....0.....0.....0.....0.....0.....1
Lỗi thể thức tại chỗ	0.....0.....0.....0.....0.....0.....1.....0
ứ mạng	0.....0.....0.....0.....0.....0.....1.....1
Lỗi thao tác DTE phía xa	0.....0.....0.....0.....0.....1.....0.....0
Lỗi thao tác mạng	0.....0.....0.....0.....1.....1.....1.....1



Đích không tương thích	0.....0.....0.....1.....0.....0.....0.....1
Mạng không làm việc	0.....0.....0.....1.....1.....1.....0.....1

**Hình 19. Mô tả thể thức các gói ngắt và xác nhận ngắt**

Bảng 6. Các trị số mã của trường lý do tái lập

Các byte		Các bit	
		8.....7.....6.....5.....4.....3.....2.....1	
	1	Nhận dạng thể thức chung	Địa chỉ nhóm kênh logic
	2	Địa chỉ kênh logic	
	3	Nhận dạng kiểu gói 0.....0.....0.....1.....1.....0.....1.....1	
	4	Lý do tái lập	
	5	Mã đoán lỗi	

a) Gói chỉ thị tái lập 1 yêu cầu tái lập

Các byte		Các bit	
		8.....7.....6.....5.....4.....3.....2.....1	
	1	Nhận dạng thể thức chung	Địa chỉ nhóm kênh logic
	2	Địa chỉ kênh logic	
	3	Nhận dạng kiểu gói 0.....0.....0.....1.....1.....1.....1.....1	

b) Gói xác nhận tái lập

**Hình 20**

Chi tiết của quá trình đăng ký trực đường sẽ không được mô tả ở sách này (xem các chi dẫn chi tiết của CCITT).

**5.8. Trường mã dịch vụ:**

Một số gói cấp mạng có trường dịch vụ trong chúng. Trường mã dịch vụ này cho phép DXE chủ gọi yêu cầu một số dịch vụ có thể dùng được trong lúc thực hiện cuộc gọi.

Trường dịch vụ này tạo thành từ một hay nhiều "phần tử dịch - vụ". Có 4 dạng phần tử dịch vụ cơ bản, được trình bày ở hình 22.

Bit 7 và 8 của byte đầu tiên của phần tử dịch vụ mã hoá kiểu dịch vụ. Trong thực tế có một số lượng khá lớn mã phần tử dịch vụ. Các kiểu nêu ra ở đây là kiểu đáng chú ý nhất để hỗ trợ dịch vụ mạng OSI.

Byte đầu, byte nhận dạng kiểu dịch vụ, rồi tiếp tới là một hay nhiều byte của trường tham số dịch vụ.

Các byte		Các bit	
		8.....7.....6.....5.....4.....3.....2.....1	
	1	Nhận dạng thể thức chung	0.....0.....0.....0
	2	Địa chỉ kênh logic	
	3	Nhận dạng kiểu gói 1.....1.....1.....1.....0.....0.....1.....1	
	4	Chiều dài địa chỉ DTE	Chiều dài địa chỉ DCE
		Địa chỉ DCE và DTE 0.....0.....0.....0	
		0	Chiều dài đăng ký
		Đăng ký	

a) Gói yêu cầu đăng ký

Các byte		Các bit	
		8.....7.....6.....5.....4.....3.....2.....1	
	1	Nhận dạng thể thức chung	0.....0.....0.....0
	2	0.....0.....0.....0.....0.....0.....0.....0	
	3	Nhận dạng kiểu gói 1.....1.....1.....1.....0.....1.....1.....1	
	4	Nguyên nhân	
	5	Phán đoán	
		Chiều dài địa chỉ DTE	Chiều dài địa chỉ DCE
		Địa chỉ DCE và DTE 0.....0.....0.....0	
		0	Chiều dài đăng ký
	Đăng ký		

b) Gói xác nhận đăng ký

Hình 21. Gói yêu cầu và xác nhận đăng ký

Các byte		Các bit	
		8.....7.....6.....5.....4.....3.....2.....1	
	0	0.....0.....x.....x.....x.....x.....x.....x	
	1	Trường tham số dịch vụ/ đăng ký	

Loại A

Các byte		Bit
		8.....7.....6.....5.....4.....3.....2.....1
	0	0.....1.....x.....x.....x.....x.....x.....x
	1	Dịch vụ/ đăng ký
	2	Trường tham số

Loại B

Các byte		Bit
		8.....7.....6.....5.....4.....3.....2.....1
	0	1.....0.....x.....x.....x.....x.....x.....x
	1	Trường tham số dịch vụ/ đăng ký
2		
3		

