

# CHƯƠNG 4

## TẦNG GIAO VẬN (UDP/TCP)

ET4230

**TS. Trần Quang Vinh**  
**BM. Kỹ thuật Thông tin**  
**Viện Điện tử - Viễn thông**  
**Đại học Bách Khoa Hà Nội**  
**[vinhtq@mail.hut.edu.vn](mailto:vinhtq@mail.hut.edu.vn)**



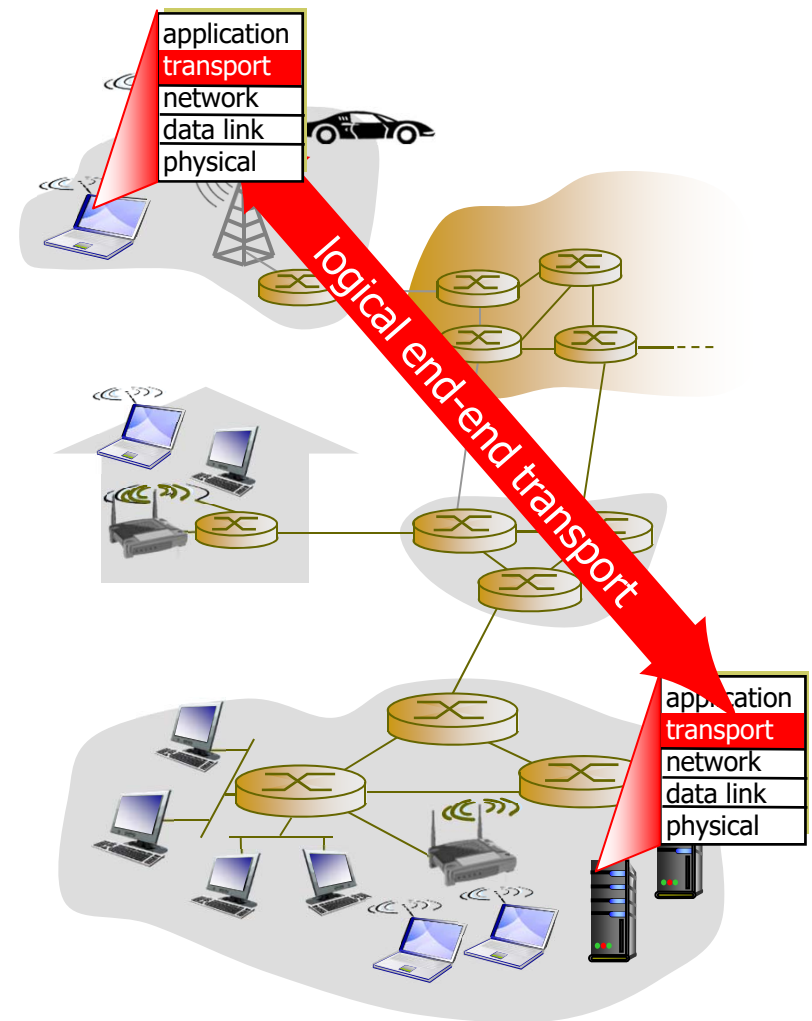
## Chương 4: Nội dung

---

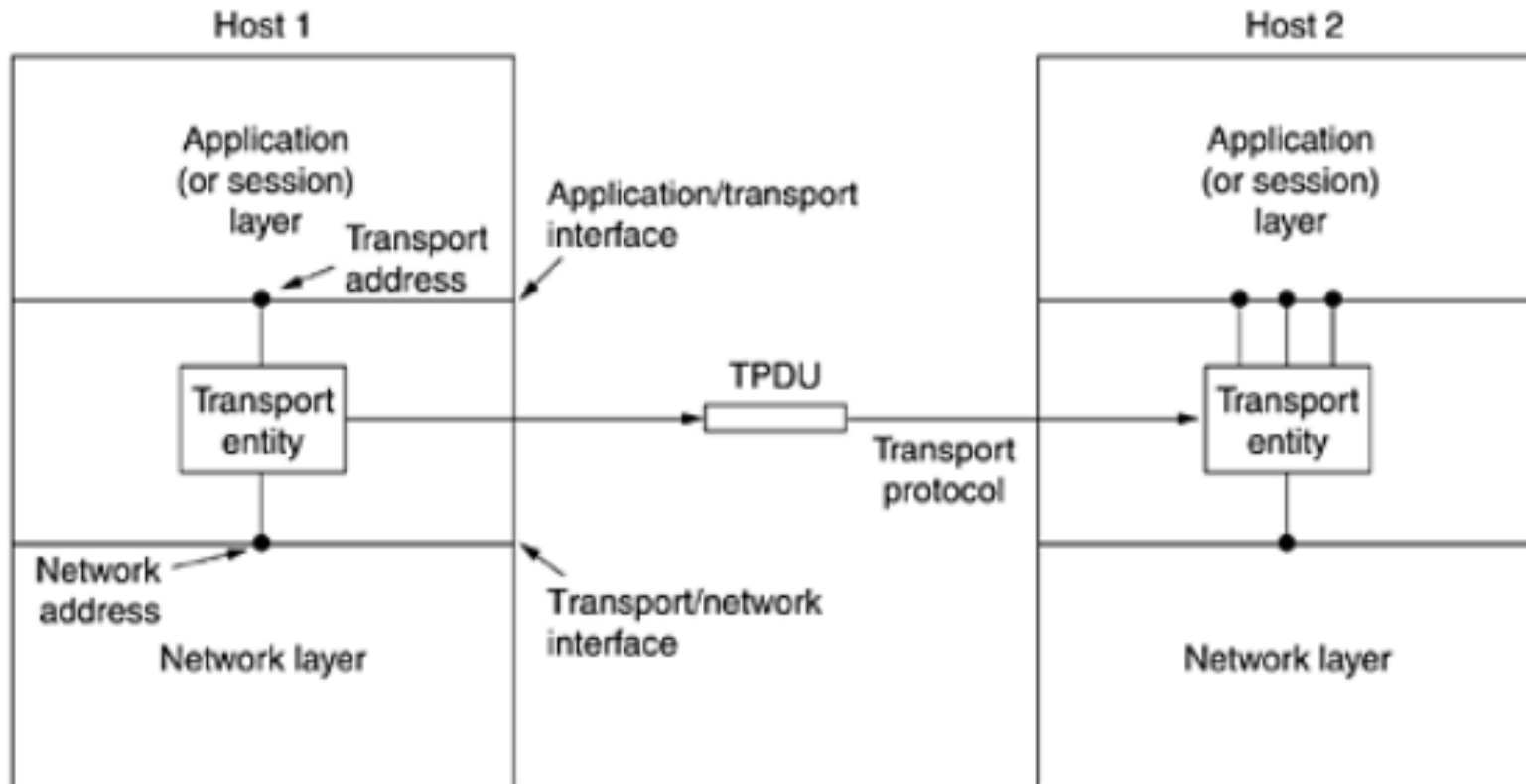
- Ví trí, vai trò, chức năng lớp giao vận
- UDP
- TCP
- Điều khiển luồng và chống tắc nghẽn trong TCP

# Transport services and protocols

- ❖ provide *logical communication* between app processes running on different hosts
- ❖ transport protocols run in end systems
  - send side: breaks app messages into *segments*, passes to network layer
  - rcv side: reassembles segments into messages, passes to app layer
- ❖ more than one transport protocol available to apps
  - Internet: TCP and UDP

