# **Chương 1**

* 1. **Mạng máy tính và Internet**
* Mạng máy tính là một hệ thống bao gồm các máy tính được nối kết với nhau để trao đổi thông tin.
* Mục đích:
  + Chia sẻ phần cứng
  + Chia sẻ dữ liệu
  + Trao đổi thông tin
  1. **Phân loại mạng máy tính**
* Phân thành các loại:
  + mạng cục bộ (LAN - Local Area Network)\*: kết nối các máy tính trong một khu vực bán kính hẹp, thông thường dưới 1 Km, có băng thông lớn
  + mạng khu vực (Campus) : thường kết nối các máy tính trong phạm vi một vài tòa nhà của một cơ quan hoặc một doanh nghiệp
  + mạng thành phố (MAN -Metropolitan Area Network) : trải rộng trên phạm vi một thành phố hoặc một khu vực
  + mạng diện rộng (WAN - Wide Area Network) \*: kết nối các máy tính có phạm vi lớn hơn 1 Km, cho phép cung cấp các dịch vụ công cộng
  1. **Hình trạng mạng**
     1. **Hình dạng vật lí**
* có 3 dạng cấu trúc cơ bản là:

|  |  |  |
| --- | --- | --- |
| Dạng tuyến (Bus Topology) | Dạng vòng (Ring Topology) | Dạng hình sao (Star Topology) \* |
| Chỉ cần một điểm trên đường dây bị đứt sẽ ngừng hoạt động của toàn bộ mạng | Đường dây phải khép kín, nếu bị ngắt ở một nơi nào đó thì toàn bộ mạng sẽ ngừng hoạt động | Trung tâm: thiết bị tập trung (hub) hoặc thiết bị chuyển mạch (switch) - Brigde.  Một thiết bị nào đó ở một nút thông tin bị hỏng thì mạng vẫn hoạt động bình thường  Trung tâm có sự cố thì toàn mạng ngừng hoạt động |

* + 1. **Hình trạng logic**
* 2 phương thức:
  + **Quảng bá:** kênh truyền được chia sẻ cho tất cả các máy tính, khi một máy tính gửi tin thì dữ liệu được chuyển lên môi trường truyền dẫn và tất cả các máy tính khác sẽ nhận được tin đó
  + **Thẻ bài:** Hình trạng này không có nút điều phối, một bản tin đặc biệt gọi là thẻ bài (token) được lần lươt chuyển cho mỗi nút mạng theo một thứ tự định trước.
    1. **Kết nối với mạng Internet**
* 3 loại dịch vụ kết nối:
  + Kết nối trực tiếp, cố định
  + Kết nối trực tiếp, không cố định
  + Kết nối gián tiếp

1. **Môi trường**

* Cáp đồng xoắn đôi: CAT5, CAT6. Cáp thẳng và cáp chéo phụ thuộc vào Tx (phát) và Rx(thu) của thiết bị. Adaptive là ngoại lệ
  + T568-A
  + T568-B
* Cáp đồng trục (hiện chỉ dùng trong mạng truyền hình cáp)
* Cáp quang
  + Cáp đa mode MM
  + Cáp đơn mode SM
* Vô tuyến (không dây)

**Cáp đồng xoắn đôi**

* Sao phải xoắn ?
  + Tăng sức bền cơ học (giống bện thừng)
  + Chống nhiễu
  + Tăng khả năng chịu uốn
* Chuẩn bấm cáp:
  + T568A
  + T568B (Tại VN, nếu dùng cáp thẳng thì dùng chuẩn T568B)
* Trong thực tế hệ thống cáp còn bao gồm:
  + Wall Plate
  + Patch panel (đặt trong tủ rack thiết bị)
  + Dây nhảy
  + Dây kết nối Wall Plate và Patch panel
* Chiều dài tối đa (Cat5e): 100m = 3+90+3+4 (dự phòng)

# **Chương 2: Kiến trúc và Hiệu năng mạng**

1. **Chuyển mạch kênh và chuyển mạch gói**

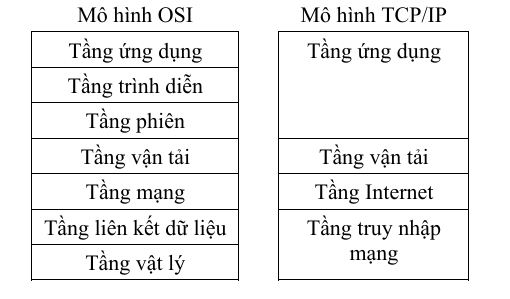
* **Chuyển mạch kênh:** 
  + Giữa chúng sẽ thiết lập một kênh cố định và được duy trì cho đến khi một trong hai bên ngắt liên lạc
  + Dữ liệu chỉ được truyền theo con đường cố định đó.
  + Việc phân chia băng thông của kênh truyền có thể thực hiện kỹ thuật đa truy nhập phân chia theo tần số (FDMA - Frequency Division Multi Access) hay đa truy nhập phân chia theo thời gian (TDMA - Time Division Multi Access).
* **Chuyển mạch thông điệp:** 
  + Thông điệp là một đơn vị thông tin với khuôn dạng nhất định, mỗi thông điệp gồm hai phần: Phần thông tin điều khiển và phần nội dung cần chuyển, phần thông tin điều khiển phải chỉ định rõ đích đến của thông điệp.
  + Căn cứ vào thông tin điều khiển, mỗi nút trung gian sẽ quyết định chuyển thông điệp tới nút kế tiếp, đường đi của các thông điệp sẽ không cố định
* **Chuyển mạch gói:**
  + Thông tin trao đổi giữa hai máy tính được phân thành *những gói tin* có kích thước tối đa xác định. Gói tin của những người dùng khác nhau sẽ chia sẻ nhau băng thông của kênh truyền
  + Tận dụng kênh truyền tốt hơn, do đó có thể đáp ứng nhiều người sử dụng hơn mà không cần phải nâng cấp hệ thống phần cứng

1. **Phân tầng và chức năng của các tầng**

* Giao thức là tập hợp các qui ước về định dạng dữ liệu và các thủ tục để hai bên có thể trao đổi thông tin với nhau

1. **Kiến trúc phân tầng (đa tầng)**

* Mục đích của việc phân tầng là để chuyên môn hóa các chức năng dịch vụ
* Giao thức (Protocol) là tập các tiêu chuẩn để trao đổi thông tin giữa hai hệ thống máy tính hoặc hai thiết bị máy tính với nhau
* Trong mỗi máy, mỗi tầng sử dụng các dịch vụ do tầng bên dưới cung cấp
* Liên kết giữa hai tầng thấp nhất gọi là liên kết vật lý, liên kết của tất cả các tầng cao hơn gọi là liên kết logic
* Giao tiếp giữa hai tầng liền kề gọi là giao diện, chúng trao đổi dữ liệu với nhau qua các điểm truy nhập dịch vụ (SAP)

****

1. **Mô hình OSI**

* Mô hình tham chiếu

**2.2.2.1. Các tiến trình ngang hàng**

**2.2.2.2. Giao diện giữa các tầng**

**2.2.2.3. Tổ chức các tầng**

* Có thể chia bảy tầng có thể thành ba nhóm:
  + Nhóm hỗ trợ mạng (3 tầng thấp nhất): tầng vật lý, tầng liên kết dữ liệu và mạng
  + Nhóm hỗ trợ người sử dụng (3 tầng cao nhất): tầng phiên, trình diễn, ứng dụng
  + Nhóm trung gian : tầng vận tải
* (PDU - Protocol Data Unit): đơn vị dữ liệu của giao thức

1. **Chức năng các tầng trong mô hình OSI**

* Mô hình OSI gồm 7 tầng sau:
  + - 1. **Tầng vật lí**
* Thực hiện các chức năng cần thiết để truyền luồng bit dữ liệu đi qua các môi trường truyền đẫn
* Chức năng:
  + Đảm bảo giao tiếp với môi trường truyền dẫn
  + Biểu diễn dữ liệu dưới dạng bit: dữ liệu là các bit, các bit được mã hóa thành các tín hiệu điện, quang
  + Tốc độ truyền dẫn: qui định b/s
  + Đồng bộ: máy gửi và nhận đồng bộ hóa ở mức bit
  + Hình trạng vật lý:
  + Chế độ truyền dẫn: đơn công, bán song công, song công.
    - 1. **Tầng liên kết dữ liệu**
* Đảm bảo truyền tin tin cậy giữa hai thiết bị vật lý kết nối trực tiếp với nhau, dữ liệu tại tầng này gọi là khung (Frame)
* Chức năng:
  + Tạo khung dữ liệu: chia gói tin từ tầng mạng thành các khung dữ liệu
  + Quản lý địa chỉ vật lý: xác định địa chỉ đich của gói tin
  + Kiểm soát lưu lượng: ngăn ngừa quá tải nơi nhận
  + Kiểm soát lỗi: Tăng tính tin cậy cho tầng vật lý bằng cách phát hiện và truyền lại các khung bị lỗi, ngăn ngừa hiện tượng lặp khung
    - Kỹ thuật kiểm tra vòng (CRC)
  + Kiểm soát truy cập: xác định thiết bị sử dụng đường truyền
    - 1. **Tầng mạng**
* Chịu trách nhiệm chuyển dữ liệu giữa các thiết bị đầu cuối của người sử dụng, đơn vị dữ liệu của tầng mạng gọi là gói tin (packet)
* Chức năng:
  + Quản lý địa chỉ logic: chứa địa chỉ logic của thiệt bị gửi và nhận – phân biệt chúng
    - Địa chỉ vậy lí của tầng lk dữ liệu đã đàm bảo tính duy nhất trong toàn mạng, ko giải quyết vấn đề định địa chỉ nội bộ
  + Định tuyến: kết nối qua các thiết bị trung gian (router) để xác định tuyến đường của gói tin đến địch – liên mạng
    - 1. **Tầng vận tải**
* Chịu trách nhiệm chuyển toàn bộ bản tin từ nơi gửi đến nơi nhận một cách toàn vẹn
* Đảm bảo liên kết giữa các tiến trình trên các máy tính khác nhau trên môi trường mạng
* Có hai loại liên kết:
  + Liên kết có hướng (Connection Oriented): tạo ra một kết nối logic giữa hai cổng đầu cuối: dữ liệu của bản tin được truyền qua cổng đó
  + Liên kết vô hướng (Connectionless)
* Chức năng:
  + Thiết lập liên kết logic giữa các tiến trình trên thiết bị đầu cuối của người dùng: Mỗi máy tính chạy nhiều chương trình -> nhiều tiến trình -> thiết lập cổng (port), mỗi cổng tương ứng với bản tin của 1 tiến trình tại phiên
  + Phân đoạn và tái hợp: dữ liệu lớn -> chia thành các đoạn nhỏ hơn, độc lập -> được gán thứ tự để phía nhận tái hợp hoàn chỉnh
  + Kiểm soát kết nối: kết nối có hướng (gửi yêu cầu kết nối) và vô hướng (datagram)
  + Kiểm soát lưu lượng:
  + Kiểm soát lỗi: đảm bảo đến đích chính xác, nếu có lỗi thì truyền lại
    - 1. **Tầng phiên**
* Đóng vai trò kiểm soát viên hội thoại giữa các tiến trình trên lớp ứng dụng qua mạng, nó đảm bảo nhiệm vụ thiết lập, duy trì và đồng bộ hóa tính tương tác giữa các tiến trình đồng cấp trên các máy tính khác nhau
* Chức năng:
  + Kiểm soát hội thoại: cho phép truyền thông giữa hai tiến trình được thực hiện hoặc theo kiểu bán song công hoặc theo kiều song công
  + Đồng bộ hóa: Tầng phiên cho phép một tiến trình thêm các mốc gọi là điểm đồng bộ (synchronization point) vào luồng dữ liệu
    - 1. **Tầng trình diễn**
* Thực hiện các nhiệm vụ liên quan đến cú pháp và ngữ nghĩa của các thông tin được trao đổi giữa hai hệ thống
* Chức năng:
  + Mã hóa/Giải mã dữ liệu (Encode/Decode): tạo khuôn thông tin trung giữa 2 tầng trình diễn của máy gửi và nhận
  + Nén/Giải nén: giảm số lượng bit -> nâng cao hiệu suất truyền tin
  + Mã hóa/Giải mã bảo mật (Encrypt/Decrypt): bảo mật
    - 1. **Tầng ứng dụng**
* Cung cấp các tiện ích để người dùng truy cập vào mạng như: các dịch vụ như gửi thư điện tử, truy cập và chuyển file từ xa….
* Cung cấp:
  + Thiết bị đầu cuối ảo của mạng: phiên bản phần mềm, cho phép đăng nhập vào một máy từ xa
  + Quản lý, truy cập và chuyển tập tin: cho phép người dùng truy cập tệp tin, quản lý trên một máy tính khác
  + Các dịch vụ khác: Hai dịch vụ phổ biến nhất là thư điện tử và web
    1. **Mô hình TCP/IP**
* Trong thực tế mô hình TCP/IP đang được sử dụng rộng rãi nhất hiện nay và còn gọi là mô hình Internet
* Bộ giao thức này được đặt tên theo hai giao thức chính của nó là giao thức điều khiển truyền (TCP - Transmission Control Protocol) và giao thức liên mạng (IP - Internet protocol)
* Mô hình TCP/IP chia theo 4 tầng: truy nhập mạng, Internet, vận tải và ứng dụng
  + - 1. **Tầng truy nhập mạng**
* Chịu trách nhiệm nhận các gói tin của tầng trên Internet và việc truyền phát chúng trên một mạng xác định
* Modem quay số: được truyền bằng cách sử dụng giao thức PPP
* Truy nhập Internet băng thông rộng: giao thức PPPoE thường được sử dụng
  + - 1. **Tầng Internet**
* Ứng với tầng mạng OSI
* Đảm bảo liên kết logic giữa hai thiết bị đầu cuối của người sử dụng
* Chọn đường để chuyển tiếp gói tin đến địa chỉ đích
* Giao thức thông điệp điều khiển Internet (ICMP - Internet Control Message Protocol) và giao thức quản lý nhóm Internet (IGMP - Internet Group Management Protocol)
  + - 1. **Tầng vận tải**
* Đảm bảo liên kết giữa các tiến trình trên các thiết bị đầu cuối của người sử dụng
* Có thể điều chỉnh lưu lượng luồng thông tin và cung cấp vận chuyển tin cậy
* Kết hợp các khả năng truyền bản tin từ đầu cuối đến đầu cuối mà không phụ thuộc vào mạng bên dưới, kiểm soát lỗi (error control), phân mảnh dữ liệu và điều khiển lưu lượng
  + Kết nối có hướng – TCP
  + Kết nối vô hướng – UDP
* RTP (Real-time Transport Protocol - giao thức vận tải thời gian thực) là một giao thức được thiết kế cho dữ liệu thời gian thực, đó là giao thức tầng phiên sử dụng định dạng gói tin UDP những vẫn đc xếp vào tầng vận tải
  + - 1. **Tầng ứng dụng**
* Nó bao gồm ba tầng trên của mô hình OSI
* Bao gồm tất cả các chuơng trình ứng dụng sử dụng các dịch vụ sẵn có thông qua một chồng giao thức TCP/IP
* (Giao thức truyền siêu văn bản HTTP dùng cổng 80; Giao thức truyền tập tin FTP dùng cổng 20 và 21, giao thức chuyển thư điện tử SMTP sử dụng cổng 25, giao thức truy nhập từ xa Telnet sử dụng cổng 23...)
  1. **Tên miền và địa chỉ**
* Vị trí của mỗi máy tính được xác định thông qua địa chỉ logic, trong mô hình TCP/IP đó là địa chỉ IP
* Địa chỉ IP gồm có 32 bit (phiên bản 4) hoặc 128 bit (phiên bản 6) và có cấu trúc phân cấp

1. **Các dịch vụ tên miền**

* Sử dụng tên miền dễ nhớ hơn số
* Hệ thống tên miền (DNS - Domain Name System) thực hiện chức năng chuyển đổi tên miền thành địa chỉ IP, nó sử dụng hệ thống các máy chủ để thực hiện nhiệm vụ này
* Giao thức trao đổi tên miền DNS thuộc tầng ứng dung và chạy trên nền giao thức UDP với số hiệu cổng là 53
* Giống như các giao thức HTTP, FPT hay SMTP, giao thức DNS nằm ở tầng ứng dụng
* Cung cấp một số dịch vụ quan trọng như đặt bí danh cho máy tên miền, phân tải xử lý tên miền …
* DNS thực hiện việc phân tán tải cho các máy chủ, đặc biệt là các máy chủ được nhân bản
* DNS được đặc tả trong RFC 1034 và RFC 1035 và cập nhật trong một số RFC khác

1. **Cơ chế hoạt động của dịch vụ tên miền**

* Vấn đề nảy sinh:
  + Điểm hỏng duy nhất: máy chủ tên miền duy nhất ngừng làm việc là toàn bộ mạng Internet ngừng hoạt động
  + Khối lượng xử lý lớn: một máy chủ xử lí tất cả các truy vấn DNS
  + Cơ sở dữ liệu tập trung ở xa: máy chủ tên miền không thể gần máy khách -> độ trễ
  + Bảo trì: cập nhập thông tin thường xuyên
* Máy chủ tên miền cục bộ: Mỗi nhà cung cấp dịch vụ Internet (ISP) đều có máy chủ tên miền cục bộ (còn được gọi là máy chủ tên miền mặc định)
* Máy chủ tên miền gốc
* Máy chủ tên miền ủy quyền
* Một đặc tính quan trọng của DNS là lưu trữ tạm thời các bản ghi DNS (DNS caching) để làm giảm độ trễ và số bản tin DNS trao đổi trên mạng