Loại C

Các byte		Bit
		8.....7.....6.....5.....4.....3.....2.....1
	0	1.....1.....x.....x.....x.....x.....x.....x
	1	Chiều dài trường tham số dịch vụ đăng ký
	2	Trường tham số dịch vụ đăng ký

Loại C

**Hình 22. Các kiểu phân tử dịch vụ**

• *Mở rộng địa chỉ bị gọi*

Dịch vụ này được đưa thêm vào chuyên đề hỗ trợ cho địa chỉ OSINSAP (điểm tiếp cận dịch vụ mạng), địa chỉ này cần nhiều vị trí hơn trường địa chỉ DTE bị gọi. Byte đầu tiên sau trường mã dịch vụ là chiều dài của trường tham số (chỉ số byte). Nó nhiều hơn số lượng byte cần để giữ phần mở rộng của địa chỉ. Byte đầu tiên của trường tham số được chia thành hai vùng. Các bit từ 1 tới 6 chỉ thị số lượng chữ số ở phần mở rộng địa chỉ (mỗi chữ số đòi hỏi 4 bit). Bit 7 và 8 mã hoá kiểu sử dụng phần mở rộng địa chỉ như sau:

Bit 8 Bit 7 Sử dụng mở rộng địa chỉ bị gọi

0 0 Để chuyển đi toàn bộ địa chỉ OSINSAP bị gọi

0 1 Để chuyển đi một phần địa chỉ OSINSAP bị gọi

1 0 Chỉ thị không chuyển đi phần mở rộng địa chỉ OSI bị gọi

1 1 Dự trữ

Số lượng cực đại các chữ số cho phép là giữa 32 và 40 tùy thuộc vào điều liên quan tới như chiều dài hợp lệ tối đa của địa chỉ OSINSAP (CCITT và ISO bất đồng). Chữ số địa chỉ mở rộng được ghép vào trường tham số giống như các địa chỉ DTE.

- *Mở rộng địa chỉ chủ gọi*

Điều này giống hệt cấu trúc dịch vụ trước, loại trừ toàn bộ các từ "bị gọi" được chuyển thành "chủ gọi".

- *Đầu tính cước và chọn nhanh*

Dịch vụ này có một trường tham số 1 byte. Các bit ở trường tham số đòi hỏi tùy theo việc sử dụng dịch vụ. Chúng được xác định như sau:

Thứ tự bit Dịch vụ yêu cầu

1 Đảo tính cước. Nếu mạng đang dùng tính cước cho chủ gọi (tức PSS), khi đảo tính cước để cước được tính cho DXE bị gọi thay vì cho DXE chủ gọi.

7 Hạn chế trả lời. Chỉ trả lời khi bit này được thiết lập để xóa cuộc gọi.

8 Chọn nhanh. Nó xác định trường số liệu mở rộng của thuê bao chủ gọi và trường số liệu của thuê bao xóa có hiệu lực sử dụng.

Tất cả các bit khác trong byte này cần lập "0" vì chúng có thể được phân phối cho các dịch vụ mới trong tương lai của nghi thức này.

- *Thu xếp kích cỡ gói*

Byte đầu của trường tham số này là kích cỡ tối đa của các gói số liệu từ DXE bị gọi, còn byte thứ 2 là kích cỡ tối đa của các gói số liệu từ DXE chủ gọi. Các trị số có thể có của trường này là:

00000100 = 16 bytes

00000101 = 32 bytes

00000110 = 64 bytes

00000111 = 128 bytes

00001000 = 256 bytes

00001001 = 512 bytes

00001010 = 1024 bytes

00001011 = 2048 bytes

00001100 = 4096 bytes

Nếu một cuộc đấu nối có tham số này ở trường dịch vụ của nó thì DXE chủ gọi yêu cầu sử dụng kích cỡ gói tối đa được chỉ ra. DXE bị gọi có thể xem xét các giá trị này nhờ chuyển về các phần tử dịch vụ đó theo gói đầu nối/tiếp nhận cuộc gọi (nếu cuộc gọi có thể được tiếp nhận). Nếu nó chấp nhận các giá trị đã yêu cầu thì nó sẽ chuyển về các giá trị giống thế. Nếu không thì nó có thể thu xếp một giá trị giảm tới một kích cỡ mà nó có thể quản lý được. Lưu ý là nó không thể tăng giá trị này loại trừ trường hợp đặc biệt.