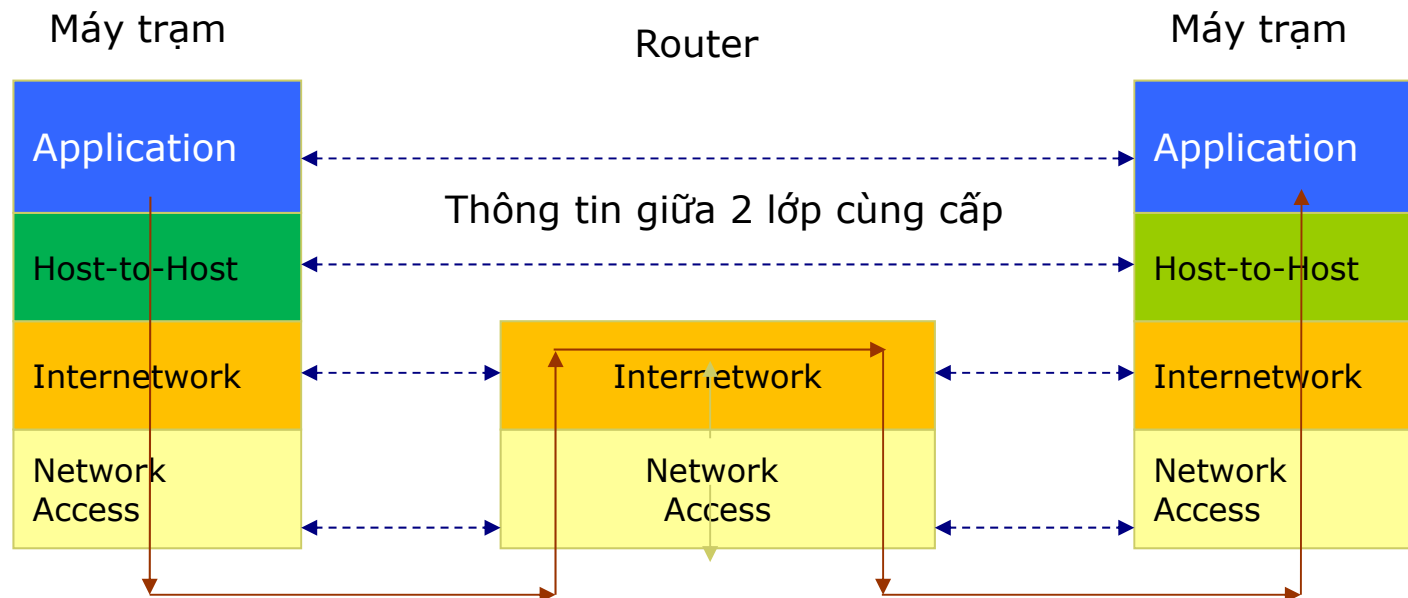


# Transport services and protocols



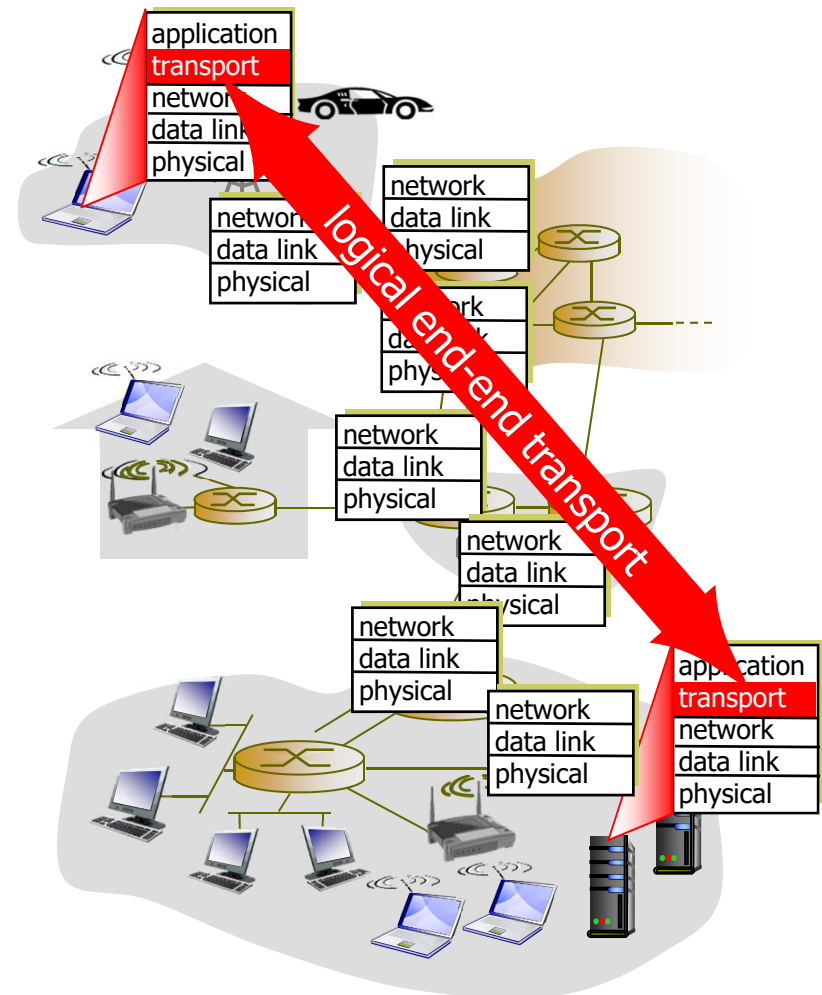
# Transport vs. network layer

- ❖ *network layer*: logical communication between hosts
- ❖ *transport layer*: logical communication between processes
  - relies on, enhances, network layer services



# Internet transport-layer protocols

- ❑ reliable, in-order delivery (TCP)
  - ❑ congestion control
  - ❑ flow control
  - ❑ connection setup
- ❑ unreliable, unordered delivery: UDP
  - ❑ no-frills extension of “best-effort” IP
- ❑ services not available:
  - ❑ delay guarantees
  - ❑ bandwidth guarantees



# Dịch vụ giao vận và ứng dụng

---

Ứng dụng	Giao thức ứng dụng	Giao thức giao vận
E-mail	SMTP	TCP
remote terminal access	Telnet	TCP
Web	HTTP	TCP
File transfer	FTP	TCP
streaming multimedia	giao thức riêng (e.g. RealNetworks)	UDP hoặc TCP
Internet telephony	giao thức riêng (e.g., Vonage, Dialpad)	UDP

# Chức năng lớp giao vận

## □ Mô hình Internet

- Chức năng chính: định tuyến và chuyển tiếp
- Truyền tin không tin cậy
  - Không có cơ chế đảm bảo thứ tự gói
  - Không có cơ chế đảm bảo gói sẽ được truyền đến nơi nhận