1. **Bản ghi dịch vụ tên miền**

* Máy chủ tên miền cũng triển khai cơ sở dữ liệu phân tán, ghi lại các bản ghi tài nguyên cho các ánh xạ tên máy một địa chỉ IP. Mỗi bản tin trả lời DNS chứa một hay nhiều bản ghi tài nguyên, chi tiết trong RFC 1034, RFC 1035
* Bản ghi tài nguyên gồm 4 trường sau: (Name, Value, Type, TTL).
  + TTL là thời gian tồn tại của bản ghi tài nguyên, dùng để xác định thời điểm có thể xoá bản ghi tài nguyên khỏi bộ nhớ lưu trữ
  + Nếu Type = A thì Name là tên máy và Value là địa chỉ IP của máy đó. Bản ghi kiểu A là ánh xạ Tên máy - Địa chỉ IP chuẩn
  + Nếu Type = NS thì Name là một miền và Value là tên máy của máy chủ tên miền ủy quyền của các máy tính trong miền đó. Bản ghi này thường được sử dụng để gửi tiếp các truy vấn DNS
  + Nếu Type = CNAME thì Value là tên đầy đủ của máy có tên bí danh đặt trong Name. Bản ghi kiểu này cho phép xác định tên đầy đủ của một máy tính từ tên bí danh
  + Nếu Type : MX thì Value là tên máy của máy chủ thư điện tử có tên bí danh đặt trong Name
* Tiện ích nslookup sẽ hiển thị máy chủ tên miền đã cấu hình trong máy tính, nslookup Tên\_Miền sẽ cho biết địa chỉ của tên miền cần phân giải
* Tiện ích ipconfig /displaydns hiển thị tất cả các bản ghi DNS hiện nay đang được lưu tạm trong hệ điều hành của máy khách
* Tìm hiểu thêm về DNS tại slide cô Thủy
  1. **Nguyên tắc thiết kế Internet**
* Mô hình phân cấp
  1. **Các yếu tố tạo nên hiệu năng mạng**
* Khái niệm cho biết hiệu suất hoạt động của hệ thống mạng
* Hiệu năng chủ yếu được xác định bởi sự kết hợp của nhiều yếu tố, có những yếu như: Băng thông, thông lượng, thời gian đáp ứng, độ trễ, độ tin cậy, tỉ lệ lỗi, tốc độ xử lý của phần mềm ứng dụng, tính sẵn sàng của hệ thống

1. **Các yếu tố đánh giá hiệu năng mạng**

* Yếu tố đánh giá hiệu năng thành hai loại: các yếu tố hướng tới người sử dụng và các yếu tố hướng tới hệ thống
* Băng thông được định nghĩa là lượng thông tin tối đa có thể chuyển tải trên mạng trong một đơn vị thời gian
* Thông lượng được định nghĩa là lượng thông tin thực tế được vận chuyển qua mạng trong một đơn vị thời gian. Đơn vị thông tin ở đây có thể là bit, byte hay gói số liệu...
* Thời gian trễ là thời gian trung bình để vận chuyển một gói số liệu qua mạng từ nguồn tới đích

1. **Vai trò của việc đánh giá hiệu năng**

* Cấu hình mạng: Nhằm đạt được sự tối ưu hoá, nhà sản xuất phải chỉ ra được các cách kết hợp và tổ chức phần cứng và phần mềm mạng để đem lại một giải pháp tốt nhất
* Điều chỉnh hiệu suất hoạt động của hệ thống
* TÍnh toán bằng các phương trình giải tích

1. **Các phương pháp đánh giá hiệu năng mạng**

* **Phương pháp mô hình giải tích:** Trong các mạng chuyển mạch gói, gói tin là các khối dữ liệu có chiều dài thay đổi được, được truyền qua mạng từ nguồn tới đích theo một con đường nào đó do hệ thống mạng quyết định
* **Phương pháp mô phỏng:** Mô phỏng là sự bắt chước một hay nhiều khía cạnh của sự vật có thực, bằng một cách nào đó càng giống càng tốt
  + Một kỹ thuật sử dụng máy tính điện tử số để làm các thí nghiệm về mạng có liên quan đến thời gian
* **Phương pháp đo lường:** phương pháp xác định hiệu năng dựa trên việc đo các tham số mạng cấu thành độ đo hiệu năng cần quan tâm trên mạng thực
  + Giám sát hiệu năng mạng
  + Thu thập số liệu để lập mô hình dữ liệu vào cho các phương pháp đánh giá hiệu năng bằng giải tích hoặc mô phỏng
  + Kiểm chứng các mô hình khác dựa trên các số liệu đo được.

**Chương 3: Tầng Ứng Dụng**

1. **Các khái niệm và cài đặt các giao thức tầng ứng dụng**

* Mặc dù chương trình ứng dụng mạng có nhiều loại khác nhau, có thể có nhiều thành phần tương tác với nhau, nhưng lõi của chúng là phần mềm
* Việc kết nối được thực hiện giữa các tiến trình chứ không phải giữa các chương trình phần mềm, tiến trình là một chương trình chạy trên thiết bị đầu cuối. Khi các tiến trình chạy trên cùng một thiết bị, chúng sẽ trao đổi dữ liệu với nhau thông qua cơ chế truyền thông liên tiến trình
* Tầng trình diễn thực hiện ba chức năng chính: mã hõa/giải mã dữ liệu, nén/giải nén và mã hóa/giải mã bảo mật
  + Mã hóa/giải mã dữ liệu:
  + Nén/giải nén
  + Mã hóa/giải mã bảo mật
  + Các giao thức: QuickTime là giao thức cho video và âm thanh, MPEG là tiêu chuẩn cho việc nén và mã hóa video
* Tầng phiên: tạo hội thoại giữa các ứng dụng nguồn và đích, nó xử lý các vấn đề liên quan đến việc khởi tạo, duy trì hoạt động và tái tạo lại các phiên trong trường hợp bị ngắt quãng hoặc tạm nghỉ trong thời gian dài

1. **Mô hình dịch vụ của tầng ứng dụng**

* Tiêu chuẩn định dạng văn bản (HTML), trình duyệt Web (Netscape Navigator hay Microsoft Internet Explorer), phần mềm máy chủ web (Apache, Microsoft, và Netscape) và giao thức tầng ứng dụng
* Giao thức tầng ứng dụng của Web - HTTP (Hypertext Transfer Protocol, RFC 2616), định nghĩa cách thức chuyển bản tin giữa trình duyệt và máy chủ web, như vậy HTTP chỉ là một phần của ứng dụng Web
* Giao thức SMTP chỉ là một phần của ứng dụng thư điện tử. Giao thức tầng ứng dụng định nghĩa cách thức truyền bản tin giữa các tiến trình ứng dụng chạy trên các thiết bị khác nhau

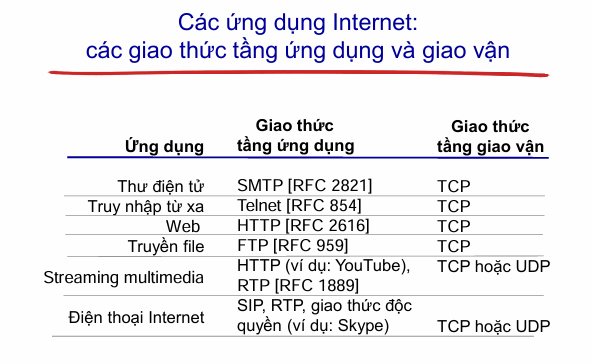
1. **Mô hình khách chủ (phân cấp)**

* Các máy tính được thiết lập để cung cấp các dịch vụ được gọi là chủ (Server)
* Các máy tính truy cập và sử dụng dịch vụ thì được gọi là khách (Client)

1. **Mô hình ngang hàng**

* Trong mô hình ngang hàng (peer-to-peer)các máy tính trong mạng có thể hoạt động vừa như một Client vừa như một Server
* FTP - được dùng để truyền tập tin giữa hai máy
* Skype- giao thức riêng(độc quyền)

1. **Các giao thức thường dùng tại tầng ứng dụng**

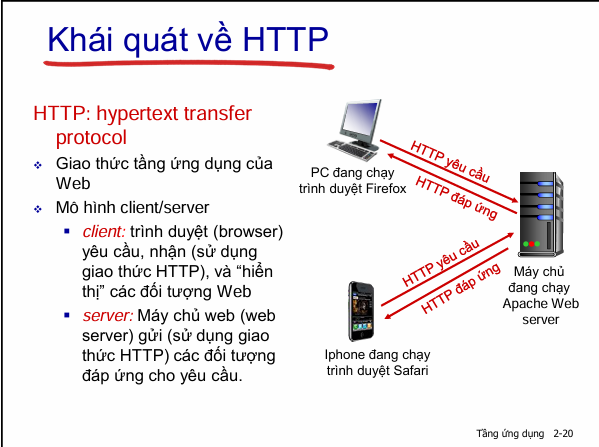
****

* Một số giao thức dùng trong các ứng dụng phổ biến: Web, truyền tập tin, thư điện tử và dịch vụ tên miền.
  + Web là ứng dụng đầu tiên và vì Web được sử dụng phổ biến và giao thức tầng ứng dụng của nó – HTTP
  + Ứng dụng truyền tập tin – trái với HTTP
  + Thư điện tử - SMTP
  + DNS: cung cấp dịch vụ đổi tên miền thành địa chỉ IP
  + Mail server \*: cổng 25

1. **Giao thức truy nhập trang web HTTP**

* World Wide Web: thay đổi cách thức tương tác giữa con người và môi trường làm việc. Chính điều này đã giúp đưa Intemet từ một trong rất nhiều mạng thông tin (ví dụ mạng trực tuyến Prodigy, American Onlie hay Compuserve, hệ thống thông tin quốc gia: Minitee/Tranpac ở Pháp, Private X25, Frame Relay) thành một mạng lớn nhất toàn cầu

1. **Tổng quan về giao thức HTTP**

****

* Giao thức chuyển dữ liệu siêu văn bản (Hyper Text Transfer Protocol HTTP) là giao thức căn bản sử dụng trong việc trao đổi thông tin giữa máy khách và máy chủ Web
* Trang Web chứa các đối tượng, đối tượng đơn giản chỉ là một tập tin như ảnh, âm thanh… Đối tượng được xác định qua định vị tài nguyên cố định (URL - Uniform Resource Locator) – chứa tên host và đường dẫn .
  + Ví dụ, [www.ptit.edu.vn/Portals/0/ptitlogo72.gif](http://www.ptit.edu.vn/Portals/0/ptitlogo72.gif) thì www.ptit.edu.vn là tên máy chủ - host và Portals/0/ptitlogo72.gif là tên đường đẫn đến ảnh ptitlogo72.gif được lưu trên máy chủ.
* Trình duyệt (Browser) là phần mềm được cài đặt trên máy khách để người dùng có thể giao tiếp với máy chủ Web
* HTTP được gọi là giao thức không trạng thái
* Giao thức HTTP không giải quyết việc mất mát dữ liệu mà việc này là trách nhiệm của giao thức TCP

**Kết nối liên tục và không liên tục**

* Kết nối liên tục – HTTP 1.1 và không liên tục – HTTP 1.0
* Liên tục là máy chủ không đóng liên kết TCP sau khi gửi bản tin trả lời
  + Không liên tục: mở và đóng liên kết TCP để lấy 1 bản tin – 2 RTT, nhiều bản tin thì kết nối nhiều TCP song song
  + Liên tục: duy trì TCP: trao đổi nhiều bản tin -> 1 TCP hay 1 RTT
* HTTP sử dụng cổng 80
* RTT: thời gian để một gói tin nhỏ đi từ client đến sever và quay lại

1. Khuôn dạng bản tin HTTP

**Bản tin yêu cầu HTTP:**

* Thường có dạng:
  + GET /english/index.html HTTP/1.1 Host: [www.ptit.edu.vn](http://www.ptit.edu.vn)
  + Connection: close
  + User-agent: Mozzilla/4.0
  + Accept-language: En
  + (extra carry return line feed)
* Được viết bằng mã ASCII vì thế bất kỳ máy tính thông thường nào cũng có thể đọc được
* Bản tin gồm 5 dòng và mỗi dòng đều kết thúc bởi lặp ký tự đặc biệt Carriage Return (CR=13h) và Line Feed (LF-10h), trong thực tế một bản tin có thể có nhiều dòng hơn.
* Dòng đầu tiên của bản tin được gọi là dòng yêu cầu, các dòng sau gọi là tiêu đề
* Dòng yêu cầu có 3 trường: trường phương thức, trường URL và trường phiên bản HTTP
* Trường phương thức gồm
  + GET: yêu cẩu của máy khách gửi đến máy chủ để lấy dữ liệu web
  + POST: dùng để gửi các thông điệp cần đẩy dữ liệu lên máy chủ
  + PUT đẩy tài nguyên hoặc nội dung từ máy khách lên máy chủ.
* Connection: close là trình duyệt yêu cầu máy chủ không sử dụng cách kết nối liên tục và yêu cầu máy chủ đóng kết nối lại sau khi đã gửi đi đói tượng được yêu cầu
* Trường User-agent là phần mềm trình duyệt của người sử dụng, phần mềm trình duyệt ở đây là Mozzila, một sản phẩm của hãng Nestcape

**Bản tin trả lời:**

* Ví dụ:
  + HTTP/1.1 200 OK
  + Connection: close
  + Date : Thu, 01 Dec 2011 11:00:15 GMT
  + Máy chủ Apache/1. 3. 0 (unix)
  + Last modified :Mon, 14 Nov 2011 19:29:04 GMT
  + Connect length : 8611
  + Connect type: text/html
  + (data .....)
* Gồm có 3 phần : Dòng đầu tiên là dòng trạng thái (status link), 6 dòng tiêu đề và cuối cùng là phần thân (Entity body) chứa đối tượng được yêu cầu đó là phần nội dung của trang.
* Dòng trạng thái có 3 trường : trường phiên bản của giao thức, mã trạng thái và trường trạng thái bản tin trả lời
  + Phiên bản HTTP 1.1 và trạng thái là sẵn sàng
    - ***200 OK***: Yêu cầu được đáp ứng và dữ liệu được yêu cầu nằm trong bản tin
    - ***301 Moved permanetly***: cho biết đối tượng đã được chuyển và URL mới của đối tượng được đặt trong trường Location của bản tin trả lời
    - ***400 Bad Request***: máy chủ không hiểu được yêu cầu từ máy khách
    - ***404 Not found***: đối tượng không được lưu trên máy chủ
    - ***505 HTTP version not support***: máy chủ không hỗ trợ giao thức của máy khách
  + Connection: close ...
  + Date cho biết thời gian khi máy chủ tạo ra bản tin và gửi đi
  + Last modified là thời gian cuối cùng đối tượng được cập nhật
  + lenght: cho biết độ dài của đối tượng được gửi
  + type xác định kiểu của đối tượng là file văn bản HTML

1. Tương tác người dùng – máy chủ

**Authentication (Xác thực):**

* Nhiều máy chủ yêu cầu người dùng phải cung cấp tênvà mật khẩu để có thể truy cập vào tài nguyên trên máy chủ. Yêu cầu này được gọi là kiểm chứng
* trường mã trạng thái là 401 đòi hỏi phải có xác thực

**Cookies:**

* Cookie là kỹ thuật khác được máy chủ sử dụng để ghi lại đấu vết của người truy cập và được đặc tả trong RFC 2109
* Tệp cookie lưu tại thư mục: c:\windows\cookies dưới dạng cặp name-value
  + Theo slide cô Thủy:
    - Tệp cookie được lưu trữ trên host người dùng, được quản lý bởi trình duyệt người dùng
    - Cơ sở dữ liệu back-end
* Mục địch:
  + Nếu máy chủ yêu cầu kiểm chứng nhưng không muốn đòi hỏi người dùng đăng nhập qua tên và mật khẩu thì có thề sử dụng cookie cho mỗi lần người dùng truy cập vào máy chủ - tự động bỏ qua đăng nhập lần sau
  + Máy chủ sử dụng cookie nếu muốn ghi nhớ các hoạt động của người dùng, phục vụ mục đích quảng cáo
  + Nếu người sử dụng mua hàng trên mạng thì máy chủ sử dụng cookie để ghi lại những gì mà người sử dụng đã mua

1. GET có điều kiện

* Lưu giữ lại các đối tượng đã từng được lấy, web cache có thể làm giảm thời gian chờ từ khi gửi yêu cầu đến khi nhận đối tượng và làm giảm lưu lượng thông tin truyền trên mạng Internet
* Cơ chế cho phép sử dụng cache trong khi vẫn đảm bảo đối tượng trong cache chưa bị cũ
* Điều kiện: (1) thông điệp sử dụng phương thức GET và (2) thông điệp có trường If- modified- since trong tiêu đề: xác định ngày của bản sao

1. Web caches – proxy server

* Web cache là thực thể đáp ứng yêu cầu từ máy khách, máy tính làm nhiệm vụ Web cache có ổ đĩa riêng lưu trữ bản sao các đối tượng đã từng được yêu cầu
* Webcache vừa là máy khách vừa là máy chủ, nó đóng vai trò máy chủ khi nhận yêu cầu và trả lời, đóng vai trò máy khách khi gửi yêu cầu và nhận bản tin trả lời
* Trình duyệt gửi tất các yêu cầu HTTP tới cache:
  + Nếu đối tượng có trong cache: cache sẽ trả về
  + Ngược lại, cache sẽ yêu cầu đối tượng từ server nguồn, sau đó trả đối tượng về client, lưu lại bản sao -> giảm thời gian chờ cho lần tiếp theo

**Cache cộng tác:**

* Có thể kết hợp nhiều webcache đặt ở các vị trí khác nhau trên mạng nhằm nâng cao hiệu suất tổng thể
  + Hệ thống cache NLANR: lấy đối tượng từ cache khác bằng cách kết hợp sử dụng giao thức HTTP và ICP (Intemet Caching Protocol). ICP là giao thức ở tầng ứng dụng cho phép một cache nhanh chóng xác định một cache khác có một đối tượng nào đó hay không, và nếu có thì cache có thể sử dụng giao thức HTTP để lấy đối tượng về
  + Cụm cache (cache cluster), thường đặt trên cùng một mạng cục bộ, trình duyệt thực hiện phép băm trên địa chỉ URL, trình duyệt sẽ căn cứ vào kết quả để gửi yêu cầu đến một trong các cache trong cụm
  + Tìm kiếm theo hàm băm là cốt lõi của giao thức Cache Array Routing (CARP)

