

TẦNG VẬN CHUYỂN (Computer Network)

Trình bày: TS Ngô Bá Hùng

Mục đích

- Chương này nhằm giới thiệu với người đọc những nội dung sau:
 - Vai trò của tầng vận chuyển và các chức năng mà tầng vận chuyển cung cấp cho tầng ứng dụng
 - Ý nghĩa và cơ chế thiết lập nối kết và giải phóng nối kết cho các nối kết điểm – điểm
 - Chi tiết về hay giao thức TCP và UDP thuộc tầng vận chuyển

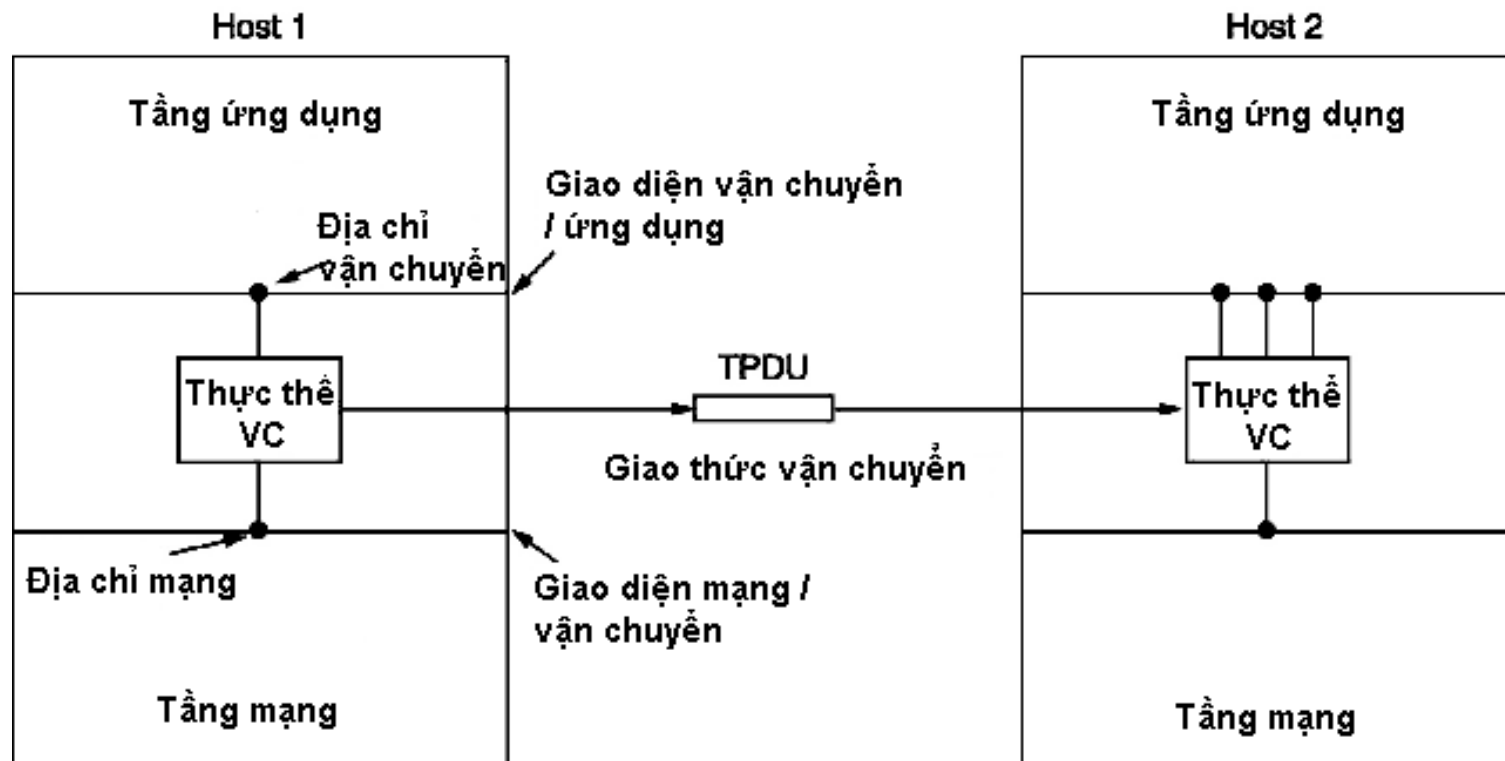
Yêu cầu

- Sau khi học xong chương này, người học phải có được những khả năng sau:
 - Biện luận được sự cần thiết của tầng vận chuyển trong một liên mạng
 - Giải thích được cơ chế thiết lập và xóa nối kết các cuộc giao tiếp điểm-điểm của tầng vận chuyển
 - Trình bày được nguyên tắc hoạt động của hai giao thức TCP và UDP của mạng Internet

Nhiệm vụ của tầng vận chuyển

- Tầng mạng đảm bảo truyền tải kiểu Host -to- Host
- Tầng vận chuyển đảm bảo truyền tải kiểu End point –to- End point
- End point là các chương trình ứng dụng
- Cấp dịch vụ vận chuyển gói tin hiệu quả, tin cậy và tiết kiệm chi phí cho người dùng

Vị trí của tầng vận chuyển



Dịch vụ cung cấp bởi tầng vận chuyển

- Hai kiểu dịch vụ
 - Có nối kết :
 - Thiết lập nối kết,
 - Truyền dữ liệu
 - Hủy nối kết
 - Không nối kết
- Các hàm dịch vụ cơ sở để triệu gọi các dịch vụ vận chuyển và các hàm này là đơn giản, duy nhất và độc lập với các hàm cơ sở ở tầng mạng