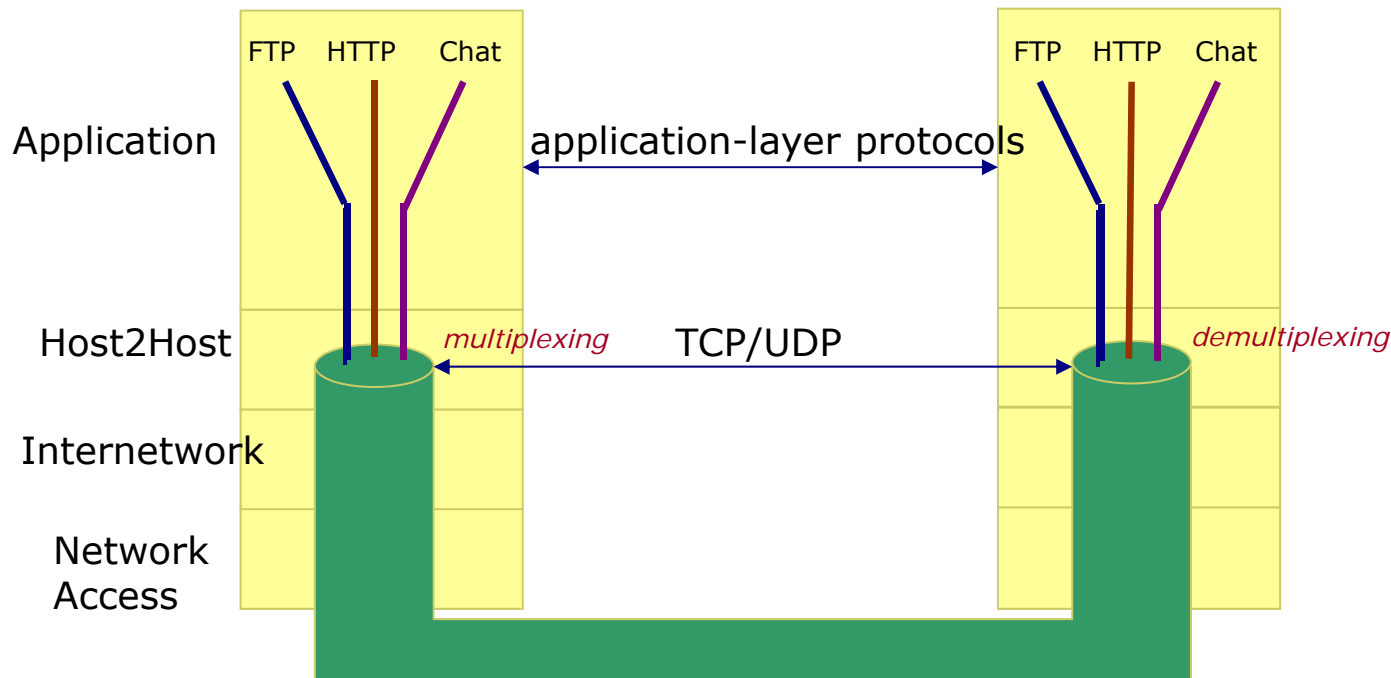
→ Cần các chức năng bổ sung giữa “Application” và “Internetwork”

Application	http, SIP, H.323, RTP, FTP, SMTP, POP, SNMP .v.v.	Ứng dụng truyền thông: truyền file, ứng dụng thời gian thực, web, email, VoIP .v.v.
Host-to-Host	TCP/UDP	
Internetwork	IP, ICMP routing protocols	Định tuyến và chuyển tiếp, truyền tin không tin cậy
Network Access	IEEE 802.x	Điều khiển truy nhập kênh, biến đổi dòng bit theo kênh vật lý



# Ghép kênh

- Chia nhỏ dữ liệu của người sử dụng thành các segment lớp host-to-host
- Ghép nhiều luồng dữ liệu lớp ứng dụng vào để truyền trên mạng IP



# Ghép kênh

---

- Phương pháp ghép kênh:
  - Mỗi ứng dụng (hay tiến trình) có một địa chỉ cổng (port number) khác nhau
    - Các cổng được định nghĩa trong RFC 1700.
  - Các ứng dụng trao đổi dữ liệu qua “socket”. Mỗi socket được phân biệt qua bộ 2 tham số:
    - Địa chỉ port: là một số 16 bit
    - Địa chỉ IP
  - Socket được quản lý bởi hệ điều hành → Có thể tạo, trao đổi dữ liệu qua socket thông qua các API của hệ điều hành

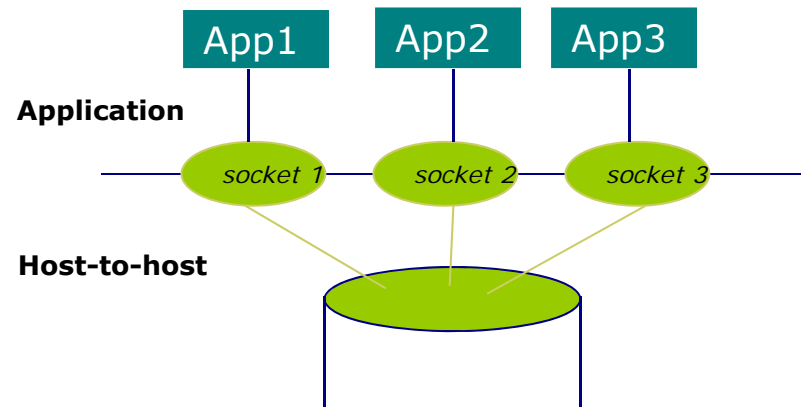
# SOCKET

## □ Một số cổng điển hình:

### □ 0 – 1023: cổng mặc định định nghĩa bởi IANA

- FTP: 20, TCP
- SSH: 22, TCP/UDP
- Telnet: 23, TCP
- SMTP: 25, TCP
- DNS: 53, TCP/UDP
- HTTP: 80, TCP/UDP

### □ Từ 1024 - $2^{16}$ : do người sử dụng tự định nghĩa



# Cơ chế tăng độ tin cậy

---

## □ Độ tin cậy:

### □ Kiểm tra và chống lỗi

- thêm các mã kiểm tra lỗi phần dữ liệu (CRC .v.v.)
  - ARQ (auto-repeat request)
  - FEC (forwarding error correction)

### □ Chống tắc nghẽn

- RED (Random Early Discarding)
- WRED (Weighted Random Early Discarding)
- ECN (Explicit Congestion Notification)

# Cơ chế tăng độ tin cậy (2)

---

## □ ARQ:

- Trao đổi dữ liệu thông qua cơ chế: gửi – phúc đáp

- Phân loại:

  - Stop-and-Wait, Go-Back-N, Selective Repeat

- Chức năng:

  - Phát hiện lỗi gói, phát hiện mất gói, sắp xếp thứ tự gói

  - Điều khiển luồng

    - Thay đổi kích thước cửa sổ trượt  $W$

    - Thay đổi thời gian ACK để điều chỉnh RTT

- Nhận xét:

  - ARQ không thích hợp cho các dịch vụ thời gian thực:

    - RTT lớn

    - Băng thông của luồng dữ liệu bị giới hạn bởi  $W$

# Cơ chế tăng độ tin cậy (3)

---

## □ FEC:

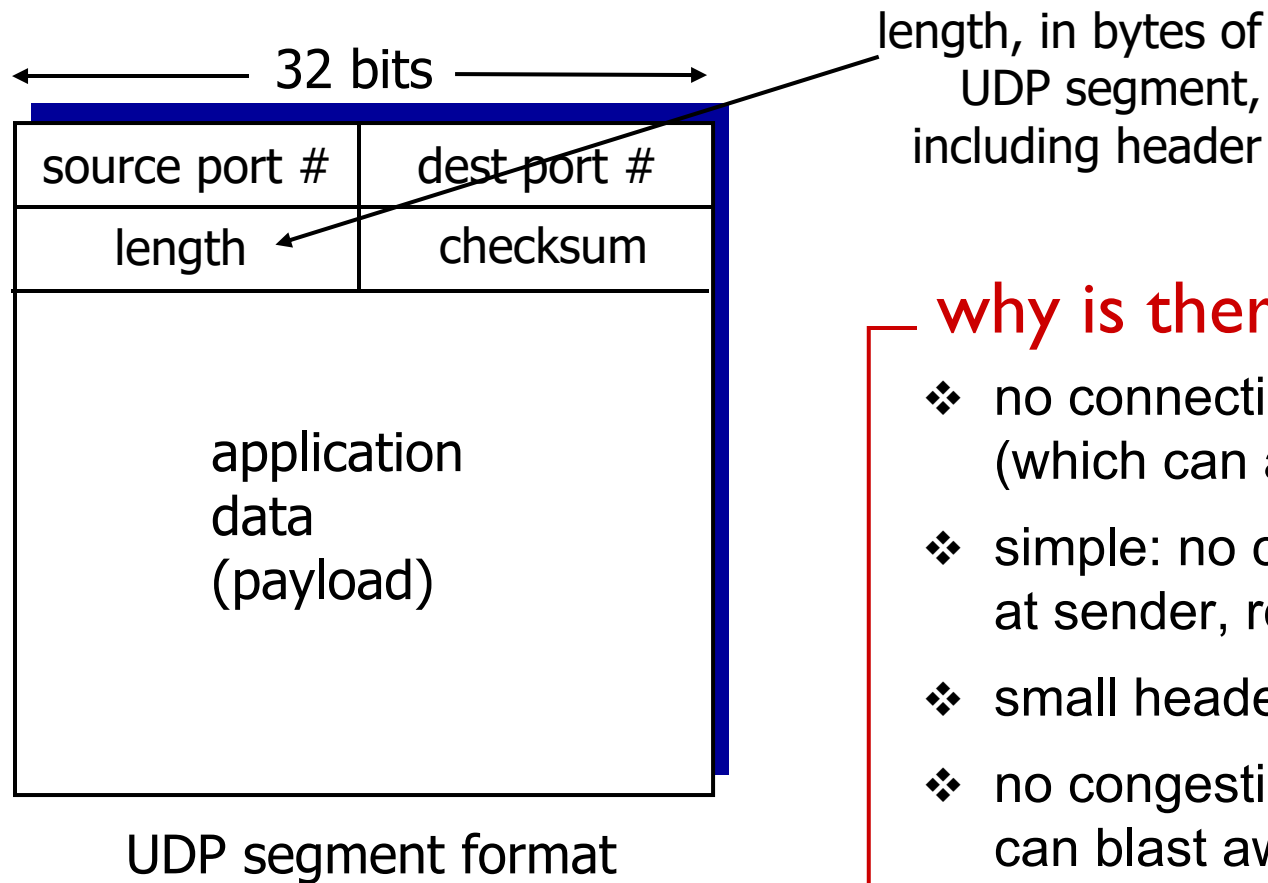
- Bên phát không phát lại gói
- Bên thu có chức năng:
  - Kiểm tra lỗi
  - Nếu phát hiện gói lỗi:
    - Bên thu sửa lỗi (xác định vị trí bit lỗi)
    - Nếu lỗi không sửa được: hủy gói
- Nhận xét:
  - FEC thích hợp cho các dịch vụ thời gian thực

# Giao thức UDP

---

- UDP – User Datagram Protocol
  - RFC 768
  - Gói tin UDP – UDP datagram
- Đặc điểm:
  - Sử dụng cơ chế không liên kết (connectionless): không cần thiết lập một phiên truyền giữa nguồn – đích
- Các ứng dụng sử dụng UDP
  - Các ứng dụng không cần độ tin cậy cao
    - Streaming Media, DNS (Domain Name System), Trivial File Transfer Protocol (TFTP)
  - Các ứng dụng thời gian thực
    - Real-time Multiplayer, Game trực tuyến
    - VoIP

# UDP: segment header



## why is there a UDP?

- ❖ no connection establishment (which can add delay)
- ❖ simple: no connection state at sender, receiver
- ❖ small header size
- ❖ no congestion control: UDP can blast away as fast as desired



# UDP checksum

---

**Goal:** detect “errors” (e.g., flipped bits) in transmitted segment

**sender:**

- treat segment contents, including header fields, as sequence of 16-bit integers
- checksum: addition (one's complement sum) of segment contents
- sender puts checksum value into UDP checksum field

**receiver:**

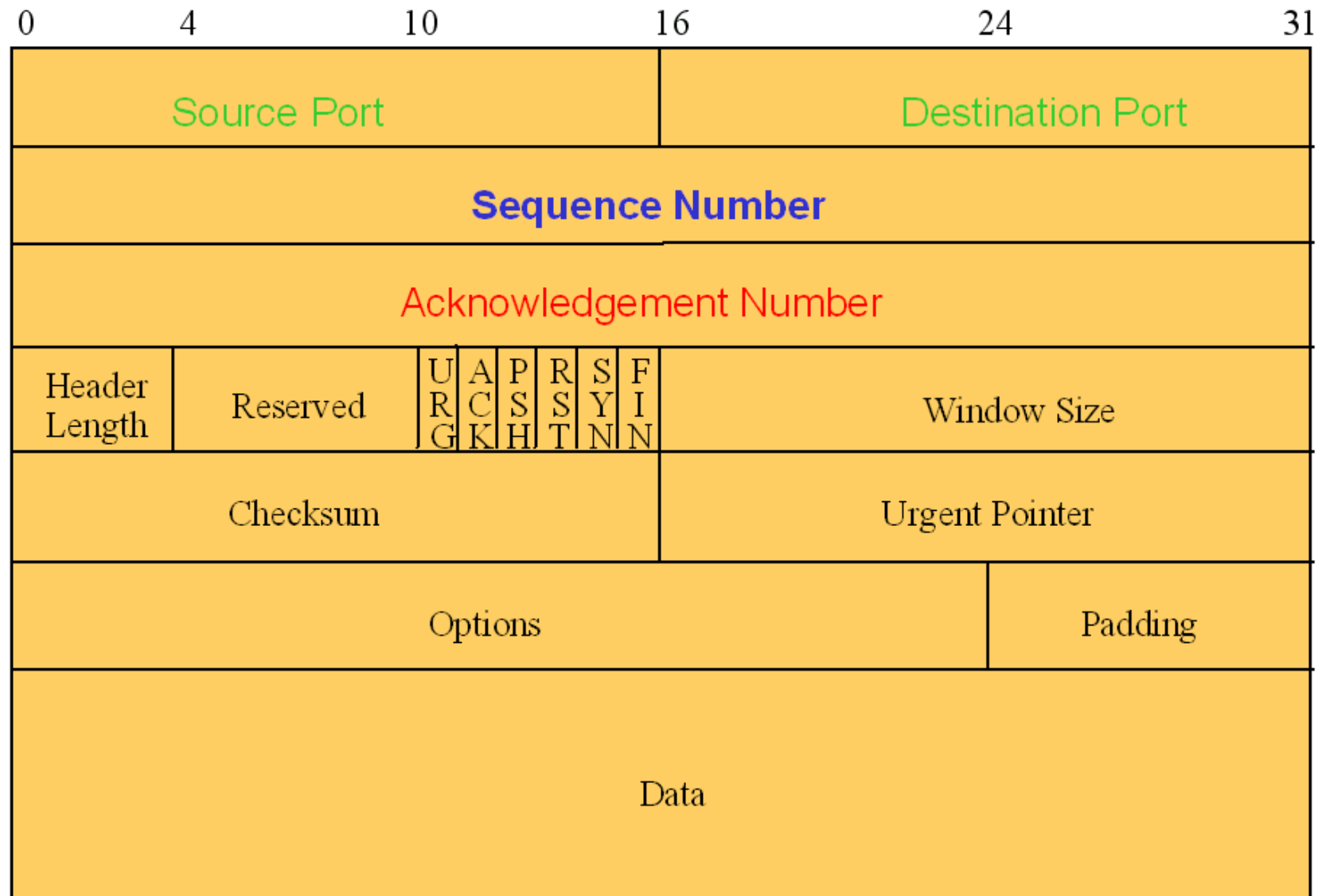
- compute checksum of received segment
- check if computed checksum equals checksum field value