1. **Giao thức truyền tệp tin FTP – truyền file đến/từ host ở xa**

* FTP (File Transfer Protocol) là giao thức truyền tập tin tin cậy giữa hai máy tính. Trong phiên làm việc của FTP, người dùng làm việc trên máy tính của mình và trao đổi tập tin với một máy tính khác
* Để truy cập tới máy tính khác, người dùng phải đăng nhập thông qua việc cung cấp tên người dùng và mật khẩu. Sau khi những thông tin này được kiểm chứng thì công việc truyền tập tin từ hệ thống tập tin trên máy tính của mình đến hệ thống tập tin ở đầu kia mới có thể thực hiện được
* FTP sử dụng hai kết nối TCP song song, một đường truyền thông tin điều khiển (lệch, tiêu đề bản tin, mật khẩu, ...) và một đường truyền dữ liệu (truyền file)
* FTP phân biệt luồng thông tin điều khiển với luồng dữ liệu nên nó dược gọi là gửi thông tin điều khiển ngoài băng
  + Giao thức RTSP dùng để truyền âm thanh và hình ảnh liên tục cũng sử dụng cách gửi thông tin điều khiển kiểu ngoài băng
  + Giao thức HTTP gửi tiêu đề của bản tin và file dữ liệu trên cùng một kết nối TCP, vì vậy mà HTTP được gọi là gửi thông tin điều khiển trong băng (in-band)
* FTP sẽ thiết lập một đường kết nối thông tin điều khiển TCP qua cổng 21
* FPT mở một kết nối TCP để truyền dữ liệu qua cổng 20
* Giữ lại các thông tin về trạng thái của người dùng

**Các lệnh FTP**

* Lệnh được mã hóa bằng ASCII 7 bit
* Lặp ký tự CR (carriage return) và LF (line feed) được sử dụng để phân biệt các lệnh (và trả lời) mỗi câu lệnh chứa 4 kí tự ASCII in hoa
  + USER
  + PASS
  + LIST: yêu cầu máy chủ gửi danh sách
  + RETR: lấy tập tin
  + STOR: tải tập tin
* Câu trả lời:
  + 331 username OK, password requiered
  + 125 connection already open; Transfer staring
  + 425 can’t open data connection
  + 452 error writing file

1. **Giao thức chuyển thư điện tử**

* Hệ thống thư điện tử gồm tiến trình thư điện tử của người dùng trên máy khách, tiến trình quản lý thư điện tử máy chủ và giao thức chuyển thư đơn giản (SMTP - Simple Mail Transfer Protocol)
* Máy chủ thư điện tử là thành phần cốt lõi trong hệ thống thư điện tử, nó có nhiệm vụ quản lý hộp thư của mỗi thành viên, thực hiện tiếp nhận các yêu cầu chuyển thư và chuyển tiếp thư điện tử giữa các máy chủ
* SMTP (Simple Mail Transfer Protocol) là giao thức gửi thư điện tử của tầng ứng dụng, sử dụng dịch vụ truyền dữ liệu tin cậy của TCP để truyền thư từ máy chủ thư điện tử của người gửi đến máy chủ thư điện tử của người nhận
* SMTP hoạt động theo mô hình khách/chủ (server/client)

1. Giao thức SMTP

* Trái tim của dịch vụ thư điện tử và được đặc tả trong RFC821
* Ra đời trước HTTP
* Không sử dụng máy chủ thư điện tử trung gian để gửi thư
* SMTP máy khách (chạy trên máy chủ thư điện tử gửi) thiết lập kết nối TCP với cổng 25 tại SMTP máy chủ (chạy trên máy chủ thư điện tử nhận)
* Máy khách sử dụng 5 câu lệnh: HELO (viết tắt của HELLO), MAIL FROM, RCPT TO, DATA và QUIT
* Thông điệp là mã ASCII 7 bit
* Sử dụng kết nối liên tục
  + HTTP là giao thức kiểu kéo (Pull protocol), máy khách kéo thông tin từ máy chủ về. Phía nhận (máy khách) là phía thiết lập kết nối TCP
  + SMTP lại là giao thức theo kiểu đẩy (Push protocol), máy khách đẩy thông tin lên máy chủ. Phía gửi (máy khách) là phía thiết lập kết nối TCP trước
  + Ngoài dữ liệu văn bản, bản tin còn có thể chứa các kiểu dữ liệu khác như âm thanh hình ảnh, HTTP đặt các đối tượng này trong các bản tin riêng rẽ để gửi, với SMTP tất cả các đối tượng này được đặt trong cùng một thư điện tử
* Tiêu đề phải có trường From: và trường To: một số trường như Subject: có thể có hoặc không
* Phần tiêu đề bản tin được đặc tả trong RFC 822 phù hợp cho việc gửi văn bản nhưng lại không đầy đủ để gửi thư chứa nội dung đa phương tiện (multimedia) - là thư có đính kèm ảnh, audio, video hoặc các thư chứa các ký tự khác tiếng Anh
* Để gửi dữ liệu không thuộc dạng văn bản ASCII 7bit, user agent gửi phải gửi thêm một số trường trong tiêu đề của thư. Những trường này được đặc tả trong RFC 2045 và RFC 2046, là phần mở rộng MIME (Multipurpose Intemet Man Extension) cho RFC 822
* Hai trường MIME hỗ trợ multimedia là Content-Type: và Content- Transfer-encoding
  + Trường Content-Type cho phép phía nhận thực hiện các thao tác thích hợp trên thư nhận được
  + Trường Content-Transfer- encoding: xác định phần thân bản tin đã được mã hoá theo bảng mã ASCII và phương pháp mã hoá được sử dụng

**Content-Type: type/subtype; parmeters**

* Kiểu cao nhất – dữ liệu (top-level/type) được sử dụng để khai báo kiểu dữ liệu chung
* Kiểu con (subtype) xác định định dạng đặc biệt trong kiểu dữ liệu chung
* Các tham số bổ nghĩa cho kiểu và không ảnh hưởng tới bản chất kiểu dữ liệu
* IAAN (Internet Assigned Numbers Authority) là cơ quan đăng ký trung tâm. Tiến trình đăng ký kiểu dữ liệu được đặc tả trong khuyến nghị RFC 2048.
* 5 nhóm dữ liệu chính:
  + ***Văn bản (Text):*** Kiểu văn bản được sử dụng để xác định thân bản tin chứa thông tin dạng văn bản một kiểu con thường gặp là plain (trơn)
  + ***Ảnh (Image):*** Kiểu ảnh được dùng để xác định thân bản tin là ảnh. Hai kiểu con thông dụng là image/gif và image/jpeg (phải giải nén GIF)
  + ***Âm thanh (Audio):*** Kiểu audio yêu cầu nội dung được gửi ra thiết bị audio (speaker hoặc telephone). Kiểu con thông dụng là basic (mã theo luật 8-bit cơ sở) và 32kadpcm (định dạng 32kps được đặc tả trong RFC 1911)
  + ***Video:*** Kiểu video có kiểu con là mpeg và quicktime
  + ***Kiểu ứng dụng (Application):*** Kiểu ứng dụng dành cho dữ liệu không thuộc bất kỳ kiểu nào khác
* Nội dung bản tin gửi và nhận ở SMTP: (đọc thêm)
* Received: trường này có thể nhìn thấy trực tiếp trên màn hình hoặc khi in thư
* 2 giao thức thông dụng bên người nhận (SMTP – bên người gửi):
  + POP3 (Post Once Protocol - Versíon 3)
  + IMAP (Internet Mail Access Protocol)
* SMTP được dùng để chuyển thư giữa các máy chủ thư điện tử hay giữa phần mềm thư điện tử của người gửi và máy chủ thư điện tử của người gửi (đẩy). POP3 hay IMAP được đùng để chuyển thư từ máy chủ thư điện tử tới phần mềm thư điện tử của người nhận (kéo).

1. POP3

* POP3 được đặc tả trong RFC 1939 là giao thức lấy thư đơn giản và có rất ít chức năng, phần mềm thư điện tử POP3 được khởi tạo kết nối TCP tới máy chủ thư điện tử qua cổng 110. Sau khi thiết lập được kết nối, POP3 thực hiện xác thực, xử lý và cập nhật
* Chế độ tải và lưu
* Không giữ trạng thái của các phiên làm việc
* Mỗi lệnh có hai trạng thái kết quả: +OK thông báo lệnh vừa gửi được thực hiện đúng và ERR thông báo lệnh vừa gửi không thực hiện được.
  + Lệnh sai : -ERR

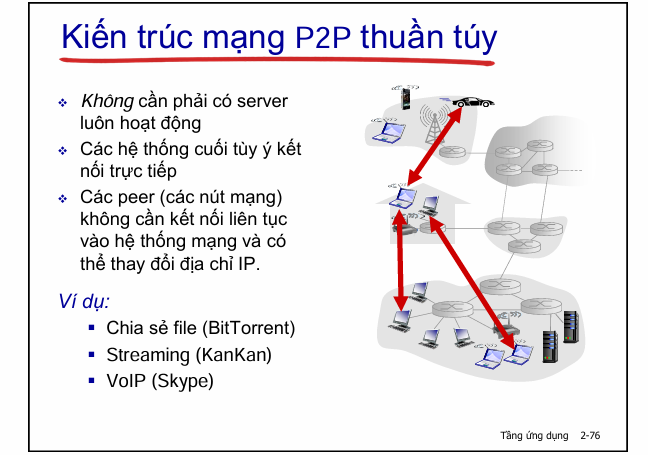
1. IMAP

* Giống như POP3, IMAP cũng là giao thức lấy thư nhưng có nhiều đặc tính và do đó phức tạp hơn, cho phép người dùng thao tác trên những hộp thư ở xa một cách dễ dàng
* Lưu trữ các thông điệp tại server
* IMAP cho phép người dùng tạo những thư mục thư khác nhau trong hộp thư, người dùng có thể đặt thư vào trong thư mục hay dịch chuyển thư từ thư mục này đến những thư mục khác
* IMAP cũng có lệnh cho phép tìm kiếm trên thư mục theo tiêu chí xác định. IMAP phức tạp hơn POP3 nhiều vì máy chủ IMAP phải duy trì hệ thống thư mục cho mọi người dùng
* Phiên làm việc IMAP gồm 3 giai đoạn: giai đoạn thiết lập kết nối giữa máy khách và máy chủ IMAP, giai đoạn máy chủ chấp nhận kết nối và giai đoạn tương tác máy khách máy chủ.
* Máy chủ luôn ở một trong bốn trạng thái: Chưa kiểm chứng (đăng nhập) -> Đã kiểm chứng -> Lựa chọn -> Thoát

1. **Một số ứng dụng quen thuộc**
2. **Trình duyệt web**
3. **Phần mềm đọc thư điện tử**
4. **Các phần mềm đa phương tiện**
5. **Tiện ích telnet, rlogin, ssh**

* Telnet (viết tắt của TErminaL NETwork) là một tiện ích mạng giao diện dòng lệnh được dùng để cung cấp những phiên giao dịch đăng nhập vào các máy trên mạng, tạo cảm giác như một thiết bị cuối được gắn vào một máy tính khác.
* Tiện ích Telnet thường sử cổng dịch vụ số 23 của giao thức TCP
* rlogin là tiện ích được cài đặt phổ biến trên hệ điều hành Unix, giao thức sử dụng cũng không được bảo mật, sử dụng cổng số 513
* SSH là một chương trình tương tác giữa máy chủ và máy khách có sử dụng cơ chế mã hoá đủ mạnh nhằm ngăn chặn các hiện tượng nghe trộm, đánh cắp thông tin trên đường truyền – ưu thế truy cập từ xa

**3.4. P2P và DHT: bảng băm phân tán của P2P**



**Chương 4: Tầng Vận Tải**

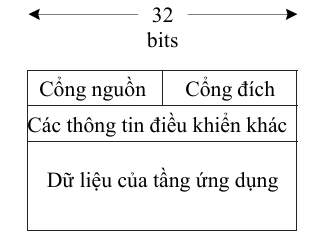
* Nằm giữa tầng ứng dụng và tầng mạng, tầng vận tải là tầng trung tâm trong kiến trúc phân tầng với nhiệm vụ cung cấp dịch vụ truyền thông giữa các tiến trình ứng dụng chạy trên các máy tính khác nhau
* Các tiến trình ứng dụng sẽ sử dụng đường truyền ảo này để trao đổi bản tin
* cho phép tại một thời điểm trên một thiết bị đầu cuối nhiều ứng dụng trao đổi thông tin qua mạng
* Truyền thông logic giữa các host
* Chức năng: đọc ở chương 1

1. **Ghép kênh và phân kênh, các giao thức TCP và UDP**

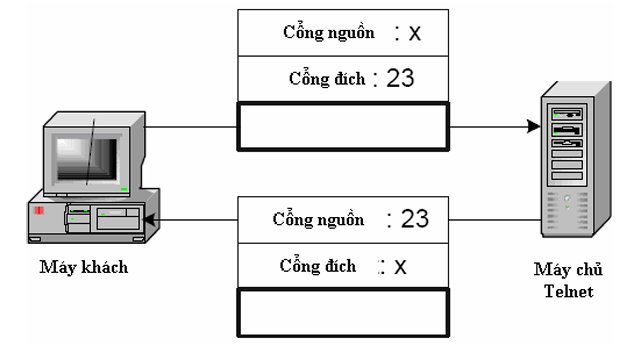
* Công việc được thực hiện bằng cách chia bản tin thành nhiều đoạn nhỏ, bổ sung vào đầu mỗi đoạn tiêu đề của tầng vận tải để tạo ra các đoạn tin và chuyển xuống tầng mạng

1. **Ghép kênh và phân kênh**

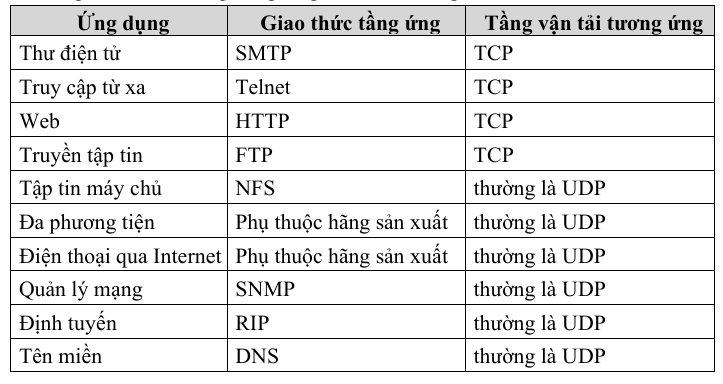
* Không phải là chức năng chính
* Tầng vận tải đảm bảo liên kết giữa các tiến trình đầu cuối với nhau
* Mỗi đoạn tin của tầng vận tải có trường xác định tiến trình nhận dữ liệu, tầng vận tải bên nhận sẽ sử dụng trường này để xác định rõ tiến trình nhận và gửi dữ liệu trong đoạn tin tới tiến trình đó, công việc chuyển dữ liệu trong đoạn tin tới đúng tiến trình ứng dụng được gọi là phân kênh
* Tại thiết bị gửi, tầng vận tải nhận dữ liệu từ nhiều tiến trình ứng dụng khác nhau, tạo đoạn tin chứa dữ liệu cùng với một số thông tin tiêu đề và cuối cùng chuyển đoạn tin xuống tầng mạng, quá trình trên được gọi là ghép kênh
* Các giao thức UDP và TCP thực hiện việc ghép kênh và phân kênh nhờ hai trường đặc biệt ở đầu đoạn tin: trường định danh cổng nguồn của tiến trình gửi và trường định danh cổng đích của tiến trình nhận, hai trường này sẽ xác định một tiến trình ứng dụng duy nhất chạy trên máy tính

****

* Số hiệu cổng được xác định bằng 16 bit nhị phân, nhận giá trị từ 0 đến 65535, các giá trị từ 0 đến 1023 là các giá trị dùng cho các dịch vụ công cộng, tức là chỉ để cho các ứng dụng thông dụng như HTTP, FTP…, HTTP sử dụng cổng 80, FTP sử dụng cổng 20 và 21
* Trong phiên làm việc khách/chủ:

****

* + 2 trường hợp : khác cổng nguồn đích và cùng cổng nguồn đích (đọc thêm)
* Bên nhận sử dụng cả ba giá trị: địa chỉ IP nguồn, số hiệu cổng nguồn và số hiệu cổng đích để xác định tiến trình ứng dụng của bên nhận
* Việc chia nhỏ dữ liệu của tầng ứng dụng thành từng đoạn để đảm bảo truyền tin cậy trên mạng và tập hợp lại thành dữ liệu nguyên bản ban đầu được các giao thức TCP và UDP thực hiện theo các hình thức hoàn toàn khác nhau. Mỗi đoạn trong giao thức TCP chứa số tuần tự để đảm bảo tính tuần tự của dữ liệu trong khi đó giao thức UDP không quan tâm tới thứ tự của các dữ liệu đó

****

1. **Giao thức TCP**

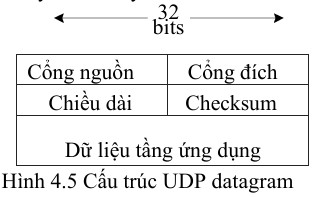
* Giao thức TCP thuộc tầng vận tải và được mô tả trong các khuyến nghị RFC 793, RFC 1122, RFC 1323, RFC 2018, RFC 2581, giao thức này có những đặc điểm sau:
  + Định hướng kết nối:Thực hiện thủ tục thiết lập liên kết và hủy bỏ
  + Đánh số tuần tự: đánh số tuần tự, dựa vào đó bên nhận sắp xếp lại chính xác và phát hiện lỗi
  + Đảm bảo tính tin cậy: nếu có lỗi, thất lạc thì TCP sẽ gửi lại
  + Điểu khiển lưu lượng: tránh quá tải bên nhận -> tràn bộ nhớ -> treo

1. **Giao thức UDP**

* Giao thức UDP là giao thức thuộc tầng vận tải và được mô tả trong RFC 768, ngoài chức năng ghép kênh/phân kênh, UDP có cũng cơ chế phát hiện lỗi nhưng rất đơn giản
* Giao thức UDP không đòi hỏi bên gửi và bên nhận phải thiết lập liên kết trước khi trao đổi dữ liệu, vì vậy UDP được xem là giao thức kết nối vô hướng hay không liên kết
* Giao thức UDP dường như không có có nhiều ưu điểm như giao thức TCP: truyền dữ liệu tin cậy, kiểm soát lưu lượng…, tuy nhiên trên thực tế giao thức UDP được sử dụng nhiều hơn vì những đặc điểm sau:
  + Không cần thiết lập liên kết: không mất thời gian thiết lập liên kết truyền
  + Không duy trì trạng thái kết nối: UDP không phải lưu giữ những thông tin như vậy, do đó nếu phía máy chủ sử dụng UDP thì có khả năng phục vụ đồng thời nhiều máy khách hơn
    - TCP ghi nhớ trạng thái kết nối trên hệ thống đầu cuối, trạng thái kết nối -> giúp tin cậy hơn
  + Thông tin điều khiển ít hơn: 8 bytes
    - TCP: 20 bytes
  + Không kiểm soát tốc độ gửi: Bên nhận không nhất thiết phải nhận toàn bộ dữ liệu, khi nghẽn mạng một phần dữ liệu có thể bị mất do tràn vùng đệm ở các thiết bị mạng, tốc độ nhận có thể bị giới hạn do tắc nghẽn ngay cả khi tốc độ gửi không bị hạn chế
* Các ứng dụng phổ biến như: Email, truy cập từ xa, Web và truyền tập tin chạy trên nền TCP vì chúng cần đến dịch vụ truyền dữ liệu tin cậy. Tuy nhiên có một số ứng dụng khác thích hợp với UDP hơn TCP như: bản định tuyến RIP, quản trị mạng SNMP, DNS
* Ngày nay UDP thường được các ứng dụng đa phương tiện như điện thoại Internet, hội thảo từ xa, các ứng dụng thời gian thực
* Nhược điểm: thiếu cơ chế kiểm soát tắc nghẽn