Các hàm dịch vụ cơ sở - Có nối kết

Hàm	Gói tin gửi đi	Ý nghĩa
LISTEN	Không có	Nghẽn cho đến khi tiến trình nào đó nối kết tới
CONNECT	Yêu cầu kết nối (Connection Request)	Chủ động yêu cầu thiết lập nối kết đến tiến trình khác
SEND	Dữ liệu (Data)	Gởi thông tin đi
RECEIVE	Không có	Nghẽn cho đến khi một gói tin đến và nhận nó
DISCONNECT	Yêu cầu hủy kết nối (Disconnection Request)	Muốn hủy kết nối với bên đối tác

Các hàm dịch vụ cơ sở - Không nối kết

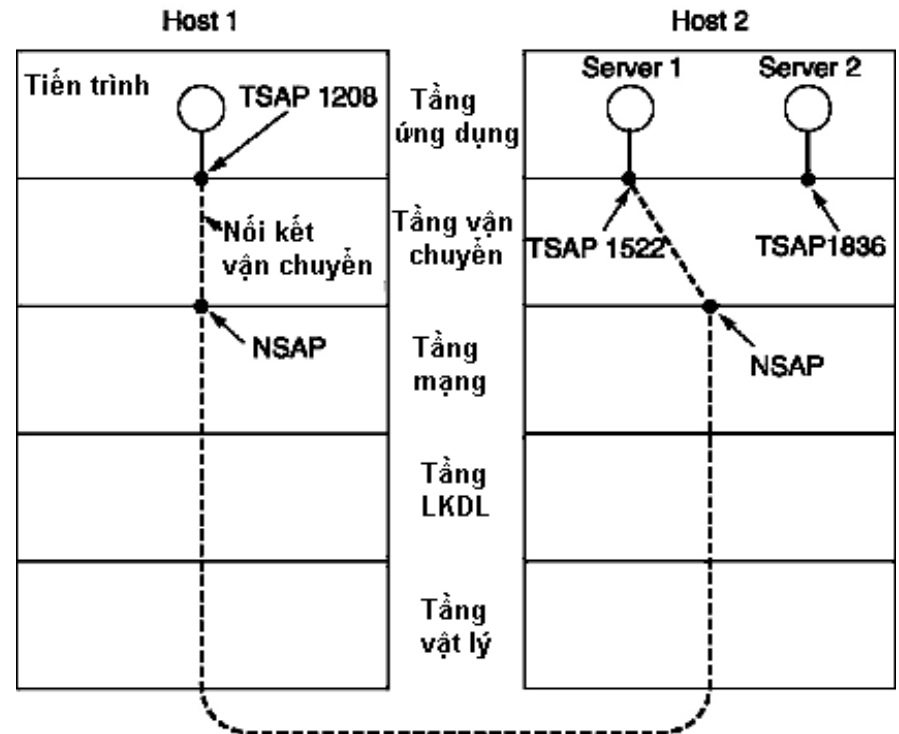
Hàm	Gói tin gửi đi	Ý nghĩa
SEND	Dữ liệu (Data)	Gửi thông tin đi
RECEIVE	Không có	Nghẽn cho đến khi một gói tin đến và nhận nó

Các yếu tố cấu thành giao thức vận chuyển

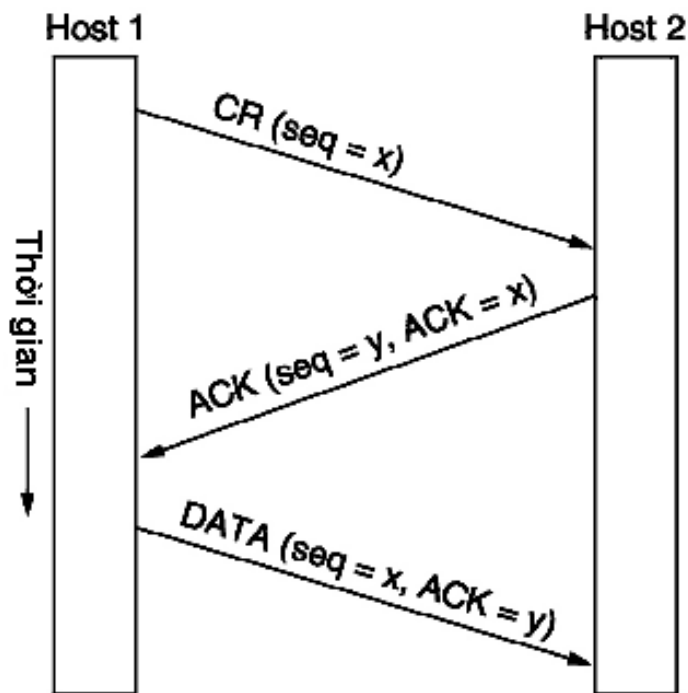
- Điều khiển lỗi, đánh số thứ tự gói tin và điều khiển luồng dữ liệu.
- Môi trường giao tiếp qua một tập các mạng trung gian
- Những vấn đề cần quan tâm:
 - Định địa chỉ các tiến trình trên các host
 - Xử lý những trường hợp mất gói tin, gói tin đi chậm dẫn đến mất kỳ và gửi thêm một gói tin bị trùng lặp,
 - Đồng bộ hóa hai tiến trình đang trao đổi dữ liệu khi mà chúng đang ở rất xa nhau