Tất cả các khoản bổ sung cần có khả năng trợ giúp cho một kích cỡ gói 128 bytes, chủ yếu chỉ có thể trợ giúp cho kích cỡ gói này. Nếu yêu cầu tạo lập kích cỡ gói cực đại ít hơn 128 bytes thì DXE bị gọi được phép trả lời theo giá trị dành cho 128 bytes. Nếu gọi đầu nối/tiếp

nhận gọi không bao hàm một đáp ứng cho cuộc thử gọi về sự thoả thuận kích cỡ gói, thì điều này hàm ý là DXE bị gọi có thể chấp nhận kích cỡ gói theo yêu cầu. Mặc dù dịch vụ này cho phép các kích cỡ gói khác nhau có thể thu xếp theo mỗi hướng nhưng thông thường thì hai kích cỡ gói này là như nhau.

- *Kích cỡ cửa sổ*

Tham số này rất giống tham số trước. Byte đầu của trường tham số này là kích cỡ cửa sổ cho gói số liệu từ DXE bị gọi. Byte thứ hai là kích cỡ cửa sổ các gói số liệu từ DXE chủ gọi.

Khi sử dụng hệ thống địa chỉ dãy thông thường thì kích cỡ cửa sổ có thể từ 1 tới 7. Còn đối với hệ thống địa chỉ dãy mở rộng thì kích cỡ cửa sổ có thể trong phạm vi từ 1 đến 127.

Hệ thống thoả thuận đối với tham số này cũng nhiều như đối với kích cỡ gói. Ngoài ra DXE bị gọi chỉ có thể chấp thuận giá trị như thế hoặc ít hơn loại trừ trường hợp khi kích cỡ cửa sổ là 1 đã yêu cầu. Tất cả các điều bổ sung cần chấp thuận cho kích cỡ cửa sổ là 2 nhưng không cần thiết cho 1. Giao thức này cho phép DXE bị gọi chuyển về giá trị 2 để trả lời cho sự yêu cầu có giá trị 1 nếu nó không chấp nhận kích cỡ cửa sổ đó.

- *Thoả thuận loại tiếp thông tối thiểu*

Điều này cho phép DXE chủ gọi yêu cầu một mức tiếp thông nào đó cho cuộc gọi. Trường hợp tham số là một byte đơn, nó mã hoá loại tiếp thông (tiềm năng khác nhau) cho mỗi hướng.

- *Trễ chuyển tiếp đầu cuối*

Khoản này cho phép các DXE thiết lập độ trễ cho gói tin từ đầu tới cuối cuộc gọi. Byte đầu của trường tham số này chỉ thị số byte như sau: nó có thể là 2; 4 hoặc 6. Một gói yêu cầu gọi vào/yêu cầu gọi có thể có tất cả 6 byte, còn gói đầu nối/tiếp nhận gọi có thể chỉ có 2 byte.

Hai byte đầu là độ trễ tích góp giữa hai DXE. Hai byte tiếp theo chứa độ trễ chuyển tiếp theo yêu cầu. Hai byte cuối cùng là độ trễ đầu - cuối tối đa có thể chấp nhận được. Toàn bộ các giá trị này được biểu thị theo ms.

- *Thoả thuận xúc tiến số liệu*

Dịch vụ này có thể được sử dụng để quyết định dịch vụ xúc tiến số liệu có hiệu lực không. Số liệu cần xúc tiến được tải đi ở các gói ngắt, vì vậy nó quyết định số liệu các gói ngắt có thể được dùng hay không.

Trường tham số này có thể phức tạp thêm. Bit 1 là 0 nếu dịch vụ không được sử dụng, hoặc bit 1 là 1 nếu sử dụng dịch vụ này. Các bit khác có thể phân phối cho các dịch vụ khác trong tương lai theo đặc tính kỹ thuật nêu ra. Hiện tại chúng cần được mã hoá toàn số 0.

## 6. Cấp X.25 - 3 - Một số hướng dẫn thực tế

Như ở cấp 2 của X.25, nếu một DTE và DCE sắp trao đổi với nhau thì chúng cần phải được cấu hình chuẩn xác. Việc cắm một PAD theo kiểu X.25 vào một cửa chuyển mạch X.25 và cho nó làm việc thì quả là tuyệt vời.

Khuyến cáo chủ yếu, ngay khi thiết lập tuyến cho một thiết bị là DCE và thiết bị khác là DTE (nên nhớ rằng đa số các trường hợp các thiết bị có thể là cái này hoặc cái kia và cần thiết thiết lập cho cái này hay cái kia) chính là các giá trị LCGN và LCN. Nhiều thiết bị có thể định rõ được một kênh logic được phân phát. Một gói gọi vào/gói yêu cầu gọi có địa chỉ kênh nằm ngoài phạm vi này có thể bị huỷ đi mà không chỉ thị vì sao.