1. Cấu trúc dữ liệu của giap thức UDP

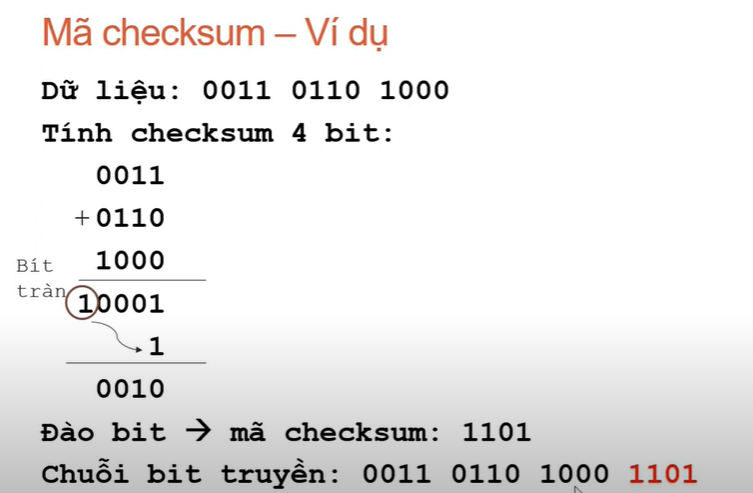
* Được mô tả trong RFC 768, dữ liệu của ứng dụng nằm trong trường dữ liệu của UDP datagram



* Phần thông tin điều khiển UDP có bốn trường, độ lớn mỗi trường là hai byte. Số hiệu cổng cho phép thiết bị gửi chuyển dữ liệu tới đúng tiến trình chạy trên thiết bị nhận. Trường Checksum được bên nhận sử dụng để kiểm tra trong datagram có lỗi hay không. Trường độ dài (Length) cho biết độ dài (tính theo byte) của toàn bộ UDP datagram kể cả phần thông tin điều khiển

1. Cách tính kiểm tra tổng trong UDP

* Kiểm tra tổng trong giao thức UDP được sử dụng để phát hiện lỗi nhưng không xử lý lỗi
* Mã checksum:



1. **Các nguyên lý truyền dữ liệu tin cậy**

* Truyền dữ liệu tin cậy là một trong những chức năng chính của tầng vận tải
* Cần đến những giao thức truyền dữ liệu tin cậy bất chấp các tầng phía dưới sử dụng các giao thức truyền tin không tin cậy
  + Bên gửi sẽ chuyển dữ liệu từ phía trên xuống qua hàm rdt-send()
  + phía nhận sử dụng hàm rdt\_rcv() để lấy gói dữ liệu từ tầng thấp hơn và thực hiện các thủ tục cần thiết để đảm bảo dữ liệu nhận được chính xác
  + chuyển dữ liệu lên tầng trên bằng cách gọi hàm deliver\_data()

1. **Xây dựng giao thức truyền dữ liệu tin cậy**

* Để tìm hiểu cơ chế truyền dữ liệu tin cậy, chúng ta sẽ mô tả trạng thái của phía nhận và phía gửi bằng kỹ thuật máy trạng thái hữu hạn (FSM - finite state machine)
* Máy trạng thái hữu hạn là một đồ thị có hướng của các sự kiện, hành động và trạng thái

1. Truyền dữ liệu tin cậy trên kênh tin cậy hoàn toàn

* Giao thức đơn giản nhất rdt 1.0 sử dụng kênh truyền dữ liệu tin cậy ở tầng thấp hơn, nghĩa là không xảy ra lỗi bit hoặc mất mát gói tin ở tầng dưới
* Với rdt1.0, việc gửi đơn giản chỉ là nhận dữ liệu từ tầng trên thông qua sự kiện rdt\_send(data), tạo ra đoạn dữ liệu bằng hàm make\_data (packet,data) và gửi đoạn dữ liệu lên kênh truyền.
* Việc nhận gói dữ liệu từ kênh truyền bằng sự kiện rdt\_rcv(packet), trích xuất dữ liệu ra khỏi gói dữ liệu bằng hàm extract (packet,data) và đưa dữ liệu lên tầng trên

1. Truyền dữ liệu tin cậy trên kênh truyền có lỗi bit

* Rdt 2.0
* Trong mạng máy tính, giao thức truyền tin cậy dựa trên cơ chế truyền lại như vậy được gọi là các giao thức yêu cầu nhắc lại tự động (ARQ - Automatic Repeat Request). Các giao thức ARQ cần phải có ba khả năng sau để xử lý trường hợp có lỗi bit:
  + Phát hiện lỗi: sử dụng kỹ thuật vòng CRC
  + Phản hồi từ phía người nhận: Phía nhận gửi thông tin phản hồi thông báo tình trạng nhận cho phía gửi
    - Báo đúng: ACK
    - Báo sai: NAK
  + Truyền lại: đoạn dữ liệu lỗi sẽ đc chuyển lại
* Trong giao thức rdt 2.0, bên gửi có hai trạng thái, trạng thái chờ dữ liệu từ tầng trên chuyển xuống và trạng thái đợi phản hồi ACK hoặc NAK từ bên nhận = rdt\_rcv
* Hoạt động thuộc kiểu dừng và chờ (stop and wait)
* Máy trạng thái hữu hạn bên nhận trong giao thức rdt 2.0 chỉ có một trạng thái duy nhất, đó là trạng thái chờ tầng dưới gọi. Khi nhận được đoạn dữ liệu, bên nhận sẽ kiểm tra xem đoạn dữ liệu nhận được có lỗi bit hay không, nếu xảy ra lỗi sẽ phản hồi NAK, ngược lại sẽ phản hồi ACK
* Rdt 2.1 là phiên bản vá lỗi của rdt 2.0 trong trường hợp ACK/NAK có thể bị lỗi bằng cách thêm một trường thứ tự cho đoạn dữ liệu
* Rdt 2.2 là giao thức truyền dữ liệu tin cậy trên kênh truyền có bít lỗi không sử dụng NAK

1. Truyền dữ liệu tin cậy khi có lỗi mất đoạn tin

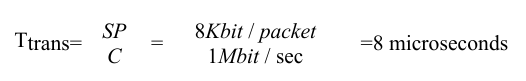
* Rdt 3.0
* Đối với bên gửi, truyền lại là giải pháp chung nhất, bên gửi không cần biết đoạn dữ liệu nay xác nhận ACK bị mất, hoặc chỉ đơn giản là chúng đến trễ, dù là trường hợp thì thao tác truyền lại sẽ là đơn giản nhất. Giao thức 2.2 sẽ sử lí dữ liệu trùng lặp
* Giao thức rdt 3.0 được gọi là giao thức một bit luân chuyển vì số thứ tự của đoạn dữ liệu thay đổi lần lượt giữa 0 và 1

1. **Điều khiển lưu lượng**

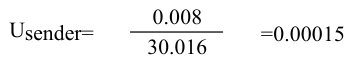
* Điều khiển lưu lượng là hành động thay đổi tốc độ chuyển dữ liệu giữa bên gửi và bên nhận để tránh hiện tượng bên nhận không kịp xử lý.
* Nhiệm vụ của nó là đảm bảo rằng bên gửi không thể tiếp tục truyền dữ liệu nhanh hơn mức mà bên nhận có thể tiếp thu được.
* Điều khiển lưu lượng được thực hiện bằng cách bên nhận thông báo cho bên gửi biết về khả năng xử lý dữ liệu của nó
* Điều khiển tắc nghẽn thực hiện nhiệm vụ đảm bảo cho mạng có khả năng vận chuyển lưu lượng đưa vào
* Điều khiển lưu lượng là để tránh tắc nghẽn, còn điều khiển tắc nghẽn là để giải quyết vấn đề tắc nghẽn hoặc có dấu hiệu tắc nghẽn sắp xảy ra
* Giải pháp điều khiển lưu lượng được sử dụng rộng rãi nhất là dùng cơ chế cửa sổ trượt
  + Cơ chế điều khiển lưu lượng bằng cửa sổ trượt cho phép bên gửi phát đi liên tiếp một số đơn vị dữ liệu nhất định rồi dừng lại và chờ thông báo về kết quả nhận trước khi tiếp tục phát các đơn vị dữ liệu tiếp theo

1. **Nâng cao hiệu năng bằng đường ống Pipeline**

* Mặc dù hoạt động đúng nhưng hiệu suất hoạt động của rdt 3.0 chưa cao, điểm yếu của vấn đề là do thao tác dừng và chờ
* Thời gian trễ giữa hai thiết bị, dù tín hiệu lan truyền với tốc độ ánh sáng, trong thực tế độ trễ trên các thiết bị mạng lớn hơn rất nhiều, là Pprop xấp xỉ 15 ms. Giả sử hai thiết bị được kết nối bằng đường truyền tốc độ C =1 Gbit/s. Kích thước của đoạn dữ liệu SP=1Kbyte/đoạn, thời gian cần thiết để truyền toàn bộ đoạn dữ liệu trên kênh truyền tốc độ 1 Gbps được tính bởi công thức:

****

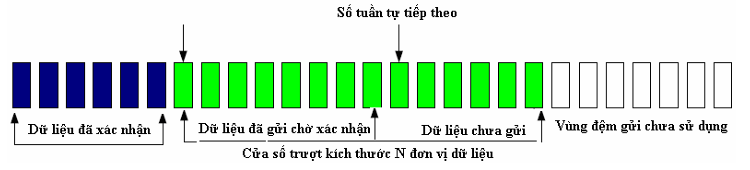
* Với giao thức dừng và chờ, nếu phía gửi bắt đầu gửi đoạn dữ liệu tại thời điểm t = 0 thì tại thời điểm t = 8 microsecond, bit cuối cùng mới được bên gửi đẩy ra đường truyền. Tiếp theo phải mất 15 ms để cả đoạn dữ liệu đi từ phía gửi sang phía nhận như vậy bit cuối cùng của gói dữ liệu đến đích tại thời điểm t = 15.008ms.
* Để đơn giản, ta giả thiết gói ACK có cùng độ dài với đoạn dữ liệu và bên nhận gửi ngay bản tin xác nhânh ACK khi nhận được bit cuối cùng của đoạn dữ liệu. Như vậy bit cuối cùng của đoạn tin xác nhận ACK được truyền tới đích tại thời điểm t = 30.016 ms, trong khoảng thời gian này bên gửi chỉ hoạt động trong 0.016 ms. Nếu định nghĩa hiệu suất của bên gửi là tỷ lệ thời gian bên gửi hoạt động chúng ta có hiệu suất Usender rất thấp:

****

* Điều đó có nghĩa là bên gửi chỉ hoạt động trong khoảng 0.15 phần nghìn thời gian. Theo cách tính khác bên gửi gửi 1 Kbyte trong 30,016 milisecond tương đương với tốc độ truyền là 33 Kbyte/s thấp hơn nhiều so với tốc độ có thể là 1 Gigabit/s
* Giải pháp cho vấn đề hiệu suất sẽ là cho phép phía gửi gửi đồng thời nhiều đoạn dữ liệu mà không cần phải đợi bản tin xác nhận – kĩ thuật pipeline
* Kỹ thuật này làm tăng hiệu suất của giao thức lên nhiều lần, tuy nhiên nó đòi hỏi những yêu cầu sau:
  + Khoảng số thứ tự phải tăng
  + Bên gửi và bên nhận có thể phải sử dụng bộ đệm cho nhiều đoạn dữ liệu
* Yêu cầu về khoảng số thứ tự cần thiết cũng như về vùng đệm phụ thuộc vào cách giao thức xử lý việc mất dữ liệu, dữ liệu bị lỗi, bị trễ. Có hai cách tiếp cận chính được trình bày ở đây: Quay lui N (Go-Back-N) và lặp có lựa chọn

1. **Giao thức quay lui-N**

* Trong giao thức quay lui-N, bên gửi cho phép truyền đi đồng thời nhiều đoạn dữ liệu mà không phải đợi xác nhận. Tuy nhiên tổng số đoạn dữ liệu không phải là vô hạn mà bị giới hạn bởi giá trị N, đó là số lượng tối đa đoạn dữ liệu chưa được xác nhận trong đường ống

****

* Định nghĩa base là số thứ tự của đoạn dữ liệu đã được truyền đi lâu nhất chưa được xác nhận và next seqnum là số thứ tự nhỏ nhất chưa được sử dụng, đó là số thứ tự cho đoạn tiếp theo sẽ gửi
  + Khoảng [0,base-1] ứng với số thứ tự của các đoạn dữ liệu đã được truyền đi và đã được xác nhận
  + Khoảng [base, nextseqnum-1] ứng với các đoạn dữ liệu đã được gửi đi nhưng chưa được xác nhận
  + Khoảng [nextseqnum, base +N- 1] có thể được sử dụng làm số thứ tự cho các đoạn sẽ được gửi nếu như có dữ liệu từ tầng trên chuyển xuống
  + Khoảng từ [base+n] trở lên chưa được sử dụng cho đến khi nhận được các bản tin xác nhận các đoạn tin đã gửi
* Giao thức GBN là giao thức cửa sổ trượt (sliding- window) – máy trạng thái hữu hạn mở rộng
* Trong giao thức GBN, bên gửi phải đáp ứng ba sự kiện sau:
  + Có dữ liệu từ trên xuống: rdt\_send() được phía trên sử dụng để chuyển dữ liệu xuống, bên gửi phải ktra xem cửa sổ đầy chưa, nếu rồi thì thông báo tầng trên và k chấp nhận dữ liệu
  + Nhận được một ACK: đoạn tin có số thứ tự <= n thì đúng
  + Hết thời gian đợi: giống giao thức dừng và chờ
* Sử dụng ACK tích luỹ là sự lựa chọn tốt nhất cho giao thức GBN. Trong giao thức GBN, bên nhận loại bỏ đoạn tin không theo thứ tự
* Có thể xem TCP là sự kết hợp của cả hai giao thức GBN và SR (Lặp có lựa chọn)

1. **Giao thức lặp có lựa chọn**

* Vấn đề: Khi kích thước cửa số và truyền bản tin lớn, có thể 1 đoạn tin bị lỗi -> GBN truyền lại nhiều đoạn tin -> giảm hiệu suất
* Giao thức lặp có lựa chọn (SR - Selective Repeat) tránh việc truyền lại không cần thiết bằng cách bên gửi chỉ gửi lại các đoạn tin mà nó cho là có lỗi hoặc mất.
* Khác với GBN, bên gửi sẽ nhận được biên nhận ACK cho một số đoạn tin trong cửa sổ
* Bên nhận của giao thức lặp có lựa chọn sẽ xác nhận cho bất kỳ đoạn tin nhận đúng, cho dù không theo đúng thứ tự. Đoạn tin không đúng thứ tự vẫn được lưu giữ lại cho đến khi tất cả các đoạn tin còn thiếu (đoạn tin có số thứ tự nhỏ hơn) được chuyển đến, khi đó tất cả các đoạn tin sẽ được chuyển lên tầng trên theo đúng thứ tự
* **Tóm lại** là nó có thông báo ACK cho từng đoạn thay vì cả đoạn như GBN và nó xác nhận các đoạn đúng ko theo trình tự như GBN và lưu lại cho đến khi các đoạn thiếu được truyền lại còn GBN là xác nhận theo trình tự nếu sai là truyền lại cả đoạn

**Sự kiện và phản ứng của bên gửi**

* Dữ liệu nhận được từ phía trên: giống GBN
  + Dùng bộ đệm thời gian đếm thời gian chuyển đi nếu lâu quá thì gửi lại còn không thì nhận ACK
* Hết thời gian đợi: sử dụng đồng hồ hệ thống giữ vai trò đồng bộ cho các bộ đếm thời gian
* Nhận được ACK:

**Sự kiện và phản ứng của bên nhận**

* Nhận đúng đoạn tin nhắn với số thứ tự trong khoảng [rcv\_base,rcv\_base+N-1]: chuyển đoạn tin nhắn
* Nhận được đoạn tin với số thứ tự trong [rcv\_base-N,rcv\_base-1]: xác nhận lại
* Các trường hợp khác: Bỏ qua đoạn tin nhắn
* Giải pháp trong thực tế là bảo đảm số thứ tự không được sử dụng lại cho đến khi bên gửi có thể tương đối chắc chắn về đoạn tin với số thứ tự X được gửi trước đây không còn tồn tại trong mạng. Điều này được thực hiện với giả thiết một đoạn tin không thể tồn tại trong mạng trong một khoảng thời gian lớn hơn một khoảng thời gian cố định nào đó, thời gian sống lớn nhất của đoạn tin xấp xỉ là 3 phút [RFC 1323]

**Chương 5: Lập trình SOCKET**

1. **Khái niệm về socket**

* Socket là một phương pháp để thiết lập kết nối truyền tin giữa một chương trình yêu cầu dịch vụ và một chương trình cung cấp dịch trên mạng và đôi khi trong máy tính
* Là cánh cửa giữa các tiến trình ứng dụng và giao thức giao vận end-to-end

