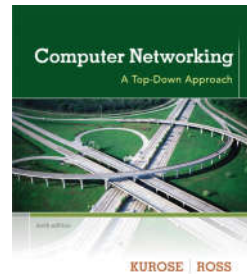


Chương 3 Tầng giao vận



*Computer
Networking: A Top
Down Approach*
7th edition
Jim Kurose, Keith Ross
Addison-Wesley
March 2017

Người dịch: Nguyễn Thanh Thủy

Tài liệu được dịch cho mục đích giảng dạy (được sự đồng ý của tác giả).

© All material copyright 1996-2012
J.F. Kurose and K.W. Ross, All Rights Reserved

Tầng giao vận 3-1

Chương 3: Tầng giao vận

Mục đích:

- ❖ Hiểu được các nguyên lý đằng sau các dịch vụ tầng giao vận:
 - Ghép kênh/phân kênh (multiplexing, demultiplexing)
 - Truyền dữ liệu tin cậy
 - Điều khiển luồng
 - Điều khiển tắc nghẽn
- ❖ Nghiên cứu về các giao thức tầng giao vận trong mạng Internet:
 - UDP: vận chuyển không kết nối
 - TCP: Vận chuyển tin cậy, hướng kết nối
 - Điều khiển tắc nghẽn trong TCP

Tầng giao vận 3-2

Chương 3: Nội dung

3.1 Các dịch vụ tầng giao vận

3.2 Ghép kênh và phân kênh

3.3 Vận chuyển không kết nối: UDP

3.4 Các nguyên lý truyền dữ liệu tin cậy

3.5 Vận chuyển hướng kết nối: TCP

- Cấu trúc đoạn dữ liệu (segment)
- Truyền dữ liệu tin cậy
- Điều khiển luồng
- Quản lý kết nối

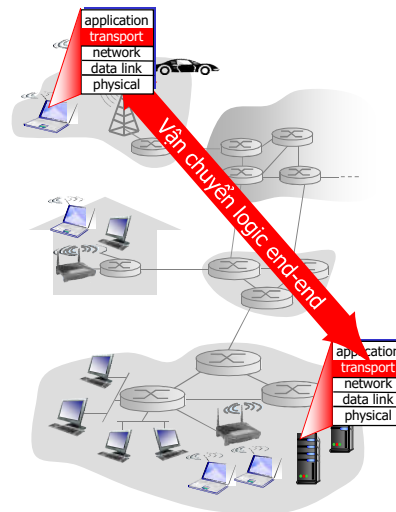
3.6 Các nguyên lý điều khiển tắc nghẽn

3.7 Điều khiển tắc nghẽn TCP

Tầng giao vận 3-3

Các dịch vụ và giao thức tầng giao vận

- ❖ Cung cấp **truyền thông logic** giữa các tiến trình ứng dụng chạy trên các host khác nhau.
- ❖ Giao thức tầng giao vận chạy trên các hệ thống đầu cuối
 - Phía gửi: cắt các thông điệp ứng dụng thành các **đoạn (segment)**, chuyển xuống tầng mạng
 - Phía nhận: Tập hợp lại các đoạn thành các thông điệp, chuyển lên tầng ứng dụng.
- ❖ Có nhiều hơn một giao thức tầng giao vận dành cho các ứng dụng
 - Internet: TCP và UDP



Tầng giao vận 3-4

Tầng giao vận và tầng mạng

- ❖ **Tầng mạng:** truyền thông logic giữa các host
- ❖ **Tầng giao vận:** truyền thông logic giữa các tiến trình
 - Dựa vào và nâng cao các dịch vụ tầng mạng

Tình huống tương tự:

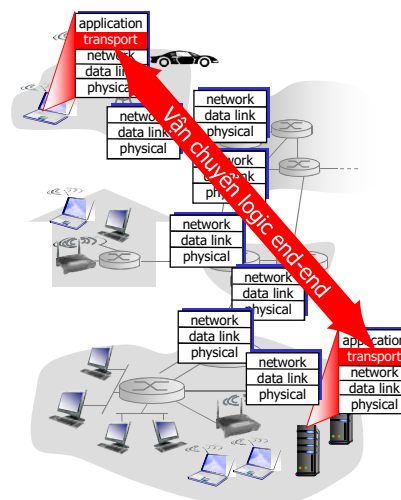
12 em bé nhà Ann gửi thư đến 12 em bé nhà Bill:

- ❖ Các host = Các ngôi nhà
- ❖ Các tiến trình = các em bé
- ❖ Thông điệp ứng dụng = Nội dung bức thư (trong bì thư)
- ❖ Giao thức giao vận = Quy ước giữa các em bé nhà Ann và nhà Bill
- ❖ Giao thức tầng mạng = Dịch vụ bưu điện

Tầng giao vận 3-5

Các giao thức tầng giao vận trên Internet

- ❖ Truyền tin cậy, theo thứ tự: TCP
 - Điều khiển tắc nghẽn
 - Điều khiển luồng
 - Thiết lập kết nối
- ❖ Truyền không tin cậy, không theo thứ tự: UDP
 - Mở rộng của giao thức IP
- ❖ Không có các dịch vụ:
 - Đảm bảo trễ
 - Đảm bảo băng thông



Tầng giao vận 3-6

Chương 3: Nội dung

3.1 Các dịch vụ tầng giao vận

3.2 Ghép kênh và phân kênh

3.3 Vận chuyển không kết nối: UDP

3.4 Các nguyên lý truyền dữ liệu tin cậy

3.5 Vận chuyển hướng kết nối: TCP

- Cấu trúc đoạn dữ liệu (segment)
- Truyền dữ liệu tin cậy
- Điều khiển luồng
- Quản lý kết nối

3.6 Các nguyên lý điều khiển tắc nghẽn

3.7 Điều khiển tắc nghẽn TCP

Tầng giao vận 3-7

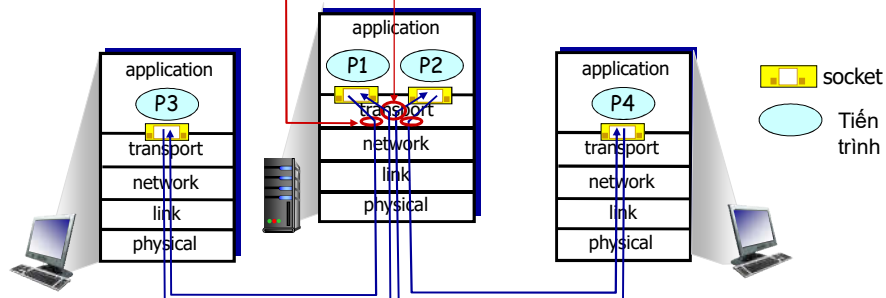
Ghép kênh/Phân kênh

Ghép kênh tại phía gửi:

Xử lý dữ liệu từ nhiều socket, thêm phần tiêu đề tầng giao vận (sau này dùng cho việc phân kênh)

Phân kênh tại phía nhận:

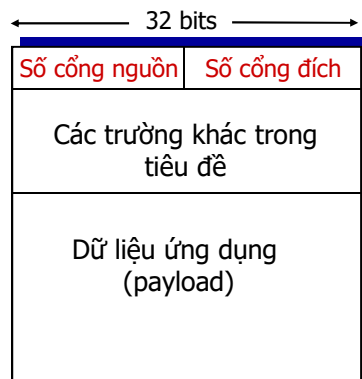
Sử dụng thông tin trong phần tiêu đề để phân phối các đoạn dữ liệu (segment) đã nhận được đến đúng socket



Tầng giao vận 3-8

Việc phân kênh được thực hiện như thế nào?