Định địa chỉ

- Địa chỉ tiến trình là TSAP (Transport Service Access Point).
 - Mạng Internet là dùng số hiệu cổng (port),
 - Mạng ATM là AAL-SAP.
- Tầng mạng được gọi là NSAP

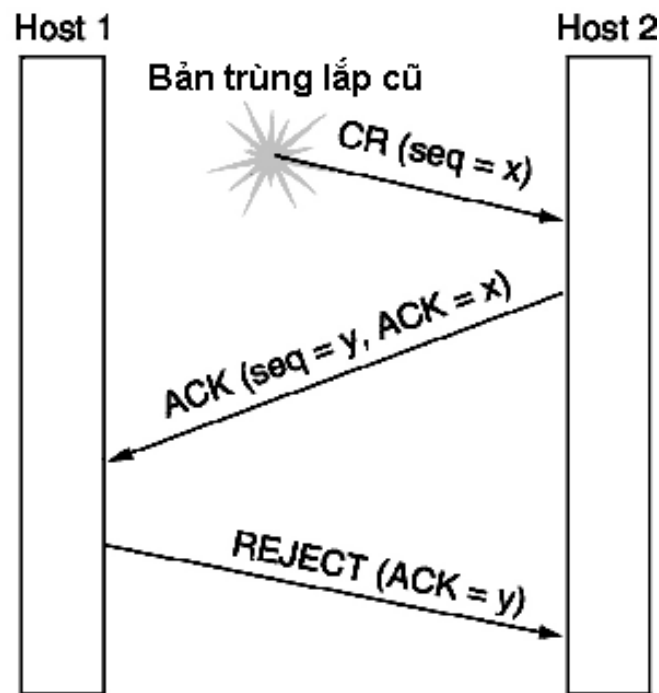


Thiết lập nối kết



(a)

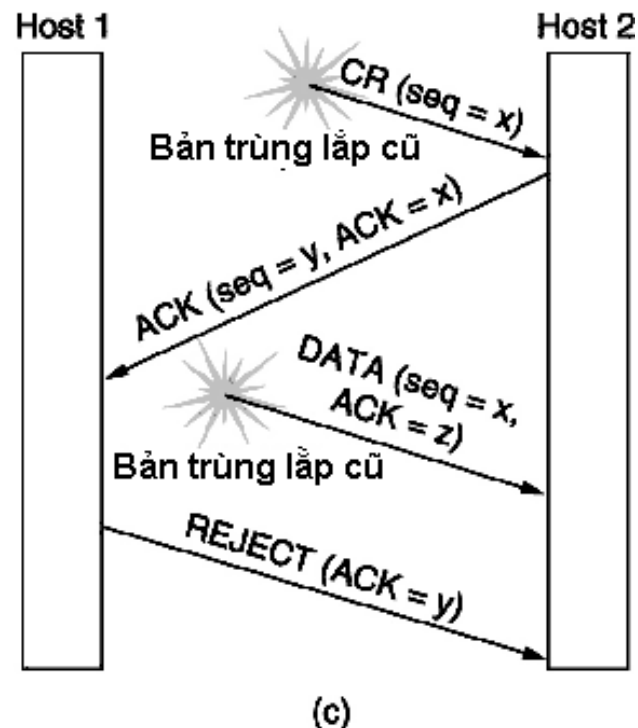
Three-way hand-shake
Hoạt động bình thường.



(b)

Bản CR bị trùng lặp

Thiết lập nối kết

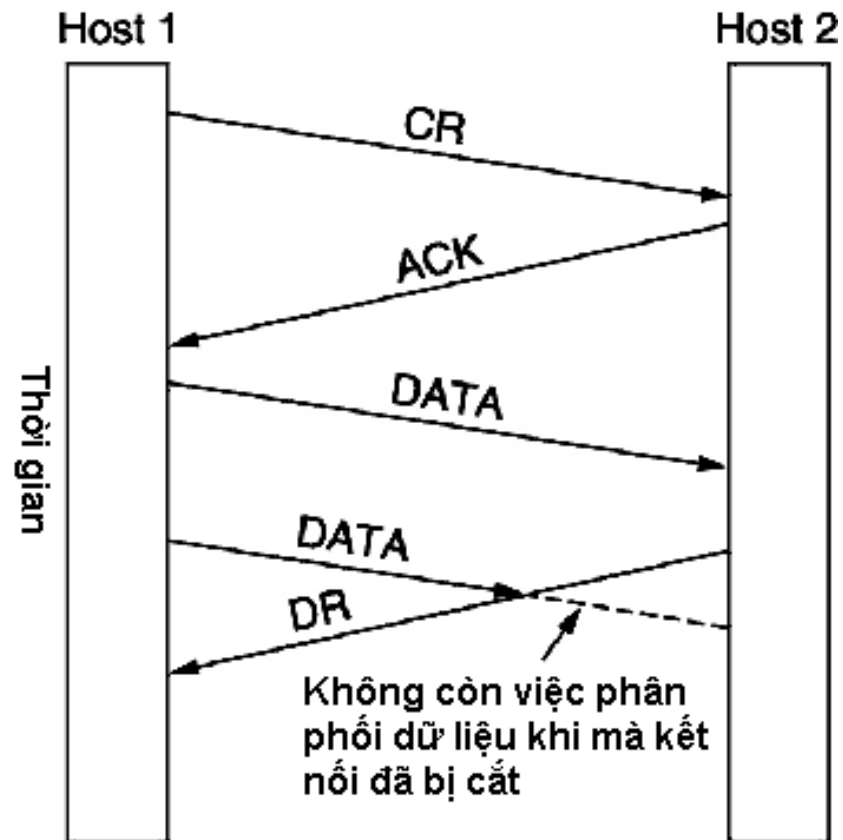


Cả CR và ACK đều bị trùng lặp

Giải phóng nổi kết

- Hai kiểu giải phóng nổi kết:
 - Kiểu dị bộ hoạt động như sau: khi một bên cắt nổi kết, kết nổi sẽ bị hủy bỏ (giống như trong hệ thống điện thoại).
 - Kiểu đồng bộ làm việc theo phương thức ngược lại: khi cả hai đồng ý hủy bỏ nổi kết, nổi kết mới thực sự được hủy