1. **Mô hình khách/chủ**

* Mô hình truyền tin khách/chủ hướng tới việc cung cấp dịch vụ, quá trình trao đổi dữ liệu bao gồm:
  + Tiến trình khách gửi yêu cầu tới tiến trình chủ
  + Tiến trình chủ kiểm tra và xử lý yêu cầu của tiến trình khách
  + Tiến trình chủ gửi kết quả xử lý yêu cầu cho khách
* Như vậy, mô hình khách/chủ thực chất gồm hai bước truyền thông điệp và một bước xử lý, như vậy nảy sinh vẫn đề đồng bộ client và server.
* Mô hình khách/chủ thường được cài đặt dựa trên các thao tác cơ bản là gửi và nhận
* Quá trình giao tiếp giữa client và server có thể diễn ra theo một trong hai chế độ: đồng bộ hoặc không đồng bộ
  + Chế độ đồng bộ: khi tiến trình client hoặc server phát ra lệnh gửi dữ liệu, việc thực thi của tiến trình sẽ bị tạm ngừng cho tới khi tiến trình nhận phát ra lệnh nhận dữ liệu. Tương tự đối với tiến trình nhận dữ liệu
  + Chế độ không đồng bộ: khi tiến trình client hay server phát ra lệnh gửi dữ liệu thực sự, việc thực thi của tiến trình vẫn được tiến hành mà không quan tâm đến việc có tiến trình nào phát ra lệnh nhận dữ liệu đó hay không. Tương tự cho trường hợp nhận dữ liệu

1. **Các kiến trúc khách/chủ**
2. Khách/chủ hai tầng

* Khối lượng công việc xử lý được dành cho phía khách trong khi tiến trình chủ chỉ đơn giản đóng vai trò như là chương trình kiểm soát luồng vào ra giữa ứng dụng và dữ liệu.
* Có thể tìm thấy những ứng dụng loại này như dịch vụ truy nhập từ xa (telnet), truyền tập tin (ftp), thậm chí là cả dịch vụ Web (web tĩnh).
* Kiến trúc này có nhược điểm sau:
  + Tăng lưu lượng thông tin lưu chuyển trên mạng: tiến trình khách cần nhiều phụ trợ -> phải gửi nhiều yêu cầu đến tiến trình chủ
  + Giảm hiệu năng xử lý: các máy trạm có cấu hình thấp hơn máy chủ
  + Khó khăn trong việc bảo trì và nâng cấp hệ thống: phải thay đổi cả phía khách lẫn chủ

1. Khách/chủ 3 tầng

* Hạn chế các nhược điểm của kiến trúc khách/chủ hai tầng bằng cách thêm một tầng mới tách biệt việc xử lý dữ liệu với việc cung cấp dịch vụ
  + Tầng đầu tiên là tầng trình diễn thường chỉ đóng vai trò hiển thị các thông tin cho người sử dụng
  + Tầng thứ hai, thực thi các công việc xử lý nghiệp vụ
  + Tầng thứ ba chứa dữ liệu cần thiết cho ứng dụng, dữ liệu này có thể bao gồm bất kỳ nguồn thông tin nào (Oracle, SQL Server hoặc tài liệu XML…)
* Sử dụng phổ biến nhất

1. Kiến trúc n-tầng

* Việc thêm mỗi tầng sẽ phức tạp hơn trong vấn đề xử lý, tăng giá thành sản phẩm phần mềm. Về cơ bản việc phân tầng phải đảm bảo các nguyên tắc sau:
  + Tầng ứng dụng: quan lý tương tác người dùng với ứng dúng
  + Tầng trình diễn: xác định cách thức hiện thị giao diện người dùng và các quản lý các yêu cầu của người dùng
  + Tầng nghiệp vụ: mô hình hóa các quy tắc nghiệp vụ
  + Tầng dịch vụ hạ tầng: cung cấp các chức năng cần thiết

1. **Mô hình truyền tin socket**

* Hai giao thức TCP và UDP là các giao thức tầng vận tải để truyền dữ liệu, mỗi giao thức có những ưu và nhược điểm riêng. Giao thức TCP là giao thức có liên kết, nó có độ tin cậy truyền tin cao nhưng tốc độ truyền tin bị hạn chế do phải có giai đoạn thiết lập và giải phóng liên kết, khi đoạn tin có lỗi hay bị thất lạc thì giao thức TCP phải thực hiện truyền lại. Giao thức UDP thuộc loại không liên kết, có tốc độ truyền tin rất nhanh vì không cần phải thiết lập và giải phóng liên kết
* Khi lập trình cho giao thức TCP thì sử dụng các socket dạng luồng, còn đối với giao thức UDP sẽ sử dụng DatagramSocket và DatagramPacket
* Truyền tin có liên kết nghĩa là cần phải có giai đoạn thiết lập và giải phóng liên kết trước và sau khi truyền tin
* Dữ liệu được truyền trên mạng máy tính dưới dạng các gói tin, mỗi gói tin chứa thông tin điều khiển bao gồm địa chỉ + cổng nguồn và đích, chiều dài hữu hạn
* Socket coi tất cả các thao tác gửi/nhận tương tự như thao tác đọc/ghi tập tin trên ổ đĩa máy tính, các thao tác mức thấp hoàn toàn trong suốt đối với người lập trình.
* Một socket có thể thực hiện bảy thao tác cơ bản:
  + Kết nối với một máy ở xa
  + Gửi dữ liệu
  + Nhận dữ liệu
  + Ngắt dữ liệu
  + Gán cổng
  + Nghe dữ liệu đến
  + Chấp nhận liên kết từ các máy ở xa trên cổng đã được gán

1. **Java sockets**

* Trong Java, thư viện Java.net.Socket là lớp được dùng trong việc tạo ra các socket phía client và java.net.ServerSocket được dùng cho phát triển ứng dụng phía server

1. **Socket cho phía máy chủ**

* Một ServerSocket mới được tạo ra trên một cổng xác định bằng cách sử dụng phương thức ServerSocket(số hiệu cổng) tuân thủ quy định IANA
* ServerSocket lắng nghe liên kết đến trên cổng đó bằng cách sử dụng phương thức accept()
* Tùy thuộc vào kiểu tiến trình chủ, hoặc phương thức getInputStream(), getOutputStream() hoặc cả hai được gọi để nhận các luồng vào ra để truyền tin với client
* Tiến trình chủ và khách tương tác theo một giao thức thỏa thuận sẵn cho tới khi một trong hai phía hoặc cả hai phía gửi yêu cầu hủy bỏ liên kết
* Tiến trình chủ, tiến trình khách hoặc cả hai gửi yêu cầu hủy bỏ liên kết
* Tiến trình chủ trở về bước hai và đợi yêu cầu thiêt lập liên kết tiếp theo

1. **Socket cho tiến trình trên máy khách**

* Hàm public Socket(string host,int port) throws UnknownHostException, IOException tạo một socket TCP, trong đó host là địa chỉ hoặc tên miền (trong trường hợp này phải sử dụng dịch vụ DNS) và port là số hiệu cổng cung cấp dịch vụ của server

1. **Máy chủ đa xử lý**

* Mỗi ứng dụng cung cấp dịch vụ trên máy chủ thường phải phục vụ đồng thời yêu cầu của nhiều tiến trình trên máy khách, các ứng dụng trên máy chủ thường được xây dựng theo kiểu đa tiến trình (multiprocessing), mỗi tiến trình lại được xây dựng dựa trên nguyên lý xử dụng đa luồng (multithread)
* Thread: trong hệ điều hành có ☺

1. **Lập trình socket với ngôn ngữ C**

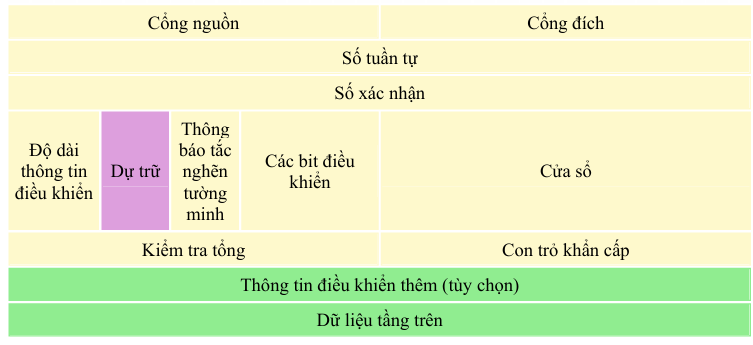
* Các hàm phục vụ cho lập trình socket với ngôn ngữ C nằm trong thư viện socket.h
* Các bước tạo lập một socket phía server gồm:
  + Tạo một socket bằng cách gọi hàm socket().
  + Nhúng socket đến địa chỉ của máy chủ sử dụng hàm bind(), bao gồm địa chỉ máy chủ và số hiệu cổng mà dịch vụ sẽ cung cấp.
  + Nghe các yêu cầu kết nối đến từ client bằng cách sử dụng hàm listen().
  + Tiếp nhận các kết nối sử dụng hàm accept().
  + Trao đổi thông tin với client bằng các hàm read() và write().
  + Đóng kết nối bằng hàm close().
* Các bước thiết lập một socket phía client gồm:
  + Tạo một socket bằng hàm socket()
  + Gửi yêu cầu kết nối đến server bằng hàm connect().
  + Trao đổi thông tin với server bằng các hàm read() và write().
  + Đóng kết nối bằng hàm close()

**Chương 6: Giao Thức TCP**

* Không những đảm bảo truyền tin mà còn hỗ trợ nhiều giao thức tầng ứng dụng cung cấp các dịch vụ phổ biến

1. **Cấu trúc đơn vị dữ liệu của giao thức TCP**

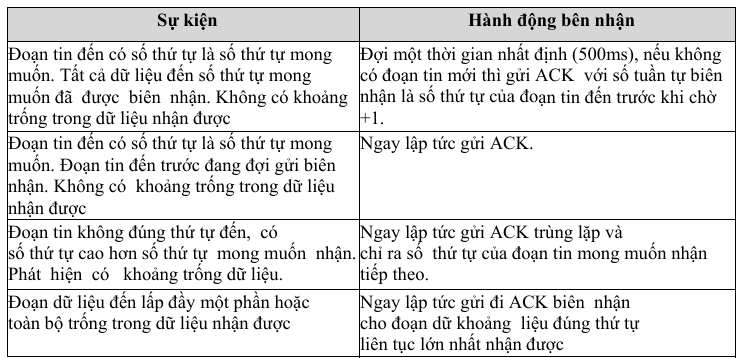
* Đơn vị dữ liệu của giao thức TCP gọi là đoạn dữ liệu (Segment), đoạn dữ liệu bao gồm các trường thông tin điều khiển và dữ liệu của lớp ứng dụng

****

* + Cổng nguồn, cổng đích (Source Port, Destination Port): Trường số hiệu cổng nguồn, số hiệu cổng đích để thực hiện dịch vụ ghép kênh/phân kênh dữ liệu cho các ứng dụng lớp trên
  + Số thứ tự (Sequence Number): Trường số thứ tự đoạn tin bên gửi dài 32 bit được sử dụng để hỗ trợ cho dịch vụ truyền dữ liệu tin cậy, trường này có quan hệ mật thiết với cờ điều khiển SYN
    - 1 thì đó là giá trị khởi tạo số tuần tự của đoạn tin đầu tiên
    - 0 thì đó là số tuần tự tích lũy của đoạn tin
  + Số thứ tự xác nhận (Acknowledgment Number): Trường số thứ tự xác nhận dài 32 bit được bên nhận sử dụng trong việc cung cấp dịch vụ truyền dữ liệu tin cậy
  + Độ dài phần thông tin điều khiển (Header Length): 4 bit xác định độ dài của phần thông tin điều khiển (đơn vị là 4 bytes), thông thường phần thông tin điều khiển chiếm 20 bytes
  + Thông báo tắc nghẽn tường minh (ECN - Explicit Congestion Notification): cho phép thông báo tường minh tình trạng tắc nghẽn trên mạng giữa đầu cuối với đầu cuối
  + Các bit điều khiển: Bit ACK được sử dụng để chỉ ra rằng giá trị đặt trong trường xác nhận là đúng. Các bit RST, SYN và FIN được sử dụng trong việc thiết lập hay đóng kết nối
  + Cửa sổ (Window): Trường độ lớn cửa sổ 16 bit được sử dụng để kiểm soát lưu lượng, đó là số lượng byte dữ liệu tối đa mà bên nhận có thể chấp nhận được
  + Kiểm tra tổng (Checksum): Giá trị kiểm tra lỗi, được tính bằng phần bù của tổng chuỗi 16 bit
  + Con trỏ khẩn cấp (Urgent Pointer): Cho biết vị trí byte cuối cùng của dữ liệu khẩn cấp
  + Thông tin điều khiển thêm: Trường này được sử dụng để bên gửi và bên nhận có thể thương lượng về giá trị MSS hoặc giá trị gia tăng của cửa sổ trong mạng cao tốc, lựa chọn nhãn thời gian
* Một phiên làm việc của TCP bao gồm ba bước: Thiết lập liên kết, truyền dữ liệu và hủy bỏ liên kết

1. **Truyền dữ liệu tin cậy**

* Tầng dưới của giao thức TCP là giao thức IP, nó không đảm bảo tính chính xác cũng như thứ tự của các đoạn tin, các gói tin IP có thể bị tràn
* Giao thức TCP tạo ra một đường truyền dữ liệu tin cậy trên nền giao thức IP không tin cậy
* Dịch vụ truyền dữ liệu tin cậy của TCP đảm bảo dòng dữ liệu tới tiến trình nhận không có lỗi, liên tục, không trùng gặp và đúng thứ tự, nghĩa là dòng byte nhận được giống hệt dòng byte gửi đi
* Giao thức TCP cung cấp dịch vụ truyền dữ liệu tin cậy bằng cách sử dụng cơ chế phối hợp số tuần tự, xác nhận số tuần tự ACK và đồng hồ xác định thời gian quá hạn mỗi đoạn tin cần phải phản hồi
* TCP cũng thực hiện việc gửi liên tục (cơ chế đường ống - pipline), cho phép bên gửi có thể gửi nhiều đoạn tin mà chưa cần nhận biên nhận ngay
* Cơ chế truyền tin:
  + ACK lần đầu tiên:

****

* + ACK trùng lặp: truyền lại nhanh:
    - GBN truyền tất cả các đoạn tin có số tuần tự nhỏ hơn hoặc bằng số tuần tự của đoạn tin bị lỗi trong khi đó TCP chỉ truyền lại bản tin bị lỗi

1. **Điều khiển luồng**

* Khi kết nối TCP nhận được các đoạn dữ liệu, nó sẽ đặt chúng vào một vùng nhớ tạm thời gọi là đệm nhận. Vì một lý do nào đó, tiến trình đọc chưa đọc kịp dữ liệu trong bộ đệm, khi đó sẽ xảy ra tình trạng tràn bộ đệm.
* Để giải quyết vấn đề này, TCP cung cấp dịch vụ điều khiển lưu lượng, thực chất đó là quá trình làm tương thích về tốc độ gửi/nhận
* Giả sử máy A gửi một tập tin đến máy B qua kết nối TCP:
  + Máy B sẽ khởi tạo bộ đệm cho kết nối này với độ lớn RcvBuffer.
  + Tiến trình ứng dụng trên B đọc dữ liệu từ bộ đệm.
  + Giả thiết LastByteread là số thứ tự của byte cuối cùng trong dòng dữ liệu mà tiến trình ứng dựng trong máy B đọc từ vùng đệm
  + LastByteRcvd là số byte cuối cùng trong dòng dữ liệu đến từ mạng và được để trong đệm nhận của máy B
* Giao thức TCP không được phép tràn bộ đệm nên LastByteRcvd - LastByteread < RcvBuffer, cửa sổ nhận RcvWindow là độ lớn vùng đệm rỗi:
  + RcvWindow = RcvBuffer - [LastByteRcvd - LastByteread]
  + RcvWindow = 0 khi bộ đệm bị đầy
* Máy A cũng có hai biến LastByteSent và LastByteAcked, độ lệch giữa hai biến này LastByteSent - LastByteAcked là số lượng dữ liệu chưa được xác nhận mà A gửi qua kết nối
* Trong suốt thời gian kết nối, A phải đảm bảo LastByteSent - LastByteAcked <= RcvWindow để không làm tràn bộ đêm bên B
* Khác với TCP, giao thức UDP không có cơ chế kiểm soát lưu lượng, nó đặt các datagram vào trong một hàng đợi có độ lớn hữu hạn. Tiến trình đọc lần lượt từng datagram trong hàng đợi, nếu tốc độ đọc của tiến trình không đủ nhanh thì hàng đợi sẽ tràn và các datagram đến sau sẽ bị mất

1. **Quản lý kết nối**

* Giao thức TCP thuộc loại kết nối có hướng
* Ba giai đoạn: thiết lập liên kết, truyền dữ liệu và hủy bỏ liên kết
* thực thể TCP máy khách khởi tạo kết nối TCP tới thực thể TCP trên máy chủ qua ba bước sau:
  + Bước 1: Máy khách tạo một đoạn tin đặc biệt: cờ SYN
  + Bước 2: Khi nhận được đoạn tin SYN
  + Bước 3: Nhận được đoạn tin SYNACK (bit SYN + bit ACK)
* Cờ Fin để gửi tin
* Quá trình trao đổi thông tin được thực hiện qua các Socket, mỗi socket có thể nhận các trạng thái sau:
  + LISTEN:
  + SYN-SENT:
  + SYN-RECEIVED:
  + ESTABLISHED:
  + TIME-WAIT:

1. **Điều khiển tắc nghẽn**

* Một chức năng quan trọng
* Cơ chế này của TCP chỉ dưa vào các thiết bị đầu cuối chứ không dựa vào cơ chế kiểm soát tắc nghẽn của tầng mạng vì tầng IP không cung cấp cho TCP các thông tin minh bạch khi có tắc nghẽn
* Kết nối TCP kiểm soát tốc độ truyền của nó bằng cách giới hạn số lượng các đoạn tin đã gửi nhưng chưa được biên nhận
* Cơ chế kiểm soát tắc nghẽn của TCP bổ sung thêm hai biến nữa: cửa sổ tắc nghẽn và ngưỡng tắc nghẽn
* Cửa sổ tắc nghẽn ký hiệu là CongWin biểu thị số lượng dữ liệu tối đa mà người gửi có thể gửi qua kết nối
  + LastByteSend - LastByteAcked <= min {CongWin,RcvWin}
* Ngưỡng ký hiệu là threshold sẽ ảnh hưởng tới quá trình tăng của cửa sổ tắc nghẽn CongWin
* Cửa sổ tắc nghẽn CongWin tăng theo hàm số mũ đến khi vượt ngưỡng thì sẽ tăng tuyến tính, giai đoạn này còn gọi là tránh tắc nghẽn. Khi hết thời gian đợi, giá trị ngưỡng bằng một nửa giá trị cửa sổ tắc nghẽn hiện thời và cửa nó nhận giá trị 1
  + TCP được coi là thuật toán AIMD (additive- increase, multiplicative-decrease)