# Giao thức TCP

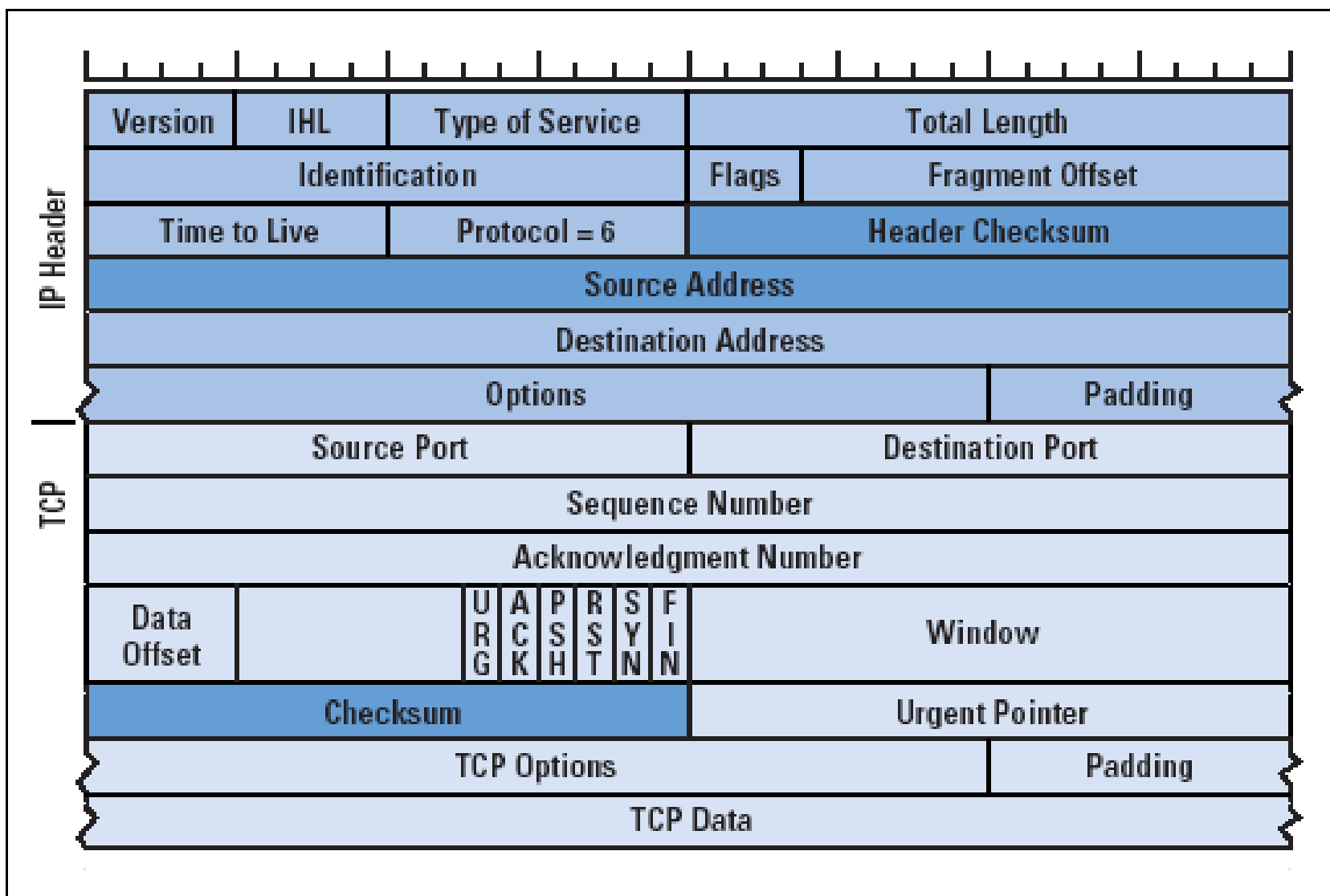
---

- TCP: giao thức truyền tin cậy (reliable):
  - Cơ chế báo nhận (ACK)
  - Mã chống lỗi để bảo vệ dữ liệu
  - Sequence number: phát hiện gói mất và gói không đúng thứ tự
  - Cơ chế timeout
  - Cơ chế sắp xếp lại thứ tự gói ở đầu thu
  - Cơ chế điều khiển luồng (flow control)
  - Cơ chế chống tắc nghẽn (congestion control)
  - một số dịch vụ tầng ứng dụng sử dụng giao thức TCP như: Email, FTP, SSH, Telnet, Web-browsers ....