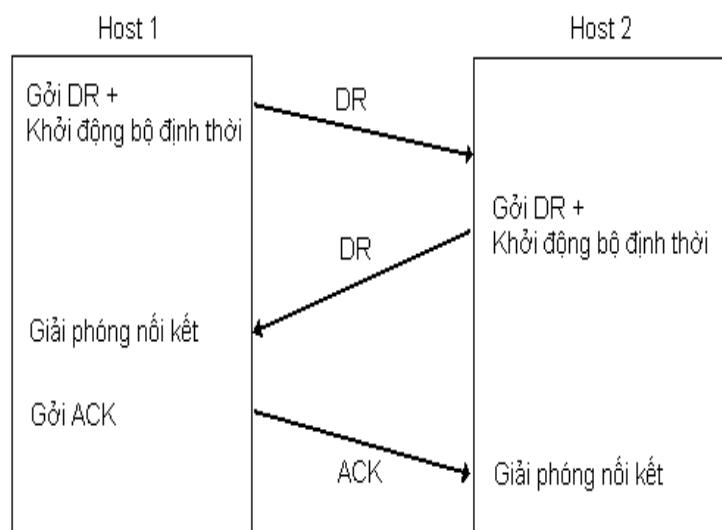
Giải phóng nối kết dị bộ



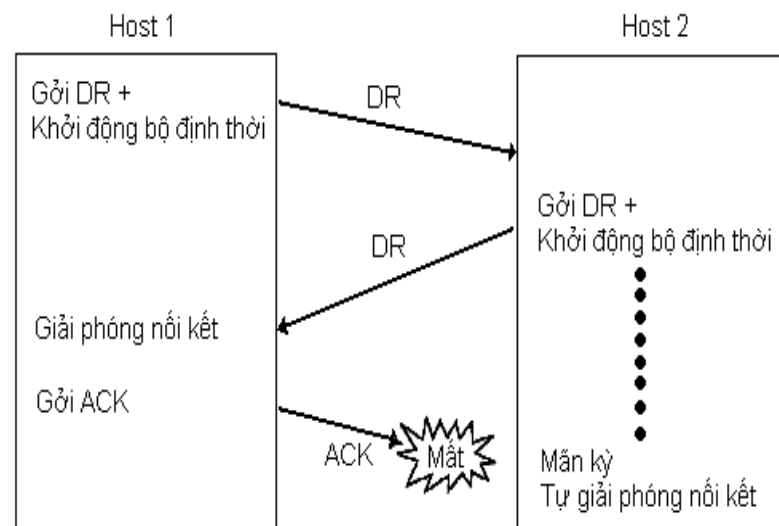
Giải phóng nối kết đồng bộ

- Một nút phải tiếp tục nhận dữ liệu sau khi đã gửi đi yêu cầu giải phóng nối kết (DISCONNECT REQUEST – CR), cho đến khi nhận được chấp thuận hủy bỏ nối kết của bên đối tác đó
- Sử dụng phương pháp hủy nối kết ba chiều cùng với bộ định thời