Một cách lý tưởng là các phạm vi kênh được sử dụng ở DTE và DCE cần phải giống nhau. Khi đó DCE sẽ bắt đầu phân bổ các kênh này từ dưới lên còn DTE thì phân bổ các kênh này từ trên xuống. Quy tắc tìm kênh tốt là cần phải bắt đầu ở LCGN4 và LCN0. Điểm cuối cùng của vùng kênh phụ thuộc số lượng tối đa của các kênh logic cần thiết. Giả thiết rằng cần có tối đa 32 kênh logic. Giới hạn trên của vùng kênh sẽ là LCGN4 LCN31.

Mặt khác cần phải xác minh là liệu dịch vụ chọn nhanh có được đáp ứng ở cả hai thiết bị hay không. Nếu một phía tạo ra các gói gọi vào chọn nhanh/gói yêu cầu gọi và phía kia không đáp ứng dịch vụ này thì mọi lần thử gọi sẽ đều bị từ chối.

## 7. X.75

X.75 là một kiểu của X.25, nó được biến đổi để tạo ra hệ thống báo hiệu liên tổng đài cho mạng chuyển mạch gói quốc tế. Thay vì DTE và DCE, X.75 có STE. STE dùng cho thiết bị đầu cuối (TE) báo hiệu. Các STE ở mỗi đầu của tuyến giống nhau về trạng thái. Để nhận dạng, khuyến nghị này gọi một thiết bị là STE-X còn thiết bị kia là STE-Y.

### 7.1. Cấp vật lý của X.75

Giao tiếp cấp vật lý của X.75 được xác định ở G.703 đối với tuyến 64 kbit/s. Nó hơi mập mờ về các tốc độ khác ngoài điều cần phải nói là nó đến tận các mạng lưới để thích ứng nếu chúng đang sử dụng với các giao tiếp khác với G.703. Giao tiếp G.703 được mô tả ở chương 10.

### 7.2. Cấp tuyến của X.75

Cấp tuyến của X.75 tương tự cấp hai của X.25. Có dạng tuyến đơn (SLP) và tuyến ghép (MLP).

### 7.3. Cấp mạng của X.75

Cũng như cấp tuyến, cấp mạng có rất nhiều điều giống cấp 3 của X.25. Các thể thức gói rất giống ở cấp 3 của X.25. Điều khác nhau chủ yếu khi thao tác là bất cứ DTE/DCE không đối xứng ở X.25 đều bị loại ra ở X.75. Điều này thể hiện là chỉ gói yêu cầu gọi thay cho một gói yêu cầu gọi và một gói gọi vào. Các gói khác thì thực sự giống nhau; chỉ có một kiểu cho mỗi gói.

Ngoài ra ở trường dịch vụ của các gói yêu cầu gọi, yêu cầu đầu nối, yêu cầu xoá còn có một trường đặc biệt để sử dụng mạng. Nó sử dụng cho báo hiệu nhờ các cơ quan quản lý mạng thay cho các DTE liên quan tới cuộc gọi.

**MỤC LỤC**

<b>X.25. Giao thức mạng chuyển mạch gói</b> .....	1
1 Mở đầu.....	2
2. X.25 (84) cấp 1 - Cấp vật lý .....	2
3. X.25 (84) Cấp 2 - Cấp tuyến số liệu.....	3
3.1. Thẻ thức khung của LAPB.....	3
3.2. Các kiểu khung LAPB .....	5
3.3 Các trường (vùng) N (R) và N (S) .....	6
3.4 Bit P.....	6
3.5. Thao tác cấp tuyến số liệu .....	8
3.6. Các tham số hệ thống .....	10
4.Cấp X.25 thứ 2 - Một số gợi ý thực tế .....	10
5. Cấp X.25 (84) cấp 3 - cấp mạng (lớp mạng).....	11
5.1. Khuôn mẫu gói cấp mạng .....	11
5.2. Các kiểu gói cấp mạng .....	12
5.3. Các địa chỉ dây cấp mạng.....	17
5.4. Trường mã nhận dạng khuôn mẫu. ....	18
5.5 Cung đoạn tái khởi động .....	18
5.6. Thiết lập các cuộc gọi thực .....	20
5.7 Cung đoạn chuyển giao tin.....	26
5.8. Trường mã dịch vụ: .....	30
6. Cấp X.25 - 3 - Một số hướng dẫn thực tế .....	34
7. X.75 .....	35
7.1. Cấp vật lý của X.75 .....	35
7.2. Cấp tuyến của X.75 .....	35
7.3. Cấp mạng của X.75 .....	35