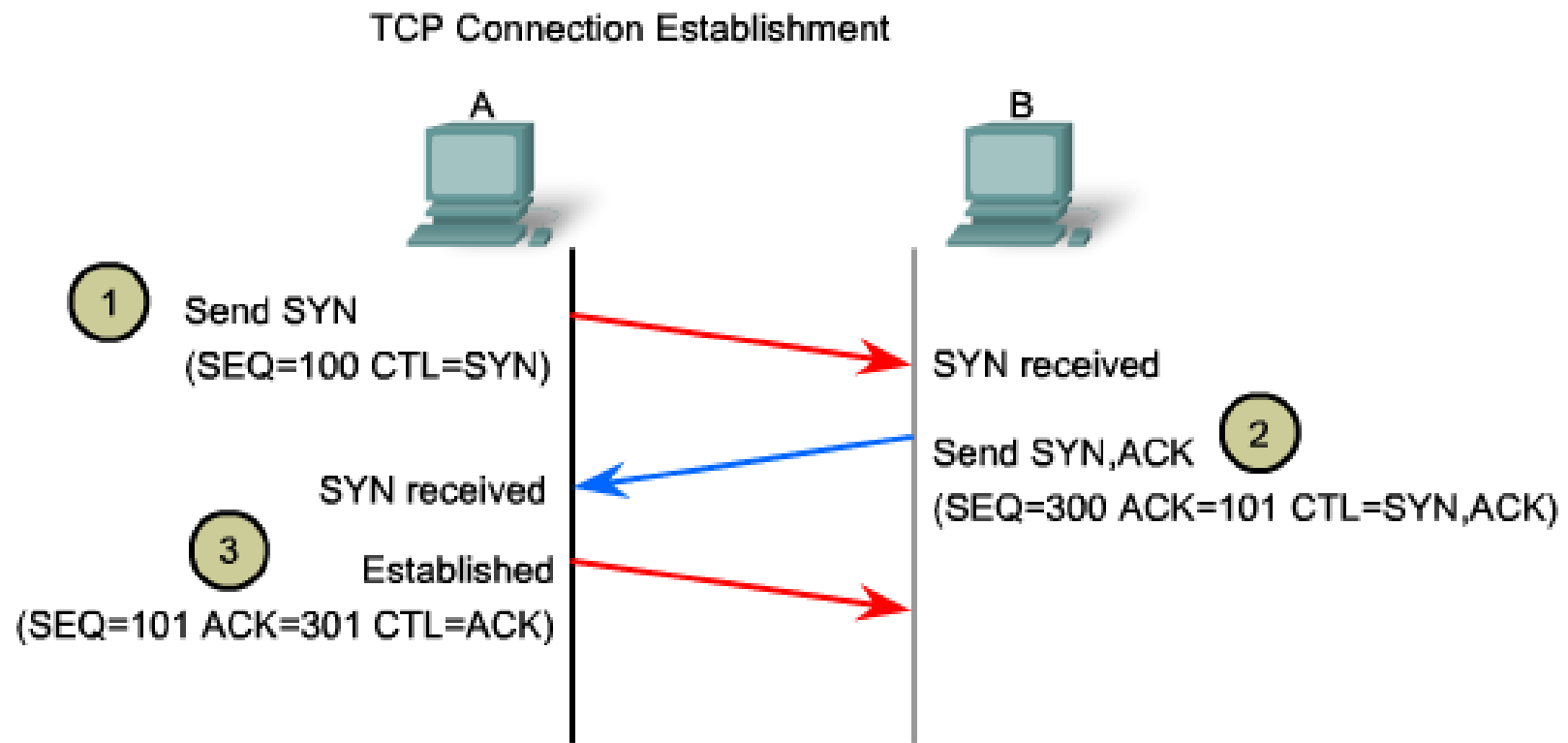
# Cấu trúc gói tin TCP



# TCP/IP HEADER



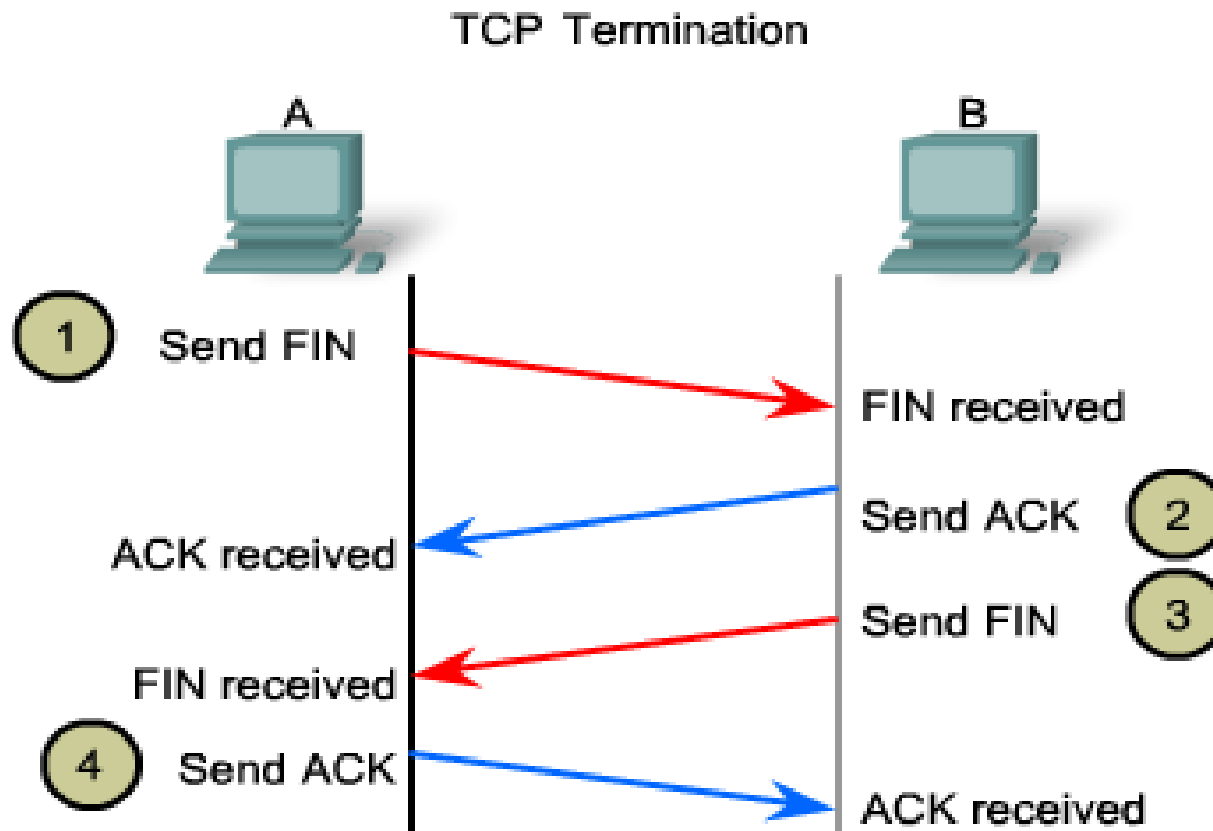
# Thiết lập kết nối: 3-way handshake



CTL = Which control bits in the TCP header are set to 1

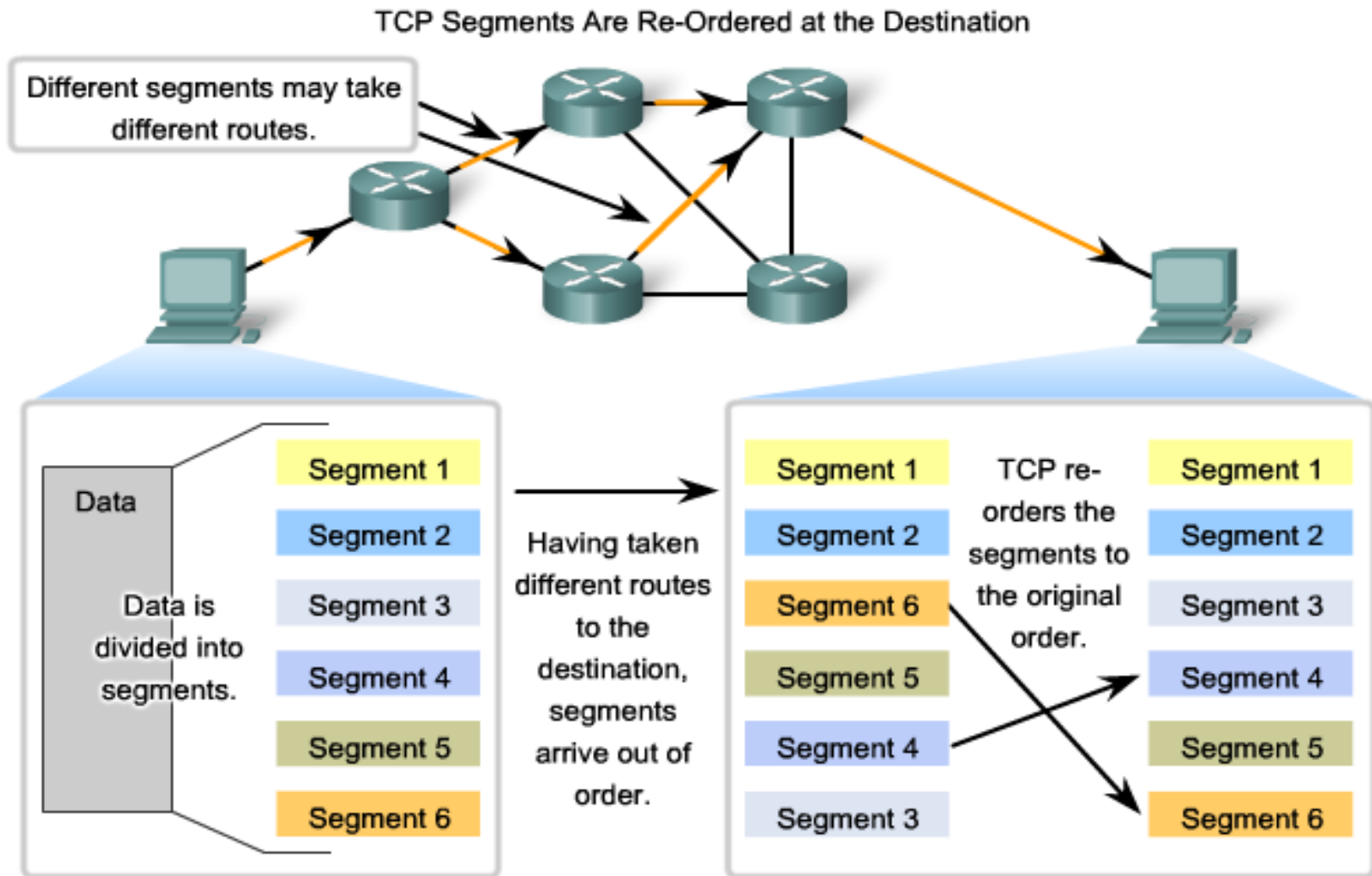
A sends ACK response to B.