**Thuật toán Tahoe,Reno và Vegas:**

* Là các biến thể của thuật toán kiểm soát tắc nghẽn

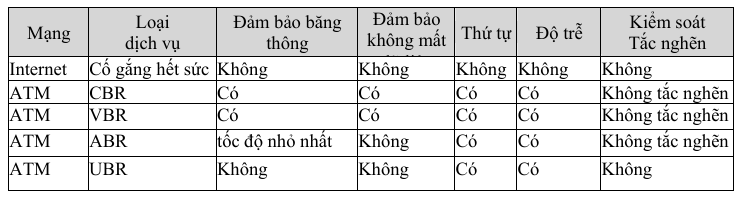
**Chương 7: Tầng Mạng và Giao Thức IP**

1. **Mô hình dịch vụ tầng mạng**

* Cung cấp dịch vụ truyền thông giữa hai tiến trình đang chạy trên hai máy tính khác nhau
* Không giống tầng vận tải, tầng mạng gồm nhiều máy tính và các thiết bị mạng trung gian, vì vậy giao thức tầng mạng là một trong những giao thức phức tạp nhất
* Vai trò chính của thiết bị định tuyến là xác định đường đi tiếp theo cho mỗi gói tin và chuyển gói tin từ một cổng giao diện sang cổng giao diện khác
* Có ba chức năng quan trọng sau đây:
  + Xác định đường đi: thuật toán định tuyến
  + Chuyển mạch: quyết định gói tin đến các cổng định tuyến hợp lý
  + Thiết lập đường truyền: 2 thực thể truyền thông phải có một giai đoạn thiết lập liên kết trước khi trao đổi dữ liệu, quá trình này cho phép bên gửi và bên nhận thiết lập các thông tin trạng thái cần thiết

1. **Nguyên lý chuyển mạch tầng mạng**

* Tầng mạng sử dụng mạch ảo (VC - Virtual Circuit) để truyền các gói tin, về khía cạnh nào đó mạch ảo tương tự mạng điện thoại truyền thống, có ba giai đoạn trong chuyển mạch ảo:
  + Thiết lập mạch ảo: yêu cầu thiết lập mạng ảo -> cập nhật bảng định tuyến và dự trữ tài nguyên
  + Truyền dữ liệu: dữ liệu được chuyển vào mạch ảo đó
  + Đóng mạch ảo: thông báo đóng -> cập nhật bảng định tuyến..
* Bản tin trao đổi giữa các thiết bị đầu cuối yêu cầu khởi tạo hay kết thúc mạch ảo, bản tin trao đổi giữa các thiết bị chuyển mạch yêu cầu cập nhật bảng chuyển mạch gọi là bản tin báo hiệu, giao thức được sử dụng để trao đổi những bản tin này gọi là giao thức báo hiệu
* Dịch vụ của mạng ATM là mạch ảo
* Dịch vụ chuyển mạch ảo được xếp vào lớp dịch vụ kết nối có hướng

****

* Tầng Internet của mô hình TCP/IP không đảm bảo thời gian gửi các gói tin giống nhau, các gói tin không đến đích theo đúng thứ tự và thậm chí không đảm bảo gói tin đến được đích
* Dịch vụ truyền với tốc độ cố định (CBR - Constant Bit Rate) là dịch vụ ATM đầu tiên được chuẩn hoá và là lựa chọn lý tưởng cho việc truyền dữ liệu đa phương tiện theo thời gian thực
  + Trong dịch vụ CBR các tế bào được truyền qua mạng với một độ trễ nào đó
  + Tỉ lệ tế bào được mất hay trễ không vượt một giá trị ngưỡng
* Dịch vụ truyền với tốc độ không xác định (UBR - Unspecified Bit Rate) chỉ đảm bảo gửi các tế bào theo đúng thứ tự
  + Không đảm bảo tốc độ cũng như độ trễ, nó chỉ đảm cố gắng hết sức phân phát các tế bào
  + Không cung cấp thông tin phản hồi
* Dịch vụ truyền với tốc độ sẵn có (ABR - Available Bit Rate) tương tự như UBR nhưng bổ sung thêm hai tính năng:
  + Tốc độ truyền tế bào nhỏ nhất (MRC) được đảm bảo cho kết nối ABR -> tốc độ cao hơn khi tải tài nguyên của mạng rỗi
  + Có phản hồi về tắc nghẽn từ tầng mạng -> điều chỉnh tốc độ gửi
    - ABR không đảm bảo một băng thông tối thiểu nhưng cố gắng truyền dữ liệu nhanh nhất có thể
* Dịch vụ truyền với tốc độ thay đổi (VBR - variable bit rate) là dịch vụ thỏa thuận trước về tỷ lệ mất gói dữ liệu, độ trễ có thể chấp nhận được (=CBR)

1. **Lịch sử chuyển mạch gói và chuyển mạch ảo**

* ATM phản ánh nguồn gốc của các nguyên lý chuyển mạch gói và chuyển mạch ảo. Hoạt động dựa trên mạch ảo và cung cấp dịch vụ tốc độ cố định CBR
* mô hình TCP/IP xuất phát từ nhu cầu kết nối các máy tính với nhau
* Tầng mạng có 4 bước thực hiện: đánh địa chỉ, chuyển đoạn dữ liệu thành các gói tin, định tuyến và cuối cùng là tập hợp các gói tin thành các đoạn tin để chuyển lên tầng vận tải

1. **Nguyên tắc định tuyến**

* Có hai cách xây dựng bảng định tuyến, định tuyến tĩnh và định tuyến động.
  + Định tuyến tĩnh thường áp dụng cho tuyến đường rất ít thay đổi hoặc vì một số lý do về an ninh mạng, trong khi đó định tuyến động phù hợp với các tuyến đường khi lưu lượng mạng hay kiến trúc liên kết mạng thường xuyên thay đổi.
  + Trong định tuyến động, một tiến tình định kỳ hoặc theo sự kiện liên tục gửi bản tin về tình trạng của các tuyến mạng
* Trọng tâm của giao thức định tuyến là thuật toán xác định đường đi cho gói tin hay còn gọi là thuật toán tìm đường hoặc thuật toán định tuyến. Mục tiêu là tìm đường đi tốt nhất thường là đường đi có giá trị nhỏ nhất xong thực tế rất phức tạp
* Thuật toán định tuyến toàn cục xác định đường đi với giá thấp nhất giữa nguồn và đích bằng cách sử dụng tất cả thông tin về tổng thể mạng
  + Phải có trước đầy đủ thông tin về đồ thị mạng
* Trong thuật toán phân tán, xác định đường đi có giá thấp nhất được thực hiện dần dần theo cách thức phân tán. Không nút nào có đầy đủ thông tin về giá của tất cả các liên kết trên mạng.Thuật toán này là thuật toán vector khoảng cách (distance vector)

1. **Thuật toán định tuyến theo trạng thái đường truyền**

* Trong thuật toán trạng thái đường truyền, cấu trúc mạng và giá của tất cả các liên kết đều phải được xác định trước được thực hiện bằng hình thức quảng bá
* Thuật toán trạng thái đường truyền thường được sử dụng là thuật toán Dijkstra, đó là thuật toán xác định đường đi có giá thấp nhất từ một nút nguồn đến tất cả các nút khác trên mạng hay thuật toán Link state (LS)
* Chúng ta cần phải kiểm tra qua tất cả các bước là n(n+l)/2 và theo đó chúng ta có thể nói rằng thuật toán Link state có độ phức tạp là O(n2). Thuật toán này có thể được cải tiến bằng cách sư dụng cầu trúc dữ liệu HEAP, độ phức tạp chỉ còn theo hàm logarit của n

1. **Thuật toán vector khoảng cách**

* Thuật toán vector khoảng cách (DV – Distance Vector) là thuật toán lặp, không đồng bộ và phân tán
* Thuật toán DV còn được gọi là thuật toán Bellman - Ford. Nó được áp dụng trong nhiều giao thức định tuyến trong thực tế như: Internet BGP, ISO IDRP...

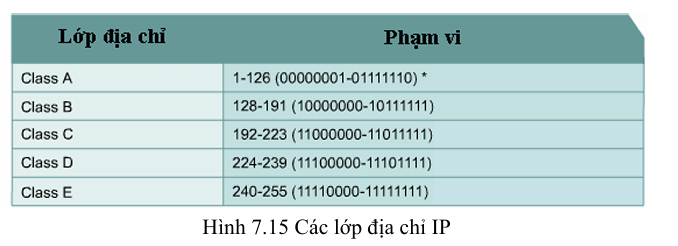
1. **Định tuyến phân cấp**

* Cần phải giảm độ phức tạp trong việc xác định đường đi trên một mạng lớn như Internet. Giải quyết bằng cách nhóm các thiết bị định tuyến thành các vùng tự quản (AS - Autonomous System)
* Các thiết bị định tuyến trong một AS sử dụng cùng một thuật toán định tuyến (ví dụ như thuật toán LS hay DV)
* Thuật toán định tuyến chạy trong mỗi vùng AS được gọi là giao thức định tuyến nội vùng
* AS có nhiệm vụ định tuyến gói tin ra ngoài bằng thiết bị định tuyến nối cầu (gateway router)

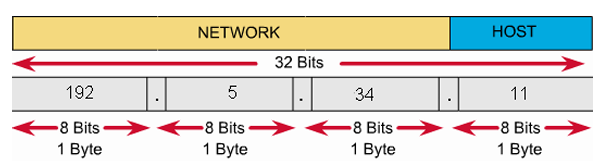
1. **Giao thức IP**

* Một số giá trị địa chỉ IP đặc biệt:
  + 0.0.0.0/8 được sử dụng cho Default Route và Defaul Network
  + 127.0.0.0/8 (127.0.0.0->127.255.255.255): dùng làm địa chỉ IP Lopback
  + 255.255.255.255 dùng làm địa chỉ Global Broadcast
* Một số giao thức dùng phổ biến trong tầng mạng bao gồm IP, IPX, AppleTalk và giao thức dịch vụ mạng không liên kết CLNS/DECNet
* Tầng mạng của Internet sử dụng dịch vụ chuvền mạch gói và sử dụng giao thức IP để vận chuyển dữ liệu của tầng vận tải
* Tầng mạng trong kiểu mạng chuyển mạch gói giống như mạng Internet có ba thành phần chính:
  + Giao thức mạng: xác định địa chỉ tầng mạng, là giao thức liên mạng hay IP
  + Bộ phận xác định đường đi: xác định tuyến đường của gói tin
  + Chức năng quản lý: giao thức bản tin điều khiển Internet – ICMP
* Giao thức IP phiên bản 4 (IPv4) đang được sử dụng rộng rãi, giao thức IP phiên bản 6 (IPv6) đã được phát triển và đang được cài đặt dần thay thế cho phiên bản 4.
* Giao thức IP được thiết kế để phân phát các gói tin qua các mạng khác nhau chứ không quản lý đường đi của các gói tin. Về cơ bản, giao thức IP thuộc loại không liên kết, nó chuyển các gói tin theo dạng cố gắng hết sức, giao thức này có thể hoạt động độc lập với môi trường truyền dẫn

1. **Địa chỉ Ipv4**

****

* Phạm vi trong các submark
* Lớp A: /8, B: /16, C: /24, D-E: ??

****

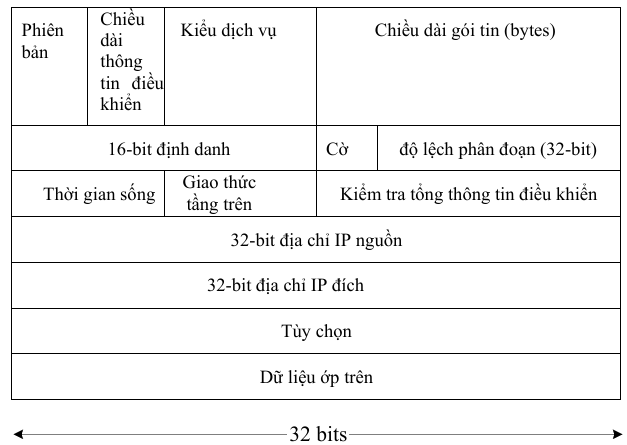
* Nằm giữa máy tính và đường kết nối vật lý là một giao diện ghép nối
* Thiết bị định tuyến khác với máy tính, chức năng của thiết bị định tuyến là chuyển một gói tin từ mạng này sang mạng khác
* Thiết bị định tuyến có thể có nối đến nhiều kênh truyền và bộ phận nằm giữa thiết bị định tuyến và một kênh truyền cũng được gọi là giao diện
* Địa chỉ IP ứng với một giao diện
* Địa chỉ IP có độ dài 32 bit (4 byte) và như vậy không gian địa chỉ có 2^32 địa chỉ IP
* Ví dụ có một địa chỉ 223.1.1.0/24:
  + Kí hiệu: /24 là mặt nạ mạng với ý nghĩa 24 bit đầu tiên trong 32 bit xác định địa chỉ mạng, cũng được xem là tiền tố mạng (network prefix)
  + /24: xác định dải mạng

1. Vấn đề địa chỉ và định tuyến

* Chuyển gói dữ liệu nhờ vào giao thức của tầng liên kết dữ liệu giữa chúng
* Cơ chế đường đi của gói dữ liệu thông qua các thiết bị định tuyến:

1. Khuôn dạng gói dữ liệu IP

* Khuôn dạng gói dữ liệu IP:



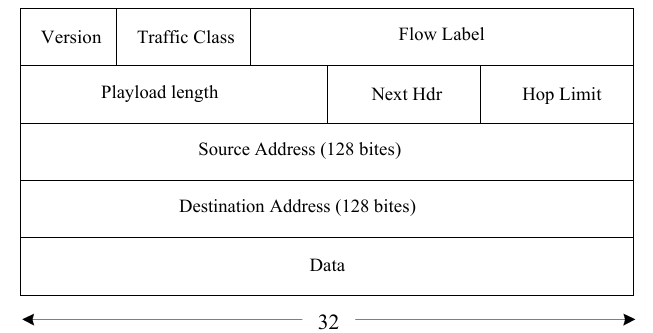
* Các trường khoá trong gói dữ liệu IPv4 như sau:
  + Phiên bản (version): 4 bit
  + Chiều dài thông tin điều khiển (Header length): 4 bit dùng để xác định vị trí bắt đầu của dữ liệu, tiêu đồ gói dữ liệu thường cố định là 20 byte
  + Kiểu dịch vụ (Type of service - TOS): phân biệt các gói dữ liệu IP
  + Chiều dài gói dữ liệu (datagram length): 16bit, kích thước tối đa là 65.535
  + Định danh, cờ và vị trí phân đoạn (Identifer, Flags, Fragmention Offset): phiên mảnh gói dữ liệu IP
  + Thời gian sống (Time-To-live-TTL): mỗi lần đi qua 1 định tuyến thì -1, nếu = 0 thì loại bỏ gói tin
  + Giao thức tầng trên (Upper Protocol): khi gói tin dữ liệu IP đến đích
  + Kiểm tra tổng của phần thông tin điều khiển (Header checksum): kiểm tra lỗi
  + Đia chỉ IP nguồn và đích: 32 bit địa chỉ IP của máy gửi và nhận, IP cùng với số hiệu cổng tạo gọi là socket
  + Tùy chọn (Option): các trường này cho phép mở rộng tiêu đề
  + Dữ liệu (data): gói dữ liệu của tầng vận tải (TCP hay UDP)

1. Phân mảnh và hợp nhất gói tin IP

* Số lượng dữ liệu tối đa của một gói tin trên một đường truyền vật lý được định nghĩa là MTU (maximum transfer unit) – giới hạn đồ dài của gói tin IP
* Đường kết nối có giá trị MTU nhỏ hơn độ dài gói dữ liệu IP, khi đó thiết bị định tuyến phải phân mảnh dữ liệu trong gói dữ liệu IP thành nhiều gói dữ liệu IP nhỏ hơn và sau đó gửi những gói dữ liệu nhỏ hơn này trên đường kết nối, mỗi gói dữ liệu IP nhỏ này được coi là một mảnh
  + Chiếm nhiều thời gian xử lý
* IPv4 quyết định việc hợp nhất các mảnh dữ liệu được thực hiện tại các thiết bị đầu cuối chứ không phải là tại các thiết bị định tuyến
* Máy tính đích sẽ sử dụng các trường identifcation, flag và fragmentation để thực hiện công việc hợp nhất này
  + Gửi dữ liệu: Khi tạo ra một gói dữ liệu IP, ngoài địa chỉ gửi và địa chỉ nhận, máy tính gửi sẽ đặt vào trường identifcation một số định danh tăng dần
  + Chia nhỏ dữ liệu: Khi một thiết bị định tuyến cần chia chia nhỏ một gói dữ liệu, thì tất cả các gói dữ liệu con được tạo ra đều có địa chỉ nguồn, địa chỉ đích và giá trị trường định danh giống hệt gói dữ liệu ban đầu
  + Kiểm tra mảnh: Khi đích nhận được một loạt các gói dữ liệu từ cùng một nơi gửi đến, nó có thể kiểm tra giá trị định danh để xác định liệu những gói dữ liệu đó có là các mảnh của một gói dữ liệu lớn hơn hay không
  + Vì dịch vụ IP không tin cậy nên một số mảnh có thể không đến được đích nên flag quy định:
    - trường cờ của mảnh cuối cùng phải có giá trị 0
    - trường cờ của các mảnh khác có giá trị là 1
* Trường Offset được sử dụng để xác định vị trí của mảnh trong gói dữ liệu IP ban đầu
* Nếu mất mát mảnh thì sẽ loại bỏ cả gói tin và không chuyển lên tầng vận tải
  + Nếu sử dụng giao thức TCP ở tầng vận tải -> khắc phục mất mát -> gửi lại đoạn tin

1. **Địa chỉ IP phiên bản 6**
2. Định dạng gói tin IP V6

* Điểm thay đổi quan trọng nhất của IPv6 chính là khuôn dạng gói tin:
  + Mở rộng khả năng đánh địa chỉ: 32 bit lên 128 bit
  + Phần thông tin điều khiển có độ dài cố định 40 byte: xử lí nhanh hơn
  + Gán nhãn luồng (flow label) và độ ưu tiên (priority): dự đoán được nhu cầu phân biệt giữa các luồng dữ liệu ngay cả khi chưa định nghĩa chính xác được luồng là gì
* IP v6 có cấu trúc đơn giản hơn



* Các trường:
  + Phiên bản (version): 4 bit
  + Traffc class: 8 bit giống TOS của v4
  + Nhãn luồng : 20 bit xác định một luồng chứa gói thông tin
  + Độ lớn dữ liệu (Payload length): độ lớn của phần dữ liệu k tính tiêu đề
  + Next header: xác định giao thức phía trên sẽ nhận dữ liệu (ví dụ tới TCP,UDP)
  + Hop limit: giá trị giảm đi 1 khi qua thiết bị định tuyến, = 0 thì loại bỏ
  + Địa chỉ nguồn và đích (source ang destination addresss): khuôn dạng 128 bit
  + Dữ liệu (data): khi gói tin tới đích, tiêu đề bị loại bỏ và chuyển dữ liệu
* Không cho phép phân mảnh và hợp nhất gói tin tại các thiết bị định tuyến trung gian
  + Nếu một gói dữ liệu IPv6 quá lớn -> thiết bị định tuyến sẽ gửi lại gói tin và báo lỗi ICMP “Packet Too Big” cho bên gửi -> gửi lại với kích thước nhỏ hơn
  + Tăng tốc độ truyền mạng
* Các giao thức tầng vận tải và tầng liên kết dữ liệu đã thực hiện kiểm tra lỗi, chức năng này không cần thiết trong tầng mạng

1. ICMP cho Ipv6

* Giao thức ICMP được sử dụng để thông báo lỗi và cung cấp một số các thông tin hạn chế tới thiết bị đầu cuối (ví dụ lệnh ping). Một phiên bản mới của ICMP được đặc tả cho IPv6 trong khuyến nghị RFC 2463
  + Kiểu mã lỗi mới“Packet Too Big” hay “Unrecognized IP v6 option”

1. **Chuyển từ Ipv4 sang Ipv6**

* Các thiết bị mạng mới đều có khả năng tương thích với IP v4 nhưng các hệ thống cũ thì không thể tương thích với IP v6
* Giải pháp: chọn một thời điểm nào đó, tắt tất cả máy để nâng cấp lên IPv6
  + Có hàng triệu máy -> không tưởng
* RFC 1933 đưa ra hai giải pháp có thể sử dụng đồng thời hay dùng riêng rẽ để dần dần tích hợp các thiết bị sử dụng IPv6 vào thế giới IPv4
* Một giải pháp khác đơn giản hơn là đưa vào các thiết bị cơ chế ngăn xếp đôi (dual-stack), các thiết bị này hỗ trợ cả IPv4 lẫn IPv6

1. **Định tuyến trên Internet**

* Hiện nay có 3 giao thức định tuyến nội miền được sử dụng rộng rãi: RIP (Routing lnformation Protocol), OSPF (Open Shortest Path First) và EIGRP (Cisco propriety Enhanced Interior Gateway Routing Protocol). Giao thức định tuyến liên miền thường được sử dụng là giao thức BGP.