Giải phóng nối kết đồng bộ

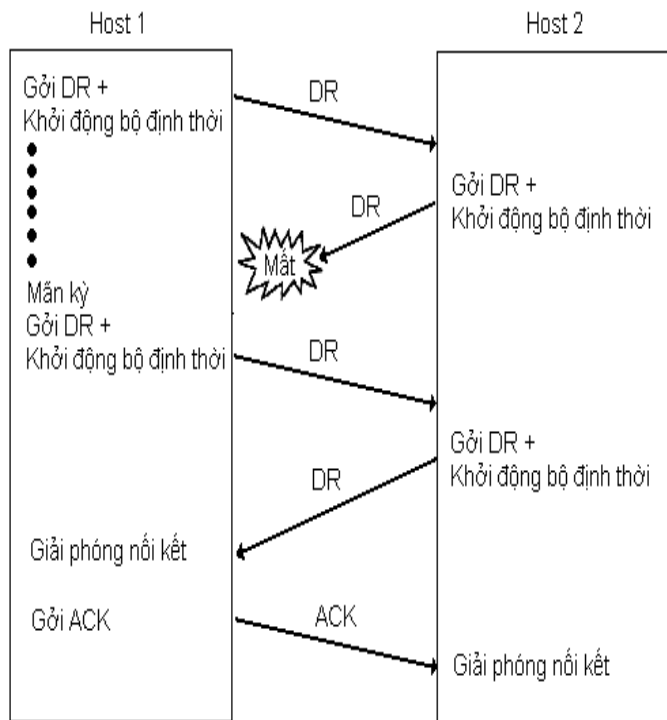


Bình thường

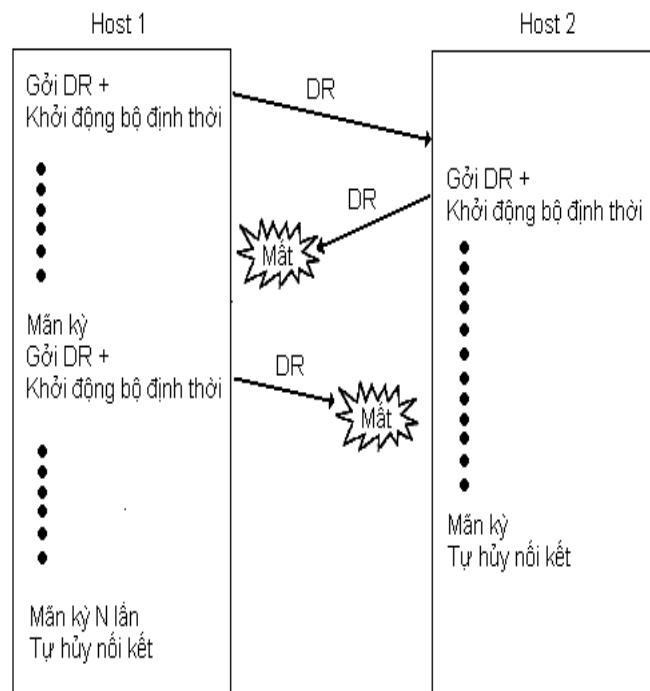


Khung ACK cuối cùng bị mất

Giải phóng nỗi kết đồng bộ



Trả lời bị mất



***Trả lời mất và các gói tin
DR theo sau cũng bị mất***

Điều khiển thông lượng

- Sử dụng giao thức cửa sổ trượt với kích thước cửa sổ của bên gửi và bên nhận là khác nhau
- Cần phải có sơ đồ cung cấp buffer động:
 - Trước tiên, bên gửi phải gửi đến bên nhận một yêu cầu dành riêng số lượng buffer để chứa các gói bên gửi gửi đến.
 - Bên nhận cũng phải trả lời cho bên gửi số lượng buffer tối đa mà nó có thể cung cấp.
 - Mỗi khi báo nhận ACK cho một gói tin có số thứ tự SEQ_NUM, bên nhận cũng phải gửi kèm theo thông báo cho bên gửi biết là lượng buffer còn lại là bao nhiêu để bên gửi không làm ngập bên nhận

Điều khiển thông lượng

	<u>A</u>	<u>Thông điệp</u>	<u>B</u>	<u>Giải thích</u>
1	→	<yêu cầu 8 buffers>	→	A muốn B cung cấp 8 buffers
2	← →	<ack = 0, buf = 4>	← →	B chỉ cấp cho A 4 buffers thôi
3	→	<seq = 0, data = m0>	→	A còn lại 3 buffers
4	→	<seq = 1, data = m1>		A còn lại 2 buffers
5	← →	<seq = 2, data = m2>	← → ...	Thông điệp bị mất, nhưng A nghĩ nó còn 1 buffer
6	→ →	<ack = 1, buf = 3>	→ →	B báo nhận cho thông điệp 0 và 1, còn 3 buffers
7	←	<seq = 3, data = m3>	←	A còn lại 1 buffer

Điều khiển thông lượng

8		<seq = 4, data = m4>		A không còn buffer nào và phải dừng
9		<seq = 2, data = m2>		Thông điệp thứ 2 của A mất kỳ và được truyền lại
10		<ack = 4, buf = 0>		Mọi thứ đã được báo nhận, nhưng A vẫn nghẽn
11	←	<ack = 4, buf = 1>	←	A có thể gửi 1 gói tin thứ 5
12	← →	<ack = 4, buf = 2>	← →	B có thêm 1 buffer nữa
13	→	<seq = 5, data = m5>	→	A còn lại 1 buffer
14	←	<seq = 6, data = m6>	← ←	A nghẽn một lần nữa
15		<ack = 6, buf = 0>		A vẫn còn nghẽn
16	...	<ack = 6, buf = 4>		Khả năng dẫn đến deadlock

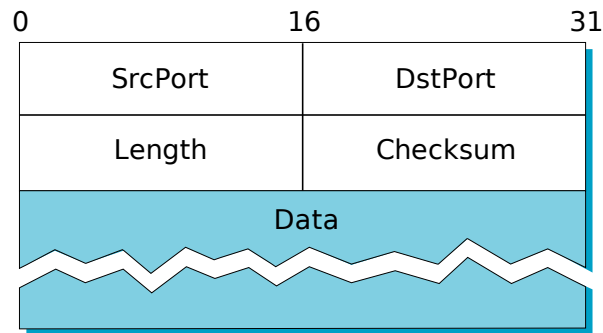
Tầng vận chuyển trong mạng Internet

- **Nhiệm vụ**
 - Đảm bảo việc phân phối thông điệp qua mạng.
 - Phân phối các thông điệp theo thứ tự mà chúng được gửi.
 - Không làm trùng lặp thông điệp.
 - Hỗ trợ những thông điệp có kích thước lớn.
 - Hỗ trợ cơ chế đồng bộ hóa.
 - Hỗ trợ việc liên lạc của nhiều tiến trình trên mỗi host
- **Hỗ trợ hai phương thức hoạt động**
 - Không nối kết (UDP)
 - Có nối kết (TCP)

Giao thức UDP (User Datagram Protocol)

- UDP là dịch vụ truyền dữ liệu dạng không nối kết.
- Không có thiết lập nối kết giữa hai bên truyền nhận,
- Gói tin UDP (segment) có thể xuất hiện tại nút đích bất kỳ lúc nào.
- Các segment UDP tự thân chứa mọi thông tin cần thiết để có thể tự đi đến đích.