- ❖ Host nhận các IP datagram
 - Mỗi datagram có địa chỉ nguồn IP và địa chỉ IP đích
 - Mỗi datagram mang một đoạn dữ liệu của tầng giao vận
 - Mỗi segment có số hiệu cổng nguồn và số hiệu cổng đích
- ❖ Host sử dụng **địa chỉ IP & số hiệu cổng** để định hướng đoạn đến socket phù hợp



Định dạng TCP/UDP segment

Tầng giao vận 3-9

Phân kênh hướng không kết nối

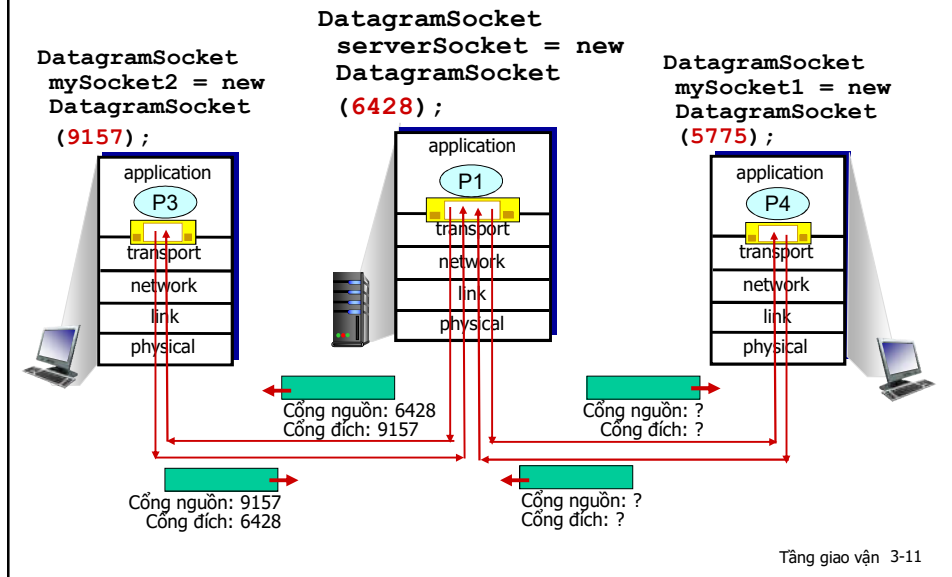
- ❖ Tạo các socket có số hiệu cổng cục bộ của host:

```
DatagramSocket mySocket1  
= new DatagramSocket(12534);
```
- ❖ Khi tạo datagram để gửi vào trong UDP socket, cần phải xác định:
 - Địa chỉ IP đích
 - Số hiệu cổng đích
- ❖ Khi host nhận UDP segment:
 - Kiểm tra số hiệu cổng đích trong segment
 - Định hướng UDP segment tới socket tương ứng với số hiệu cổng đó

Các IP datagram với **cùng số hiệu cổng đích**, nhưng có địa chỉ IP nguồn và/hoặc các số hiệu cổng nguồn khác nhau sẽ được định hướng tới **cùng socket** tại đích