1. **Giao thức RIP**

* RIP là một trong những giao thức định tuyến nội miền đầu tiên, nó có một số đặc điểm sau:
  + Định tuyến nội miền: Cho phép trao đổi thông tin giữa các thiết bị định tuyến trong một miền
  + Đo khoảng cách bằng chặng đường đi: Bằng số lượng thiết bị định tuyến trung gian
  + Truyền thông không tin cậy: RIP sử dụng UDP để chuyển bản tin về các tuyến đường
  + Gửi tin dạng quảng bá và nhóm: RIP v1 – truyền quảng bá, RIP v2 – nhóm
  + Thuật toán vector khoảng cách: RIP sử dựng thuật toán vector khoảng cách.
  + Các máy tính có thể thụ động nhận thông tin từ các thiết bị định tuyến: RIP cho phép các thiết bị đầu cuối (chủ yếu là máy tính) lắng nghe và cập nhật bảng định tuyến
* Ưu điểm chính của RIP là tính đơn giản, không đòi hỏi cấu hình nhiều
* Nhược điểm: đỗ trễ nhất định, chỉ phù hợp các mạng kích thước nhỏ

1. **Giao thức OSPF**

* Có các đặc điểm:
  + Định tuyến nội miền: Cho phép trao đổi thông tin giữa các thiết bị định tuyến trong một miền
  + Hỗ trợ CIDR: hỗ trợ việc phân mạng, tạo lập mạng con.
  + Trao đổi các thông tin đã được kiểm chứng: xác minh nhận bản tin, ngăn ngừa cuộc tấn công giả mạo
  + Sử dụng thuật toán tìm đường dựa trên trạng thái kênh truyền
  + Hỗ trợ phân cấp trong miền:
    - Ưu điểm chính của OSPF là cho phép tiếp tục phân một miền thành nhiều miền con

1. **Giao thức BGP**

* Giao thức cổng biên (BGP - Border Gateway Protocol) được mô tả trong các khuyến nghị RFC 1771, 1772, 1773, được xem là một chuẩn hiển nhiên trong định tuyến giữa các vùng trên mạng Internet
* Nhiệm vụ của nó là định tuyến giữa các vùng được quản trị độc lập với nhau
* BGP cho phép cung cấp các thông tin định tuyến giữa các vùng, mỗi tuyến đường được xem là một chuỗi các vùng tự quản liên tiếp nhau
* Giao thức này cũng hỗ trợ việc thiết lập chính sách như ggười quản trị có thể áp dụng những chính sách nào đó
* Sử dụng kết nối TCP có độ tin cậy cao

1. **Các giao thức khác**
2. **Giao thức ICMP**

* ICMP được các thiết bị đầu cuối, thiết bị định tuyến sử dụng để trao đổi các thông tin tầng mạng với nhau (chủ yếu cho việc báo lỗi), giao thức này được đặc tả trong RFC 792
* ICMP thường được coi là một phần của bộ giao thức IP
* Thông báo ICMP có trường kiểu (type) và trường mã (code) và chứa 8 byte đầu tiên của gói dữ liệu IP gây ra lỗi (nguyên nhân để tạo ra thông báo ICMP)

1. **Cấp phát địa chỉ IP**

* Câu lệnh traceroute cho phép người sử dụng xác định tất cả các thiết bị định tuyến trên một tuyến đường giữa bất kỳ hai thiết bị đầu cuối nào
* 2 cách gán địa chỉ IP:
  + Gán thủ công: Người quản trị hệ thống thiết lập cấu hình
  + Gán tự động: dịch vụ cấp phát địa chỉ động, sử dụng giao thức: RARP, BOOTP, DHCP

1. Giao thức RARP

* Là giao thức dùng để tìm địa chỉ IP khi biết địa chỉ vật lý MAC

1. Giao thức BOOTP

* Hoạt động ở lớp ứng dụng
* Phục vụ cho việc trong việc cấp phát địa chỉ IP động

1. Giao thức DHCP

* Dùng để gán các địa chỉ IP động cho máy trạm khi tham gia vào mạng

1. **Chuyển đổi địa chỉ**
2. Giao thức ARP

* Trên các mạng theo mô hình TCP/IP, giao thức ARP được dùng để tìm một địa chỉ MAC tương ứng với một địa chỉ IP

1. Chuyển địa chỉ mạng

* Chuyển địa chỉ mạng (NAT - Network Address Translation) là cơ chế dùng để chuyển đổi địa chỉ IP trong mạng nội bộ với mạng bên ngoài (IP công cộng). Cơ chế này được dùng chủ yếu cho hai mục đích:
  + Giải quyết vấn đề thiếu hụt địa chỉ IP
  + Che giấu địa chỉ IP trong mạng nội bộ nhằm tăng cường khả năng bảo mật mạng
* Có 2 kiểu cơ bản chuyển địa chỉ IP là NAT (Network Address Translation) và PAT (Port Address Translation).
  + Trong khi cơ chế NAT chỉ cho phép ánh xạ ở mức địa chỉ IP
  + PAT cho phép ánh xạ thêm tới mức cổng, do đó tại một thời điểm số lượng liên kết ra mạng bên ngoài sẽ nhiều hơn

1. **Chia mạng**
2. **Địa chỉ MAC**

* Ipconfig/all
* Bao gồm 48 bit. Chia thành 2 phần: 24 bit đầu là Provider ID, 24 bit sau là Product ID 🡪 địa chỉ MAC có tính đơn nhất
* Địa chỉ có dạng không phân cấp
* Biểu diễn dưới dạng hệ cơ số 16
* Mỗi cổng được gán 1 MAC: không thay đổi
  + Mạng Lan: FF-FF-FF-FF
* Cài trong ROM của NIC
* Địa chỉ MAC Broadcast: 48x”1” hoặc 12x”F”

**Chương 8: Tầng Liên Kết**

1. **Mô hình dịch vụ tầng liên kết dữ liệu**

* Giao thức tầng liên kết dữ liệu được sử dụng để truyền các khung dữ liệu trên một môi trường vật lý
* Chức năng của giao thức tầng liên kết dữ liệu khi gửi và nhận khung dữ liệu bao gồm: phát hiện lỗi truyền lại, điều khiển lưu lượng và truy cập ngẫu nhiên.
* Giao thức tầng liên kết dữ liệu rất đa dạng: Ethernet, token ring, FDDI và PPP, đôi khi ATM và frame relay
* Nếu nhiệm vụ của tầng mạng là chuyển gói dữ liệu của tầng vận tải từ máy gửi từ máy nhận thì giao thức của tầng liên kết dữ liệu có nhiệm vụ chuyển gói dữ liệu tầng mạng giữa hai nút kế tiếp trên đường truyền
* Cung cấp những dịch vụ sau:
  + Đóng khung dữ liệu (frame) và truy cập kênh truyền (link access): Khung gồm gói dữ liệu và một trường tiêu đề khác
  + Dịch vụ truyền tin cậy: nếu cung cấp dịch vụ truyền tin cậy, các giao thức tầng liên kết dữ liệu đảm bảo chuyền chính xác khung dữ liệu
    - Xác nhận và trả lại nếu lỗi
  + Tầng liên kết dữ liệu thường được sử dụng trên đường truyền có tỉ 1ệ lỗi cao, mục đích là sửa lỗi ngay trên đường truyền lỗi chứ không truyền lại
  + Kiểm soát lưu lượng: Khả năng lưu trữ tạm thời các khung dữ liệu tại các nút trên mỗi phía của đường truyền không phải là vô hạn. Nếu không kiểm soát lưu lượng, bộ đệm phía nhận có thể bị tràn và khung dữ liệu sẽ bị mất
  + Phát hiện lỗi: Nút nhận có thể nhận được bit 0 trong khi phía gửi gửi bit 1 hay ngược lại, nguyên nhân bit bị lỗi có thể do tín hiệu bị suy hao hay nhiễu điện từ
    - Phức tạp và được triển khai bằng phần cứng
  + Sửa lỗi: xác định được nơi lỗi và sửa
  + Bán song công và song công (Half duplex, full duplex)

1. **Giao thức đa truy nhập**

* Có hai loại đường truyền: truyền điểm-điểm và truyền quảng bá
  + Đường truyền điểm-điểm chỉ có một bên gửi và một bên nhận duy nhất ở hai đầu của đường truyền.
    - Nhiều giao thức tầng liên kết dữ liệu đã được thiết kế cho đường truyền điểm-điểm như HDLC (High Data Link Control) và PPP (point-to-point Protocol)
  + Quảng bá cho phép có nhiều nút gửi và nút nhận cùng kết nối đến một kênh truyền duy nhất (1 gửi tất cả đều nhận)
    - Ethernet là công nghệ quảng bá được triển khai rộng rãi nhất
* Phát thanh, truyền hình là những dịch vụ quảng bá, nhưng đó chỉ là quảng bá một chiều
* Quảng bá dễ gây xung đột đường truyền
* Để hiệu suất của kênh truyền quảng bá đạt giá trị tối đa khi nhiều nút muốn gửi dữ liệu cần phải có cơ chế phối hợp giữa những nút có nhu cầu truyền, cơ chế phối hợp này là trách nhiệm của giao thức đa truy cập
* Phân loại các giao thức đa truy cập vào ba loại: giao thức phân chia kênh truyền, giao thức truy cập ngẫu nhiên và giao thức truy cập lần lượt
* Giao thức đa truy cập trên kênh truyền quảng bá với tốc độ R b/s lý tưởng (đọc thêm)

1. **Giao thức phân chia kênh truyền**

* Phân kênh theo thời gian (TDM) và theo tần số (FDM) là hai kỹ thuật có thể được sử dụng để phân chia băng thông của kênh truyền giữa tất cả các nút dùng chung kênh truyền đó
  + Kỹ thuật TDM đã loại trừ được xung đột và hoàn toàn công bằng: mỗi nút có được tốc độ truyền riêng trung bình R/N b/s
  + Phương pháp FDM chia kênh truyền R b/s thành các tần số khác nhau (mỗi tần số có băng thông R/N) và gán một tần số đó cho mỗi nút
  + Nhược điểm chung: mỗi nút bị giới hạn bởi tốc độ trung bình R/N b/s và phải đợi đến thời gian truyền của mình ngay cả khi nó là nút duy nhất có nhu cầu gửi
* Giao thức phân chia kênh truyền thứ ba là chia mã (CDMA - Code division multiple access)
  + Nếu TDM và FDM gán khoảng thời gian và tần số cho các nút thì CDMA gán cho mỗi nút một mã khác nhau
  + Nút sử dụng mã duy nhất này để mã hoá dữ liệu gửi đi
  + CDMA cho phép nhiều nút gửi đồng thời và các nút nhận tương ứng nhận đúng dữ liệu gửi cho mình, miễn là nó biết được mã của nút gửi
  + từng được sử dụng trọng hệ thống quốc phòng nhờ đặc tính chống nhiễu, đa truy cập kênh không dây (ngày nay)

1. **Giao thức đa truy cập ngẫu nhiên**

* Nút truyền luôn luôn truyền dữ liệu với tốc độ cao nhất của kênh truyền R b/s
* Khi có xung đột, nút liên quan đến xung đột truyền lại khung dữ liệu theo khoảng thời gian ngẫu nhiên cho đến khi khung dữ liệu đó đến đích an toàn
* Hai loại đại diện: giao thức ALOHA và giao thức đa truy cập cảm nhận sóng mang (CSMA/CD)

1. Slotted ALOHA (thu gọn = chat gpt)

* Cơ chế truyền dữ liệu:
  + Dữ liệu được chia thành các khung có kích thước cố định L bit.
  + Thời gian chia thành các khoảng L/R giây (đủ để truyền một khung).
  + Các nút đồng bộ, chỉ truyền tại đầu mỗi khoảng thời gian.
* Phát hiện và xử lý xung đột:
  + Nếu có xung đột (nhiều nút cùng truyền), tất cả nút đều nhận biết trong khoảng thời gian đó.
  + Nút gặp xung đột sẽ chờ và thực hiện truyền lại với xác suất p tại các khoảng thời gian sau cho đến khi thành công.
* Xác suất thành công:
  + Xác suất một nút truyền thành công khi N nút hoạt động là Ps​=Np(1−p)^N−1
  + Hiệu suất đạt giá trị cực đại khi N tiến đến vô hạn, đạt 1/e≈0.37
* Hiệu suất kênh:
  + Hiệu suất tối đa là 0.37R, tức chỉ 37% thời gian được sử dụng hiệu quả.
  + 37 % thời gian kênh rỗi, 26% thời gian xảy ra xung đột.
* Tính chất:
  + Slotted ALOHA là thuật toán phân tán, mỗi nút tự quyết định truyền lại.
  + Đòi hỏi đồng bộ thời gian giữa các nút.
* Tóm tắt: Slotted ALOHA cho phép các nút truyền dữ liệu hiệu quả nhưng giới hạn thông lượng tối đa ở mức 37% tốc độ kênh do xung đột và thời gian kênh rỗi.

1. ALOHA thuần túy (rút gọn = chat gpt)

* Giao thức slotted ALOHA đòi hỏi tất cả các nút đồng bộ việc truyền tại đầu mỗi khoảng thời gian.
* Giao thức ALOHA thuần túy không chia khoảng thời gian, khi có dữ liệu từ lớp mạng chuyển xuống nó chuẩn bị các khung dữ liệu và ngay lập tức chuyển khung dữ liệu đầu tiên lên kênh truyền quảng bá
* Nếu xảy ra xung đột thì ngay sau khi truyền xong khung dữ liệu này, nút đó ngay lập tức truyền lại khung dữ liệu đó với xác suất p
* Xung đột xảy ra nếu bất kỳ nút nào khác bắt đầu truyền trong khoảng thời gian từ t0​−1 đến t0+1
* Xác suất truyền thành công của một nút: Ps​=p(1−p)2(N−1)
* lớn là 1/(2e)≈0.18,bằng một nửa Slotted ALOHA.