# Hủy bỏ kết nối: 2-way handshake



A sends ACK response to B.

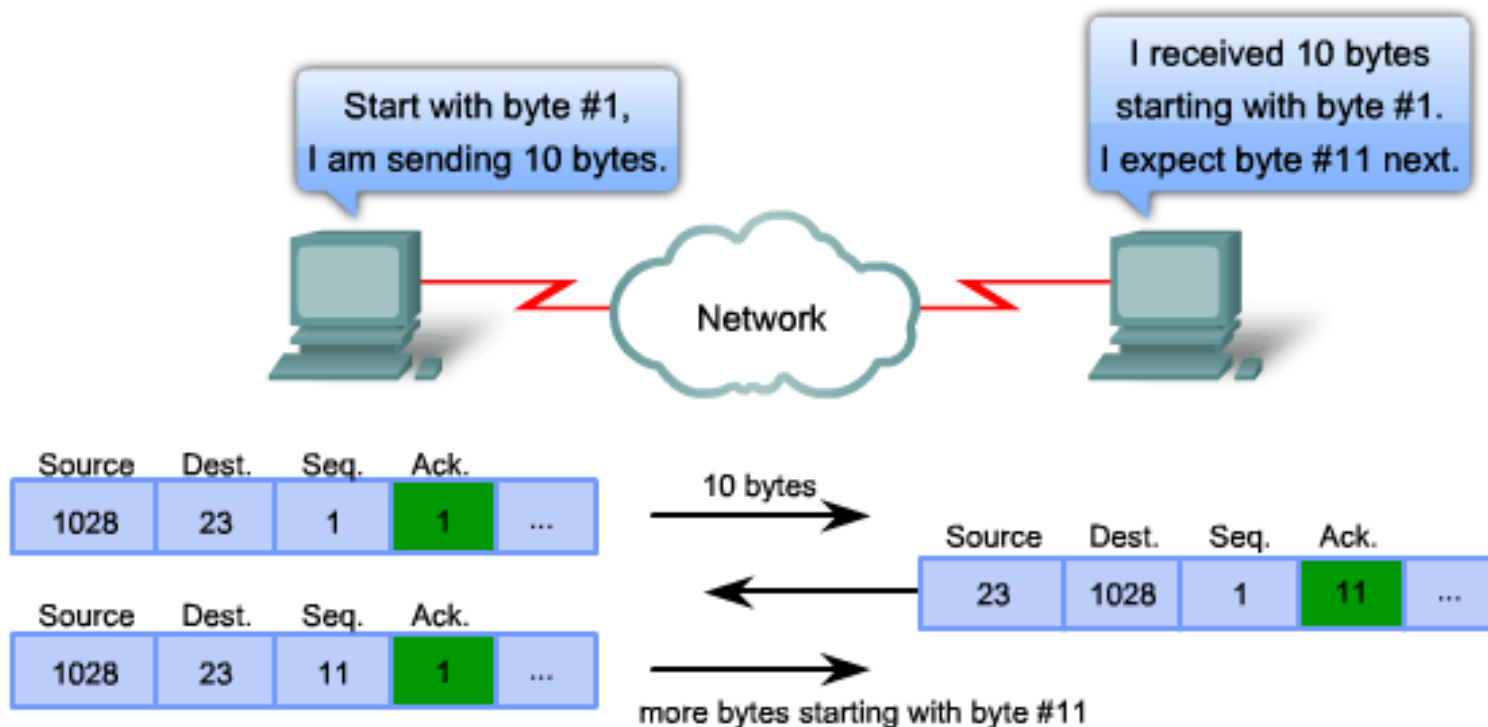
# Sắp xếp lại gọi tin trong TCP



# Cơ chế truyền có xác nhận

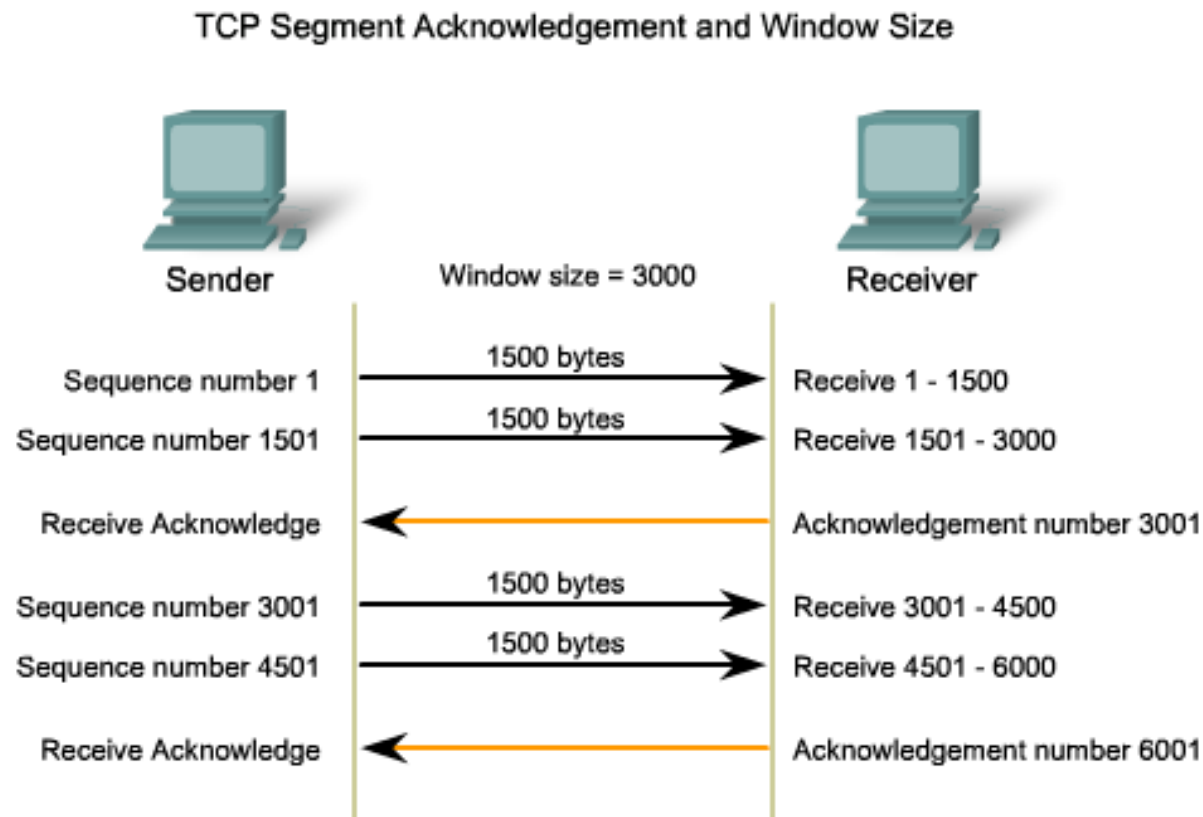
Acknowledgement of TCP Segments

Source Port	Destination Port	Sequence Number	Acknowledgement Numbers	...
-------------	------------------	-----------------	-------------------------	-----