Tầng giao vận 3-10

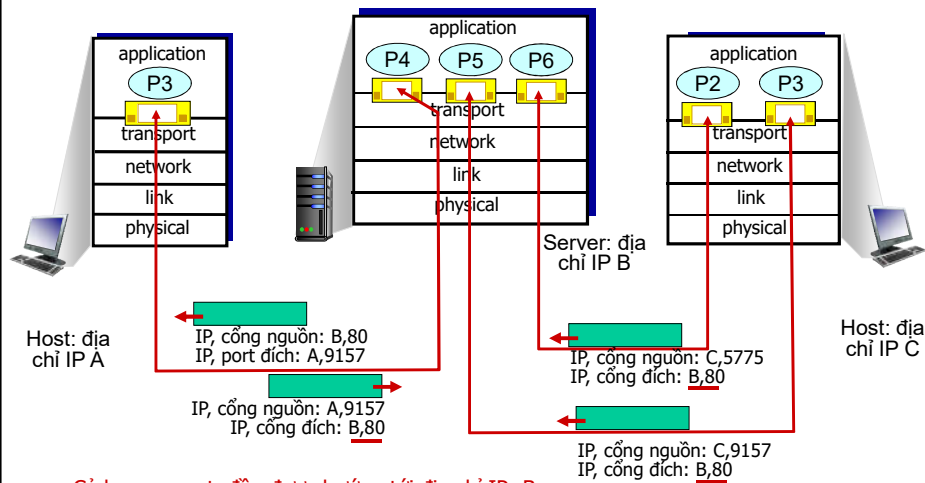
Ví dụ phân kênh hướng không kết nối



Phân kênh hướng kết nối

- ❖ TCP socket được xác định bởi bộ-4 giá trị:
 - Địa chỉ IP nguồn
 - Số hiệu cổng nguồn
 - Địa chỉ IP đích
 - Số hiệu cổng đích
- ❖ Phân kênh: Phía nhận sử dụng cả bốn giá trị này để định hướng segment tới socket phù hợp
- ❖ Host server có thể hỗ trợ nhiều TCP socket đồng thời:
 - Mỗi socket được xác định bởi bộ-4 giá trị của nó
- ❖ Web server có các socket khác nhau cho mỗi kết nối từ client
 - Kết nối HTTP không bền vững sẽ có các socket khác nhau cho mỗi yêu cầu.

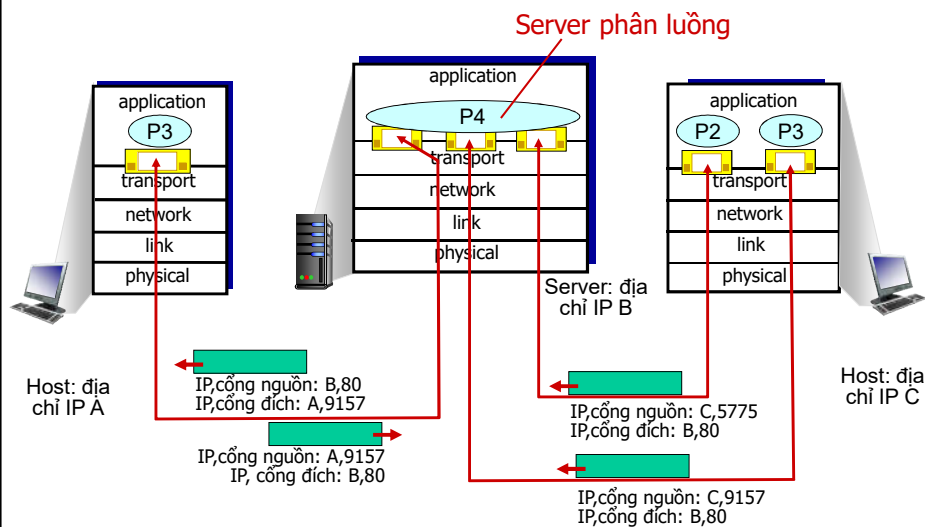
Ví dụ phân kênh hướng kết nối



Cả ba segment, đều được hướng tới địa chỉ IP: B, cổng đích: 80, nhưng được phân kênh tới các socket khác nhau

Tầng giao vận 3-13

Ví dụ phân kênh hướng kết nối



Tầng giao vận 3-14

Chương 3: Nội dung

3.1 Các dịch vụ tầng giao vận

3.2 Ghép kênh và phân kênh

3.3 Vận chuyển không kết nối: UDP

3.4 Các nguyên lý truyền dữ liệu tin cậy

3.5 Vận chuyển hướng kết nối: TCP

- Cấu trúc đoạn dữ liệu (segment)
- Truyền dữ liệu tin cậy
- Điều khiển luồng
- Quản lý kết nối