1. Đa truy cảm nhận sóng mang

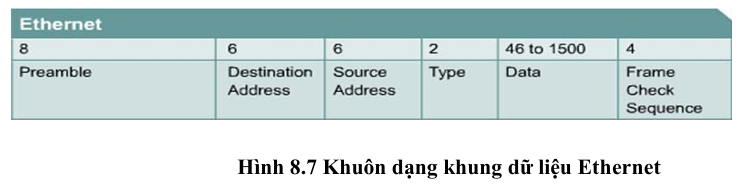
* Giao thức CSMA (Carrier Sense Multuple Access) và CSMA/CD (CSMA with Collision Detection):
  + Nghe trước khi nói và nghe trong khi nói
  + Công nghệ Ethernet sử dụng
  + Xung đột vẫn xuất hiện do độ trễ của tín hiệu

1. **Các công nghệ kết nối**
2. **Công nghệ Ethernet**

* Thống trị thị trường mạng cục bộ
  + Tốc độ cao
  + Dễ lắp và rẻ
  + Liên tục nâng cấp về tốc độ và đồng thời đảm bảo tính tương thích giữa các tốc độ khác nhau
* Các máy tính được kết nối với nhau bằng một kênh truyền chung tương tự như hệ thống ống dẫn nước sinh hoạt trong một khu tập thể
  + Mỗi máy tính kết nối vào mạng phải có bộ phận tiếp giáp gọi là giao diện mạng (NIC – Network Interface Controler)

1. Cấu trúc khung Ethernet

* Sử dụng cáp đồng trục hay cáp quang, chạy với tốc độ 10 Mbps, 100 Mbps hay 1 Gbps

****

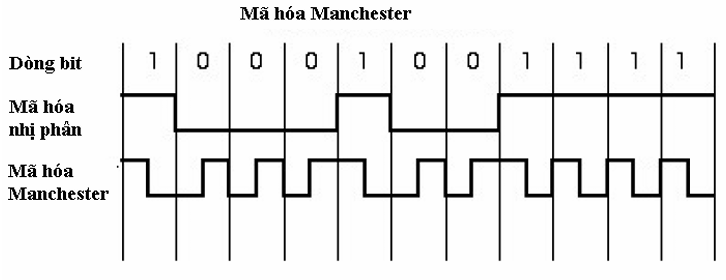
* + Preamble (mào đầu, 8 byte): bằng đầu bằng preamble dài 8 byte, trong đó
    - 7 byte đầu có giá trị: 10101010
    - byte thứ tám có giá trị 10101011
  + Destination Address (Địa chỉ đích, 6 byte): Trường này chứa địa chỉ vật lý của Card mạng nhận
    - BB-BB- BB-BB-BB-BB(chính nó) hoặc FF-FF-FF-FF-FF-FF(LAN)
  + Source Address (Địa chỉ nguồn, 6 byte): Trường này chứa địa chỉ vật lý của Card mạng gửi khung dữ liệu
  + Type (Trường kiểu, 2 byte): Trường này cho phép Ethernet hỗ trợ nhiều giao thức tầng mạng khác nhau
  + Data (Trường dữ liệu, từ 46 đến 1500 byte): chứa gói dữ liệu của IP
    - <46: chèn thêm bit giả, > 1500: chia nhỏ
  + CRC (Mã kiểm tra dư thừa vòng - Cyclic Redundancy Check, 4 byte): phát hiện lỗi do suy hao năng lượng điện từ của tín hiệu

1. Dịch vụ truyền số liệu không liên kết

* Tất cả công nghệ Ethernet cung cấp cho tầng mạng dịch vụ không liên kết – không tin cậy

1. Dải tần cơ sở và mã hóa Manchester

* Ethernet sử dụng băng tần cơ sở (baseband) nghĩa là Card mạng gửi tín hiệu số trực tiếp vào kênh truyền dùng chung
  + Card mạng không chuyển tín hiệu sang đại tần số

****

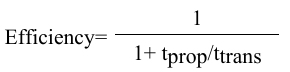
* Đồng bộ hóa card mạng nhận và gửi, card nhận thu được tín hiệu của mỗi bit là 0 hoặc 1
* Thường được sử dụng ở tầng vật lý hơn

1. CSMA/CD

* Vì các nút trên mạng cục bộ Ethernet kết nối một kênh truyền bá dùng chung, vì vậy khi card mạng gửi đi một khung dữ liệu, tất cả các card mạng trên LAN đều nhận được khung đó, Ethernet dùng thuật toán đa truy cập CSMA/CD
* Hình dạng Ring
* Hiệu suất cải thiện hơn ALOHA
* Giao thức CSMA/CD làm việc như sau:
  + Card mạng nhận gói tin từ tầng mạng, tạo ra khung dữ liệu Ethernet và đặt vào trong bộ đệm
  + Nếu Card mạng thấy kênh rỗi thì truyền còn bận thì đợi đến khi rỗi
  + Trong khi chuyền, kiểm tra có tin hiệu nào không, nếu không thì truyền thành công
  + trong khi đang truyền thì lập tức nó dừng lại không truyền và gửi đi tín hiệu báo nhiễu 32 bit (tín hiệu JAM) đảm bảo các mạng khác đều phát hiện xung đột
  + Sau khi dừng phát và gửi tín hiệu báo nhiễu, Card mạng sẽ thực hiện thuật toán exponential backoff (đọc thêm)

1. Hiệu suất Ethernet

* Hiệu suất của Ethernet là tỉ lệ thời gian không có xung đột trên kênh truyền khi có nhiều nút truyền số liệu, mỗi nút cần truyền nhiều khung dữ liệu trong một khoảng thời gian nhất định
* Giả sử tprop là thời gian lớn nhất năng lượng tín hiệu lan tỏa giữa hai Card mạng hay độ trễ. Giả sử ttrans là thời gian để truyền đi một khung dữ liệu Ethernet với độ lớn cực đại xấp xỉ 1,2 ms với Ethernet 10 Mbps, ta sử dụng công thức:

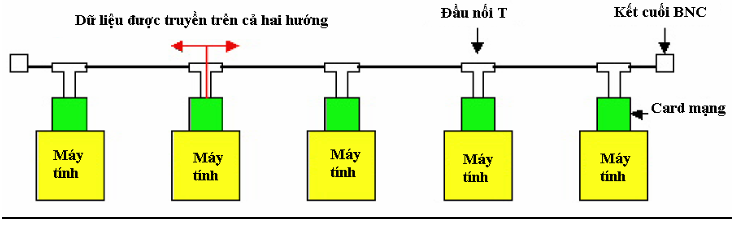
****

* + Nếu tprop = 0 thì hiệu suất đạt 1

1. Các công nghệ Ethernet

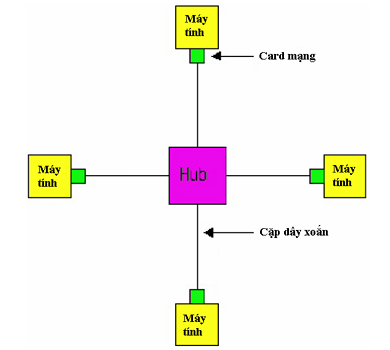
* Công nghệ Ethernet sử dụng cáp đồng trục (coaxial cable) có hình trạng dạng bus, tốc độ truyền là 10 Mbps
  + Sử dụng cặp dây xoắn, hình trạng hình sao, tốc độ truyền từ 10 đến 1000Mbps. Gigabyte Ethernet sử dụng cả sợi quang hay dây đồng xoắn, truyền với tốc độ 1 Gbps
  + Những công nghệ Ethernet này được chuẩn hóa bởi IEEE 802.3
* Bộ lặp (Repeater): thiết bị tái tạo lại tín hiệu trước khi không nhận ra tín hiệu đó – tránh mất tín hiệu do khoảng cách
  + Bộ lặp là thiết bị tầng vật lý xử lý trên từng bit riêng lẻ chứ không phải trên khung dữ liệu

**Ethernet 10BASE2**

****

* 10Base2 là một công nghệ Ethernet rất phổ biến trong những năm 90 của thế kỷ trước
* Số 10 trong 10base2 có nghĩa là tốc độ truyền tối đa là 10 Mbit/s, số 2 có ý nghĩa khoảng cách tối đa giữa hai trạm không có bộ lặp ở giữa không vượt quá 200m
* Dùng chung cáp đồng trục
* Hình dạng bus

**Ethernet 10BaseT và 100BaseT**

****

* 10BaseT và 100BaseT là hai công nghệ tương tự nhau, điểm khác biệt quan trọng nhất là tốc độ truyền của 10BaseT là 10Mbit/s còn kia là 100Mbit/s
* Hình dạng sao
* một thiết bị trung tâm được gọi là hub (đôi khi gọi là bộ tập trung – concentrator), thực chất đó là bộ lặp nhiều cổng, khoảng cách đến các nút max là 200m
* Tại mỗi đầu của kết nôi có một bộ nối theo chuẩn RJ-45 – tương tự như đầu nối chuẩn RJ-11 được sử dụng cho điện thoại thông thường
* Chữ “T” là viết tắt của “Twisted pair” nghĩa là cặp dây xoắn

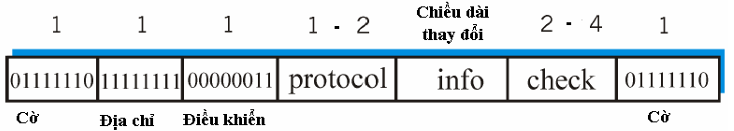
**Gigabit Ethernet**

* Truyền 1Gbit/s

1. **Kết nối mạng diện rộng**
2. Giao thức PPP

* PPP là giao thức được sử dụng chủ yếu khi người dùng truy cập Internet từ nhà thông qua đường điện thoại quay số
* Một trong những giao thức tầng nên kết dữ liệu được sử dụng nhiều nhất nhất ngày nay
  + Giao thức quan trọng thứ hai là HDLC (High Level Data Link Control).
* Giao thức PPP là giao thức tầng liên kết dữ liệu trên kênh truyền nối trực tiếp giữa hai nút - mỗi nút ở một đầu của đường truyền
* Đường truyền PPP có thể là đường điện thoại quay số (ví dụ kết nối modem 56k), đường truyền SONET, kết nối X.25 hoặc mạng ISDN
* Chủ yếu được lựa chọn để kết nối máy tính gia đình đến ISP thông qua đường dây điện thoại. IETF đã đặt ra cho mọi thiết kế của PPP :
  + Đóng khung dữ liệu (Framing): phía gửi đóng khung gói dữ liệu, phía nhận xác định được vị trí bắt đầu và kết thúc của khung
  + Tính trong suốt: có thể truyền bất kỳ gói tin nào
  + Hỗ trợ nhiều giao thức tầng mạng: Giao thức PPP phải có khả năng hỗ trợ nhiều giao thức tầng mạng (ví dụ, IP và DECnet) trên cùng đường truyền vật lý tại cùng một thời điểm
  + Hỗ trợ nhiều kiểu đường truyền: Ngoài khả năng hỗ trợ nhiều giao thức ở tầng cao hơn, PPP phải có khả năng vận hành trên nhiều kiểu đường truyền khác nhau bao gồm đường truyền tuần tự (truyền lần lượt từng bit một) hoặc song song (truyền nhiều bit cùng một lần), đồng bộ (truyền tín hiệu đồng hồ cùng với bit dữ liệu) hoặc dị bộ, truyền với tốc độ chậm hoặc cao, tín hiệu điện tử hoặc quang học
  + Phát hiện lỗi: PPP phía nhận có khả năng phát hiện lỗi bit trong khung dữ liệu nhận
  + Thời gian kết nối: PPP phải có khả năng phát hiện đường truyền bị lỗi ở mức link
  + Thoả thuận địa chỉ tầng mạng: PPP phải cung cấp cơ chế cho phép hai thực thể tầng mạng tham gia truyền thông (IP) có thể học, đặt cấu hình mạng cho nhau
  + Đơn giản:
* Tuy nhận PPP không yêu cầu
  + Sửa lỗi
  + Kiểm soát lưu lượng
  + Đánh số thứ tự
  + Đường truyền đa điểm

**Khuôn dạng khung dữ liệu**



* Trường giao thức (protocol): Trường giao thức cho PPP xác định giao thức tầng trên sẽ nhận dữ liệu trong khung dữ liệu PPP, các mã 16bit được sử dụng cùng PPP
  + Giao thức IP (dữ liệu trong khung dữ liệu PPP là gói dữ liệu IP) ứng với giá trị 21h, giao thức AppleTalk là 29h, DFCnet là 27h, giao thức điều khiển đường truyền PPP là C021h và giao thức điều khiển IP là 8021
* Thông tin: Trường này chứa gói tin được giao thức tầng mạng gửi đi trên đường truyền PPP (là gói tin IP).
* Checksum: Trường checksum được sử dụng để phát hiện các bit bị lỗi trong khung dữ liệu nhận được

**Chèn Byte**

* Tránh các byte mở đầu và kết thúc (01111110)
* Chèn byte điều khiển đặc biệt 01111101 vào trước dữ liệu thực không phải cờ để phân biệt dữ liệu với cờ

1. Giao thức điều khiển đường truyền PPP

* Quá trình khởi tạo, duy trì, báo lỗi và đóng đường truyền PPP được thực hiện nhờ giao thức điều khiển đường truyền (LCP -Linh Con trol Protocol) và các giao thức điều khiển mạng của PPP
* Gói tin IPCP được đặt trong khung dữ liệu PPP. IPCP cho phép hai thực thể IP thay đổi hoặc đặt cấu hình địa chỉ IP hay thỏa thuận có nén gói dữ liệu IP không
  + Những giao thức kiểm soát mạng tương tự được đưa ra cho những giao thức tầng mạng khác như DECNET [RFC 1762] và AppleTalk [RFC 1378].

1. **Các thiết bị mạng nội bộ**
2. **Bộ tập trung**

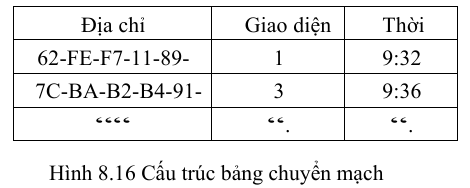
* là một thiết bị đơn giản sao chép tín hiệu đến từ một cổng ra tất cả các cổng còn lại, bản chất của hub là bộ lặp thao tác trên bit, vì thế chúng là thiết bị ở tầng vật lý
* Khi bit đi vào một cổng, hub sẽ truyền bit này qua tất cả các cổng khác.
* Xoăn đôi truyền tối đa 100m

1. **Cầu nối**

* Khác với hub là thiết bị chỉ hoạt động ở tầng vật lý, cầu nối (bridge) xử lý trên khung dữ liệu Ethernet -> tầng liên kết dữ liệu
* Thực tế, cầu nối chính là thiết bị chuyển mạch thực hiện việc chuyển và lọc các khung dữ liệu căn cứ trên địa chỉ vật lý

1. Nguyên lí lọc và chuyển tiếp

* Lọc là khả năng xác định liệu sẽ chuyển tiếp khung dữ liệu đến cổng nào đó hay loại bỏ, chuyển tiếp là khả năng xác định cổng kế tiếp để chuyển khung dữ liệu đi
* Bridge thực hiện hai chức năng này nhờ bảng chuyển mạch, mỗi hàng trong bảng ứng với một nút đích trên mạng LAN
  + mỗi hàng trong bảng chuyển mạch gồm địa chỉ vật lý của nút và cổng bridge có thể dẫn đến nút đó, thời gian thiết lập hàng đó trong bảng



1. Xây dựng bảng chuyển mạch

* Một đặc tính quan trọng của bridge là khả năng tự học, đó là bảng chuyển mạch của bridge được xây dựng tự động

1. Spanning Tree

* Nếu bridge bị hỏng thì các phân đoạn mạng sẽ không kết nối được với nhau, để dự phòng rủi ro, người ta thường xây dựng mạng với nhiều đường nối giữa các phân đoạn mạng -> loạn -> tạo vòng lặp -> khung tin không thể đến đích
* Trong giao thức spanning tree, bridge liên lạc với bridge khác trên mạng nội bộ để xác định tập con của hình trạng ban đầu không có vòng lặp. Sau khi xác định được spanning tree, bridge chỉ kết nối với các cổng phù hợp để tạo spanning tree từ hình trạng ban đầu

1. So sánh cầu nối và thiết bị định tuyến

* Thiết bị định tuyến hoạt động theo kiểu kiểu lưu và chuyển, nó chuyển tiếp gói tin dựa trên địa chỉ tầng mạng
* Thiết bị chuyển mạch ở tằng liên kết dữ liệu kiểu lưu và chuyển
* Bridge là thiết bị kiểu “cắm vào là chạy”, tính năng được tất cả các nhà quản trị mạng ưa thích. Bridge cũng có tốc độ cao khi lọc và chuyển khung dữ liệu
* Giao thức STP đảm bảo các khung dữ liệu chỉ được chuyển trên một đường duy nhất, thậm chí khi có nhiều đường dẫn trực tiếp nhưng bị phong tỏa giữa nguồn và đích
  + Sự hạn chế của giao thức STP nằm ở khả năng tải trên đường truyền khi nó có thể đã được lan truyền đến tất cả các đường truyền khác của mạng cũ
* bridge không đưa ra bất cứ sự bảo vệ nào để chống lại sự phát tán hàng loạt nếu máy tính bị lỗi và truyền đi một luồng dữ liệu liên tục, bridge sẽ chuyển tất cả những khung dữ liệu này khiến toàn bộ mạng bị tắc nghẽn

1. Kết nối các đoạn mạng qua đường trục
2. **Switch**

* Thực chất đó là bridge nhiều cổng
* Ưu điểm của switch nhiều cổng và ở chỗ dễ dàng kết nối trực tiếp giữa các máy tính với switch. Khi một máy tính có đường kết nối trực tiếp song công với switch, nó có thể truyền và nhận dữ liệu ở tốc độ truyền tối đa của Card mạng và không bao giờ xảy ra xung đột

**Các loại chuyển mạch:**

* Giả sử khung dữ liệu đến switch từ một cổng nào đó, switch chỉ cần phải đọc 6 byte đầu tiên là đã có thể xác định được cần phải chuyển dữ liệu đó đến cổng nào
* thiết kế ba loại: chuyển mạch lưu và chyển tiếp, chuyển mạch xuyên suốt và chuyển mạch Fragment-Free
  + Lưu và truyển tiếp: Độ trễ tương đối lớn, đảm bảo độ chính xác
  + Xuyên suốt: có tốc độ cao, không chính xác
  + Dung hòa cả 2: swich chỉ chuyển tiếp các khung dữ liệu sau khi đã nhận được ít nhất 64 bytes..

1. **Kết nối không dây**

* Kết nối không dây là hình thức kết nối giữa các thành phần trong mạng không sử dụng các loại cáp như một mạng thông thường, môi trường truyền thông của các thành phần trong mạng là không khí. Các thành phần trong mạng sử dụng sóng điện từ để truyền thông với nhau.
* Có thể truyền nhận thông tin ở 2 dãy tần 2.4 Ghz và 5Ghz

1. **Các mô hình kết nối mạng không dây**

**Mô hình mạng độc lập:**

* Các nút di động tập trung lại trong một không gian nhỏ để hình thành nên kết nối ngang hàng, chúng có thể trao đổi thông tin trực tiếp với nhau

**Mô hình mạng cơ sở:**

* Bao gồm các điểm truy nhập (AP - Access Point) gắn với mạng đường trục hữu tuyến và giao tiếp với các thiết bị di động trong vùng phủ sóng, nó đóng vai trò chuyển tiếp cho các thiết bị di động của người sử dụng

**Mô hình mạng mở rộng:**

* AP thực hiện việc giao tiếp thông qua hệ thống phân phối, đó là hệ thống được tiếp sóng trở lại một đích trong cùng một mạng cơ sở, chuyển tiếp trên hệ thống phân phối tới một AP khác
* Các thông tin nhận bởi AP từ hệ thống phân phối được truyền tới BSS sẽ được nhận bởi trạm đích

1. **Ưu và nhược điểm của kết nối không dây**
   * Sự tiện lợi:
   * Hiệu quả
   * Triển khai:
   * Khả năng mở rộng:

* Nhược điểm:
  + Bảo mật
  + Phạm vi
  + Độ tin cậy
  + Tốc độ