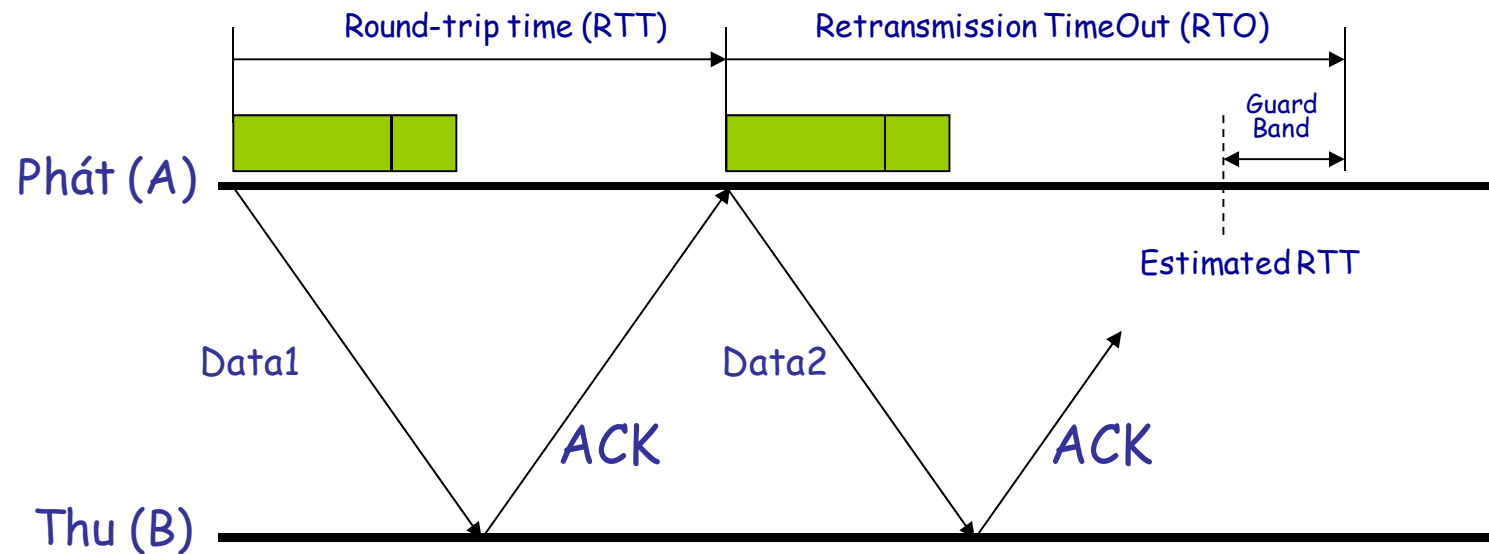
# Cơ chế truyền có xác nhận (tiếp)



The **window size** determines the number of bytes sent before an acknowledgment is expected.

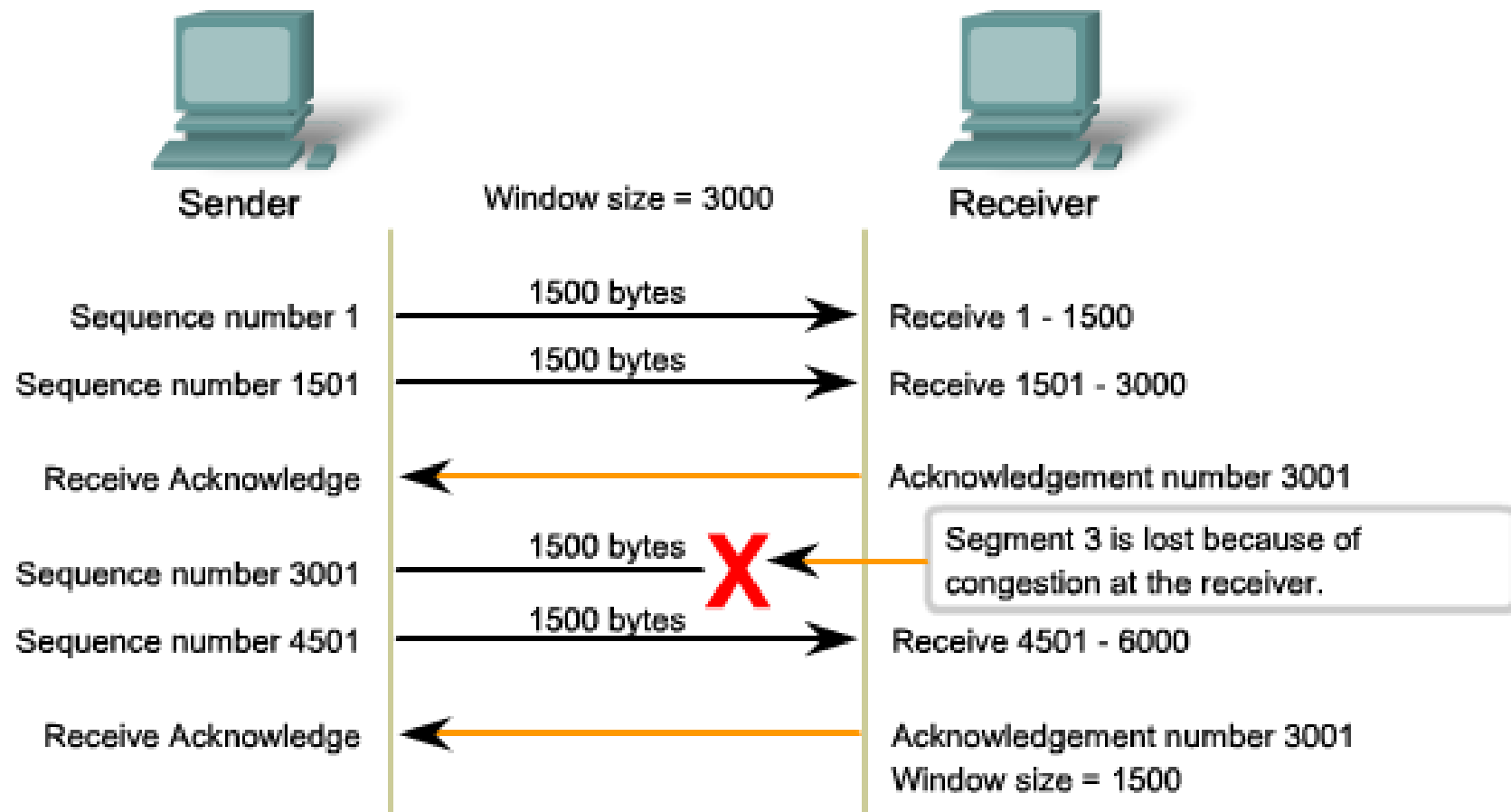
The **acknowledgement** number is the number of the next expected byte.

# Quá trình điều khiển luồng tránh tắc nghẽn



# Quá trình điều khiển luồng tránh tắc nghẽn

## TCP Congestion and Flow Control



# BÀI TẬP

---