Giao thức UDP (User Datagram Protocol)



- Checksum: Là phần kiểm tra lỗi tổng hợp trên phần header, phần dữ liệu và cả phần header ảo.
- Phần header ảo chứa 3 trường trong IP header: địa chỉ IP nguồn, địa chỉ IP đích, và trường chiều dài của UDP.

Giao thức UDP (User Datagram Protocol)

- Phương pháp kiểm tra lỗi

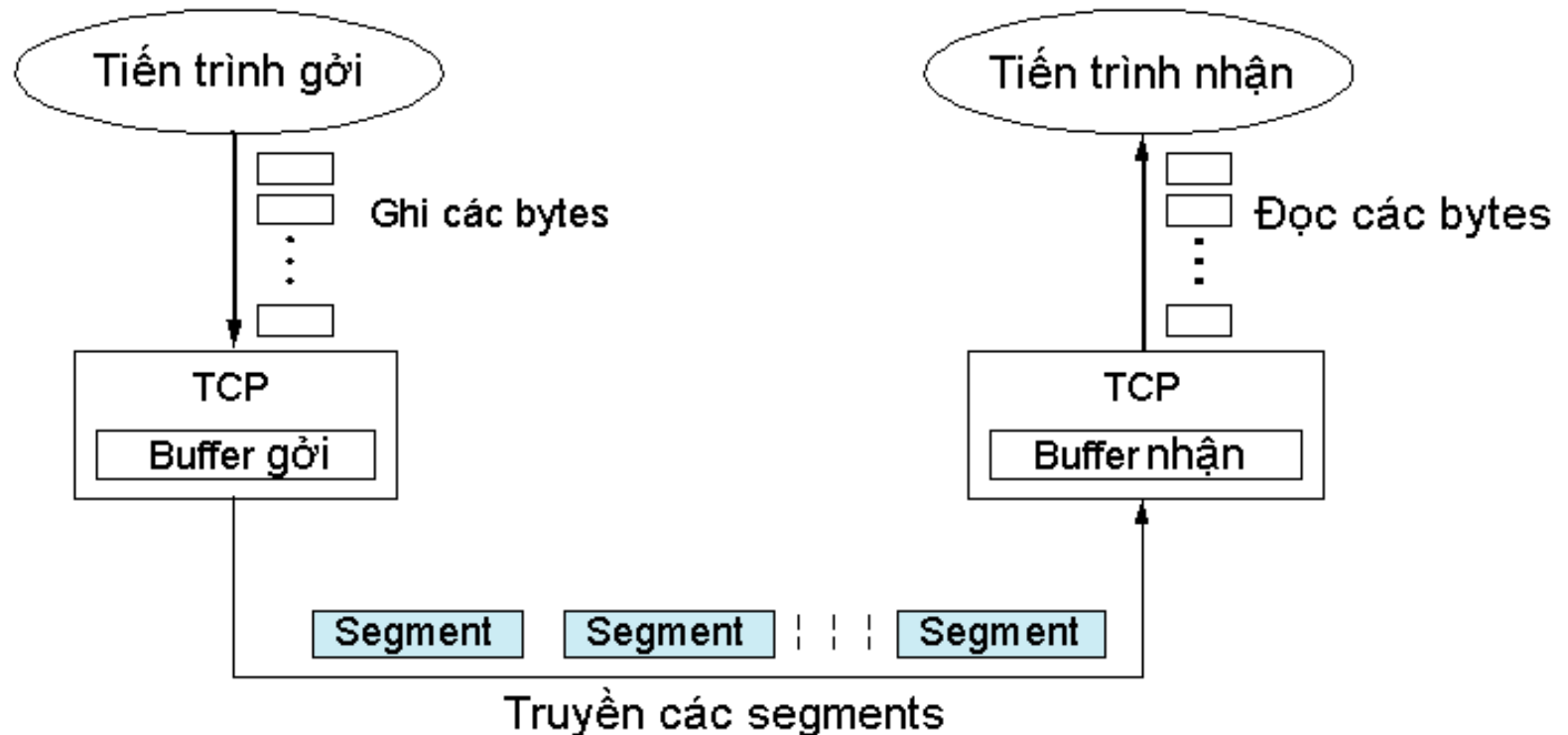
- `u_short`
- `cksum(u_short *buf, int count)`
- `{`
- `register u_long sum = 0;`
- `while (count--)`
- `{`
- `sum += *buf++;`
- `if (sum & 0xFFFF0000)`
- `{`
- `/* bit carry xuất hiện, vì thế gấp và cộng dồn nó lại */`
- `sum &= 0xFFFF;`
- `sum++;`
- `}`
- `}`
- `return ~(sum & 0xFFFF);`
- `}`

Giao thức TCP

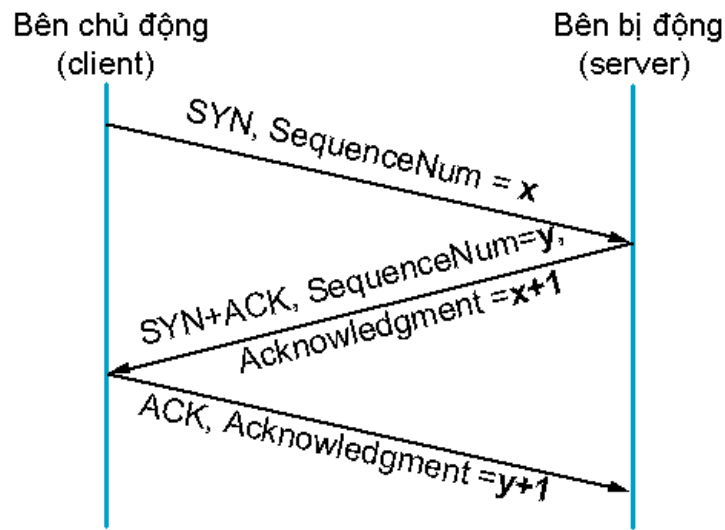
(Transmission Control Protocol)