3.6 Các nguyên lý điều khiển tắc nghẽn

3.7 Điều khiển tắc nghẽn TCP

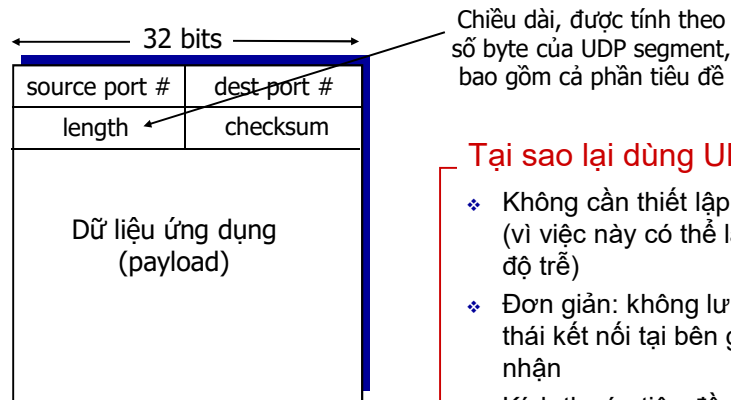
Tầng giao vận 3-15

UDP: User Datagram Protocol [RFC 768]

- ❖ Là giao thức tầng giao vận của mạng Internet
- ❖ Dịch vụ “best effort”, các UDP segment có thể:
 - Bị mất
 - Được vận chuyển không đúng thứ tự tới ứng dụng
- ❖ *Hướng không kết nối:*
 - Không có giai đoạn bắt tay giữa bên gửi và bên nhận của UDP
 - Mỗi UDP segment được xử lý độc lập với các segment khác
- ❖ UDP được dùng trong:
 - Các ứng dụng streaming multimedia (chịu mất mát dữ liệu, bị ảnh hưởng bởi tốc độ)
 - DNS
 - SNMP
- ❖ Truyền tin cậy trên UDP:
 - Bổ sung đặc tính tin cậy vào tầng ứng dụng
 - Khôi phục lỗi cụ thể của ứng dụng

Tầng giao vận 3-16

UDP: Tiêu đề segment



Định dạng UDP segment

Tại sao lại dùng UDP?

- ❖ Không cần thiết lập kết nối (vì việc này có thể làm tăng độ trễ)
- ❖ Đơn giản: không lưu trạng thái kết nối tại bên gửi, bên nhận
- ❖ Kích thước tiêu đề nhỏ
- ❖ Không điều khiển tắc nghẽn: UDP có thể gửi nhanh theo mong muốn

Tầng giao vận 3-17

UDP checksum

Mục tiêu: Phát hiện các “lỗi” (ví dụ: các bit bị bật lên) trong các segment được truyền đến

Bên gửi:

- ❖ Xử lý nội dung các đoạn, bao gồm cả các trường trong tiêu đề, như là chuỗi các số nguyên 16-bit
- ❖ checksum: bổ sung thêm (tổng bù của 1) vào nội dung segment
- ❖ Bên gửi đặt giá trị checksum vào trong trường checksum của UDP

Bên nhận:

- ❖ Tính toán checksum của segment đã nhận được
- ❖ Kiểm tra xem checksum đã tính có bằng giá trị của trường checksum hay không:
 - KHÔNG – phát hiện có lỗi
 - CÓ – không phát hiện lỗi. *Nhưng vẫn có thể có lỗi mà chưa được phát hiện? Xem thêm phần sau*

Tầng giao vận 3-18

Ví dụ: checksum trên Internet

Ví dụ: Cộng hai số nguyên 16-bit

| | | | | | | | | | | | | | | | | |
|----------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| Bit dư | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 |
| Tổng | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| checksum | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |

Chú ý: Khi cộng các số nguyên, một bit nhớ ở phía cao nhất cần phải được thêm vào kết quả