- TCP là giao thức cung cấp dịch vụ vận chuyển tin cậy, hướng nối kết theo kiểu truyền thông tin bằng cách phân luồng các bytes.
- TCP là giao thức truyền song công, hỗ trợ cơ chế đa hợp
- TCP là giao thức hướng bytes

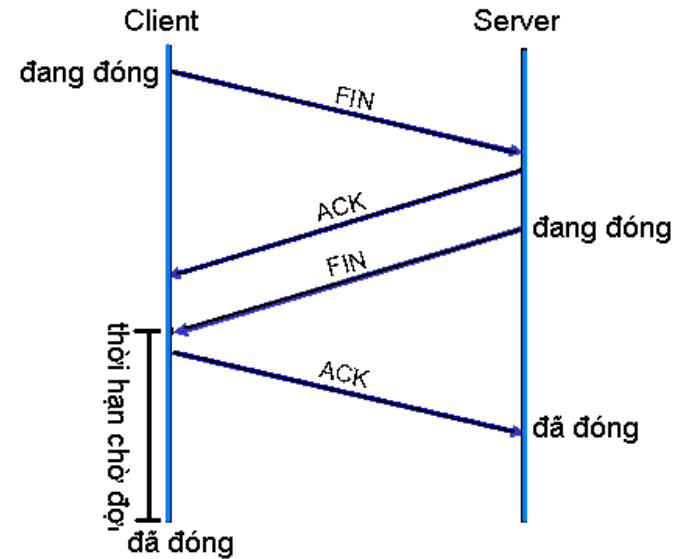
Giao thức TCP (Transmission Control Protocol)



Giao thức TCP (Transmission Control Protocol)



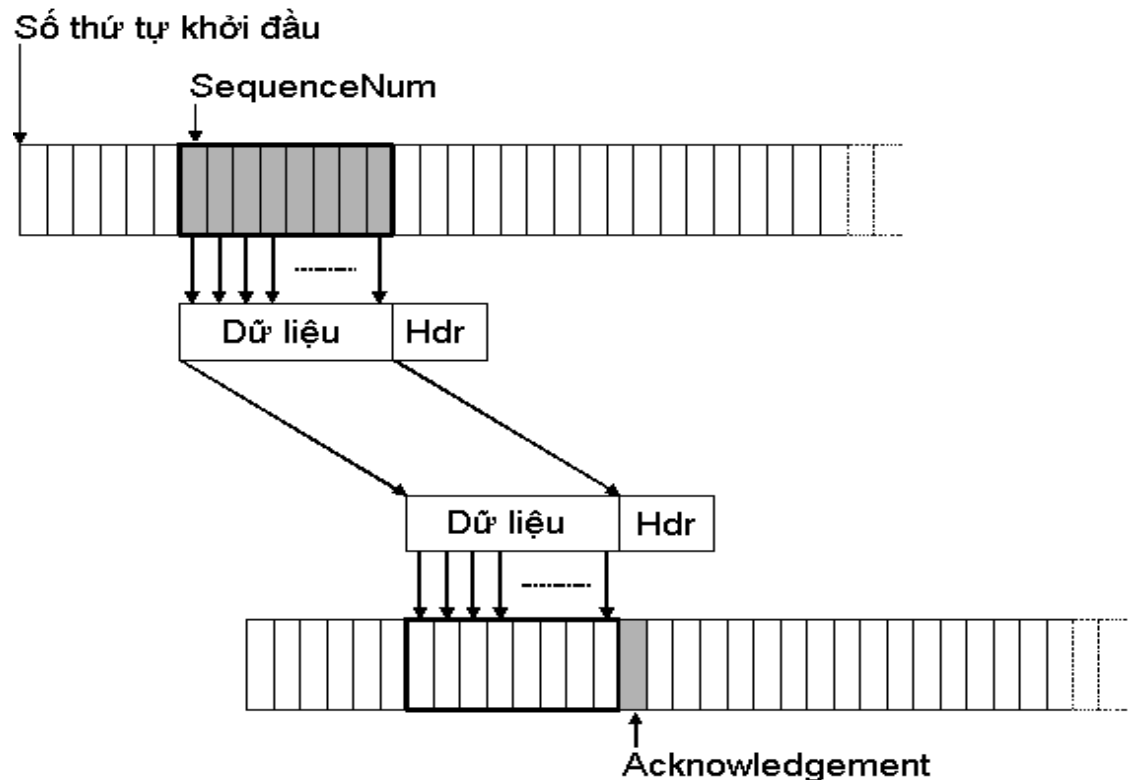
Bắt tay trong TCP



Hủy bắt tay trong TCP

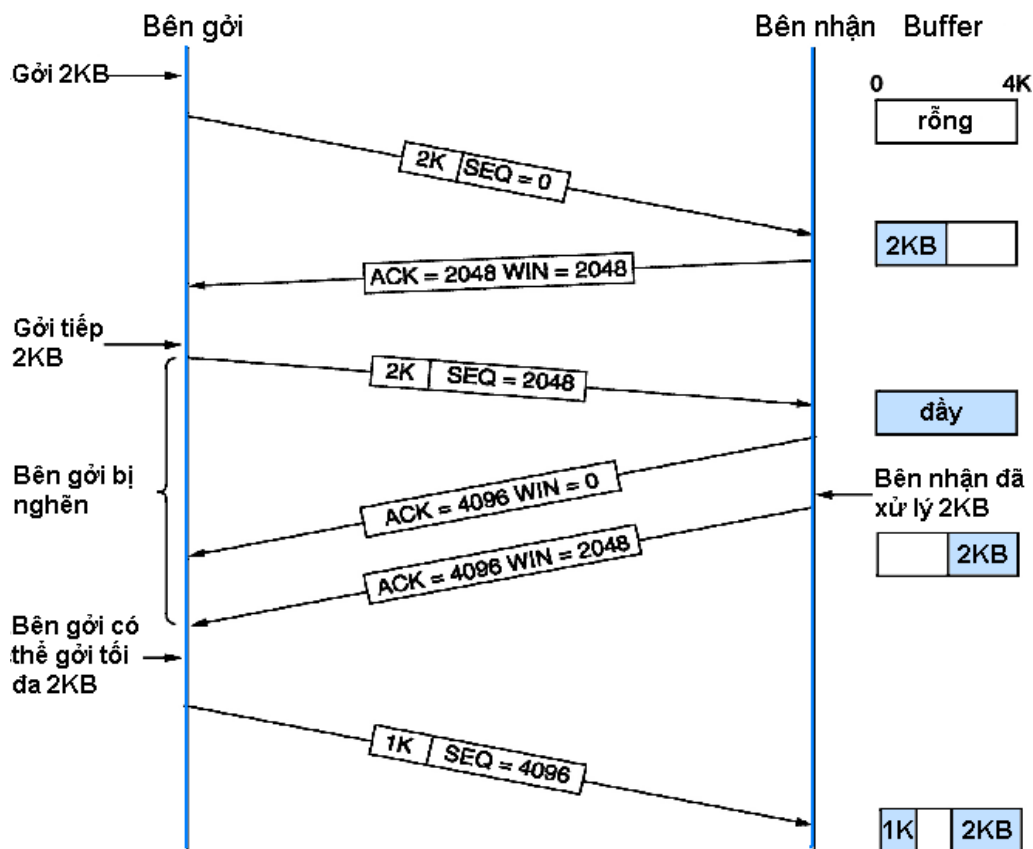
Điều khiển thông lượng trong TCP

- Là giao thức truyền hướng bytes
- Mỗi lần truyền đi một Segment

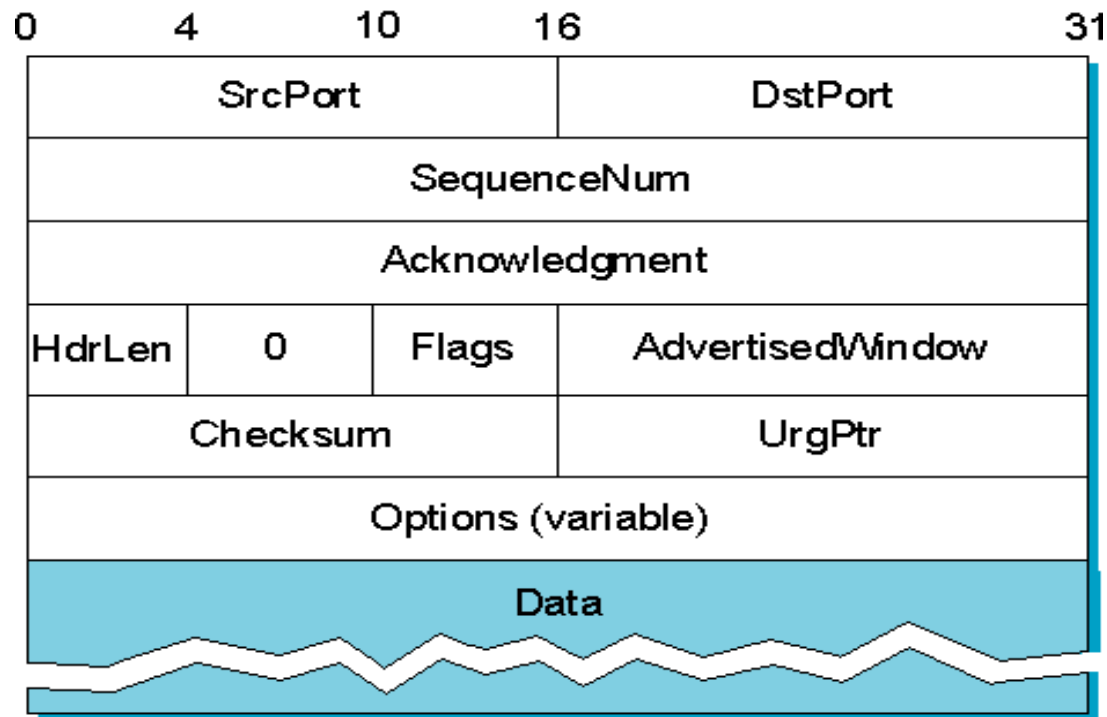


Điều khiển thông lượng trong TCP

- Sử dụng giao thức cửa sổ trượt



Giao thức TCP (Transmission Control Protocol)



Flags = [SYN, FIN, RESET, PUSH, URG, ACK]